



HAL
open science

Contribution à l'ingénierie dirigée par les modèles en EIAH : le cas des situations-problèmes coopératives

Thierry Nodenot

► **To cite this version:**

Thierry Nodenot. Contribution à l'ingénierie dirigée par les modèles en EIAH : le cas des situations-problèmes coopératives. Autre [cs.OH]. Université de Pau et des Pays de l'Adour, 2005. tel-00160740

HAL Id: tel-00160740

<https://theses.hal.science/tel-00160740v1>

Submitted on 6 Jul 2007

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

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

Habilitation à Diriger des Recherches en Informatique

présentée par

Thierry Nodenot

Laboratoire d'Informatique de l'Université de Pau et des Pays de l'Adour (LIUPPA)
Avenue de l'Université, B.P. 1155, 64013 Pau Cedex, France
Courriel : Thierry.Nodenot@iutbayonne.univ-pau.fr

Contribution à l'ingénierie dirigée par les modèles en EIAH : le cas des situations-problèmes coopératives

Soutenue le 30 novembre 2005 à l'IUT de Bayonne

Après avis de :

Alain DERYCKE, Professeur, Université de Lille 1
Monique GRANDBASTIEN, Professeur, Université de Nancy 1
Pierre TCHOUNIKINE, Professeur, Université du Maine

Rapporteur
Rapporteur
Rapporteur

Devant la commission d'examen formée des rapporteurs et de :

Franck BARBIER, Professeur, Université de Pau et des Pays de l'Adour
Mauro GAIO, Professeur, Université de Pau et des Pays de l'Adour

Président
Examineur

Table des matières

INTRODUCTION	1
1 POSITIONNEMENT DES TRAVAUX	5
1.1 LE PROJET : PRODUIRE DES EIAH METTANT EN ŒUVRE DES METHODES PEDAGOGIQUES ACTIVES ET SITUEES	5
1.1.1 <i>Des EIAH proposant une utilisation effective des technologies</i>	6
1.1.1.1 Exemples d'utilisation réussie de la technologie au service de l'apprentissage humain.....	7
1.1.1.2 Exemples d'exploitation des outils de type TIC au service de l'apprentissage et de l'enseignement.....	9
1.1.2 <i>Des EIAH reposant sur des théories d'apprentissage et des stratégies pédagogiques plus complémentaires qu'en opposition</i>	11
1.1.2.1 Vers une approche intégratrice des théories d'apprentissage que les enseignants utilisent	11
1.1.2.2 De la complémentarité de différentes méthodes actives.....	12
1.2 CADRE THEORIQUE DE REFERENCE POUR LA CONCEPTION D'EIAH	15
1.2.1 <i>Présentation de la théorie empirique de l'apprentissage</i>	16
1.2.1.1 Tout apprentissage est recherche de sens	16
1.2.1.2 Tout apprentissage est le résultat d'une mise en œuvre par l'apprenant d'une habileté cognitive définie à partir de son système de représentation de la situation	17
1.2.1.3 Seule l'activité de transfert de l'habileté cognitive permet de s'assurer que l'on a appris.....	20
1.2.2 <i>Fondements de la théorie empirique de l'apprentissage</i>	20
1.3 QUESTIONS DE RECHERCHE POSEES PAR LA PROBLEMATIQUE CHOISIE A LA COMMUNAUTE EIAH	21
1.3.1 <i>Verrous liés aux caractéristiques des environnements supports utilisés</i>	23
1.3.2 <i>Verrous liés aux principes d'ingénierie utilisés</i>	24
1.4 CONCLUSIONS.....	25
2 PROBLEMATIQUE ET DEMARCHE DE RECHERCHE	27
2.1 VERS DES ENVIRONNEMENTS-SUPPORTS AU SERVICE DES SITUATIONS-PROBLEMES COOPERATIVES.....	28
2.1.1 <i>Cahier des charges</i>	28
2.1.2 <i>Travaux menés au niveau national et international pour promouvoir les environnements supportant l'apprentissage à base de situations-problèmes coopératives</i>	29
2.1.2.1 Un point de départ : les plates-formes de FOAD.....	29
2.1.2.2 Une autre approche possible : les portails et plates-formes de travail coopératif	30
2.1.2.3 Des plates-formes de travail coopératif aux plates-formes de FOAD à base de composants.....	32
2.1.3 <i>Approche de la problématique des environnements-supports au LIUPPA</i>	33
2.1.3.1 Travaux sur les plates-formes de FOAD	34
2.1.3.2 Travaux utilisant des plates-formes pour le travail coopératif.....	34
2.1.3.3 Travaux sur les plates-formes de FOAD à base de composants	35
2.1.3.4 Bilan de ces travaux et orientations de recherche.....	35
2.2 MODELES ET LANGAGES POUR L'INGENIERIE DES SITUATIONS-PROBLEMES COOPERATIVES	38
2.2.1 <i>Place des modèles et méta-modèles en Ingénierie (des EIAH)</i>	38
2.2.1.1 Modèles descriptifs et modèles prescriptifs.....	38
2.2.1.2 Modèles et méta-modèles	39
2.2.1.3 L'exploitation des modèles.....	41

2.2.2	<i>Du scénario pédagogique aux modèles de description d'un EIAH</i>	42
2.2.2.1	Présentation de MISA	43
2.2.2.2	Présentation du standard IMS-LD	45
2.2.2.3	Apports des approches basées sur la scénarisation pédagogique.....	48
2.2.2.4	Limites des approches basées sur la scénarisation pédagogique	51
2.2.2.4.1	Problèmes opérationnels pouvant se poser au concepteur qui utilise la spécification IMS-LD	51
2.2.2.4.2	Problèmes plus généraux posés par les approches centrées sur la scénarisation pédagogique.....	53
2.2.3	<i>Orientations de recherche au LIUPPA pour la modélisation des situations-problèmes coopératives</i>	57
2.3	CONCLUSION	60
3	RESULTATS ET PERSPECTIVES.....	61
3.1	TRAVAUX SUR LES MODELES ET META-MODELES UTILES POUR DECRIRE DES SITUATIONS-PROBLEMES	61
3.1.1	<i>Langages et formalismes pour la modélisation des situations-problèmes</i>	62
3.1.1.1	Travaux autour du langage « Spec »	62
3.1.1.2	Travaux visant à définir des modèles compréhensibles par des non informaticiens	65
3.1.1.3	Directions de recherche résultant de ces travaux.....	68
3.1.2	<i>Un méta-modèle et un profil UML pour décrire des situations-problèmes coopératives</i>	69
3.1.2.1	La situation Smash	69
3.1.2.2	Les éléments du modèle conceptuel.....	71
3.1.2.3	Du modèle conceptuel au profil CPM	73
3.1.2.3.1	La syntaxe abstraite.....	74
3.1.2.3.2	La syntaxe concrète.....	77
3.1.2.4	Examen de quelques modèles produits avec le profil CPM	78
3.1.2.4.1	Modèle d'information associable à une activité	78
3.1.2.4.2	Modèle d'information associable à une activité collaborative CPM	79
3.1.2.4.3	Structuration et dynamique des scénarios	80
3.1.2.5	Outillage du profil et bilan des expérimentations menées.....	82
3.1.2.6	La vérification et la validation des modèles produits	84
3.2	TRAVAUX EN INGENIERIE DIRIGEE PAR LES MODELES	85
3.2.1	<i>Des modèles de situations d'apprentissage aux modèles de processus</i>	85
3.2.1.1	La formalisation de chaque étape du processus de conception.....	86
3.2.1.2	Différents processus possibles	87
3.2.1.3	Le processus de conception des situations-problèmes vu par les enseignants.....	88
3.2.1.4	L'environnement d'aide à la spécification des situations-problèmes proposé aux concepteurs	90
3.2.1.5	La prise en compte de l'environnement d'exploitation des situations-problèmes coopératives.....	91
3.2.2	<i>Le problème de l'exploitation des modèles de conception</i>	93
3.2.2.1	Transformation des modèles CPM en représentations IMS-LD.....	94
3.2.2.2	Du profil CPM aux modèles de composants éducatifs	95
3.3	PERSPECTIVES DE RECHERCHE	98
3.3.1	<i>Bilan et perspectives immédiates de ces travaux</i>	98
3.3.2	<i>Projet scientifique : modèles et outils pour la co-conception des EIAH</i>	102
3.3.2.1	Problématique et pistes d'investigation.....	102
3.3.2.2	Propositions d'actions de recherche à court terme	105
3.4	CONCLUSION	106
4	BILAN DE MON ACTIVITE D'ENSEIGNANT-CHERCHEUR A L'UPPA	107
	CURRICULUM VITAE.....	111
	REFERENCES BIBLIOGRAPHIQUES.....	121
	ANNEXE A : TABLEAUX SYNTHETIQUES ANALYSANT LE META-MODELE IMS-LD DU POINT DE VUE DE LA CONCEPTION DES SITUATIONS D'APPRENTISSAGE.....	137
	ANNEXE B : EXTRAITS DES DOCUMENTS RELATIFS A LA SITUATION-PROBLEME SMASH	143
	ANNEXE C : LES PAQUETAGES DU META-MODELE CPM.....	153
	ANNEXE D : EXTRAITS DU META-MODELE D'UML UTILES POUR LA COMPREHENSION DU PROFIL CPM	157
	ANNEXE E : EXTRAITS DES MODELES REALISES AVEC LE PROFIL CPM POUR LA SITUATION SMASH.....	161

Glossaire des termes utilisés et définitions

- ADL :** Advanced Distributed Learning
L'initiative ADL vise à exploiter au mieux la puissance des Technologies de l'Information au service de l'apprentissage et de la formation.
- AFNOR :** Association Française de Normalisation
- AGL :** Atelier de Génie Logiciel
- Capacité :**
Activité intellectuelle stabilisée et reproductible dans des champs divers de connaissance ; terme utilisé souvent comme synonyme de « savoir faire ». Aucune capacité n'existe « à l'état pur » et toute capacité ne se manifeste qu'à travers la mise en œuvre de contenus (Meirieu 1994).
- CASE :** Computer Aided Software Engineering
- CNRS :** Centre National de Recherche Scientifique
- Compétence :**
Savoir identifié mettant en jeu une ou des capacités dans un champ notionnel ou disciplinaire déterminé. Plus précisément, on peut nommer compétence la capacité d'associer une classe de problèmes précisément identifiée avec un programme de traitement déterminé (Meirieu 1994).
- Conflit socio-cognitif :**
Interaction cognitive entre des sujets ayant des points de vue différents. Pour que l'interaction ait réellement lieu, il convient que chaque sujet prenne en compte le point de vue d'autrui et intériorise le conflit socio cognitif. Il y a alors conflit de centrations, contradiction et, si elle est surmontée, progression intellectuelle. On observe que de nombreuses situations de communication ne sont pas interactives dans la mesure où les sujets y abandonnent leur représentation ou l'imposent à autrui. La mise en groupe d'apprentissage constitue un dispositif où le mode de fonctionnement sollicite une véritable interaction (Meirieu 1994).
- Consigne-but :**
Définition d'un projet à réaliser dans une situation didactique en termes de « produit fini » et renvoyant essentiellement au registre des motivations des apprenants.
- Consigne-critère :**
Consigne permettant au sujet de déterminer si la tâche (ou le produit) qu'il doit réaliser est conforme aux attentes que le formateur manifeste à son égard.
- Consigne-procédure :**
Consigne proposant des stratégies différenciées permettant d'effectuer, selon des itinéraires multiples, l'opération mentale requise par le dispositif didactique mis en place.
-

Consigne-structure :

Consigne ayant un caractère obligatoire pour tous les apprenants dans un dispositif didactique déterminé ; pour le formateur, elles incarnent la structure de l'opération mentale à effectuer ; pour le formé, elles décrivent le mode de fonctionnement du dispositif.

Contrainte UML :

Ce procédé d'extension est utilisé pour ajouter une nouvelle règle d'utilisation à un élément UML (ou modifier une contrainte existante déjà). Les contraintes peuvent s'exprimer en langage naturel ou encore avec le langage OCL.

CPM : Cooperative Problem-based learning Metamodel

Critère de réussite formel :

Éléments permettant au sujet de vérifier qu'il a bien réalisé la tâche proposée et que le produit de son activité est conforme à ce qu'il devait obtenir. La liste de ces critères peut être établie avec les apprenants comme le proposent les théoriciens de l'évaluation formatrice ; elle est parfois nommée alors « carte d'étude » (Meirieu 1994).

CSCL : Computer-Supported Cooperative Learning

CSCW : Computer-Supported Cooperative Work

Décontextualisation :

Opération par laquelle un sujet utilise une acquisition dans un autre contexte que celui qui en a permis l'apprentissage. Elle est ainsi la première phase de l'identification d'un acquis, qui doit se prolonger par la mentalisation. Il faut préférer la notion de décontextualisation à celle de transfert, dans la mesure où cette dernière suppose une antériorité de l'acquisition, alors que c'est au terme du processus de décontextualisation et mentalisation que l'acquisition est véritablement stabilisée (Meirieu 1994).

Différenciation :

Différencier, c'est organiser les interactions et les activités de sorte que chaque élève soit constamment ou du moins très souvent confronté aux situations didactiques les plus fécondes pour lui (Perrenoud 1997).

EIAH : Environnement Informatique pour l'Apprentissage Humain

EJB : Enterprise Java Bean

EML : Educational Modeling Language

Enigme :

Savoir entrevu qui suscite le désir de son dévoilement. L'énigme naît ainsi de ce que l'apprenant sait déjà et dont le formateur sait montrer le caractère partiel, ambigu, voire mystérieux. Le désir de savoir peut ainsi émerger face à une situation problème si celle-ci est construite à partir d'une évaluation diagnostique des compétences et capacités d'un sujet. Le déjà-là problématisé offre la possibilité de son dépassement (Meirieu 1994).

FOAD : Formation Ouverte et A Distance

IDEE : Interaction – Document Electronique – Education (équipe du laboratoire LIUPPA)

IDM : Ingénierie Dirigée par les Modèles

IMS :

Le consortium IMS propose des spécifications techniques ayant pour finalité de favoriser l'interopérabilité des technologies éducatives.

J2EE : Java2 Enterprise Edition

LIUPPA : laboratoire d'Informatique de l'Université de Pau et des pays de l'Adour

MDA : Model-Driven Architecture

MDE : Model-Driven Engineering

Méta-modèle :

Un méta-modèle est un modèle d'un langage de modélisation (OMG 2002b).

MISA : Méthode d'Ingénierie de Systèmes d'Apprentissage

Cette méthode a été conçue et mise en œuvre au laboratoire LICEF (Québec).

Modèle :

Un modèle est une description de tout ou partie d'un système à partir d'un langage clairement défini (Kleppe, Warmer et al. 2003).

MOF : Meta-Object Facility. Voir (OMG 2002b)

MOT : Modélisation par Objets Typés

Langage développé au laboratoire LICEF (Québec) permettant de produire des représentations graphiques servant à exprimer sous forme de modèles, les connaissances de divers champs de savoir. Ces modèles rendent compte de façon dynamique un ensemble de connaissances, pouvant être nombreuses et complexes d'un domaine particulier et des liens qui existent entre ces connaissances.

Notion :

A un certain niveau, une notion est explicative d'un ensemble de faits. Pour être assimilée, une notion doit s'ancrer sur un ensemble de faits dont elle cherche la cohérence. Ainsi, ce sont deux opérations d'abstraction étroitement liées. On ne peut pas enseigner de notions (exemple : la respiration) indépendamment de faits (exemple : le poisson respire par les branchies, l'homme par les poumons). Et un fait ne prend de sens que par rapport à la notion qui l'englobe (Develay 1993).

Objectif-obstacle :

Objectif dont l'acquisition permet au sujet de franchir un palier décisif de progression en modifiant son système de représentations et en le faisant accéder à un registre supérieur de formulation.

Obstacle :

Difficulté émergeant dans la réalisation d'une tâche et qui permet d'engager un apprentissage pour la surmonter. Mais l'émergence de l'obstacle, si elle permet de finaliser la poursuite d'un objectif d'apprentissage, n'engage pas le sujet de manière automatique vers celui-ci : le sujet peut tenter de ne pas apprendre, en contournant l'obstacle, ou bien renoncer à poursuivre la tâche (Meirieu 1994).

OCL : Object Constraint Language

Pratique sociale de référence :

Activités par rapport auxquelles un apprentissage prend du sens pour un sujet apprenant. Ces activités peuvent être extra scolaires (ainsi l'arpentage peut-il être une pratique sociale de référence pour la géométrie), ou intra scolaires (ainsi un journal scolaire peut-il être une pratique sociale de référence pour l'acquisition de l'orthographe, un élevage de cobayes dans la classe, le devenir pour l'acquisition de notions biologiques, une situation problème mathématique pour l'acquisition de concepts, ...). La notion de pratique sociale de référence permet de comprendre en quoi un savoir scolaire ne peut pas simplement être compris comme transposition d'un savoir savant (Meirieu 1994).

Profil pédagogique :

Mode de représentation de l'activité cognitive des sujets.

Profil UML :

Un profil UML est un paquetage qui contient des éléments de modélisation qui ont été adaptés pour un domaine spécifique en étendant le méta-modèle grâce à des stéréotypes, des valeurs marquées et des contraintes (OMG 2003).

Programme de traitement :

Ensemble d'opérations mentales appliqué à une classe de problèmes déterminée et permettant d'obtenir la solution de chacun d'eux. Le théorème de Pythagore est ainsi un programme de traitement permettant de résoudre les problèmes appartenant à une même classe se caractérisant par quatre indicateurs de structure : 1) l'existence d'une situation géométrique ou d'un possible codage géométrique d'une situation, 2) l'existence d'un angle droit, 3) la possibilité de construire un triangle rectangle, 4) la possibilité de connaître la mesure de deux des trois côtés de ce triangle... (Meirieu 1994).

Registre de formulation :

Système de représentations d'un phénomène situé à un certain niveau d'abstraction ou de modélisation. Le passage d'un registre de formulation à un autre s'effectue par l'identification et l'acquisition de l'objectif obstacle (Meirieu 1994).

Représentation :

Dans le domaine de l'apprentissage, désigne la conception que le sujet a, à un moment donné, d'un objet ou d'un phénomène. Si l'on retient l'hypothèse piagétienne qui fait de l'accès à l'abstraction le vecteur central de la construction de l'intelligence, on peut considérer que l'apprentissage consiste à passer d'une représentation de type métaphorique à une représentation de plus en plus conceptualisée. Par ailleurs, les représentations qu'un sujet se fait, à un moment donné, de plusieurs types de « réalités », appartenant même à des disciplines différentes, sont vraisemblablement articulées autour de principes explicatifs communs ou paradigmes (Meirieu 1994).

RTP39 : Réseau Thématique Pluridisciplinaire centré sur les EIAH

SDV : Sciences de la Vie

SHS : Sciences Humaines et Sociales

Situation d'apprentissage :

Situation (ensemble de dispositifs) dans laquelle un sujet s'approprie de l'information à partir du projet qu'il conçoit. Il s'appuie, pour ce faire, sur des capacités et des compétences déjà maîtrisées qui lui permettent d'en acquérir de nouvelles. Les situations d'apprentissage peuvent ainsi apparaître en dehors de toute structure scolaire et de toute programmation didactique. Contrairement aux situations d'enseignement, il n'y a ici aucune action du pédagogue (Meirieu 1994).

Situation didactique :

Situation d'apprentissage élaborée par le didacticien qui fournit, d'une part, des matériaux permettant de recueillir l'information et, d'autre part, une consigne but permettant de mettre le sujet en situation de projet. Une évaluation diagnostique dans le champ socio affectif permet de s'assurer que la consigne but est effectivement susceptible de mobiliser le sujet. Une évaluation diagnostique dans le champ cognitif permet de s'assurer que le sujet dispose bien des capacités et compétences lui permettant de traiter l'information. La situation ainsi didactisée permet de faire échapper l'apprentissage à l'aléatoire de rencontres et concordances fortuites (Meirieu 1994).

Situation-problème :

Situation didactique dans laquelle il est proposé au sujet une tâche qu'il ne peut mener à bien sans effectuer un apprentissage précis. Cet apprentissage qui constitue le véritable objectif de la situation problème, s'effectue en levant l'obstacle à la réalisation de la tâche. Ainsi la production impose l'acquisition, l'une et l'autre devant faire l'objet d'évaluations distinctes. Comme toute situation didactique, la situation problème doit être construite en s'appuyant sur une triple évaluation diagnostique (des motivations, des compétences et des capacités).

Stéréotype UML :

Appliquer un stéréotype consiste à étiqueter un élément afin de signaler son appartenance à un groupe d'éléments possédant une sémantique particulière. Le stéréotype est rattaché à un méta-élément du méta-modèle qu'il étend.

STIC : Sciences et Technologies de l'Information et de la Communication

UML : Unified Modeling Language

Valeur marquée (Tagged Value) :

Son utilisation consiste à associer une ou plusieurs propriétés supplémentaires à un méta-élément. Une valeur marquée peut s'appliquer aux stéréotypes UML et est représentée sous forme (étiquette, valeur)

Introduction

L'objet de ce document est de retracer mon parcours d'enseignant chercheur depuis ma thèse de doctorat à l'Université Paul Sabatier de Toulouse soutenue en novembre 1992, jusqu'à la fin de l'année universitaire 2004-2005. Une dizaine d'années au cours desquelles j'ai bénéficié d'une grande liberté d'entreprendre au sein de l'IUT de Bayonne ainsi que d'un contexte de recherche rendu favorable par les efforts de structuration du Laboratoire d'Informatique de l'Université de Pau et des Pays de l'Adour (LIUPPA).

Comme l'annonce le titre de ce document : « Contribution à l'Ingénierie dirigée par les Modèles en EIAH : le cas des situations-problèmes coopératives », mes travaux trouvent leurs fondements dans l'ingénierie des situations d'apprentissage et visent à élaborer / valider des modèles de représentation de ces situations pour accroître la qualité de leur conception. La conception est vue comme un processus qui va de l'éllicitation des besoins pédagogiques jusqu'à l'élaboration de modèles permettant d'entreprendre l'implémentation d'un environnement support aux apprentissages de type EIAH. Il s'agit donc de caractériser un environnement informatique support pour la situation-problème coopérative proposée dans lequel les apprenants vont être amenés à réaliser des activités propices à l'apprentissage de concepts et compétences précis. Les environnements ciblés sont donc plus des systèmes support à la performance que des systèmes plus classiques de FOAD qui seraient basés sur une simple exposition de supports de cours.

L'étape de conception est une étape difficile de l'ingénierie d'un EIAH, et ce pour plusieurs raisons :

- C'est lors de la conception que se nouent ou se dénouent souvent les fils du travail pluridisciplinaire que l'équipe de développement devrait mener : l'analyse de la connaissance (domaine, apprenant), l'analyse de l'interaction, l'articulation entre connaissance et interaction sont autant de dimensions devant faire l'objet d'échanges, de négociation entre les acteurs de l'équipe de développement.
- Les spécifications d'un EIAH sont souvent évolutives, longues à stabiliser. Explicitation des théories et travail expérimental doivent être menés de front pour stabiliser les caractéristiques de l'EIAH à développer, et ceci rend encore plus difficile le travail pluridisciplinaire de conception.

Aussi, l'étape de conception est souvent une étape inaboutie, de nombreux éléments restent implicites, ce qui a des conséquences importantes sur le reste du processus de développement d'un EIAH : les échanges entre acteurs de l'équipe pluridisciplinaire ont tendance à porter non pas sur les modèles de conception mais plutôt sur l'interprétation de ces modèles et surtout sur l'évaluation des applicatifs produits par les informaticiens. Faute d'avoir pu échanger sur des modèles explicites de conception, l'équipe échange (et se heurte souvent) sur la base du code exécutable produit qui doit souvent être remis en cause du fait du caractère évolutif des spécifications en EIAH.

Les travaux présentés dans ce mémoire ont donc pour objectif :

- d'améliorer la qualité des modèles de conception. Dans différents domaines tels que la gestion de production, l'informatique industrielle, ..., des indicateurs comme le coût, les délais, l'adéquation aux besoins sont des indicateurs pertinents. Le domaine des EIAH n'est probablement pas suffisamment mûre pour justifier actuellement l'utilisation de tels critères d'évaluation des solutions proposées. En revanche, si l'équipe de concepteurs est réellement pluridisciplinaire, le critère d'acceptabilité de l'EIAH par cette équipe me paraît pouvoir s'appliquer dès la conception, les critères d'acceptabilité et d'utilisabilité par les apprenants se posant après implémentation de l'EIAH (Tricot, Plébat-Soutjis et al. 2003). Pour l'équipe de concepteurs, des modèles acceptables d'un EIAH sont des modèles dont il est possible d'évaluer la pertinence (Howard 1994), (Denois 1997). Ceci signifie que les modèles sont explicites, non ambigus et qu'ils rendent donc possible des échanges entre les acteurs de la conception pour faire évoluer des modèles d'un état inacceptable à un état acceptable (cf techniques de prototypage par exemple).
- de capitaliser sur la base des modèles de conception plutôt que sur le code des EIAH produits. L'explicitation des modèles de conception (théories d'apprentissage sur lesquelles s'appuie l'EIAH à construire, scénarios d'apprentissage et principes d'interaction) doit permettre de mieux partager les expertises au sein des équipes de développement et donc de diffuser les savoirs et bonnes pratiques. Par ailleurs, il s'agit d'examiner les conditions de transformations des modèles de conception en modèles computationnels susceptibles de faciliter l'activité d'implémentation des EIAH.

Pour atteindre ces objectifs, je me suis fixé un certain nombre d'hypothèses de travail qui m'ont permis d'une part de cerner le champ des actions de recherche à mener et d'autre part d'organiser ces actions :

- La première hypothèse porte sur le public-cible de tels travaux et sur l'usage fait des modèles de conception. L'idée est de pouvoir intégrer les enseignants de terrain dans le processus de conception, le langage d'expression des modélisations ne devant donc pas être compréhensible par les seuls informaticiens de la communauté EIAH. Un autre objectif consiste à ne pas cibler des EIAH implémentés de manière ad hoc mais d'examiner comment les modèles de conception proposés vont pouvoir se tisser (par ajustement mutuel) avec les caractéristiques de plates-formes de formation à distance. Pour de telles plates-formes, l'enjeu est d'opérationnaliser les modèles de conception produits à la fois côté Front-Office (apprenants et tuteurs) et côté Back-Office (enseignants, spécialistes du domaine, concepteurs, ...). Nos travaux doivent donc tenir compte des travaux de standardisation très importants qui sont menés au niveau international pour favoriser l'interopérabilité des plates-formes et des ressources pédagogiques : il faut tout à la fois s'inscrire dans un tel processus de généralisation de la formation à distance sans renier les repères culturels des enseignants en terme d'enjeux de formation et d'apprentissage.
- La deuxième hypothèse porte donc sur les EIAH ciblés et les théories de référence sur lesquelles bâtir de tels EIAH. Il me semble important, si l'on veut intégrer les enseignants de terrain dans le processus de conception, de commencer par des modèles proches des séquences d'enseignement qu'ils ont l'habitude de mettre en place dans leurs classes. Ceci explique que nos travaux s'appuient sur une théorie particulière, la théorie empirique de l'apprentissage qui sert de référence (Develay 1993), (Meirieu 1994) dans les Instituts Universitaires de Formation des Maîtres (IUFM) ; il s'agit d'une théorie systémique -proche des travaux de Brousseau (Brousseau 1998)- qui emprunte à différents courants au service du didactique : la notion de situation-problème (donc de conflit socio-cognitif) est centrale dans cette approche largement utilisée par les professeurs des écoles dans leurs classes.
- La troisième hypothèse porte sur la façon d'intégrer les utilisateurs (enseignants-chercheurs dans le domaine des Sciences Humaines et Sociales ou SHS, spécialistes du domaine à enseigner, enseignants, ...) dans la boucle d'ingénierie des EIAH. L'orientation proposée consiste à s'appuyer sur les langages, modèles et outils du Génie

Logiciel pour faire avancer la connaissance dans le domaine des EIAH. Il ne s'agit pas de conduire une recherche en Génie Logiciel mais d'examiner en quoi des principes de Génie Logiciel comme l'ingénierie dirigée par les modèles fournissent les bases théoriques et l'outillage pour favoriser l'interdisciplinarité nécessaire à la conception des EIAH (ce qui passe par l'explicitation des théories cognitives et des décisions de conception).

Les propositions et résultats proposés dans ce mémoire sont le fruit des travaux menés dans le cadre de deux thèses soutenues à l'UPPA entre 1998 et 2004. L'encadrement de la thèse de C. Marquesuzaà m'a permis d'approfondir le problème de l'ingénierie des connaissances éducatives. La recherche avait pour objectif de faciliter la tâche de l'enseignant dans la préparation de ses séquences pédagogiques. Le travail de doctorat a permis de définir un environnement informatique meta-CASE pour la spécification des connaissances nécessaires à la formalisation de situations-problèmes : l'accent a été mis sur l'adaptation des éléments du langage aux besoins de modélisation du pédagogue face à une situation-problème qu'il veut décrire.

L'encadrement du doctorat de P. Laforcade a permis de définir un profil UML pour la conception de situations d'apprentissage à base de situations-problèmes coopératives. Ici, le problème de la modélisation a été étendu à la prise en compte des activités d'apprentissage coopératives afin de couvrir le spectre des méthodes pédagogiques dites actives : apprentissage par résolution de problèmes, apprentissage coopératif et pédagogie par le projet. Fortement marqué par les avancées en Génie Logiciel, ce travail de recherche vise :

- une conception de situations d'apprentissage dirigée par des modèles afin de pouvoir réutiliser / transformer ces modèles vers des modèles exploitables par des plates-formes de formation à distance et faciliter ainsi le travail de mise en œuvre,
- la mise en œuvre d'EIAH par agrégation / spécialisation de composants logiciels pour respecter le scénario pédagogique et les activités d'apprentissage proposées par la situation-problème coopérative à mettre en œuvre.

Dans ce document, je positionnerai tout d'abord les travaux proposés au sein de la communauté EIAH (chapitre 1) ce qui me permettra d'explicitier les théories d'apprentissage auxquelles ces travaux se réfèrent ainsi que les principaux verrous scientifiques posés par l'ingénierie des EIAH. Le chapitre 2 présentera ma démarche de recherche au cours de ces travaux ainsi que les questions de recherche spécifiques qui sont au cœur des activités menées. Ce chapitre me permettra donc de positionner mes travaux par rapport à ceux que mènent d'autres équipes nationales et internationales sur des problématiques identiques. Je présenterai ensuite (chapitre 3) les principaux résultats et perspectives qui portent sur l'ingénierie de modèles de conception pour les situations d'apprentissage de type situations-problèmes coopératives, mais aussi sur l'usage de ces modèles et la démarche de conception qu'ils supposent. Le chapitre 4 me permettra enfin de dresser un bilan de mon activité d'enseignant chercheur à l'Université de Pau et des Pays de l'Adour.

Chapitre 1

Positionnement des travaux

Ce chapitre a pour ambition de situer les activités de recherche que j'ai menées depuis plus de dix ans ; la communauté EIAH est par construction très diverse puisqu'elle regroupe des chercheurs issus aussi bien des STIC¹ (informaticiens, ergonomes) que des SHS (didacticiens, pédagogues) ou des SDV (psychologues). Même au sein de la communauté des chercheurs en EIAH se réclamant des STIC, l'éventail des centres d'intérêt est très large car les STIC (et plus particulièrement l'Informatique) doivent apporter des réponses à des questions de recherche portant tout autant sur la conception, la mise en œuvre que l'évaluation des EIAH.

Caractériser un EIAH (ou une classe d'EIAH) sans rentrer dans des détails inutiles, exprimer en quoi un développement forcément spécifique peut participer à l'ingénierie de cette classe d'EIAH constituent autant de points-clé pour caractériser un travail de recherche dans ce domaine. Dans les prochains paragraphes, je m'attacherai à faire ce travail d'introspection sur ma propre activité de recherche. Je m'appuierai pour cela sur les éléments de la démarche proposée dans Platon-1 (Tchounikine, Baker et al. 2004), synthèse de travaux menés dans le cadre de l'Action Spécifique « Fondements théoriques et méthodologiques de la conception des EIAH » du département STIC du CNRS.

Dans un premier paragraphe, j'essaierai de caractériser le type d'EIAH qui a motivé les actions de recherche que j'ai pu mener (cf paragraphe 1.1 et suivants assimilables à la dimension A de Platon-1 : définition du projet de recherche). Puis je décrirai sur quel cadre théorique en Education reposent ces actions de recherche² (cf paragraphe 1.2 et suivants se référant à la dimension B de Platon-1 : Cadre théorique de la recherche). J'aborderai en fin de chapitre des questions de recherche difficiles que pose l'ingénierie de tels systèmes à la communauté des chercheurs en EIAH.

1.1 Le projet : Produire des EIAH mettant en œuvre des méthodes pédagogiques actives et situées

Le terme EIAH est maintenant clairement défini bien qu'il recouvre une multiplicité de systèmes. Par Environnement Informatique pour l'Apprentissage Humain (EIAH), on entend « un environnement conçu dans le but de favoriser l'apprentissage humain, c'est-à-dire la construction de connaissances chez un apprenant. Ce type d'environnement mobilise des agents humains (élève, apprenant, tuteur) et artificiels (agents informatiques, qui peuvent

¹ Voir la signification de ces acronymes dans le glossaire au début de ce manuscrit.

² L'examen des dimensions d'analyse A, B et D sera poursuivi au chapitre 2 consacré à la problématique et à la démarche de ma recherche. Le chapitre 3 sera pour sa part consacré à la dimension C : résultat de la recherche.

eux aussi tenir des rôles différents) et leur offre des situations d'interaction, localement ou à travers des réseaux informatiques, ainsi que des conditions d'accès à des ressources formatives (humaines et/ou médiatisées) » (Tchounikine, Baker et al. 2004).

L'EIAH, en tant qu'artefact informatique supportant des activités d'apprentissage humain, peut donc prendre différentes formes en fonction :

- o des orientations pédagogiques de départ,
- o du rôle dévolu à chacun des agents (humains et artificiels) considérés,
- o de la manière dont l'EIAH mobilise ces agents,
- o des activités et outils proposés aux agents et des conditions d'accès aux ressources que ces activités supposent (ou imposent).

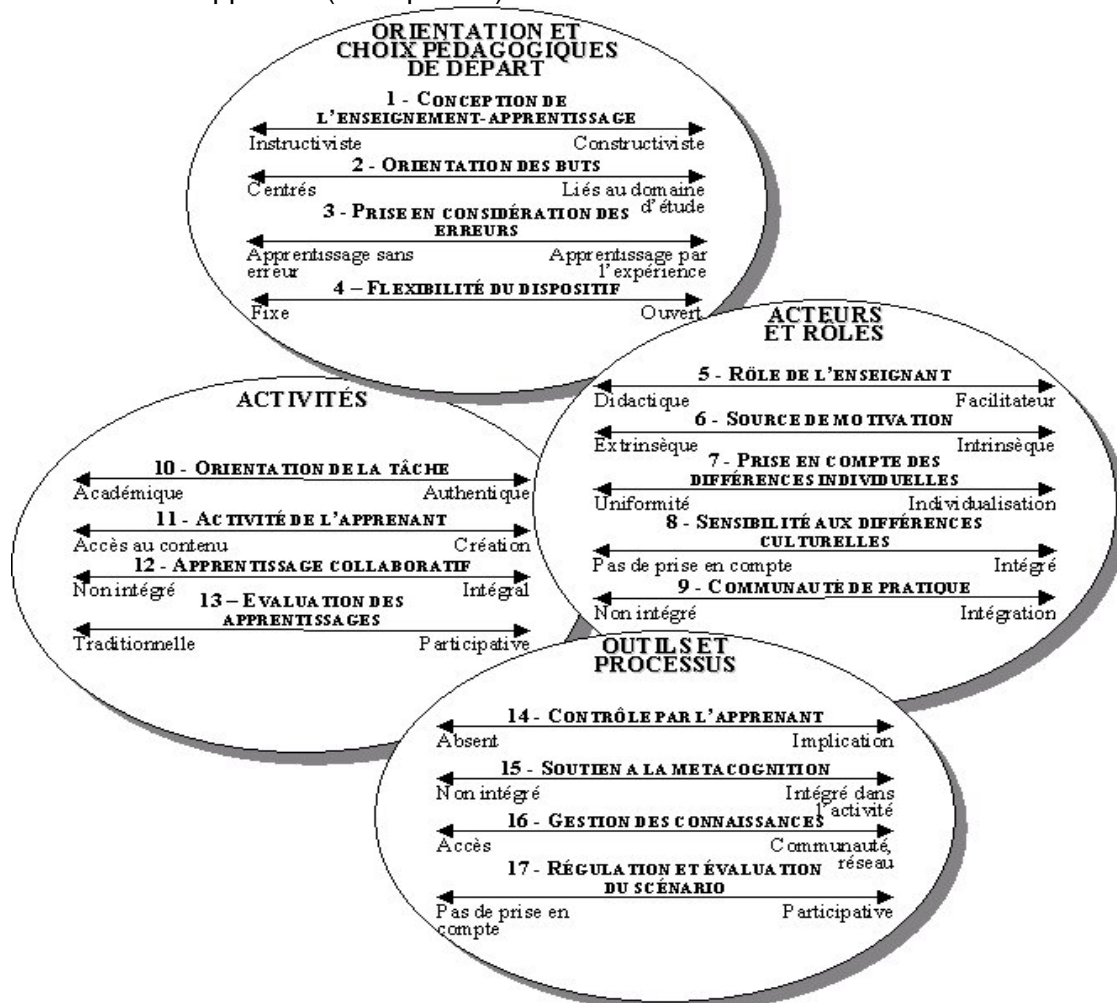


Figure 1.1 : Typologie des degrés de liberté d'un EIAH (Brassard et Daele 2003)

Dans ce document, je ne propose pas ma propre typologie des EIAH en fonction des degrés de liberté de la figure précédente. De nombreux auteurs ont déjà mené ce type de travail sous l'angle historique et/ou sous l'angle des théories d'apprentissage sous-jacentes (Wenger 1987), (Bruillard 1997), (Schneider 2003). L'idée est plutôt de caractériser le type d'EIAH que visent les divers travaux que j'ai pu engager jusqu'ici.

1.1.1 Des EIAH proposant une utilisation effective des technologies

Sans vouloir passer pour un affreux technologue, j'ai choisi ce premier point d'entrée pour étudier les caractéristiques des EIAH qui me semblent pouvoir être mis au service des apprenants et des enseignants. Le paragraphe 1.2.1 montrera que cette technologie doit et peut se mettre au service d'approches pédagogiques qui collent aux aspirations et pratiques de la communauté enseignante.

Il est clair que plusieurs auteurs en Sciences de l'Education ont une vision négative des EIAH. Dans son livre consacré aux théories contemporaines de l'Education, Yves Bertrand dresse en 1993 un bilan peu flatteur et conclut son chapitre consacré au « salut par la technologie » en renvoyant dos à dos les systèmes relevant d'une approche systémique de l'Education et ceux relevant plutôt d'une approche hypermédiatique (Bertrand 1993). Pourtant, comme le souligne Daniel Schneider, « les nouvelles technologies de l'Information et de la communication semblent avoir un potentiel pour la quasi-totalité des fonctions d'un système d'enseignement » (Schneider 2003). Un examen rapide des capacités pédagogiques de quelques EIAH récents suffit pour s'en convaincre³.

1.1.1.1 Exemples d'utilisation réussie de la technologie au service de l'apprentissage humain

En France, trois EIAH me paraissent révélateurs d'un transfert technologique réussi entre la recherche en EIAH et les établissements d'enseignement : Aplusix (Nicaud 1987), Cabri-Géomètre (Baulac 1990) et Roboteach (Leroux 1996; Leroux 2002). L'une des caractéristiques de ces EIAH est qu'ils n'ont pas été imaginés à partir de considérations technologiques mais dans le cadre d'une réflexion globale et expérimentale mêlant considérations pédagogiques et possibilités informatiques. Pour ne parler que de Cabri Géomètre, le travail de recherche mené par les équipes du CNRS et de l'Université Joseph Fournier de Grenoble a permis d'allier transfert technologique réussi (3 millions de produits vendus) et un continuum de recherche sur près de 15 ans. Au départ, Cabri-géomètre (Baulac 1990), (Laborde et Capponi 1991) est un micromonde qui permet, à partir de primitives d'assemblage ou de transformations d'objets géométriques élémentaires, la construction de figures géométriques complexes et leurs manipulations directes pour l'observation de leurs propriétés. Sur ce noyau se sont greffés de nombreux modules et nouveaux développements imaginés et mis au point dans les laboratoires de recherche :

- le tuteur intelligent Mentoniez (Py 1996) qui guide l'élève durant les deux phases de la résolution du problème: le tracé de la figure et l'élaboration de la démonstration,
- la plate-forme Baghera (Pesty, Webber et al. 2003) qui est un environnement de formation à distance dans le domaine de la preuve en géométrie. La plate-forme vise à créer les conditions d'un véritable partenariat entre agents humains (élèves et enseignants) et artificiels (agents assistants, médiateurs et tuteurs) répartis sur un réseau, pour parvenir à susciter, par leurs interactions, l'apprentissage de la construction d'une démonstration par l'élève.
- ...

Au niveau international, les équipes de K. Koedinger et J. Anderson à l'Université Carnegie Mellon de Pittsburgh ont imaginé et mis au point toute une série de tuteurs intelligents dans le domaine de l'aide à la résolution de problèmes mathématiques. Ces tuteurs tels Geometry Tutor et Algebra Tutor sont basés sur la théorie ACT-R d'acquisition de compétences cognitives (Anderson, Corbett et al. 1995) ; les capacités pédagogiques de ces tuteurs vont de l'aide à la résolution de problèmes à l'acquisition de compétences métacognitives (Aleven et Koedinger 2000). Un système comme Geometry Tutor a ainsi fait l'objet de près de 10 ans de développements et évaluations scientifiques (Aleven et Koedinger 2001); il est maintenant utilisé dans près de 350 collèges américains pour l'enseignement de la géométrie.

Ecolab (Luckin et du Boulay 1999) est un autre exemple d'EIAH dont les principes de conception sont fondés sur une théorie d'apprentissage, ici la théorie de Vygotsky relative à la zone proximale de développement. Pour l'enfant (10-11 ans), Ecolab apparaît comme un laboratoire d'écologie permettant de comprendre la notion de chaîne alimentaire à travers l'étude des relations entre organismes vivants. Dans sa version la plus évoluée (VIS), le

³ La liste des EIAH récents présentés ici ne se veut pas exhaustive ; parmi les critères de choix, j'ai privilégié les EIAH ayant fait l'objet d'un continuum de recherches sur de nombreuses années ainsi que ceux qui ont conduit à une diffusion de produits aboutis dans les établissements scolaires.

système construit au fur et à mesure des interactions un modèle de l'apprenant (sa Zone Proximale de Développement ou ZPD) que le système exploite pour déterminer l'activité la plus adaptée à proposer ensuite à l'apprenant. Là encore, plusieurs versions de cet EIAH existent et font l'objet d'évaluations scientifiques menées par le laboratoire IDEAS de l'Université de Sussex.

Le système AutoTutor (Graesser, VanLehn et al. 2001) est un tuteur intelligent (le domaine concerne l'apprentissage des principes de base des réseaux informatiques et d'Internet) capable de dialoguer avec un apprenant en langage naturel. Le système qui a été évalué dans 200 collèves américains est capable de comprendre les contributions en langage naturel d'un apprenant, de répondre à ses questions et surtout d'inciter cet apprenant à étayer son raisonnement pour faire apparaître ses savoirs et savoir-faire. Les expérimentations menées en collège ont clairement identifié les apports de ce tuteur intelligent par rapport à des formes plus classiques d'enseignement (Graesser, Moreno et al. 2003).

Le système Belvedere (Suthers 1999) a été conçu et mis au point à l'Université d'Hawaï à Manoa. Le système offre un espace partagé pour permettre à des apprenants de construire des diagrammes de croyance ou « inquiry diagrams » reliant des faits constatés et des hypothèses des apprenants pour expliquer ces faits (représentations collectives). Le système permet ainsi à des apprenants de partager des représentations, d'échanger à propos de ces représentations et d'argumenter leurs hypothèses et critiques (Suthers 2003); le système les assiste dans leur recherche de représentations acceptables en leur suggérant une démarche scientifique, en mettant en évidence des hypothèses des apprenants non encore vérifiées par des faits, ... Le logiciel Belvedere a fait l'objet de nombreuses expérimentations en collège notamment dans le domaine de l'enseignement des sciences (cf expérimentation portant sur l'évolution des espèces, celle portant sur la disparition des dinosaures, ...)

L'équipe de recherche menée par Lewis Johnson au laboratoire CARTE (Center for Advanced Research in Technology for Education) sait maintenant mettre en œuvre des agents pédagogiques animés suffisamment évolués pour interagir de manière naturelle avec un apprenant (communication verbale et communication gestuelle) (Johnson 2001). Ces agents tels Steve, Adele, Pat, Carmen ont à la fois des capacités de tuteurs intelligents (aspects cognitifs) et des capacités d'interaction tenant compte de la motivation et du côté affectif des apprenants (Johnson et Rizzo 2004).

Les travaux de Ami Soller à l'Université de Pittsburgh (Soller 2002) ont permis de produire l'environnement COMET, un système dans lequel des équipes d'apprenants peuvent résoudre de manière collaborative des problèmes portant sur la conception orientée objet. L'intérêt de ce système est qu'il permet une analyse très fine des interactions entre groupes d'apprenants portant sur les informations partagées qui sont au cœur des exercices proposés. A cette fin, le système exploite principalement les ouvreurs de phrase (« sentence openers ») choisis par des groupes d'apprenants pour échanger avec les autres ; à l'aide de techniques d'analyse sophistiquées de ces ouvreurs de phrases, le système est capable (Soller et Lesgold 2003) :

1. de reconnaître les situations dans lesquelles les apprenants n'arrivent pas à appréhender des concepts,
2. d'expliquer pourquoi ils sont en difficulté.

Notons que des systèmes comme ICLS (McManus et Aiken 1995), C-CHENE (Baker et Lund 1997), TecfaMOO (Jermann et Schneider 1997) ou OSCAR (Delium 2003) reposent sur les mêmes principes et ont des objectifs comparables.

Les exemples précédents le montrent, les EIAH actuels essaient d'exploiter au mieux les avancées en matière de recherche et de technologie ; ces environnements apportent une vraie plus-value pour les apprenants comme l'ont montré les expérimentations menées pour chacun d'eux. Pourtant, malgré les capacités pédagogiques indéniables de ces environnements, les critiques de Y. Bertrand ne sont pas non plus totalement infondées :

- les EIAH présentés ne sont que la face émergée d'un iceberg de systèmes qui n'ont pas fait la preuve de leur validité pédagogique (ou dont on sait qu'ils vont à l'encontre de bien des théories d'apprentissage).
- Les EIAH présentés ont nécessité des années de recherche / développements et l'on voit bien que l'on est très loin de pouvoir capitaliser les avancées de ces quelques EIAH pour produire et diffuser à grande échelle des systèmes qui auraient les mêmes capacités.
- Compte tenu de la complexité technologique des EIAH présentés, les enseignants ne sont pas pour l'instant en mesure de participer à une généralisation de produits ayant des capacités comparables à ceux présentés.

Cependant, le prochain paragraphe montrera que les capacités actuelles des outils de type TIC peuvent permettre la mise au point d'environnements utiles en éducation sans pour autant nécessiter des développements qui décourageraient le plus entreprenant des enseignants.

1.1.1.2 Exemples d'exploitation des outils de type TIC au service de l'apprentissage et de l'enseignement

« L'effet principal des technologies dans le tandem enseignement-apprentissage se manifeste dans les environnements pédagogiques nouveaux plus proches de la manière par laquelle l'individu apprend » écrit (Lebrun 2002). Sans vouloir rentrer ici dans le problème des théories d'apprentissage, force est de constater que de nombreux auteurs sont d'accord pour considérer que « l'intégration des technologies éducatives dans les pratiques peut aller de pair avec une meilleure prise en compte par le système éducatif des approches pédagogiques innovantes » (SEC 1996). Les systèmes « orientés performance » (Tchounikine 2002a) qui visent à créer une situation d'apprentissage individuelle ou collective pour favoriser l'acquisition de compétences précises, constituent une cible à privilégier pour l'usage de ces nouvelles technologies. Dans ce cadre d'une approche constructiviste de l'apprentissage, (Wilson et Lowry 2000) assigne trois objectifs pour les outils technologiques (en particulier sur Internet) :

- Ils facilitent l'accès à des sources d'information riches et diversifiées,
- Ils facilitent les interactions entre les apprenants et ces contenus,
- Ils mettent en relation les apprenants et leur permettent de se questionner, de s'aider et de se répondre.

La littérature du domaine EIAH est riche en propositions d'outils TIC censés répondre à ces objectifs. Les portails collaboratifs proposés par plusieurs auteurs, notamment (Brusilovsky 2003) et (Schneider 2003) apparaissent comme étant les propositions les plus argumentées et les plus à même d'amener des enseignants de terrain à investir dans les technologies éducatives. C'est donc l'exemple des portails collaboratifs qui sera détaillé dans la suite de ce paragraphe.

Ces dernières années sont donc apparus des outils modulaires permettant de configurer des sites Web communautaires interactifs ; en dehors des plates-formes commerciales type WebCT ou Blackboard, on peut citer en France le « cartable électronique de l'Université de Savoie (cf <http://www.univ-savoie.fr/Portail/Help/>) bâti à partir de l'outil Zope (cf <http://www.zope.org/>), en Suisse le portail TecfaSeed (cf <http://tecfaseed.unige.ch/door/>) et bien sûr Explor@ développé au laboratoire LICEF (cf <http://www.licef.teluq.quebec.ca/francais/>); on peut également citer des plates-formes « OpenSource » pour la mise en œuvre de tels sites Web communautaires interactifs à vocation pédagogique : en Europe, les plates-formes ARIADNE (cf <http://www.ariadne-eu.org/>) et OpenUSS (cf <http://openuss.sourceforge.net/openuss/>) et aux USA, la plate-forme uPortal (cf <http://www.uportal.org/>). (Schneider 2003) utilise le terme C3MS (Community, Content and Collaboration Systems) pour nommer ces outils modulaires. La boîte à outils proposée permet de constituer assez facilement un portail, sorte de site Web qui intègre un certain nombre de ressources (« briques d'informations et de services »). Parmi les outils, citons les carnets de bord (mini-weblogs), les hypertextes collaboratifs (wikis), les forums, les chats, les calendriers, les outils de gestion de projets, les systèmes

de quiz, ... Bien que les C3MS ne couvrent pas tous les besoins en matière de support technologique, le potentiel actuel de ces systèmes est déjà intéressant, potentiel pouvant être largement étendu par la communauté des développeurs (aspects « OpenSource des C3MS ») (Brusilovsky et Nijhawan 2002). Un portail de ce type est donc en mesure de fournir un support pour une large variété de formes discursives, ce support étant placé au service du « document vivant » qui est un élément central pour l'enseignement et la vie des communautés virtuelles (Wenger 1998), (Schneider 2003) et (Wilson 2004a). La figure suivante synthétise les fonctions supportées par les C3MS.

Fonction	Module C3MS (outils du Portail)
Gestion de contenu	<ul style="list-style-type: none"> • Content Management Systems (CMS) • Système de nouvelles annotables (y compris organisation en rubriques) • Hypertextes collaboratifs (Wikis) • Albums d'images, • Gestionnaires de glossaires et autres outils spécialisés
Echange de connaissances	<ul style="list-style-type: none"> • Weblogs personnels (carnets de bord) • Syndication des nouvelles (entre portails)
Echange d'arguments	<ul style="list-style-type: none"> • Systèmes de partage de fichiers • (tous les outils de gestion de contenu)
Gestion et support de projets	<ul style="list-style-type: none"> • Forums et/ou nouvelles • Chats • Outils de gestion de projets • Outils "Virtual Teams" • Calendriers
Gestion de connaissances	<ul style="list-style-type: none"> • Gestionnaires de FAQ, de Liens web, etc. • Outils de recherche pour tous les contenus • Box "top 10", système de "rating" des commentaires • Box "quoi de neuf" (forums, liens, downloads, etc.) • Présence, profil et identification des membres
Gestion de la communauté	<ul style="list-style-type: none"> • Shoutbox (mini-chat intégré dans les marges du portail) • Systèmes de réputation • Trace d'activités des membres (contributions) • Calendrier d'événements

Figure 1.2 : Fonctions et outils d'un C3MS (Schneider 2003)

L'approche et les outils présentés dans ce paragraphe correspondent à un usage des nouvelles technologies totalement complémentaire aux EIAH présentés dans le paragraphe 1.1.1.1 :

- Les outils de type C3MS peuvent être mis dans les mains d'enseignants pour qu'ils bâtissent, de manière empirique, un environnement support à leurs activités ; au contraire, les EIAH présentés au paragraphe 1.1.1.1 sont le résultat d'un travail de recherche faisant la plupart du temps appel à des technologies très évoluées qui rend ensuite difficile toute appropriation de ces systèmes par un enseignant.
- Les outils de type C3MS sont des outils transversaux qui peuvent être mis à la disposition des apprenants via le portail. Une fois installés dans le portail, ces outils sont disponibles à tout moment quelle que soit l'activité dans laquelle est engagé l'apprenant. Les EIAH présentés au paragraphe 1.1.1.1 me semblent avoir pour ambition de mieux gérer le contexte d'apprentissage précis dans lequel se situe l'apprenant car les fonctionnalités des outils (de communication, de production, ...) offertes à l'apprenant évoluent en fonction de ce contexte (dynamiquement, l'EIAH peut brider des fonctionnalités, interdire ou renforcer l'usage d'un outil).

Ce tour d'horizon m'a permis de montrer les potentialités des outils pour l'Education qu'ils soient construits sur mesure (cf EIAH du paragraphe 1.1.1.1) ou qu'ils soient le fruit d'un agencement de services (cf portails du type C3MS). Il est cependant clair que les technologies ne sont qu'un moyen, que les bénéfices de l'utilisation des technologies ne doivent pas être attendus dans « la seule sphère cognitive réduite aux connaissances et aux savoirs à redire » (Lebrun 2002). « Just remember: in most cases, effective teaching with

technology is effective teaching by any means » peut-on lire dans (Grabe et Grabe 1996). Cette formule est sans doute un peu péremptoire mais elle oblige à considérer d'abord l'objet d'enseignement/d'apprentissage sur lequel porte l'effort technologique. Les prochains paragraphes me permettront de préciser le cadre dans lequel j'ai souhaité insérer les outils technologiques issus de mes travaux.

1.1.2 Des EIAH reposant sur des théories d'apprentissage et des stratégies pédagogiques plus complémentaires qu'en opposition

Les discussions autour des théories d'apprentissage sont souvent prétextes à incompréhension et opposition. Je ne souhaite pas ici prendre une position de principe mais constater que, dans leurs pratiques quotidiennes, les enseignants utilisent avec discernement le potentiel de chacune de ces théories. Les EIAH que je souhaite promouvoir ne doivent donc pas non plus reposer sur une théorie de référence qui serait universelle et à laquelle il faudrait forcément adhérer ; je crois plutôt au métissage et fais le pari que l'analyse des situations d'apprentissage et des apprenants peut conduire les créateurs d'EIAH à choisir les théories et stratégies pédagogiques adaptées. Dans les deux prochains paragraphes, je rappelle très succinctement ces théories en insistant sur l'intérêt des méthodes actives à partir desquelles des pédagogies plus traditionnelles et plus transmissives peuvent se mettre en place.

1.1.2.1 Vers une approche intégratrice des théories d'apprentissage que les enseignants utilisent

De très nombreuses écoles de pensée ont fortement influencé notre perception sur l'acte d'apprendre. Dans sa classification, Yves Bertrand dénombre sept écoles différentes à partir d'une analyse des quatre composantes des théories de l'Education : le sujet (apprenant), le contenu (matières, disciplines), la société (les autres, l'environnement), les interactions pédagogiques entre ces trois pôles (l'enseignant, les technologies de communication).

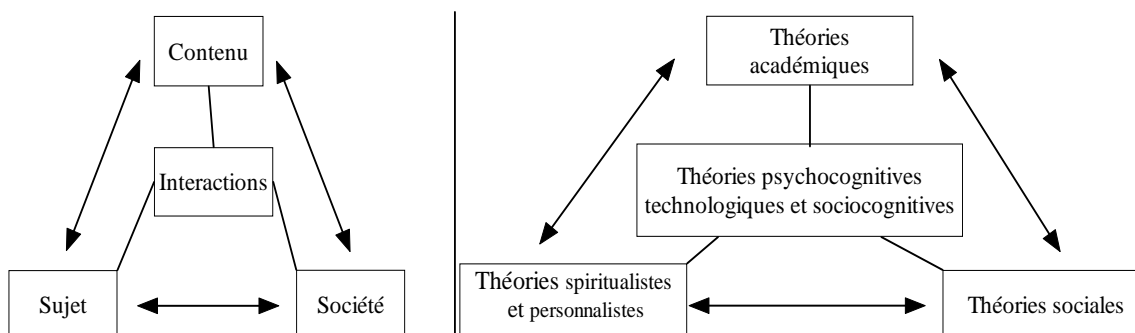


Figure 1.3 : Théories contemporaines de l'Education selon (Bertrand 1993)

Les technologies éducatives qui ont émergé depuis les années 70 ont également subi cette influence au point que dans les publications scientifiques les critiques des technologies éducatives portent autant sur les théories de l'éducation sous-jacentes que sur les technologies elles-mêmes. La typologie souvent utilisée est la suivante ; on distingue :

- Les théories behavioristes ou théories empiristes du comportement. Ces théories se concentrent sur les comportements observables des apprenants lorsqu'ils sont face à des situations bien définies (Skinner 1968). L'apprentissage est vu comme un processus de modification du comportement par établissement et renforcement de nouvelles associations entre des stimuli et des réponses. Ces théories souvent critiquées aujourd'hui ont cependant permis de mettre en valeur l'importance de la boucle de rétroaction dans les apprentissages qui a été ensuite reprise par la plupart des autres théories. Malgré leurs limites ces théories sont aujourd'hui au cœur de la grande majorité des plates-formes de formation à distance.

- Les théories (psycho)-cognitivistes (Piaget 1967) qui, en réaction aux théories behavioristes considèrent que ce qui se passe dans la tête d'un apprenant est tout aussi important que ce qu'il manifeste par ses interactions avec un système. L'idée est donc de formaliser les fonctions cognitives de l'apprenant mises en jeu lors d'apprentissages (images mentales, représentations et stades de développement, conceptions préalables) puis d'encourager dans des systèmes la mise en œuvre et l'évaluation de ces fonctions cognitives (McGilly 1994).
- Les théories connexionnistes qui contestent dans l'hypothèse cognitive le caractère symbolique des représentations et expliquent la cognition par l'émergence d'états globaux stables dans un réseau d'unités élémentaires de traitement de l'information en interconnexion mutuelle (Rumelhart et Mc Clelland 1986), (Laks 1996). De nombreux travaux récents, notamment en intelligence artificielle, sont basés sur ces théories et modèles auto-organisationnels basés sur des réseaux de neurones formels (cf http://fr.wikipedia.org/wiki/Intelligence_artificielle#Le_connexionnisme),
- Les théories constructivistes qui relèvent du courant cognitiviste et en sont les représentantes les plus référencées en Education. Ces théories mettent l'accent sur la construction de la connaissance en examinant tout particulièrement deux aspects : les conceptions préalables des apprenants (Bachelard 1938) et leurs profils pédagogiques (Garanderie 1980). Le constructivisme postule qu'on apprend en faisant, en construisant des objets (Papert 1981), la notion de conflit cognitif étant souvent utilisée pour mettre l'apprenant en situation d'apprentissage.
- Les théories sociocognitives insistent sur les facteurs sociaux et culturels dans la construction de la connaissance (Vygotsky 1985), (Bandura 1986). Ces théories cherchent à analyser les interactions sociales de coopération dans la construction de savoirs et proposent des pédagogies à base de coopérations pour amener les apprenants à construire ensemble leurs connaissances ; ici le conflit socio-cognitif est souvent point de départ des apprentissages et les interactions sociales qu'un apprenant expérimente contribuent à son développement par un effet d'échafaudage (scaffolding) (Slavin 1987).
- Les théories de la cognition-située postulent quant à elles que la cognition et l'apprentissage se situent toujours par rapport à un environnement physique et social concret (Grison 2002). Ce courant est généralement associé aux travaux de Lucy Suchman qui mettent en évidence le caractère opportuniste et improvisé de l'action, son ancrage matériel et social, et par voie de conséquence le rôle fondamental joué par les facteurs contextuels et les mécanismes de production d'intelligibilité mutuelle entre acteurs (Suchman 1987).

Comme (Lebrun 2002) et (Schneider 2003), je considère que cette typologie est trop souvent analysée plus comme une hiérarchisation que comme une classification. «Nous postulons que ces écoles ne représentent pas forcément une évolution naturelle de nos connaissances sur l'apprentissage et la pédagogie. Elles sont plutôt représentatives de catégories naturelles de différents types de situations d'apprentissage » peut-on lire en page 3 de (Schneider 2003). Selon moi, chacune de ces théories peut se révéler adaptée à un besoin d'enseignement précis et des pédagogies plutôt transmissives sont souvent nécessaires car plus efficaces pour la mise en place d'apprentissages de base. Il n'y a qu'à observer les pratiques des enseignants dans leurs classes pour s'en apercevoir, étant entendu que ces derniers insèrent ces pratiques transmissives dans un processus plus global mettant à profit les méthodes actives qui seront décrites dans le prochain paragraphe.

1.1.2.2 De la complémentarité de différentes méthodes actives

Dans son livre consacré aux technologies pour apprendre (Lebrun 2002), l'auteur consacre un chapitre aux méthodes actives pour une utilisation effective des technologies. Ces méthodes actives font clairement référence au constructivisme ainsi qu'aux théories socio-cognitives rapidement présentées dans le paragraphe précédent.

La notion de projet est donc centrale au socio-constructivisme. L'apprenant va se mettre, dans certaines circonstances, en situation de projet : les buts qu'il se fixe, les découvertes résultant de l'interaction avec son environnement vont l'amener à utiliser stratégiquement ses ressources cognitives de façon à créer de nouvelles connaissances en extrayant l'information de l'environnement et en l'intégrant dans sa structure informationnelle déjà présente en mémoire (Kozman 1991). Les projets soumis initialement résultent d'une analyse préalable de l'enseignant sur les buts à atteindre par l'apprenant qu'il s'agisse de gains dans les connaissances, d'un développement de compétences en résolution de problèmes ou encore des changements dans les perceptions, les attitudes, les valeurs et le comportement (Brown et Atkins 1988).

Ces méthodes actives placent donc l'apprenant en situation de piloter ses apprentissages en s'attachant à résoudre les problèmes posés par le projet (ou l'activité) proposé(e) : formulation des obstacles à la résolution du problème, recherche et analyse de ressources utiles, estimation du temps et planification des tâches permettant la résolution du problème, collaboration éventuelle avec d'autres apprenants, ... L'enseignant (tuteur) devient pour sa part un chef d'orchestre qui facilite l'activité cognitive de l'apprenant. L'enseignant peut aider l'apprenant pour chacune des phases de résolution du problème, proposer des scénarios amenant l'apprenant à se concentrer sur des problèmes plus simples utiles à la résolution du problème posé, fournir des outils et ressources facilitant l'activité cognitive de l'apprenant.

Les méthodes actives se caractérisent donc par le caractère personnel de l'apprentissage, le rôle catalyseur des connaissances antérieures, l'importance des « ressources » mises à disposition, le rôle du contexte et de l'expérience concrète, les compétences de haut niveau que l'apprenant doit exercer, la « démarche de recherche » dans l'apprentissage, le changement conceptuel imposé à l'apprenant (prise de conscience du problème, conflit cognitif, reformulation), le lien entre projets personnel, professionnel et d'étude, l'importance d'une construction voire d'une production par l'apprenant, le rôle de la réflexion sur l'apprentissage (notamment de la méta-cognition) (Lebrun 2002).

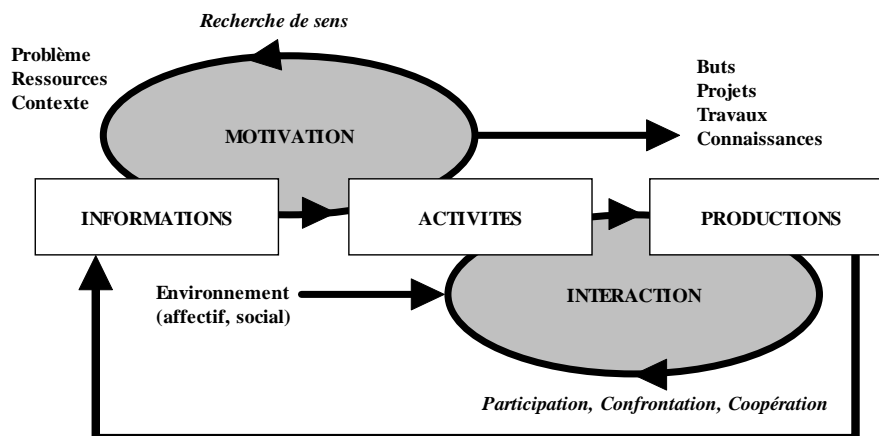


Figure 1.4 : Méthodes actives et cycles d'apprentissage

Trois types de méthodes actives me semblent particulièrement intéressantes pour une mise en œuvre via les Technologies de l'Information et de la Communication :

- L'apprentissage par résolution de problèmes (en anglais Problem-Based Learning ou PBL) est une méthode pédagogique qui est basée sur la résolution de problèmes concrets extraits de situations de la vie quotidienne ou professionnelle. Les problèmes posés aux apprenants sont pensés pour déclencher des activités d'ordre cognitif conduisant l'apprenant à poser le problème, le comprendre, formuler des hypothèses, entreprendre des actions (individuelles / collectives), interpréter et analyser les résultats des actions.
- La pédagogie du projet ou par le projet est une démarche permettant à l'apprenant de se mettre en situation de projet, de se construire un projet pour aboutir à un produit comme résultat du projet. Souvent utilisée dans le cadre d'apprentissages professionnels, cette

pédagogie est souvent assimilée à de l'apprentissage adulte (mais je considère qu'on ne doit pas l'y cantonner). Une caractéristique importante de la pédagogie du projet est qu'elle est soucieuse à la fois des produits de l'apprentissage et de la réflexion sur la démarche ayant conduit au passage d'un produit à un autre pour arriver au produit final livré.

- L'apprentissage coopératif qui vise à faire travailler les apprenants en groupes suffisamment restreints pour que chacun participe à la tâche collective assignée. Le rôle de l'enseignant (ou de l'EIAH) est alors plus de piloter les groupes (temps de parole, apport d'informations nouvelles, recours à l'argumentation, rappel des objectifs, ...) que d'intervenir sur les contenus d'apprentissage. Notons que tout apprentissage en groupe n'est pas de l'apprentissage coopératif. Les apprenants sont censés réaliser leur tâche sans supervision directe et immédiate de l'enseignant et la tâche doit être formulée de manière à ce qu'il y ait dépendance des apprenants les uns par rapport aux autres.

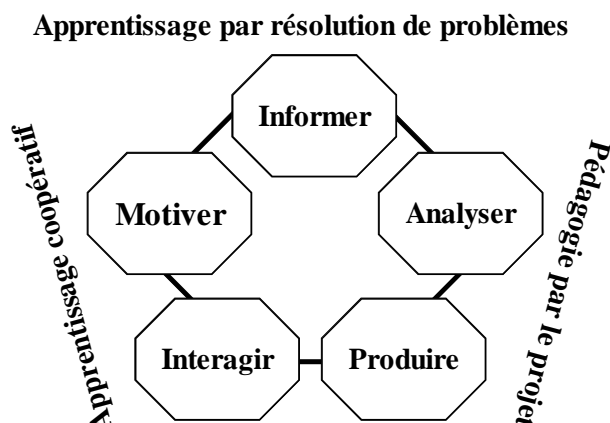


Figure 1.5 : Positionnement des méthodes pédagogiques actives selon (Lebrun 2002)

Je reviendrai plus en détail sur les principes de ces méthodes actives, tout particulièrement dans le prochain paragraphe consacré au cadre théorique qui sert de référence à ces travaux. Il est d'ores et déjà utile d'insister sur la complémentarité de ces méthodes actives :

- L'apprentissage par résolution de problèmes et la pédagogie du projet partagent une philosophie commune visant à proposer à l'apprenant des situations proches (ou issues) de la vie réelle et donc motivantes pour l'apprenant. Une situation bien conçue place l'apprenant en situation de conflit cognitif qui est un moteur de l'apprentissage. La pédagogie du projet se distingue cependant de la pédagogie par résolution de problèmes par le fait qu'en pédagogie du projet, le produit à atteindre est le plus souvent défini explicitement dans l'énoncé (en résolution de problèmes, l'objectif à atteindre est souvent masqué et le problème est ouvert : « ill-structured problems »).
- L'apprentissage coopératif n'est pas obligatoire en apprentissage par résolution de problèmes ou en pédagogie du projet. Ses apports sont cependant significatifs tant sur les aspects liés à la motivation de l'apprenant que sur l'analyse des situations d'apprentissage et les productions réalisées (cf Figure 1.4) : la coopération argumentative est un élément important pour amener les apprenants à discuter sur leurs représentations et leurs avis, pour les amener à confronter leurs idées sur le projet mené (du conflit cognitif au conflit socio-cognitif), pour les amener à atteindre des représentations communes et validées.
- Avec les méthodes actives, les documents et les ressources de l'apprentissage changent de statut : ces documents sont eux-mêmes des documents actifs intimement liés à l'activité de l'apprenant qu'ils supportent ou dont ils rendent compte. Dans ces conditions, les documents deviennent des ressources cognitives situées, des vecteurs ou des preuves (traces) de l'apprentissage.
- Enfin, il est très important de noter que les méthodes actives, si elles sont fermement ancrées dans la théorie (socio) constructiviste de l'apprentissage (cf paragraphe 1.1.2.1),

ne font pas table rase des formes plus traditionnelles d'enseignement qui trouvent toute leur place lorsqu'on les situe correctement dans le cadre de ces activités socio-constructivistes.

Ce qui caractérise aussi ces méthodes actives, c'est qu'elles ne sont pas forcément difficiles à mettre en œuvre technologiquement mais elles requièrent un véritable design pédagogique en amont des situations proposées aux apprenants. Côté outillage, les fonctionnalités de type C3MS (Community, Content and Collaboration Systems) présentées en Figure 1.2 paraissent couvrir le spectre des besoins et peuvent jouer un rôle important d'amplificateur si elles sont mises au service de tâches porteuses de sens pour les apprenants. Constituer un EIAH exploitant les méthodes actives présentées est cependant complexe car les fonctionnalités si elles existent doivent être assemblées en un tout cohérent du point de vue de l'apprenant (et du tuteur) exploitant cet EIAH. Ceci pose le problème de la conception de ce type d'environnement et tout particulièrement celui de la gestion du « workflow pédagogique » porté par cet EIAH. Le terme workflow pédagogique proposé par (Schneider 2003) correspond ici à « un scénario qui impose certains buts, qui organise le projet en plusieurs modules plutôt séquentiels et qui eux même peuvent être scénarisés si l'enseignant (le concepteur) le juge utile ». Ce scénario doit notamment expliciter les modalités de tutorat intégrées dans l'EIAH, modalités pouvant être réparties en trois types d'intervention (Schneider 2003) :

- Interventions de type « manager » pour veiller à ce que les apprenants soient actifs et impliqués,
- Interventions de type « facilitateur » pour aider les apprenants dans le choix et l'utilisation des ressources (documents, logiciels) utiles à la résolution des problèmes,
- Interventions de type « orchestrateur » pour étayer les apprentissages sur des compétences affirmées de l'apprenant, pour éviter une trop grande disparité entre le niveau de l'apprenant et le niveau de difficulté de la tâche proposée (cf zone proximale de développement), pour encourager les apprenants à décomposer les problèmes en sous-problèmes.

Au final, les méthodes actives présentées dans ce paragraphe paraissent être en mesure de répondre à deux difficultés auxquelles se trouvent confrontés les enseignants et les concepteurs : ces méthodes placent l'apprenant en situation d'être acteur de ses apprentissages tout en laissant à l'enseignant (au tuteur) la possibilité de contrôler les apprentissages (cf les types d'interventions énoncées ci-dessus). Elles permettent en effet de mêler pédagogie du problème (guidage par la situation et les activités collectives) et pédagogie du sujet (prise en compte des tâches particulières menées par chaque apprenant pour étayer les apprentissages et proposer des formes d'enseignement plus transmissives). Ainsi, les méthodes actives semblent pouvoir donner lieu à des EIAH qui sont à la fois des systèmes orientés performance mais aussi des systèmes orientés design pédagogique (Murray 1999), (Tchounikine 2002a). Cette position sera développée dans le prochain paragraphe consacré à la présentation du cadre théorique servant de référence à mes travaux.

1.2 Cadre théorique de référence pour la conception d'EIAH

Les paragraphes précédents m'ont amené à proposer l'ingénierie d'EIAH reposant principalement mais non exclusivement sur des méthodes actives d'apprentissage. Ce paragraphe propose des références bibliographiques démontrant que les méthodes actives se prêtent à un tel métissage de méthodes et que ce métissage est possible et souhaitable.

1.2.1 Présentation de la théorie empirique de l'apprentissage

La Théorie Empirique de l'Apprentissage telle que définie dans (Develay 1993) est empirique dans le sens où elle est fondée sur l'expérience et parce qu'elle relève d'emprunts à des courants divers.

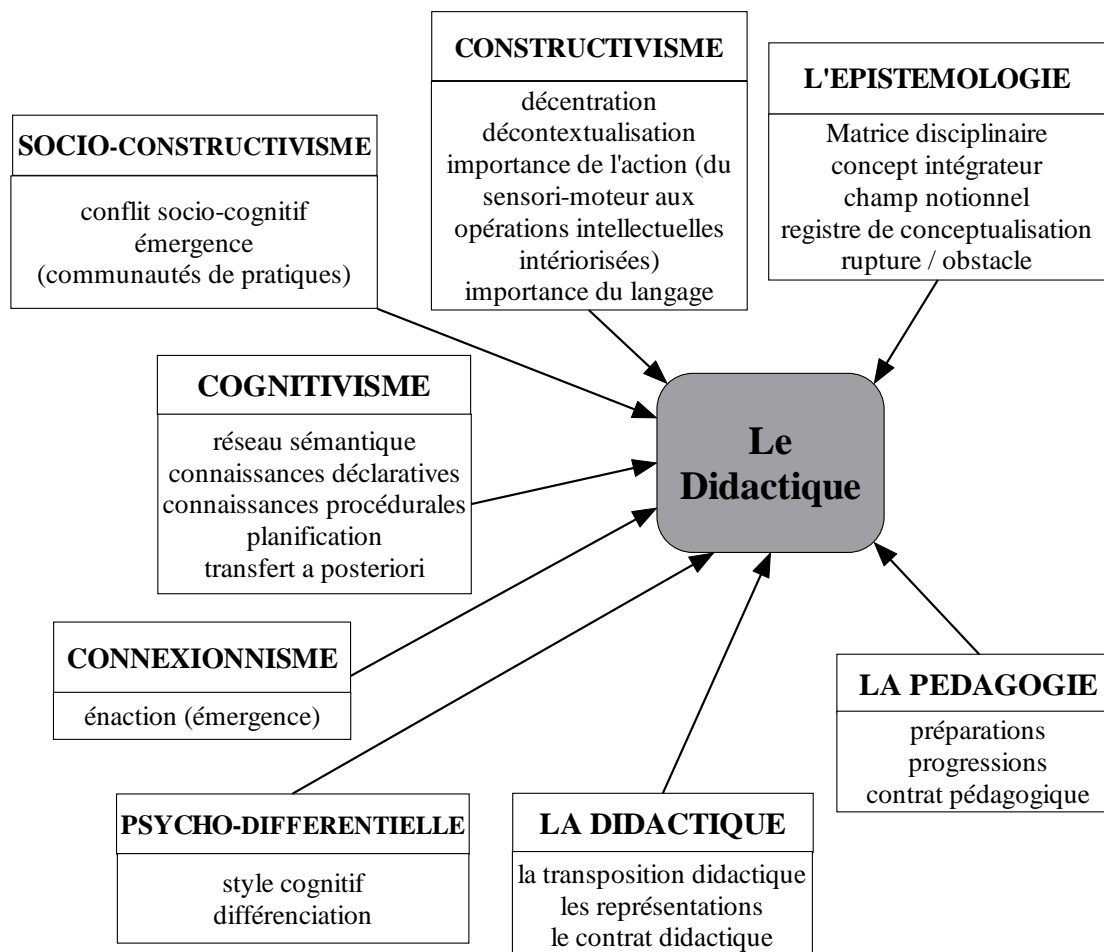


Figure 1.6 : Les emprunts de la théorie empirique de l'apprentissage (Develay 1993)

Ainsi, Michel Develay définit le didactique en s'appuyant sur des concepts⁴ relevant de différents domaines d'emprunt comme le montre la figure précédente. Trois principes sont cités par l'auteur pour résumer cette théorie empirique :

- Apprendre, c'est trouver du sens dans une situation qui n'en possède pas forcément au départ.
- Tout apprentissage est le résultat d'une mise en œuvre par l'apprenant d'une habileté cognitive définie à partir de son système de représentation de la situation.
- Apprendre, c'est être capable de transférer l'habileté cognitive.

1.2.1.1 Tout apprentissage est recherche de sens

Chercher un sens à une situation, c'est relier l'existant à une autre situation rencontrée auparavant et pour laquelle il est en général possible d'anticiper les critères de réussite de la tâche. Cette situation, si elle est ancrée dans une pratique sociale de référence (secteur professionnel par exemple) rend la tâche proposée utile et motivante car en liaison avec l'usage social ou domestique que l'apprenant pressent. C'est le sens des pédagogies du projet qui visent cette articulation tâche / pratique sociale de référence.

⁴ Les concepts apparaissant dans cette partie du manuscrit sont définis dans le glossaire qui donne la définition de chacun des termes utilisés (exemple : le terme « pratique sociale de référence »).

Face à une tâche complexe, l'apprenant va devoir se représenter le but à atteindre et les propriétés des objets sur lesquels il doit travailler. Il doit également anticiper la démarche à suivre, anticiper les résultats des opérations qu'il prévoit d'engager, évaluer ses productions et ses stratégies pour réguler son activité. Dans cette quête, il s'appuie sur des critères de réussite formels et fonctionnels qu'il s'est donnés ou qui lui ont été fixés lors de la présentation de la situation grâce aux consignes qui lui ont été données. En ce sens, le rôle de l'enseignant qui propose la situation est de faire émerger le désir d'apprendre en créant l'énigme et les conditions de sa résolution (Meirieu 1994). L'énigme et les critères de réussite sont fonction des représentations initiales des apprenants sur un phénomène ou un objet, du registre de conceptualisation à atteindre par les apprenants pour chaque notion abordée.

L'évaluation formatrice va permettre de réguler les démarches de l'apprenant en l'amenant d'abord à se donner une représentation correcte des critères de réussite des tâches auxquelles il doit se confronter et ensuite à planifier ses actions. Philippe Perrenoud utilise la métaphore du « voyage dans le voyage » pour désigner le parcours individuel d'un apprenant au regard d'une situation d'apprentissage donnée qui est proposée au groupe classe par exemple (Perrenoud 1997). Pour cette différenciation intégrée à la gestion d'une situation-problème, il s'agit d'observer l'élève :

- soit pour réguler son activité dans le contexte de la tâche associée à la situation-problème,
- soit pour l'orienter vers d'autres situations d'apprentissage, à l'intérieur du même espace-temps de formation,
- soit pour l'orienter vers d'autres espaces-temps de formation.

Ce voyage dans le voyage proposé à l'apprenant est difficile à organiser par l'enseignant qui doit amener chacun à la maîtrise de l'essentiel dans le même temps. Mais cette différenciation à partir de la situation-problème initiale contribue à cette recherche de sens à la fois pour l'apprenant et l'enseignant.

La planification pédagogique faite par l'enseignant trouve ici toute son importance. Comme d'autres situations didactiques, les situations-problèmes exigent des préparations très précises pour que l'enseignant puisse ensuite réguler l'activité individuelle des apprenants. La situation-problème initiale, les situations d'application, de remédiation et de transfert doivent être construites, organisées selon des progressions basées sur des prédictions relatives aux comportements observables des apprenants. Pour la théorie empirique de l'apprentissage, Pédagogie et Didactique contribuent donc aussi à cette recherche de sens qui va motiver les apprenants.

1.2.1.2 Tout apprentissage est le résultat d'une mise en œuvre par l'apprenant d'une habilité cognitive définie à partir de son système de représentation de la situation

« La situation d'apprentissage qu'un enseignant propose recèle selon lui des objectifs qui correspondent aux obstacles qu'il présuppose que les élèves auront à surmonter. C'est la raison pour laquelle toutes les taxonomies d'objectifs peuvent être abordées en termes d'objectifs enseignants et de comportements attendus de l'élève. L'obstacle présent pour l'élève, dans la situation proposée par le maître, est le déclencheur espéré de son apprentissage ».

Chaque problème à résoudre pour surmonter un obstacle nécessite que l'apprenant utilise un programme de traitement approprié à ce problème. Chaque programme de traitement consiste le plus souvent en un travail d'adaptation d'un programme de traitement utilisé dans une autre situation. Ainsi, un travail cognitif important, dans une situation d'enseignement, est l'identification et la caractérisation des familles de problèmes présents. Non seulement pour optimiser son enseignement, l'enseignant devrait parvenir à identifier les tâches que sa discipline nécessite de résoudre au niveau où il enseigne, mais il devrait aussi avoir une idée

relativement précise, à travers ces obstacles, des familles de problèmes à résoudre. Il devrait, de surcroît, aider ses élèves à identifier les activités mentales (correspondant aux programmes de traitement précédents) qu'ils ont dû mettre en œuvre afin d'espérer le réinvestissement ou le transfert ultérieur. Identifier une famille de problèmes revient donc à prélever, dans une situation, les indices pertinents pour sa résolution. Une situation-problème est un cas particulier de situation pour laquelle on connaît à la fois une procédure amenant à la solution et les propriétés relationnelles qui permettent de justifier cette procédure.

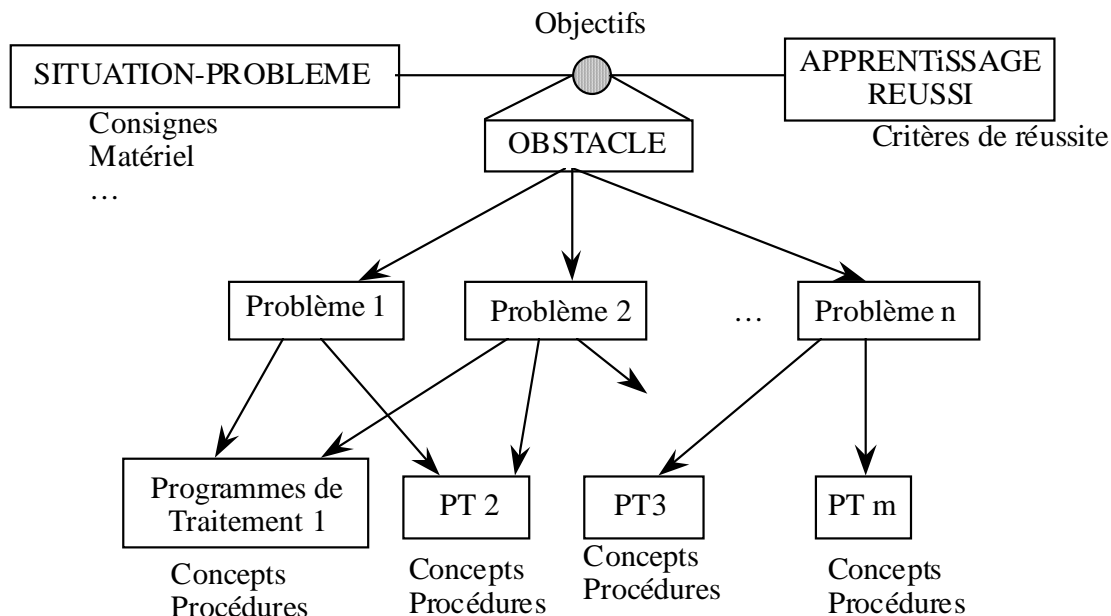


Figure 1.7 : De l'obstacle aux programmes de traitement à mettre en jeu par l'apprenant

Ainsi, comme toute situation didactique, la situation-problème doit être construite en s'appuyant sur une triple évaluation des motivations, des compétences et capacités des apprenants. L'apprentissage correspond alors à une modification de la structure cognitive de l'apprenant, l'obstacle surmonté étant la cause de cette transformation : «au départ, la tâche d'apprentissage déclenche dans la mémoire à court terme (MCT) une représentation des buts à atteindre, des critères de réussite, en même temps qu'elle conduit à s'interroger sur la planification qu'elle impose (gestion du temps, ordre de succession des actions, ...). Simultanément sont activées en mémoire à long terme (MLT) des connaissances déclaratives et procédurales. L'intrication des connaissances déclaratives et procédurales activées entraîne une rétroaction au niveau des représentations et de la planification. L'ensemble permettra finalement de résoudre le problème (espérons-le) » écrit (Develay 1993).

La dimension affective de l'apprentissage est corrélative à sa dimension cognitive. A l'instant où l'apprenant est confronté à une situation d'apprentissage, une position de déséquilibre s'installe en lui. Le système de représentation qui jusque là lui permettait de comprendre le monde se trouve mis en défaut. Ce déséquilibre initial (inhérent à la situation d'apprentissage si elle est bien construite) va s'accompagner d'une déstabilisation affective jusqu'à ce que l'apprenant parvienne à trouver de nouveaux repères. C'est par l'hypothèse que l'apprenant va tenter d'échapper au déséquilibre initial, en faisant émerger (cf rôle de la décentration) une restructuration cognitive et simultanément une restabilisation affective. L'apprentissage coopératif peut être un élément important dans ce processus parce qu'il permet l'émergence d'hypothèses, de points de vues, et impose l'argumentation entre apprenants (conflit socio-cognitif).

Ainsi, Michel Develay propose de modéliser les éléments constitutifs d'une situation d'apprentissage de type situation-problème :

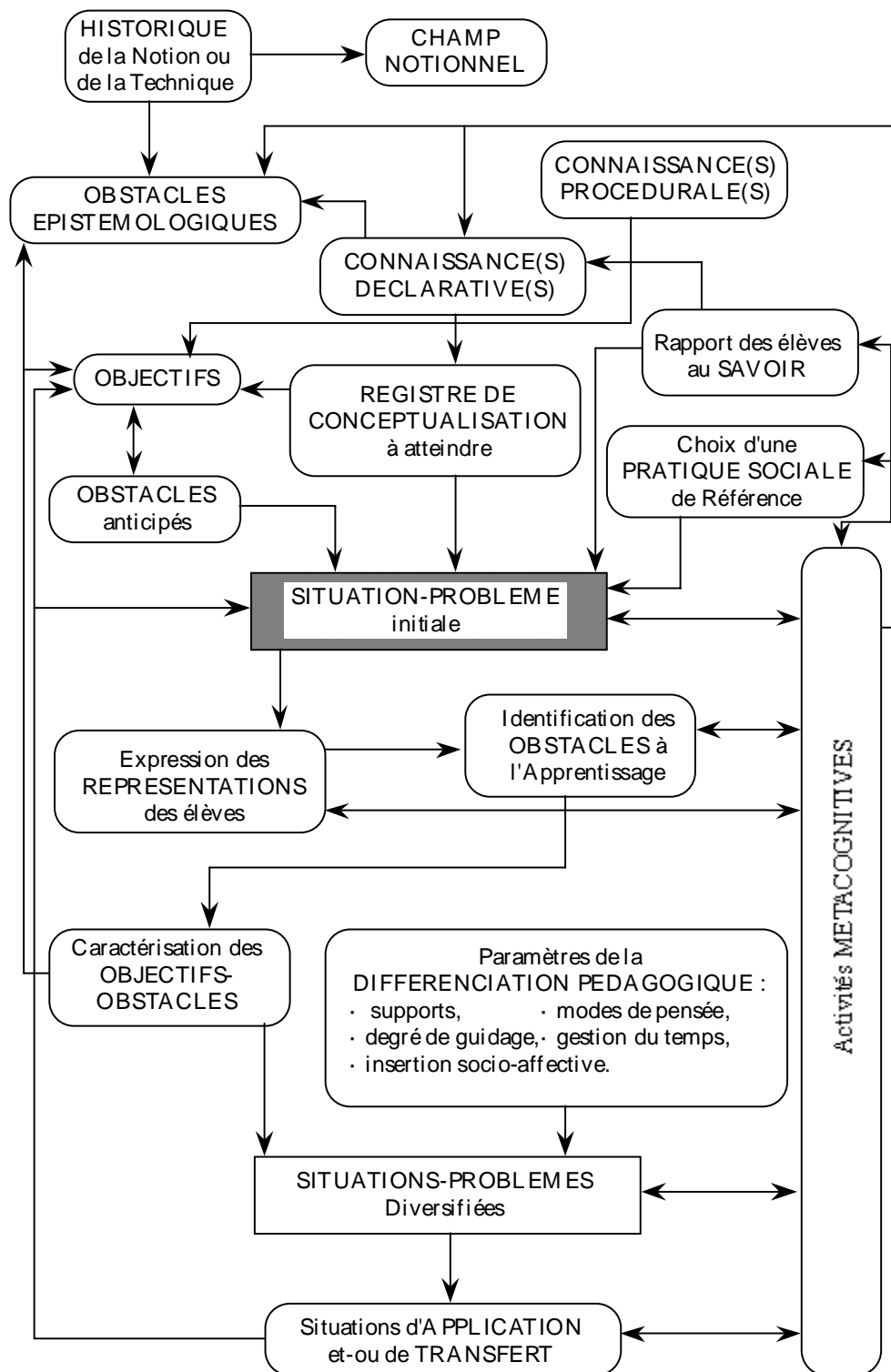


Figure 1.8 : Eléments à prendre en compte pour concevoir une situation d'enseignement

De nombreuses boucles de *rétroaction* introduisent constamment des *régulations*, de sorte que ce schéma est à analyser de manière *systémique* et non de manière linéaire. Un enseignant envisagera d'abord, par exemple, les épreuves d'évaluation qu'il peut proposer à ses élèves, avant de songer à la situation d'apprentissage proprement dite. Un autre songera plutôt aux représentations de ses élèves avant de s'intéresser à définir précisément le registre de conceptualisation de la notion qu'il souhaite faire acquérir.

1.2.1.3 Seule l'activité de transfert de l'habileté cognitive permet de s'assurer que l'on a appris

Le transfert postule que des connaissances ou des compétences acquises dans un contexte donné peuvent être utilisées dans un autre contexte. Le transfert peut se produire d'un apprentissage à un autre et d'une connaissance à une situation de résolution de problèmes. Le transfert se produit donc lorsque des connaissances acquises antérieurement influencent la façon avec laquelle de nouvelles connaissances sont acquises. « Réussir, c'est comprendre en action une situation donnée à un degré suffisant pour atteindre les buts proposés. Comprendre, c'est parvenir à dominer en pensée les mêmes situations jusqu'à pouvoir résoudre les problèmes qu'elles posent quant au pourquoi et au comment des liaisons constatées et par ailleurs utilisées dans les actions » (Piaget 1974).

Pour que le transfert s'effectue, il est important d'amener l'apprenant à une certaine réflexion cognitive sur les situations proposées et son comportement particulier. Le transfert est bien sûr facilité par les activités métacognitives : on considère qu'il y a activité métacognitive chaque fois qu'il y a recul par rapport à l'action pour analyser cette dernière, ce recul pouvant porter sur trois aspects : l'apprenant lui-même, les tâches, les stratégies mises en œuvre (Flavell 1976).

Le transfert des connaissances est donc un domaine dans lequel l'enseignant doit intervenir directement et explicitement car il ne se produit pas sans actions spécifiques. A l'issue de la résolution d'une situation-problème, il est important de proposer plusieurs exemples, y compris des contre-exemples. C'est à partir de ceux-ci que l'élève parvient à une certaine décontextualisation de ce qui est appris afin de pouvoir le transférer ensuite.

1.2.2 Fondements de la théorie empirique de l'apprentissage

La théorie empirique développée par Michel Develay est un cadre conceptuel assez ouvert qui paraît intéressant car non recroquevillé sur des principes qu'il faudrait absolument appliquer en éducation. De telles propositions sont formulées de plus en plus clairement par différents auteurs. Ainsi, répondant au manifeste de David Merrill (Merrill 1996), Brent Wilson écrit : « My intent is to work toward a conception of Instructional Design that is inclusive of different perspectives while preserving a coherent identity through a shared history and common purposes » (Wilson 2004b).

Aussi, malgré son qualificatif d'empirique (en référence à ses différents emprunts –cf Figure 1.6) la théorie proposée par Michel Develay paraît aussi apparentée à la tendance systémique de l'éducation selon la définition de (Bertrand 1993). De manière générale, les approches de type systémique sont basées sur la volonté de décrire la globalité des structures et planifier l'ensemble des opérations (Morin 1977), (De Rosnay 1975). Il s'agit d'une théorie importante que des auteurs comme Gagné (Gagné, Briggs et al. 1988), Dick et Carey (Dick et Carey 1990), et Merrill (Merrill 1996) ont popularisé en éducation. Le modèle systémique de base est un cadre établi pour organiser et tenir compte des entrées, des composantes de processus et des résultats de l'enseignement. Il aide à avoir en tête les différentes interactions et à poursuivre les buts de la formation. Il peut aussi faciliter la planification de l'enseignement en faisant apparaître et en organisant les facteurs dont on doit tenir compte pour une efficacité maximale.

Si les approches de type Apprentissage situé se focalisent sur la situation et donnent la primauté de la communauté de pratique sur l'activité de l'individu (l'émergence de connaissances résultant de l'improvisation des interactions), si les approches de type cognitiviste considèrent l'interaction humaine comme une interaction entre deux systèmes de représentations symboliques et de traitement de l'information (basées sur des connaissances déclaratives et procédurales), « les approches systémiques conduisent non seulement à prendre en compte la situation mais permettent aussi de traiter dans un cadre

théorique homogène les activités et la cognition individuelle, et celles de l'action collective » (Rogalski 2004). Michel Develay souligne cependant que « toutes les modélisations de l'action éducative ou des situations éducatives attestent de la complexité du système en jeu, d'autant que ces modélisations concernent la prévision de l'action et l'action effective » (Develay 1993). Dans cette perspective, (Hacker 1985) a développé une théorie de l'action, centrée sur un modèle de système de double régulation de l'activité, avec une insistance sur la finalité de l'action (sujet) et son caractère social (division du travail, besoins et valeurs insérées dans un cadre social). Ce modèle souligne les processus de double détermination de l'activité par la situation et le sujet, et de double impact sur les composants de la situation et sur le sujet ; ces impacts modifient en retour les conditions de l'activité, dans un processus dynamique de régulation où interagissent des composants structurels de la situation (les tâches, la division du travail, les ressources et contraintes pour la réalisation des tâches) et les propriétés (relativement) stables du sujet (sa posture par rapport à la situation de travail, ses compétences) et des composants circonstanciels liés à la situation singulière et à l'état actuel du sujet (cognitif, affectif, physiologique).

Le modèle de « double germination » des concepts défendu dans (Vygotsky 1985) offre un cadre pour analyser un développement des compétences articulant les apports de l'expérience en situation de résolution de problème et ceux d'une forme centrée sur les savoirs de référence (profession). Le modèle propose un double mouvement de construction : une émergence de concepts à partir de l'action, via la médiation sociale (des pairs ou des anciens), et une intériorisation des savoirs de référence, explicitement proposés par les formateurs. La notion de zone proximale de développement (ZPD) pointe sur l'existence de conditions pour l'efficacité de la formation : en deçà de sa ZPD, la formation n'apporte rien au sujet qu'il n'ait déjà construit, au delà, l'appropriation des concepts professionnels comme scientifiques ou techniques ne peut avoir lieu.

Le cadre de la double régulation de l'activité permet donc de prendre en compte les évolutions de la conceptualisation au niveau individuel comme au niveau collectif, les activités métacognitives constituant un composant important de l'activité (Pastré 1999).

1.3 Questions de recherche posées par la problématique choisie à la communauté EIAH

Le cadre théorique dans lequel j'ai inscrit mes travaux étant maintenant défini, je vais m'attacher dans la dernière partie de ce chapitre à énoncer les questions de recherche posées par l'objet d'étude choisi : l'ingénierie d'EIAH reposant sur des méthodes actives conformes à la théorie empirique de l'apprentissage présentée au paragraphe 1.2.1.

La situation actuelle me paraît paradoxale, sans doute plus encore que la situation dans laquelle nous étions il y a une dizaine d'années⁵. La présentation des outils faite dans le paragraphe 1.1.1.1 a montré que la communauté EIAH sait développer des outils cognitifs performants (Belvedere, Algebra et Geometry tutor, Ecolab, ...), y compris des chaînes de produits tels Aplusix, Cabri Géomètre ou Roboteach diffusés dans les établissements scolaires. Par ailleurs, les fonctionnalités de base permettant de supporter des pédagogies actives sont maintenant assez bien connues et je les ai rappelées dans la Figure 1.2. Enfin, même si l'on peut considérer que les principes théoriques présentés en paragraphe 1.2 sont pour certains plus descriptifs que prescriptifs (Dessus 2002), la communauté EIAH peut s'appuyer sur ces principes pour développer et diffuser des environnements-supports utiles pour les enseignants et les apprenants.

⁵ Lorsque j'ai terminé mon doctorat informatique (Nodenot 1992a) et initié mes travaux sur l'ingénierie des situations-problèmes (Nodenot 1995), le domaine n'était pas réellement porteur. En France, les recherches en Sciences de l'Education relatives à l'apprentissage par situations-problèmes étaient en pleine effervescence (Develay 1993), (Meirieu 1994) mais il n'y avait pas d'outils-supports et les travaux structurés de la communauté de recherche (dite aujourd'hui communauté EIAH) étaient centrés principalement sur les systèmes tuteurs intelligents.

Par ailleurs la communauté éducative est maintenant largement familiarisée aux TIC (ce qui n'était pas le cas il y a dix ans) et beaucoup d'enseignants ont en partie intégré dans leurs pratiques ces méthodes actives pour lesquelles les outils-supports seraient bien utiles. Mais force est de constater le faible transfert de compétences et de technologie entre communauté EIAH et communauté éducative. En tant que référent (informel) TICE au niveau de l'IUT et en tant que Responsable Formation continue au niveau de l'IUT, je suis souvent sollicité pour conseiller mes collègues-enseignants qui ont envie d'utiliser des environnements informatiques comme moyen de facilitation de leurs enseignements divers. Ces échanges sont très enrichissants mais peuvent se révéler frustrants :

- Du côté de l'enseignant, les demandes ont plusieurs caractéristiques récurrentes : l'enseignant verbalise son besoin dans son vocabulaire en s'appuyant sur des éléments d'un vocabulaire technique dont il ne connaît pas forcément tout le sens. Craignant d'être éconduit s'il demande des choses qu'il juge complexe, il présente la plupart du temps son projet de manière simplificatrice se disant qu'une fois ce projet simple mis en place, il sera facile de le modifier pour étendre les fonctionnalités du système informatique initial. En général, l'enseignant est prêt à consacrer une partie de son temps personnel sur ce projet important mais se refuse à utiliser des systèmes génériques (de type « usine à gaz ») dans lesquels les fonctionnalités dont il a besoin seraient noyées au milieu d'un environnement complexe. Il attend donc des conseils précis du type : « Quel environnement de développement dois-je utiliser ou faire utiliser ? Combien de temps faut-il pour développer un système aussi simple que celui que je te demande ? Qui peut prendre en charge ce développement ? ». Une discussion approfondie avec l'enseignant amène souvent à relativiser les besoins énoncés, à prendre conscience de l'importance d'autres aspects non verbalisés initialement, l'EIAH demandé réclamant la plupart du temps une partie Back-Office (pour la gestion des contenus et des traces des apprenants) et une partie Front-Office (au service du suivi par les apprenants).
- Au cours de ces échanges, le chercheur en EIAH a parfois du mal à interpréter et à répondre simplement aux questions posées, à argumenter ses réponses faute de support conceptuel et technique vers lequel orienter l'enseignant. Il hésite entre la peur de décourager les initiatives et la nécessité de promouvoir des développements informatiques pérennes qui seuls pourront conduire à des EIAH acceptables par l'enseignant. Dans ma tête, les questions fusent parfois, souvent sans réponse formelle autres que mon expérience ou mon intuition : « Faut-il cadrer le projet avant tout démarrage du développement ou faut-il encourager une démarche qui consiste à faire quelque chose qui sera peut-être jeté ou repris ensuite ? Comment expliquer simplement que de mauvais choix initiaux peuvent conduire à des développements en pure perte ? Quelles sont les conditions et la démarche pour développer une série de prototypes que l'on va faire évoluer de l'EIAH annoncé vers le besoin réel non forcément verbalisé par l'enseignant ? ». Au final, enseignant et chercheur en EIAH ressortent souvent de tels échanges assez frustrés ; l'un est frustré de ne pas avoir obtenu la réponse exacte qu'il était venu chercher (« le projet qui te tient tant à cœur, on va le mettre en route car c'est techniquement, humainement et financièrement possible »). L'autre (le chercheur) est frustré de ne pouvoir répondre simplement à des projets pertinents qui mettent en jeu des connaissances en ingénierie d'EIAH pour lesquelles il n'existe guère de documents lisibles par un public non averti.

Pour dépasser ces difficultés qui limitent bien des synergies entre les chercheurs issus de différents domaines, entre les chercheurs et les enseignants, la communauté EIAH doit d'abord identifier clairement les verrous scientifiques avant de proposer des solutions. Dans le prochain paragraphe, j'essaie d'identifier les principaux verrous en m'appuyant sur divers travaux de référence notamment (Balacheff 2001), (Downes 2002), (Paquette 2002), (Tchounikine 2002a), (Linard 2003), (Wiley 2003) et (Balacheff 2004).

1.3.1 Verrous liés aux caractéristiques des environnements supports utilisés

Le terme environnement support désigne ici une plate-forme logicielle (parfois un langage de programmation, plus souvent un CMS pour Content Management System) permettant à des développeurs (informaticiens, enseignants) d'implémenter et maintenir un EIAH tant du point de vue des fonctions offertes aux usagers (apprenants, tuteurs) que des contenus et activités proposés lors de l'exploitation de l'EIAH. De manière générale,

Verrou n°1

La communauté EIAH n'arrive pas à capitaliser dans les environnements supports les fonctionnalités que l'on trouve dans les meilleurs prototypes de recherche

Face à la diversité et la complexité des technologies à mettre en œuvre pour implémenter un environnement support de formation à distance (aspects réseau, aspects bases de données et de connaissances, aspects d'Interfaces Homme-Machine, ...),

- la plupart des équipes développent d'abord un prototype pour valider un point particulier de leurs recherches, les fonctionnalités intégrées dans ce prototype ayant pour but de constater la faisabilité ou évaluer le potentiel d'un développement. Mais les chercheurs se contentent souvent de ces premiers résultats car la programmation des mêmes fonctionnalités pour un environnement support donné imposerait une refonte technologique complète du prototype pour tenir compte des contraintes d'exploitation imposées (montées en charge, interfaces, contraintes technologiques, ...). Bien des fonctionnalités implantées dans les prototypes ne sont donc jamais re-développées compte tenu des coûts et de la faible valeur ajoutée de ce travail pour le chercheur.
- d'autres équipes s'appuient sur un environnement support de référence tels Explora (Paquette, De la Teja et al. 2000), TecfaSeed (Schneider 2003), Formid (Guéraud, Adam et al. 2004), BSCW (Sikkel, Gommer et al. 2001) qu'ils expérimentent et essaient de faire évoluer en fonction des besoins ; seules les équipes structurées disposant des ressources humaines et savoir-faire technologiques en terme de réseaux, Bases de données, Génie Logiciel, ... peuvent faire un tel choix. Cette orientation est de toute façon risquée car le potentiel d'évolution repose sur les fonctionnalités de base de l'environnement support de référence (donc des choix technologiques initiaux) : comme pour les prototypes, faire migrer des fonctionnalités développées d'un environnement support de référence vers un autre peut nécessiter des développements conséquents (Oubahssi, Grandbastien et al. 2004), (Ngomo, Oubahssi et al. 2005), (Caron, Le Pallec et al. 2005) et ce, malgré les travaux actuels menés par différents consortiums, tel IMS (IMS 2001) ou ADL (SCORM 2001) pour proposer des modèles facilitant l'interopérabilité des ressources et des plates-formes.

La communauté EIAH a donc une certaine incapacité à promouvoir et diffuser de manière réactive de nouveaux outils venant étendre des environnements supports existants. Ceci est très dommageable car cela contribue à laisser croire aux utilisateurs potentiels (enseignants, responsables de formation) que les savoir-faire actuels se limitent encore à des fonctionnalités de tutorat et autres exercices disponibles dans des systèmes comme WebCT ou Blackboard. Par voie de conséquence, les domaines de l'apprentissage à base de situations-problèmes et de l'apprentissage coopératif sont bien peu outillés dans des environnements supports alors que de nombreux prototypes existent (cf paragraphe 1.1.1.1).

Verrou n°2

La communauté EIAH a des difficultés pour évaluer les dispositifs d'apprentissage conçus à partir de ces environnements supports ainsi que les apprentissages réalisés

Les environnements d'apprentissage que nous développons dans le cadre de nos recherches sont des produits souvent inaboutis qui ne sont pas acceptables par les enseignants car la partie back-office permettant de paramétrer les EIAH développés en fonction des besoins (adaptation à une classe donnée, choix des traces à recueillir, ...) n'est pas suffisamment finalisée et conviviale. Le coût de mise en œuvre de ce back-office est important mais les dégâts dus à une absence de back-office le sont également : faute d'être en mesure de mettre entre les mains des enseignants des outils finalisés, nous manquons souvent cruellement de retours d'usage sur les environnements que nous construisons, tout particulièrement sur ceux qui laissent une part de liberté importante aux acteurs impliqués.

1.3.2 Verrous liés aux principes d'ingénierie utilisés

Les deux verrous qui viennent d'être énoncés ont porté sur les environnements supports utilisés pour produire et exploiter des EIAH. Ces verrous ne sont malheureusement que la face émergée de problèmes d'ingénierie plus profonds qui tiennent pour une bonne part à l'incapacité relative de notre communauté à maîtriser la complexité intrinsèque d'un EIAH ainsi que la diversité des acteurs à impliquer dans l'ingénierie d'un tel système.

Verrou n°3	La communauté EIAH a des difficultés pour identifier, mettre en œuvre et faire savoir les bonnes pratiques en matière d'ingénierie
------------	--

L'ingénierie de situations d'apprentissage reposant sur ces méthodes actives pose des problèmes en tant que telle puisque de telles situations s'articulent autour de différents pôles : savoir – professeur – élèves – instruments mis au service de l'activité d'apprentissage. Ces pôles font l'objet de recherches approfondies mais « ils fonctionnent sur le principe du tiers exclu, les modèles qui en naissent sont centrés sur une relation privilégiée entre deux de ces termes » (Houssaye 1988). Ces relations privilégiées correspondent au moins en partie à la dichotomie proposée par T. Murray et P. Tchounikine entre les systèmes orientés Design Pédagogique (privilégiant la relation Professeur - Savoir) et les systèmes orientés Performance (privilégiant plutôt la relation Savoir – Apprenant ou la relation Enseignant – Apprenant) (Murray 1999), (Tchounikine 2002a). Cette diversité des points de vue sur un même objet scientifique est pour l'instant peu prise en compte, les travaux visant souvent à approfondir une de ces dimensions plutôt que de proposer des méthodes et outils pour établir des ponts entre différents points de vue sur un EIAH.

Une autre difficulté semble venir de l'incapacité de la communauté à intégrer dans ses pratiques le caractère évolutif des spécifications en EIAH. Les chercheurs continuent à essayer de capitaliser et faire évoluer le code des prototypes qu'ils développent alors que l'enjeu me semble d'abord de capitaliser et faire évoluer sans rupture les spécifications de ces mêmes prototypes :

- Partager et échanger des modèles représentatifs de telles spécifications me semble aussi important et plus efficace que s'approprier le code de telle ou telle fonctionnalité d'un prototype de recherche pour essayer de l'intégrer dans un autre.
- Mettre en place des procédures pour évaluer la qualité de ces modèles, proposer des principes méthodologiques pour mettre en œuvre des applications exploitant ces modèles, proposer des exemples d'implémentation réussies sur la base de ces modèles, ... constituent également des pistes pour identifier et disséminer les bonnes pratiques d'ingénierie.

Enfin, la conception d'un EIAH est un travail transdisciplinaire (Tchounikine, Baker et al. 2004) mais les langages que les chercheurs utilisent pour décrire et raisonner sur un tel système sont souvent centrés sur une discipline. Les chercheurs en EIAH manquent donc d'un cadre d'analyse, de méthodes et de langages :

- permettant aux divers acteurs impliqués dans l'ingénierie des EIAH de partager la sémantique des modèles et croiser les points de vue représentés par ces modèles (facilitant ainsi le dialogue entre acteurs et la contextualisation des modèles).
- rendant possible l'opérationnalisation des modèles produits à la fois côté Front-Office (apprenants et tuteurs) et côté Back-Office (enseignants, spécialistes du domaine, concepteurs). Il s'agit donc de mettre en correspondance les modèles résultant de l'activité de conception pédagogique avec des modèles exploitables par les plateformes de formation à distance (le langage IMS-LD qui sera présenté dans le prochain chapitre étant un exemple de langage conçu pour être exploité par de telles plateformes).

Il faut enfin noter que les chercheurs ne sont pas en mesure pour l'instant de faire évoluer les fonctionnalités des plateformes de formation à distance en fonction des besoins qui apparaissent dans les modèles de conception décrivant un EIAH ; compte tenu de la complexité des plateformes, il est pour l'instant difficile de générer rapidement des fonctionnalités d'outils permettant aux utilisateurs d'évaluer la pertinence de chaque modèle (et la cohérence de l'ensemble), de paramétrer ces modèles en fonction des besoins constatés lors de l'usage de l'EIAH (analyse de traces ou autres), ...

1.4 Conclusions

J'ai introduit ce chapitre en référence aux dimensions d'analyse des travaux en EIAH proposées dans (Tchounikine, Baker et al. 2004). L'objectif était de situer le type d'EIAH qui a motivé mes actions de recherche ainsi que le cadre théorique en Education dans lequel ce type d'EIAH prend son sens (cf dimensions A et B du document Platon-1).

Après avoir défini le projet, produire des EIAH mettant en œuvre des méthodes pédagogiques actives et situées, j'ai listé des exemples d'EIAH et d'outils démontrant le potentiel des technologies au service des pédagogies actives et situées. J'ai ensuite défini les principes théoriques des méthodes pédagogiques actives ce qui m'a permis de définir un cadre théorique de référence pour la conception d'EIAH au service des méthodes pédagogiques actives et situées : la notion de situation-problème est au cœur de ce cadre théorique. J'ai enfin proposé un certain nombre de questions de recherche posées par la problématique choisie à la communauté EIAH : les verrous scientifiques listés ne sont pas spécifiques aux EIAH mettant en œuvre des méthodes pédagogiques actives et situées, mais se posent avec plus d'acuité, vu la richesse des interactions (et donc la diversité des modèles) qu'imposent de telles méthodes d'apprentissage.

Dans le prochain chapitre, je poursuivrai l'analyse de mon projet de recherche puisque je traiterai de la problématique et de ma démarche pour répondre aux questions de recherche posées en fin de ce chapitre (dimensions A, B et D du document Platon-1 (Tchounikine, Baker et al. 2004)).

Chapitre 2

Problématique et démarche de recherche

Ce chapitre a pour but de positionner mes travaux de recherche en ingénierie des EIAH à base de situations-problèmes et de les situer dans le contexte national et international. Les verrous scientifiques énoncés à la fin du chapitre 1 (inadaptation relative des environnements-support, difficultés de mise en œuvre de pratiques d'ingénierie intégrant la diversité des acteurs, ...) me conduisent à étudier les conditions d'une ingénierie des situations-problèmes coopératives ayant pour finalité l'exploitation de ces situations sur des plates-formes de formation à distance. Deux raisons au moins justifient cette orientation :

- Au-delà des critiques que l'on peut formuler sur les plates-formes de formation à distance disponibles, les chercheurs ne doivent pas ignorer cette orientation actuelle car le développement d'EIAH pour des plates-formes contribue aussi à l'ingénierie des EIAH en essayant de dépasser les développements artisanaux. S'inscrire dans une perspective d'exploitation de situations-problèmes coopératives par des plates-formes, c'est considérer les plates-formes dans ce qu'elles ont de positif (dispositif mûre pouvant être exploité en vraie grandeur par des apprenants et des enseignants) en les ouvrant à d'autres formes d'apprentissage et d'enseignement (via l'ajout de nouvelles fonctionnalités).
- Les plates-formes de formation à distance sont aussi des vecteurs importants pour les industriels (cf formations en aéronautique) et les institutions (cf projets de Campus ouverts). Les enjeux justifient les nombreux travaux de standardisation menés par des consortiums tels IMS, ADL, IEEE, AFNOR. Envisager l'ingénierie de situations-problèmes coopératives pour ces plates-formes est donc un moyen d'étudier les forces et limites des standards proposés et contribue à l'effort de validation/d'amélioration de ces standards.

Dans ce chapitre, je traite tout d'abord des environnements-supports mis au service des situations-problèmes coopératives. J'examine les conditions de leur mise en place, les travaux menés actuellement, avant de définir les orientations prises au laboratoire LIUPPA. La deuxième partie de ce chapitre porte sur les modèles et langages de description de situations-problèmes coopératives : les standards utilisés pour modéliser des situations d'apprentissage sur des plates-formes de formation à distance sont tout d'abord décrits avant d'examiner d'autres modèles qui paraissent très importants pour prendre en compte les situations-problèmes coopératives : cette analyse me conduit en fin de chapitre à proposer les orientations prises au laboratoire LIUPPA pour la modélisation des situations-problèmes coopératives.

2.1 Vers des environnements-supports au service des situations-problèmes coopératives

Comme le chapitre 1 l'a montré, il existe de nombreux environnements informatiques supportant l'enseignement et l'apprentissage mais peu d'entre eux proposent une approche centrée apprenant ; la plupart des plates-formes de formation à distance proposent une approche centrée enseignant qui permet aux apprenants d'accéder à des informations en ligne qu'ils parcourent selon un ordonnancement de l'information plus ou moins prédéfini. D'autres systèmes développés souvent de manière plus artisanale utilisent les technologies issues de l'intelligence artificielle pour orienter plus individuellement les apprenants via des outils (agents, systèmes tuteurs intelligents) capables de simuler le comportement d'enseignants ou d'experts (cf les exemples du paragraphe 1.1.1.1).

Au cours des années 90 sont apparus des dispositifs informatiques issus du CSCL/CSCW supportant l'apprentissage coopératif/collaboratif à base de situations-problèmes via une approche centrée sur l'activité de l'apprenant : CCL (Koschmann, Feltovich et al. 1990), (Koschmann, Feltovich et al. 1992), CSILE (Scardamalia, Bereiter et al. 1994), CALE (Mahling, Sorrows et al. 1995), CNB (Edelson, O'Neill et al. 1995), Belvedere (Suthers et Weiner 1995), (Suthers 1997), McBagel (Guzdial, Kolodner et al. 1996), Web-SMILE (Guzdial, Hmelo et al. 1997), CROCODILE (Miao, Holst et al. 2000), STEP (Steinkuehler, Derry et al. 2002), COMET (Soller 2002), Lab@Future (Mwanza et Engeström 2003), COLER (De los Angeles et Suthers 2003), COLDEX (Baloian, Breuer et al. 2004).

Plusieurs études ont porté sur ces environnements notamment celles de (Naidu et Olliver 1996), (Albion 2000), (Roschelle, Pea et al. 2000), (Steinkuehler, Derry et al. 2002). Partant d'une synthèse de ces travaux, je propose dans le prochain paragraphe d'établir le cahier des charges d'un environnement-support susceptible d'exploiter des situations-problèmes coopératives.

2.1.1 Cahier des charges

Une plate-forme de formation à distance devrait offrir quatre grands types de fonctionnalités aux acteurs engagés dans une situation-problème coopérative (Miao 2000) :

1. Fournir un support permettant aux acteurs de percevoir l'environnement distribué dans lequel ils sont engagés et d'interagir en tant que groupe (connaître à tout moment son rôle au sein du groupe et le rôle des autres, pouvoir échanger avec les autres de manière adaptée à la situation de communication rencontrée, pouvoir personnaliser son environnement de travail),
2. Fournir un support pour faciliter la compréhension mutuelle entre acteurs et la construction de connaissances partagées (possibilité d'exprimer et de mettre en relation différents types d'idées, possibilité de négocier le statut des connaissances partagées),
3. Fournir un support pour guider les apprenants en fonction de leur rôle, pour faciliter l'allocation et le changement de rôles (définition de rôles, guidage basé sur des règles de collaborations associées à chaque rôle défini),
4. Fournir un support pour faciliter l'apprentissage autodirigé (possibilité pour les acteurs de définir leurs propres plans d'action en allouant des ressources aux actions définies puis en contrôlant leur exécution).

Dans sa thèse, Y. Miao a détaillé ces grands types de fonctionnalités en fonctionnalités plus élémentaires qui lui ont ensuite permis de comparer différents environnements supportant des situations-problèmes coopératives, tels CCL, CSILE, Belvedere, Web-SMILE, ... (Miao 2000). De manière générale, Y. Miao constate qu'aucun environnement ne répond à l'ensemble de ces critères, le type de fonctionnalité le mieux pris en compte correspondant au point 2 : fournir un support pour faciliter la compréhension mutuelle entre acteurs et la construction de connaissances partagées.

Compte tenu de l'état actuel des connaissances et de l'ampleur de la tâche, il est sans doute trop tôt pour s'engager dans le développement d'une plate-forme de formation à distance qui disposerait de toutes ces fonctionnalités puisque même les prototypes les plus aboutis dans le domaine des situations-problèmes coopératives ne couvrent pas l'ensemble du cahier des charges proposé. En revanche, il est important de disposer de tels critères pour envisager l'évolution souhaitable des plates-formes de formation à distance actuelles. Les prochains paragraphes dressent un état des lieux des travaux visant à faire évoluer les capacités de ces plates-formes.

2.1.2 Travaux menés au niveau national et international pour promouvoir les environnements supportant l'apprentissage à base de situations-problèmes coopératives

2.1.2.1 Un point de départ : les plates-formes de FOAD

Les plates-formes de formation à distance (ou Learning Management Systems) participent à l'effort international de diffusion des savoirs et de formation des personnes. En tant que :

- « dispositif organisé, finalisé, reconnu comme tel par ses acteurs,
- prenant en compte la singularité des personnes dans leurs dimensions individuelle et collective,
- reposant sur des situations d'apprentissage complémentaires et plurielles en termes de temps, de lieux, de médiations pédagogiques humaines et technologiques, de ressources » (FFFOD 2002),

une plate-forme de formation à distance est un type d'environnement à considérer pour l'apprentissage à base de situations-problèmes coopératives. Le marché des plates-formes est cependant très fragmenté (en terme de fonctionnalités d'architectures logicielles, de formats de données ...) et il bénéficie actuellement des travaux structurés de nombreux organismes internationaux spécialisés dans le domaine. Ces organismes proposent des normes et standards qui visent à répondre aux questions relatives à l'interopérabilité des technologies et des contenus entre plates-formes de formation à distance.

Des organismes tels IEEE LTSC, IMS, ADL, ... proposent donc des solutions technologiques pour garantir (MASIE 2003) :

- L'interopérabilité entre contenus manipulés/échangés par différentes plates-formes,
- La réutilisation afin de faciliter l'assemblage rapide de contenus préexistants offerts par une ou plusieurs plates-formes,
- L'adaptabilité afin de pouvoir configurer une plate-forme et la doter de fonctionnalités visant de nouveaux buts.

Les propositions de standards vont de la description des méta-données éducatives (cf Dublin Core, LOM) aux architectures des plates-formes (cf architecture IEEE LTSA ou architecture OKI), en passant par la description du modèle de l'apprenant (cf les spécifications IEEE PAPI, IMS Learner information Profile et modèle GESTALT), la description des contenus de formation (cf IMS Content Packaging et ADL SCORM) et des compétences (cf IMS Reusable Competency Definition Specification), la scénarisation et le séquençement des activités (cf IMS Learning Design et IMS Simple Sequencing Specification), l'évaluation des apprentissages (cf IMS Question & Test Specification), ...

Cependant, l'ensemble de ces propositions correspond à une approche centrée contenus qui vise plus à pousser des supports de formation vers l'apprenant qu'à proposer des activités engageant l'apprenant dans des apprentissages. Des consortiums comme ADL et IMS ont senti les limites des standards proposés actuellement et s'intéressent maintenant à l'exploitation/l'évolution de leurs modèles pour permettre : a/ des apprentissages basés sur la construction par l'apprenant de ses savoirs (Hummel, Manderveld et al. 2004), b/ une plus grande personnalisation des contenus dispensés en fonction des compétences et comportements constatés de l'apprenant/des groupes d'apprenants (Koper, Giesbers et al. 2003), (ADL 2004). Il est difficile de savoir ce que donneront ces travaux qui ne font que

démarrer et sur lesquels les chercheurs n'ont que peu de prise, vu les enjeux économiques. On peut en tout cas s'interroger sur le processus engagé qui consiste à rajouter des fonctionnalités relevant d'une approche constructiviste de l'apprentissage à des systèmes pensés initialement dans une approche béhavioriste, et donc à prétendre couvrir avec une même plate-forme de formation à distance des modèles théoriques aussi différents (Paraskevi 2002), (Nodenot, Marquesuzaà et al. 2002), (Nodenot, Marquesuzaà et al. 2004).

2.1.2.2 Une autre approche possible : les portails et plates-formes de travail coopératif

Une autre façon d'aborder l'avenir des plates-formes de formation à distance est d'examiner les fonctionnalités que développent les équipes de recherche spécialisées dans les environnements collaboratifs dédiés aux situations-problèmes. Des travaux très intéressants sont menés dans les quatre directions énoncées au paragraphe 2.1.1.

En ce qui concerne les outils proposant un support permettant aux acteurs de percevoir l'environnement distribué dans lequel ils sont engagés et d'interagir en tant que groupe, les travaux récents proposent différents types de fonctionnalités :

- Des outils permettant à l'apprenant d'explorer l'environnement distribué dans lequel il est impliqué afin de pouvoir suivre (prendre conscience) et comprendre les activités menées par les autres acteurs, de pouvoir relier sa propre activité avec les leurs (Scardamalia et Bereiter 2003).
- Des indicateurs visuels rendant compte de la participation des acteurs à la fois côté apprenant et côté enseignant (Barros et Verdejo 2000), (Jermann 2002), (Scardamalia et Bereiter 2003), (De los Angeles et Suthers 2003) qui aident les apprenants à évaluer la façon dont ils collaborent et l'enseignant (tuteur) à évaluer l'interaction au niveau du groupe,
- Des éléments d'interface pour maintenir la motivation et l'engagement des apprenants dans les activités coopératives, pour tracer leur activité cognitive : boutons structurant les échanges et correspondant à des actes de dialogue simples, analyse du discours marquant l'accord, le désaccord, les propositions, ... (Baker et Lund 1997), (George 2001), (Baker, De Vries et al. 2001), (De los Angeles et Suthers 2003), (Delium 2003).

Pour ce qui est des outils facilitant la compréhension mutuelle entre acteurs et la construction de connaissances partagées, de nombreux travaux ont été conduits qui ont permis de proposer :

- Des outils pour partager et négocier des représentations entre apprenants et pour synthétiser la connaissance partagée par le groupe ; certains outils permettent à l'apprenant de choisir parmi différentes formes de représentation de la connaissance partagée et de manipuler ces représentations dans différents contextes d'interaction (Scardamalia et Bereiter 2003).
- Des outils orientés collaboration permettant par exemple de publier (dans un espace public) une production individuelle afin d'alimenter les discussions au niveau du groupe (Guzdial, Hmelo et al. 1997). Les discussions (chat, news, ...) sont donc souvent contextualisées (ex : résolution de conflits) par rapport aux productions disponibles dans les espaces privé et partagé des acteurs (De los Angeles et Suthers 2003), (Betbeder et Tchounikine 2004).
- Des outils permettant de citer et référencer le travail d'un autre pour que l'évolution des idées soit plus évidente pour le groupe et puisse devenir un objet de discussion (Scardamalia et Bereiter 2003).
- Des outils permettant de mémoriser une résolution de problème (évolution des représentations, interactions) afin de pouvoir bénéficier ensuite de l'expérience acquise. (Kolodner, Hmelo et al. 1996) et (Brusilovsky 1999) ont ainsi proposé et développé des outils d'apprentissage à partir de cas et d'exemples.
- Des outils permettant de proposer une assistance aux apprenants sur la base d'une représentation computationnelle du domaine de la situation-problème. Ces outils souvent issus des travaux menés autour des ITS (cf systèmes « coach ») permettent de mieux

comprendre les solutions proposées par les apprenants et de les comparer à celles proposées par des experts du domaine ce qui permet de commenter et proposer des améliorations aux solutions proposées par les apprenants (Guzdial, Hmelo et al. 1997), (Suthers 2003), (De los Angeles et Suthers 2003).

Concernant les outils facilitant le guidage des apprenants en fonction de leur rôle et l'allocation (dynamique) des rôles aux acteurs impliqués, les propositions d'outils sont moins nombreuses. Les travaux portent pour l'instant sur la formalisation de différents protocoles de résolution de situations-problèmes et des rôles associés. Un outil comme CROCODILE (Miao, Holst et al. 2000) permet ensuite aux apprenants de consulter ces protocoles, d'en choisir un ce qui permet ensuite une assistance personnalisée. De manière générale, le but est donc de proposer des modèles de rôles spécifiques, de pouvoir associer dynamiquement des personnes à des rôles en s'appuyant sur les contributions et les buts partagés par les acteurs pour guider la résolution de la situation-problème (Jermann 2002), (Zumbach et Reimann 2003), (Scardamalia et Bereiter 2003).

Enfin, le dernier type de fonctionnalité important pour les situations-problèmes concerne la définition d'outils facilitant l'apprentissage autodirigé. Dans le cadre de la résolution d'une situation-problème, les apprenants ont pour responsabilité de définir les buts d'apprentissage, de proposer des plans d'action, de chercher et exploiter des ressources d'apprentissage utiles. Il est donc important qu'ils disposent d'outils leur permettant de définir et mettre en œuvre ces plans d'actions (et leurs ressources associées). On peut citer :

- des outils orientés création et gestion de processus (workflow) qui sont mis en œuvre par les apprenants pour définir, exécuter leurs plans d'action (Guzdial, Hmelo et al. 1997),
- des outils métacognitifs pour analyser les actions engagées par rapport aux plans définis et adapter ces plans d'action aux contraintes de la situation-problème à résoudre (Lin, Hmelo et al. 1999), (Harrer, Bollen et al. 2004).

Comme le note (Guzdial, Hmelo et al. 1997), le maître mot est l'intégration réussie de tout ou partie de ces fonctionnalités au service d'une situation d'apprentissage donnée : cette intégration peut passer par la proposition de modèles généraux sur lesquels sont développés les outils permettant la perception, la gestion de l'activité mais aussi le soutien à l'apprenant (cf le modèle SAAD (Després 2001)). Le mot intégration recouvre différentes acceptions suivant le point de vue où l'on se place : l'enseignant et les apprenants recherchent des outils intégrés c'est-à-dire pouvant être utilisés de manière naturelle dans différents contextes et différentes situations-problèmes ; il est par exemple assez facile pour les informaticiens de créer un nouvel outil chaque fois qu'un nouveau besoin de coopération apparaît. Il est plus difficile mais plus pertinent d'enrichir des outils existants pour les ouvrir à d'autres usages utiles que les apprenants et enseignants pourront exploiter si nécessaire : il ne s'agit cependant pas de développer des outils toujours plus riches et donc plus complexes à utiliser mais de proposer des outils offrant une palette d'usages plus importante ce qui contribue à « l'enactment » des acteurs, qui vont pouvoir s'approprier les outils en adaptant les usages à l'activité concrète de résolution de la situation-problème.

Les développeurs d'environnements-supports pour les situations-problèmes prennent en compte cette dimension de l'intégration orientée « Utilisateur » mais sont en plus confrontés à d'autres acceptions de ce terme : l'intégration des données, l'intégration fonctionnelle (partie métier), l'intégration de la présentation (interfaces permettant d'accéder aux fonctionnalités des outils) ne sont pas des problèmes spécifiques des environnements d'apprentissage bien que la communauté EIAH apporte parfois ses propres solutions.

Certains travaux consistent à généraliser un environnement d'outils construit initialement comme support d'une situation-problème particulière. C'est le cas de Belvedere (Suthers 1997) qui est devenu une plate-forme d'apprentissage coopératif à partir de laquelle a été implémenté le système COLER (COLlaborative Learning Environment for Entity-Relationship modelling) ; COLER permet de guider à la manière d'un ITS la collaboration entre apprenants (De los Angeles et Suthers 2003).

Une approche plus répandue consiste à spécialiser des environnements offrant des fonctionnalités de base utiles pour supporter les activités de travail coopératif et la diffusion d'informations : identifiés en général sous l'acronyme de CMS (pour Content Management Systems) (Lahaye 2004) ou parfois de C3MS (Community, Content and Collaboration Management Systems) (Schneider 2003), ces environnements qui ont été présentés dans le chapitre 1 (cf paragraphe 1.1.1.2 et figure 1.2) reposent souvent sur des produits OpenSource mûres, par exemple Zope, postNuke, PNphpBB, ..., et sont développés pour la plupart sur des architectures n-Tiers (ce qui facilite le travail d'adaptation aux besoins d'une situation d'apprentissage donnée). Des environnements tels IPAL (Pinkwart, Harrer et al. 2004) ou Tecfa-SEED (Synteta 2002), (Schneider 2003) sont donc des sortes de portails riches (Dillenbourg, Schneider et al. 2002) que les développeurs enrichissent en fonction des besoins des situations d'apprentissage à exploiter. Dans le cadre de l'apprentissage à base de projets, P. Synteta a ainsi proposé différents outils (outil de Quizz, extension ePBL) pluggés sur le portail Tecfa-SEED, M. Chakroun a pour sa part implanté le scénario Argue&graph (Jermann et Dillenbourg 1999), (Chakroun 2003). P. Brusilovsky propose une approche similaire consistant à développer/étendre un CMS en tant que portail supportant des interactions sociales pour guider à la manière des tuteurs intelligents les activités menées par des apprenants (Brusilovsky 2003) : dans le cadre de la résolution de problèmes de programmation centrés sur le langage LISP, le portail KnowledgeTree offre ainsi un point d'accès unique aux acteurs (apprenants et concepteurs de situations d'apprentissage) qui est capable d'interroger différents serveurs d'activités distribués sur le Web en fonction des activités à réaliser (Brusilovsky 2004) :

- le serveur ELM-ART pour générer des problèmes centrés sur le langage LISP, vérifier les solutions proposées et accompagner les apprenants dans l'amélioration de leurs solutions,
- le serveur CUMULATE pour disposer d'un modèle de l'apprenant dans un contexte d'environnement distribué. Sur la base des événements significatifs détectés quand l'apprenant interagit avec d'autres serveurs, CUMULATE est capable de centraliser ces événements, de les exploiter pour faire évoluer la représentation du modèle de chaque apprenant identifié et utiliser ce modèle pour guider l'apprenant dans ses apprentissages,
- des serveurs plus classiques tels QUIZZGUIDE et QUIZZPACK capables de proposer de simples supports de cours ou des exercices comme le feraient des plates-formes de formation à distance,

La plate-forme KnowledgeTree (cf <http://www.sis.pitt.edu/~taler/KnowledgeTree.html>) est en exploitation et contribue à démontrer qu'une architecture de type portail permet de personnaliser la formation des apprenants (via des serveurs adaptés) ce qui constitue donc une alternative crédible aux plates-formes de formation à distance de type tout en un.

2.1.2.3 Des plates-formes de travail coopératif aux plates-formes de FOAD à base de composants

Finalement assez proches d'un portail au sens de D. Schneider et P. Brusilovsky, les plates-formes à base de composants se veulent également ouvertes et flexibles ; les objectifs visés sont d'appliquer les savoir-faire technologiques (EJB, Web Services par exemple) et fonctionnels (structuration des applications en couches logicielles) pour étendre la palette de services offerte par les plates-formes de formation à distance (Schneemayer 2002). CROCODILE (Miao 2000) est un exemple de plate-forme développée pour la seule mise en œuvre de situations-problèmes coopératives. Elle offre quatre catégories de services mis à disposition des apprenants par un ensemble de composants logiciels :

- Les services de base intégrés dans le module Virtual Institute exploitent assez classiquement des composants « Agent », « Place », « Tool » et « Document ».
- Le module PBL-net offre principalement un composant « Language » qui fournit les opérations de base pour que les apprenants représentent, consultent et négocient des connaissances partagées,

- Le module PBL-protocol propose le composant « Rule » qui spécifie (et fait respecter) les conditions selon lesquelles un acteur va pouvoir, en fonction de son rôle, agir sur les représentations du module PBL-net.
- Enfin le module PBL-plan propose les composants « Action » et « Work-description » qui permettent à l'apprenant / au groupe de définir ses buts et ses plans d'actions, puis de les mettre en œuvre.

Les composants proposés doivent se plugger dans une infrastructure support pour pouvoir être exploités. Certains travaux sur les composants éducatifs ont proposé une intégration par le langage, les composants sont alors stockés sur un serveur et s'exécutent côté client sous forme d'applets java par exemple (cf projet ESCOT, projet E-Slate et projet Exploratories, projet Formid). Pour favoriser l'intégration et l'interopérabilité des composants aussi bien au niveau fonctionnel qu'au niveau des interfaces-usagers, deux propositions concurrentes ont émergé dans le cadre de projets ambitieux menés aux Etats-Unis (projet OKI) et en Europe (projet CampusSource) : l'infrastructure OKI (Thorne, Shubert et al. 2002) sur laquelle est bâtie la plate-forme SAKAI (Counterman, Golden et al. 2004) et l'infrastructure EJOSA (Dewanto 2004) sur laquelle est bâtie la plate-forme OpenUSS (Dewanto 2002), (Grob, Bensberg et al. 2004). Ces infrastructures sont finalement assez proches conceptuellement :

- Elles s'appuient sur des technologies java basées sur des architectures multi-tiers qui séparent le niveau présentation du niveau fonctionnel (règles métier) et du niveau des données persistantes.
- Elles identifient deux types de composants : a/ des composants de fondation offrant les fonctionnalités de base de toute plate-forme de formation à distance (composant d'authentification des usagers, composant d'administration des formations, composant de gestion des contenus de formation, ...) et b/ des composants que l'on peut ajouter (« pluggables ») en fonction des besoins de la situation d'apprentissage à mettre en œuvre (divers composants de coopération, composants de guidage de l'apprenant, composants exercices, ...).

Des plates-formes comme SAKAI ou OpenUSS sont donc conçues pour être évolutives et supporter les composants que les développeurs souhaitent mettre en œuvre au service d'une situation d'apprentissage donnée. Les réelles capacités d'interopérabilité entre composants qu'offrent ces infrastructures vont cependant de pair avec la difficulté de compréhension et d'exploitation des « frameworks⁶ » sur lesquels ces composants doivent être construits pour être interopérables ; cette approche à base de composants a cependant tendance à se généraliser (Roschelle, DiGiano et al. 1999), (Repenning, Ioannidou et al. 2001) car les techniques sont maintenant mieux maîtrisées par les équipes de développeurs et les chercheurs en EIAH : (Vantroys 2003) a proposé des composants de gestion des processus d'apprentissage (« learnflows ») développés pour l'infrastructure J2EE, (Caron 2004) propose un ensemble de composants MDA simples pour le e-learning, des composants capables d'exploiter des modules SCORM sont maintenant disponibles pour plusieurs plates-formes telles ATutor et OpenUSS, ...

2.1.3 Approche de la problématique des environnements-supports au LIUPPA

Nous avons expérimenté au LIUPPA les trois types d'environnements supports présentés au paragraphe précédent dans le cadre de projets académiques et industriels de mise en œuvre de situations d'apprentissage. La plupart de ces situations étaient des situations-problèmes coopératives mais nous avons aussi implanté des situations d'apprentissage plus

⁶ Le développement de composants pour la plate-forme OpenUss est basé sur le Framework EJOSA (Enterprise Java Open Source Architecture) proposé par (Dewanto 2004) pour exploiter au mieux l'architecture J2EE et le modèle de composant EJB (Brethes, Hisquin et al. 2000). Le développement de composants pour la plate-forme SAKAI est basé sur des APIs dont les spécifications respectent l'architecture à base de services de la plate-forme SAKAI (Counterman, Golden et al. 2004).

classiques dans le cadre de contrats avec des entreprises. Partant d'une description des projets menés (contexte et historique), ce paragraphe explicite l'approche de cette problématique des environnements-supports au LIUPPA.

2.1.3.1 Travaux sur les plates-formes de FOAD

Depuis 1998, nous suivons l'évolution des plates-formes de FOAD, ce qui nous a d'abord conduit à exploiter et évaluer plusieurs d'entre elles dans le cadre de projets menés en collaboration avec des enseignants-chercheurs de différentes universités européennes (Faculté d'Informatique de San Sebastian, Université de Brighton, Université de Gran Canaria) et d'Amérique Latine (Instituto Varona de la Havane, CUJAE). Fort de cette expérience, nous avons participé entre 2000 et 2002, à un gros projet industriel piloté par le GRETA Pays Basque ; ce projet consistait à produire 180 heures d'enseignement du BTS audiovisuel pour la plate-forme Academynet recommandée par le CAFOC du rectorat de Bordeaux. Ces développements ont été menés suite à une commande passée au GRETA Pays-Basque par les grandes chaînes de télévision française (Canal+, France Télévision, TF1) ainsi que par l'Union Européenne de Radiodiffusion (UER). Cette expérience d'un développement industriel de modules de formation à distance nous a permis de travailler avec une équipe structurée d'enseignants, de scénaristes et développeurs recrutés au titre du projet, ce qui a été très enrichissant pour nous forger une connaissance des capacités et limites des plates-formes commerciales de type tout en un. Cette expérience a également été utile d'un point de vue méthodologique car nous avons dû travailler sous contraintes techniques (de la plate-forme WBT Manager -cf <http://www.x-perteam.com/produits/wbt/>- utilisée pour ce projet, du standard AICC choisi pour tracer les parcours des utilisateurs sur la plate-forme) et humaines (cycles de développement courts obligeant l'équipe de développeurs à formaliser les procédures de développement, nécessité de dialogue au sein de l'équipe pluridisciplinaire impliquée dans le développement, obligation de mise en place d'outils méthodologiques et logiciels permettant d'obtenir des retours d'usage sur les modules d'audiovisuel développés).

2.1.3.2 Travaux utilisant des plates-formes pour le travail coopératif

Nos travaux sur les situations-problèmes coopératives nous ont par ailleurs conduits, entre 2001 et 2003, à développer des outils de travail coopératif au service des situations-problèmes ; dans le cadre de la mise en œuvre de la situation-problème Smash (qui sera présentée plus en détail dans le paragraphe 3.1.2.1), nous avons ainsi développé des outils mêlant des fonctionnalités de "tableau blanc" et de "chat" qui visaient à faire apparaître, expliciter et réguler les conflits potentiels entre les représentations d'apprenants en interaction. Compte tenu des spécifications fonctionnelles de ces outils, il était impossible de s'appuyer sur une plate-forme de formation à distance pour les développer et nous avons donc fait appel à des infrastructures supportant le travail coopératif. Dans un premier temps, nous avons utilisé l'outil Flash Communication Server (Manapragada 2004) pour développer nos composants éducatifs et gérer les flux d'informations partagés (Goyhenex et Capliez 2004).

Dans un deuxième temps, nous avons eu recours aux technologies java, J2EE (Servlet et EJB) et XML en appuyant le développement de nos composants éducatifs sur la plate-forme Laszlo (Grosso 2005). Dans les deux cas, nous avons pu obtenir des composants de coopération fonctionnels, la solution Laszlo présentant l'avantage d'être non propriétaire et OpenSource. Nous avons cependant constaté que l'intégration de ces composants dans un module de formation n'est pas chose aisée tout particulièrement avec des plates-formes dont l'architecture n'est pas pensée en termes de composants à interconnecter : l'intégration que nous avons faite du composant Tableau blanc Smash dans le plate-forme Moodle est minimale car bien que fonctionnel, l'outil tableau blanc n'est pas capable d'interopérer au niveau de la couche Métier avec les autres fonctionnalités de Moodle. Comme le prochain paragraphe le montrera, il est possible de dépasser ce niveau d'intégration si l'on fait

interopérer ce même composant avec une plate-forme de formation à distance basée composants.

2.1.3.3 Travaux sur les plates-formes de FOAD à base de composants

Depuis 2002, je suis de près les développements menés autour de la plate-forme OpenUSS. Cette plate-forme est conçue et développée dans le cadre du programme allemand CampusSource (cf <http://www.campussource.de/org/about/>) qui fédère depuis 2000 de nombreux projets ayant pour ambition de produire des environnements et outils OpenSource pour le elearning. Afin de pouvoir supporter les montées en charge (13000 utilisateurs à l'Université de Duisbourg et 1000 connexions d'étudiants simultanées certains jours), OpenUSS est construit à partir de produits OpenSource mûres : Jonas qui est un container EJB développé par le consortium ObjectWeb (<http://jonas.objectweb.org/>) et Enhydra (cf <http://enhydra.objectweb.org/>) qui est un serveur d'application capable de générer des pages dynamiques à partir de documents via des outils comme XMLC (compilateur XML) et Barracuda (framework MVC) : ces outils permettent de séparer la partie applicative (en Java) du contenu Web (généré en XML) ce qui évite d'inclure du code java dans des fichiers HTML ou XML (cf Technologie JSP) et se traduit par une gestion plus facile des applications générées.

Par rapport aux autres plates-formes OpenSource, OpenUSS fait d'abord un peu peur car ses fonctionnalités sont (pour l'instant) à peu près équivalentes à ce que savent faire les autres plates-formes alors que les efforts nécessaires pour installer et exploiter OpenUSS sont incroyablement plus importants (l'architecture à base d'EJB complexifiant, au premier abord, la compréhension de l'application). Après une première installation réussie de la plate-forme et quelques expérimentations basiques, j'ai d'abord suivi de loin l'évolution de la plate-forme au travers de la liste de diffusion à laquelle j'étais abonné. J'ai très vite été surpris par la réactivité de l'équipe de développement, pourtant peu nombreuse, à solutionner les bugs détectés et surtout aux demandes de nouvelles fonctionnalités par les utilisateurs. Assez rapidement, des développeurs se sont manifestés pour proposer tel ou tel outil à intégrer dans la plate-forme (composant BabylonChat, composant SCORM, composant Openvc pour le travail coopératif, composant WAP, ...) et en seulement quelques jours, B. Dewanto et son équipe parvenaient à intégrer ces fonctionnalités en tant qu'outils de la plate-forme.

Suite à de nombreux échanges avec l'équipe de développeurs qui faisait évoluer les fonctionnalités de cette plate-forme à base de composants, j'ai participé aux activités de l'équipe visant à localiser l'interface d'OpenUSS pour des utilisateurs français, puis j'ai initié un développement visant à intégrer les capacités de travail coopératif offertes par Laszlo dans la plate-forme OpenUSS. Ce projet validé par L Dewanto (chef du projet OpenUss) ne fait que commencer et vise à réaliser l'intégration de notre composant Tableau blanc dans un module de formation à la sécurité routière (cf situation Smash) exploitant également les composants de la plate-forme OpenUSS. Ce travail sur le développement et l'interopérabilité entre composants éducatifs constitue une poursuite des développement mis en œuvre dans le cadre d'un projet⁷ soutenu par le Conseil Régional d'Aquitaine, le niveau d'interopérabilité cherché se situant cette fois au niveau fonctionnel.

2.1.3.4 Bilan de ces travaux et orientations de recherche

Ces expériences successives ont conforté l'équipe IDEE dans son orientation Génie Logiciel ; cette orientation conduit à privilégier les plates-formes à base de composants pour réaliser des environnements-supports adaptés aux situations-problèmes coopératives imaginées par les concepteurs. Mais ces expériences nous ont aussi montré que les techniques et outils du Génie Logiciel ne sont qu'une pierre à l'édifice visant à généraliser la

⁷ Projet CRA n°20020204006 et n°20030204009 sur le thème : Modélisation et mise en oeuvre de composants d'apprentissage coopératif pour la plate-forme J2EE (Java2 Enterprise Edition).

production et l'exploitation de composants en EIAH. Aux problèmes que sait traiter le Génie Logiciel se superposent en EIAH des problèmes d'un autre ordre :

- La communauté EIAH maîtrise encore assez mal les problèmes particuliers d'ingénierie des logiciels éducatifs (problèmes d'ordre pédagogique, d'évaluation et de mise en œuvre dans la classe) ce qui rend difficile une analyse en terme de composants.
- la production par/pour la communauté éducative diffère beaucoup de ce qui se fait dans d'autres domaines. Les problèmes sociaux (dialogue difficile entre développeurs et chercheurs en Sciences Humaines et Sociales, en Informatique, en ergonomie), la complexité des plates-formes à produire, le faible financement et les durées de développement excluent l'utilisation de bien des pratiques de Génie Logiciel, en particulier la mise en place d'équipes chargées de mettre en œuvre la réutilisation en plus des équipes de développeurs proprement dites. Même si des composants éducatifs sont créés par une équipe pour un certain domaine (cf par exemple les composants centrés sur la production de simulations scientifiques développés avec l'environnement AgentSheets), les problèmes d'interopérabilité avec d'autres composants issus d'autres équipes en EIAH (ou d'interopérabilité avec des composants COTS d'autres domaines) restent importants car ces composants ont rarement la même granularité (DiGiano, Yarnall et al. 2002), (Asencio, Dimitriadis et al. 2004).

Aussi, même si la prise en compte des connaissances en Génie Logiciel par les équipes en EIAH est meilleure (et continue à s'améliorer), il est peu probable qu'une masse critique de composants éducatifs interopérables puisse voir le jour sans des efforts coordonnés de la communauté EIAH et de la communauté Génie Logiciel. Le fait que le Département STIC-CNRS (RTP39) ait engagé les équipes françaises (dont le LIUPPA) à mener une Action Spécifique « Plate-forme pour la Recherche en EIAH » démontre l'importance de cette problématique du point de vue recherche fondamentale (définition de la notion de composant éducatif, problèmes de granularité des composants, problèmes de perception pédagogique/technique d'un composant) et du point de vue pratique (référencement des bonnes pratiques, expérimentation et usage de composants à l'échelle nationale, ...).

Dans ce domaine, l'équipe IDEE du LIUPPA est engagée à part entière dans les travaux actuels de la communauté française. Personnellement, j'ai participé :

- à l'ensemble des travaux et à la rédaction des rapports et publications scientifiques (ASPF 2005) liés à l'Action Spécifique « Plate-forme pour la Recherche en EIAH » ;
- à l'organisation, à la préparation de deux cours que j'ai ensuite dispensés lors de l'école d'été 2005 du RTP39 sur le thème « Modèles, Architectures Logicielles et Normes pour le Développement et l'Intégration des EIAH »⁸.

Par rapport aux travaux de la communauté sur les environnements-supports aux situations d'apprentissage, les travaux spécifiques que j'ai menés ou encadrés depuis 1995 (deux doctorats à l'UPPA) considèrent les effets réciproques de la technologie sur les modèles de conception et des modèles sur l'évolution technologique des plates-formes ; trois orientations permettent de caractériser ces travaux :

1. Les plates-formes à base de composants offrent d'ores et déjà les services de base (dits de fondation) pour la formation ouverte et à distance ; leurs spécifications détaillent des mécanismes d'extension par lesquels de nouveaux composants vont pouvoir être créés et interopérer avec les composants de base de ces plates-formes. Compte tenu de la solidité de leurs fondations techniques, nous considérons donc que ces plates-formes de FOAD à base de composants constituent le meilleur point de départ pour mettre en œuvre des situations-problèmes coopératives, et ce même si ces plates-formes n'offrent pas pour l'instant de composants éducatifs à vocation socio-constructivistes. Notre démarche n'est donc pas de développer de nouvelles plates-formes qui répondraient spécifiquement aux besoins des situations-problèmes coopératives, mais nous

⁸ Pour cette école d'été, j'étais responsable du contenu pédagogique des cours « Ingénierie de modèles en EIAH » et « Questions vives de recherche » (voir http://www-rtp39.imag.fr/Autrans_2005/Present.html).

cherchons plutôt à spécialiser, lors de la conception d'une situation d'apprentissage particulière, l'environnement de travail à offrir aux apprenants et aux enseignants par ces plates-formes à base de composants. Cela suppose :

- d'ancrer / contextualiser la partie métier des composants existants par rapport aux besoins pédagogiques exprimés par l'équipe de conception,
 - de faire apparaître les spécifications de nouveaux composants (et leur contexte d'exploitation) au cours de la phase de conception des situations d'apprentissage.
2. Les premières plates-formes à base de composants sont apparues très récemment, nous n'avons que peu de recul sur leurs usages. Il est donc difficile pour l'instant de comparer les potentialités de plates-formes OpenSource différentes comme OpenUSS ou SAKAI et donc difficile de choisir entre différentes mises en œuvre d'un même ensemble de services. Face à cette difficulté, notre démarche consiste à considérer des abstractions des composants offerts par ces plates-formes qu'il s'agit de rapprocher des besoins pédagogiques qui vont apparaître lors de la spécification / de la conception d'une situation-problème coopérative. Cette démarche également choisie par (Renaux, Caron et al. 2005) et (Dewanto 2005) consiste à raisonner sur des situations concrètes modélisées aussi complètement que possible du point de vue pédagogique (modèles du domaine, modèles de l'apprenant, scénario d'apprentissage, modèles d'interaction, ...). Il s'agit :
- de mettre en correspondance ces modèles de la situation d'apprentissage avec des modèles abstraits des composants offerts par les plates-formes et avec des modèles abstraits de composants éducatifs à développer spécifiquement,
 - de transformer les modèles abstraits en modèles concrets pour une plate-forme donnée via des techniques de transformation de modèles tenant compte des aspects technologiques de la plate-forme choisie (Dewanto 2005). Cette transformation peut consister : a/ à spécialiser ou produire des spécifications détaillées de composants logiciels qui vont venir enrichir une plate-forme donnée, b/ à transformer les modèles issus de la conception pédagogique en une représentation exploitable par une plate-forme donnée (les représentations à privilégier devant tenir compte des propositions sur lesquelles des travaux de standardisation sont menés).
3. Les deux orientations précitées requièrent un travail en lien étroit avec la communauté éducative. Depuis plusieurs années, les nombreux contacts que l'équipe IDEE entretient avec plusieurs instituteurs / professeurs des Hautes-Pyrénées et des Pyrénées Atlantiques nous permettent d'expérimenter dans les classes les situations d'apprentissage et les outils que nous concevons en recueillant les réactions des enfants et des enseignants. Ce travail expérimental qui a été mené de manière systématique dans le cadre de l'étude de la situation-problème Smash nous a été particulièrement utile : a/ pour recueillir les spécifications informelles des enseignants, b/ pour décrire cette situation-problème conformément aux principes pédagogiques énoncés par les enseignants, c/ pour identifier et adapter les outils informatiques proposés aux besoins des enfants et des enseignants sensés exploiter la situation-problème Smash. Ce dialogue sur la durée entre chercheurs et usagers aussi bien pour la conception des environnements-supports aux situations-problèmes coopératives que pour l'analyse des usages constitue un élément important des travaux menés.

La démarche proposée consiste donc à essayer de rapprocher les besoins nés de la conception et les fonctionnalités offertes par les plates-formes en considérant des modèles de la situation d'apprentissage à exploiter, des abstractions de composants existants et des abstractions de composants à développer. Ceci m'amène à la deuxième partie de ce chapitre consacrée aux travaux sur les modèles de description des situations d'apprentissage (et des situations-problèmes coopératives en particulier).

2.2 Modèles et langages pour l'ingénierie des situations-problèmes coopératives

L'activité de modélisation est centrale en ingénierie du logiciel mais elle l'est également en EIAH. Durant des décennies, beaucoup d'informaticiens ont critiqué les travaux centrés sur l'activité de modélisation considérant (un nombre non négligeable d'informaticiens le considère parfois encore) que les modèles produits sont des outils contemplatifs alors que le code est le cœur de la production logicielle ; les difficultés rencontrées pour faire évoluer le code en fonction de l'évolution des besoins (fonctionnels et technologiques), les travaux engagés et les résultats scientifiques assez récents en ingénierie dirigée par les modèles (Selic 2003), (Bézivin, Blay et al. 2005) redonnent depuis quelque temps ses lettres de noblesse à l'activité de modélisation et plus généralement à la notion de modèle.

Les modèles sont également au cœur de bien des discussions entre chercheurs en EIAH : les discussions portent principalement sur les formalismes à utiliser, l'expressivité / la complétude des modèles proposés et les usages à faire de ces modèles. Les paragraphes suivants donnent des définitions et présentent des points de vue utiles pour comprendre les travaux de modélisation engagés en EIAH.

2.2.1 Place des modèles et méta-modèles en Ingénierie (des EIAH)

La notion de modèle existait bien avant que n'apparaissent les ordinateurs et beaucoup d'activités scientifiques ou d'ingénierie ont souvent recours aux modèles pour mieux étudier des systèmes du monde réel (systèmes existants ou à construire).

(Seidwitz 2003) définit un modèle comme «... a set of statements about some system under study ». (Bézivin et Gerbé 2001) propose une définition plus opérationnelle : « A model is 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 ».

2.2.1.1 Modèles descriptifs et modèles prescriptifs

Comme ces deux définitions le suggèrent, il existe un consensus sur l'existence d'une relation appelée « ReprésentationDe » (Seidwitz 2003), (Atkinson et Kühne 2003), (Bézivin, Blay et al. 2005) entre le modèle et le système étudié. Cette relation a été étudiée par de nombreux auteurs aussi bien dans le domaine des EIAH que dans celui de l'Ingénierie Dirigée par les Modèles :

- On peut utiliser un modèle pour décrire un système existant, le modèle proposé est considéré comme correct si ses caractéristiques et son comportement évoluent dans le temps de la même manière que le système réel. Les modèles sont descriptifs dans la plupart des disciplines scientifiques et servent en général à évaluer des théories⁹ : la théorie est alors considérée comme correcte si les déductions que l'on fait sur le modèle en utilisant cette théorie sont conformes au système existant observé.
- On peut créer des modèles pour prescrire un système à développer. Le système sera considéré comme valide par rapport à cette spécification (représentée par le modèle) si aucune caractéristique du modèle n'est contradictoire avec le système produit. Pour un modèle de spécification, la théorie est considérée correcte a priori et les déductions que l'on peut faire sur le modèle à partir de cette théorie deviennent des éléments du modèle de spécification. Le domaine de l'ingénierie fait souvent appel aux modèles prescriptifs pour définir les caractéristiques des systèmes avant production industrielle (cf modèles et prototypes utilisés en aéronautique, dans l'industrie automobile, en informatique, ...).

Dans le domaine de l'Education et des EIAH, de nombreux auteurs ont recours aux modèles descriptifs pour établir ou mettre à l'épreuve des théories ; citons les travaux sur le

⁹ Une théorie est une façon de déduire de nouvelles caractéristiques d'un système en exploitant les caractéristiques d'un modèle de ce système.

modèle de l'apprenant présentés dans (Webber 2003), les travaux décrivant des modèles d'interaction (Baker, De Vries et al. 2001), (Ohl 2001), (Séjourné, Baker et al. 2004), les modèles de planification pédagogique décrits dans (Dessus et Maurice 1998), ... Enseignants et chercheurs ont aussi très souvent recours à des modèles prescriptifs puisque concevoir des apprentissages ou des EIAH relève d'une démarche d'ingénierie. « Décrire, c'est rendre compte, par une méthode d'observation, de la situation ou de l'activité mise en œuvre par des sujets, en identifiant leurs caractéristiques et leurs conditions d'apparition ou de changement. Prescrire, c'est préconiser la situation optimale, ou encore l'activité que les sujets devraient – ou auraient dû – mettre en œuvre, compte tenu de la situation » (Dessus 2000), (Dessus 2002).

2.2.1.2 Modèles et méta-modèles

Qu'ils soient descriptifs ou prescriptifs, les modèles sont exprimés par des langages, ils s'appuient sur ces langages : « a model is a description of (part of) a system written in a well-defined language » (Kleppe, Warmer et al. 2003). L'informatique donne une place prépondérante aux langages utilisés pour produire les modèles car la nature du langage conditionne les capacités d'exploitation des modèles par une machine. Steve Cook identifie trois aspects principaux d'un langage de modélisation : la syntaxe concrète, la syntaxe abstraite et la sémantique (Cook 2002) : la syntaxe concrète peut être définie sous la forme d'une grammaire, d'un dialecte XML ou d'un ensemble de formes graphiques. La syntaxe abstraite définit les concepts de base du langage. Enfin la sémantique définit la signification des modèles. Un méta-modèle est un modèle d'un langage de modélisation, c'est donc un outil conceptuel qui permet de caractériser un langage: « a meta-model is a model that defines the language for expressing a model » (OMG 2002b). Comme le précise le rapport de synthèse présenté dans (Bézivin, Blay et al. 2005) entre un modèle et un méta-modèle peut exister une relation particulière, la relation « Conforme ». Cette relation est particulièrement importante pour la création des modèles et leur l'interprétation : un modèle correspond à un point de vue sur un système et selon les objectifs du concepteur, certains aspects sont pris en compte et d'autres sont omis (Breton 2002). Le modèle est donc le résultat de l'application d'un filtre sur un système. Ce filtre est le méta-modèle qui établit les caractéristiques de ce qui peut être exprimé par les modèles valides c'est-à-dire respectant la relation « Conforme ». Un méta-modèle est composé d'une terminologie, d'assertions et d'une partie pragmatique ; la terminologie est l'ensemble des concepts, des propriétés et de leurs relations alors que les assertions sont des règles supplémentaires permettant de contraindre les éléments de la terminologie. La partie pragmatique porte une partie de la sémantique et elle est souvent formalisée en langue naturelle ce qui peut conduire à des interprétations divergentes de mêmes modèles (Breton 2002).

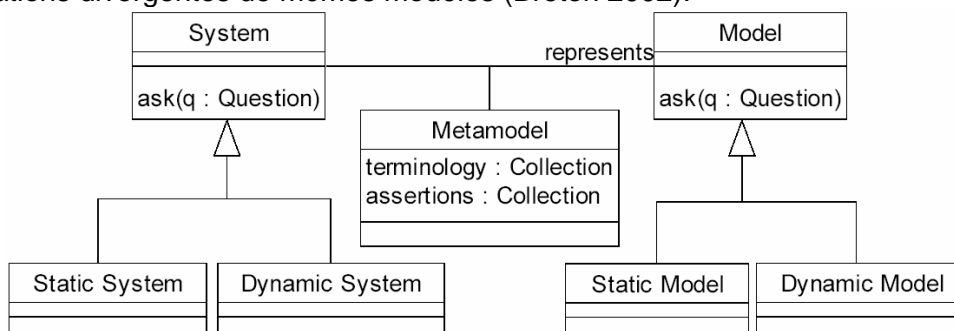


Figure 2.1 : Système, modèle et méta-modèle selon (Breton et Bézivin 2001)

L'OMG a proposé le MOF ou Meta-Object Facility (OMG 2002b) comme modèle général de description des méta-modèles (rendant possible l'unification de tous les méta-modèles, ce qui facilite leur interopérabilité).

Le domaine des EIAH (tout particulièrement celui des Systèmes Tuteurs Intelligents) a traditionnellement recours à des modèles assez diversifiés pour décrire des apprentissages :

modèle du domaine (modèle de l'expert), modèle des interactions (modèle pédagogique/modèle du tuteur), modèle de l'apprenant et modèle des interfaces (interfaces semi-structurées en apprentissage collaboratif, représentations intermédiaires, ...). Mais ces modèles ne s'appuient pas toujours sur des méta-modèles explicites¹⁰ (Dillenbourg 1999), (Baker 2000), et très rarement sur des méta-modèles spécifiques à l'Education et l'Apprentissage : les synthèses (Wenger 1987), (Nodenot 1992a), (Pernin 1996), (Baron, Bruillard et al. 1999), (Dillenbourg 1999) présentant les types de modèles couramment utilisés en ingénierie des EIAH mettent en évidence des approches fonctionnelle (priorité aux traitements), par flots de données (priorité aux données), de type « Modélisation de l'information » (modèles entité-association, réseaux sémantiques, ...), par objets (approche événementielle faisant la synthèse des approches précédentes), ou par objets prototypes comme dans (Pernin 1996). Ces mêmes études montrent également que :

- les formalismes d'expression de la dynamique (des apprentissages, des tâches, des interactions) utilisés en EIAH reposent en général sur des automates à états finis, des tables de décision, des StateCharts de Harel, et des réseaux de Petri.
- les tâches et processus dans lesquels sont engagés les apprenants sont rarement formalisés. Pourtant, que ce soit à des fins d'information, d'analyse, de simulation ou d'assistance, l'exploitation d'un processus implique que ce dernier soit décrit.
- les processus de développement proposés en EIAH reprennent les propositions faites en ingénierie du logiciel mais sont souvent adaptés pour tenir compte de la diversité des acteurs et de leurs compétences (Nodenot 1992a), (Pernin 1996), (Marquesuzaà 1998). Comme pour les processus d'apprentissage, la description des processus de développement est souvent implicite ou en langue naturelle ce qui n'est guère exploitable.

Je tire de cette analyse plusieurs constatations. En premier lieu, il est regrettable que les chercheurs en EIAH n'appuient pas les modèles qu'ils produisent sur des méta-modèles spécifiques au domaine traité : pour être efficaces, les chercheurs comme les ingénieurs doivent sans cesse se recentrer sur le cœur de leur recherche / de leur métier ; en parallèle, les EIAH sont aussi des logiciels et ceci amène les chercheurs en EIAH à être des experts dans leur domaine tout en essayant d'être aussi des experts en développement du logiciel. Cette situation n'est pas satisfaisante :

- Pour soulager les chercheurs en EIAH non informaticiens du poids des développements, il est urgent de leur demander de développer et réutiliser des modèles plutôt que du code.
- Pour faciliter le dialogue entre ces modèles et les modèles que savent produire les informaticiens, pour rendre les modèles exploitables par les informaticiens, il est nécessaire d'explicitier les méta-modèles qui permettent d'interpréter les divers modèles proposés.

Par ailleurs, je constate que différents types de méta-modèles sont en jeu lors du développement d'un EIAH car le terme modélisation désigne à la fois le résultat de l'activité de modélisation et l'activité elle-même (le processus) : les méta-modèles seront utiles pour définir la sémantique des différents modèles de l'EIAH qui est conçu mais aussi pour caractériser le processus engagé (responsabilités et centres d'intérêt complémentaires) par les acteurs participant à la conception. Le chercheur en Sciences de l'Education, l'ergonome, l'enseignant, l'ingénieur pédagogique, l'informaticien ont chacun leur point de vue sur le processus de conception d'un EIAH. A chacun son domaine particulier et donc à chacun son ou ses méta-modèles spécifiques : l'ergonome sera plus intéressé par l'utilisabilité, l'acceptabilité des interfaces en relation avec le modèle des tâches, l'enseignant par la relation savoirs<->apprenant à travers le scénario pédagogique envisagé, l'ingénieur pédagogique par la planification et l'affectation des ressources, l'informaticien par l'automatisation et la connexion avec des composants logiciels issus d'une plate-forme particulière. Ces points de vue ne sont bien sûr pas disjoints et représentent différents aspects d'un même système. Les processus peuvent donc être considérés comme des

¹⁰ Les modèles implicites sont nombreux en EIAH et sont à l'origine de nombreux problèmes d'interprétation (cf paragraphes 2.2.1.3 et 2.2.2.4.2 consacrés à l'exploitation des modèles et aux spécificités des modèles en EIAH).

vecteurs d'intégration des différentes compétences en ingénierie des EIAH : si l'on considère que chaque compétence / dimension est représentée par un méta-modèle indépendant, il est alors utile de préciser leurs interactions respectives : un méta-modèle de processus peut alors être couplé à un méta-modèle de produit ainsi qu'à d'autres méta-modèles de processus.

2.2.1.3 L'exploitation des modèles

Les modèles ne sont l'apanage ni des humains, ni des machines ; tous deux créent des modèles à partir d'autres modèles et tous deux exploitent des modèles créés par d'autres : l'interprétation et l'exploitation de ces modèles est donc un sujet essentiel.

Les modèles descriptifs sont en général utilisés pour évaluer la pertinence d'une théorie : les comportements respectifs du modèle du système et du système réel peuvent ainsi être analysés. Au contraire, les modèles prescriptifs sont utilisés pour spécifier le comportement que devra avoir le système réel. Cette spécification passe par une activité de modélisation, différents modèles pouvant être créés (ou inférés) durant cette activité ou à partir de cette activité. La modélisation (voir Figure 2.2) se fait selon trois axes (Bouzeghoub 2005) :

- L'axe de l'univers du discours qui est constitué de l'ensemble des objets et faits tangibles que l'on souhaite étudier. Les choix concernant cet univers sont arbitraires et dépendent notamment de l'objectif visé et des acteurs impliqués.
- L'axe des abstractions qui est constitué des artefacts créés lors de l'analyse de cet univers pour en simplifier la compréhension. Cet axe, souvent organisé en niveaux, permet de gérer la complexité du système observé ; il est constitué de l'ensemble des artefacts que l'on peut créer pour classer ou décrire les objets et les faits pertinents dans l'univers observé,
- L'axe des formalismes qui est constitué des langages de représentation de ces artefacts et qui vont servir de support de communication et de raisonnement.

La notion de modèle se définit par rapport à ces trois dimensions : chaque modèle élaboré concerne un réel perçu de manière particulière par un observateur qui élabore un certain nombre d'abstractions représentées dans un formalisme donné.

Les modèles prescriptifs sont donc créés pour affirmer un point de vue particulier sur un système et ces modèles vont donc pouvoir être exploités pour a/ passer d'un univers du discours à un autre (exemples : utiliser le modèle du domaine d'un EIAH pour spécifier le modèle du tuteur, transformer une représentation du modèle du domaine centrée «Expert » en une représentation « naïve ») et b/ changer de niveau d'abstraction (exemples : transformer un modèle conceptuel du domaine en une base de faits et de règles, transformer un modèle de l'apprenant en une spécification java).

L'exploitation d'un modèle repose sur l'interprétation de ce modèle par un homme ou une machine. L'interprétation qui sert à transformer un modèle en d'autres modèles –cf relation « TransforméEn » dans (Bézivin, Blay et al. 2005)- repose sur la sémantique du modèle considéré qui est définie par son méta-modèle. L'existence et le choix d'un méta-modèle sont donc très importants puisque les concepts du système considéré « sont interprétés à travers les concepts proposés par le méta-modèle, quitte quelquefois à tordre cette réalité pour l'exprimer dans le langage » (Bézivin, Blay et al. 2005). Le choix du langage est également sensible pour l'exploitation des modèles : selon les bases théoriques du langage adopté, les tâches de validation de la conformité des modèles par rapport au méta-modèle, de raisonnement sur les modèles (inférences de connaissances) sont plus ou moins faciles à réaliser, et parfois impossibles si le langage manque de rigueur. A l'opposé, des langages trop formels limitent les capacités d'expression de concepteurs non spécialistes.

(Baker 2000) a proposé une étude sur le rôle et les usages des modèles en EIAH, ce qui l'a conduit à définir les trois rôles suivants :

1. modèle comme outil scientifique : les modèles (interprétables ou non par la machine) sont utilisés comme support de compréhension, communication et prédiction de certains

aspects d'une situation d'apprentissage ; par exemple, un modèle pour comprendre les le mécanisme de l'auto-explication.

2. modèle comme composant d'un système : ces modèles interprétables sont utilisés comme composants de l'artefact éducatif (qui peut être un composant du dispositif informatique ; par exemple un modèle de l'apprenant embarqué dans un EIAH).
3. modèle comme une base pour la conception : il s'agit de modèles décrivant un processus éducatif (accompagnés de leur théorie d'enseignement/apprentissage) formant la base pour la conception d'un artefact informatique pour l'éducation.

Son étude argumentée (voir Figure 2.2) confirme l'importance des théories (méta-modèles) et du champ expérimental (système réel perçu) dans l'activité de modélisation.

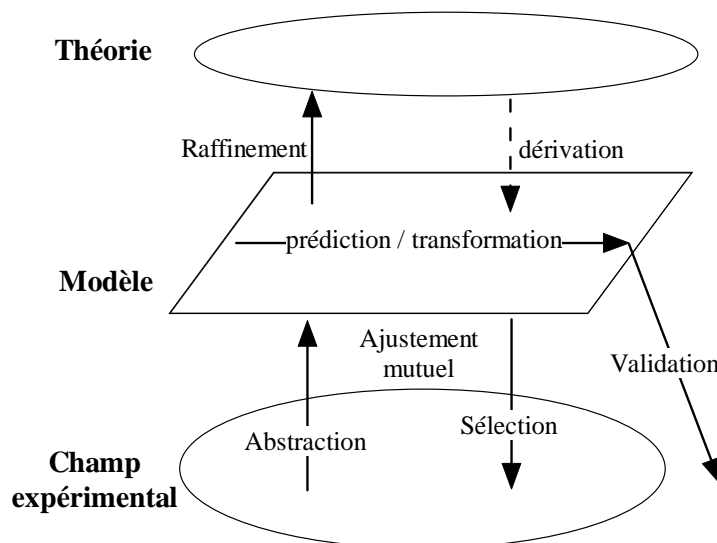


Figure 2.2 : Relations générales entre théorie, modèle et champ expérimental

Cette étude et d'autres travaux de Mickael Baker (Baker, De Vries et al. 2001), (Séjourné, Baker et al. 2004) démontrent ainsi que d'autres modèles que les modèles computationnels sont utiles en ingénierie des EIAH pour guider la production informatique (ces modèles sont souvent ignorés des informaticiens car jugés trop contemplatifs). Je considère cependant que tous les modèles, y compris ceux-ci, doivent s'appuyer sur des méta-modèles explicites qui seuls rendent possible l'interprétation par des tiers. C'est l'une des difficultés actuelles en ingénierie des EIAH car cette imprécision sur les méta-modèles limite/brouille le dialogue entre membres de la communauté et empêche l'exploitation des modèles à des fins de transformation et de réutilisation.

2.2.2 Du scénario pédagogique aux modèles de description d'un EIAH

Dans un article publié en 1995 (Nodenot 1995), je m'interrogeais sur la définition de langages et d'outils permettant la spécification de logiciels éducatifs par des pédagogues, reprenant en cela un vœu de Nicolas Balacheff : « ... Le problème de l'explication du tuteur en direction du maître, dont l'objectif serait la nature des apprentissages et des interactions didactiques avec l'apprenant, n'a pas été envisagé. Pourtant la capacité de la machine à trouver une place dans le système didactique comme collaborateur du maître dépend de sa capacité à communiquer dans le registre didactique ... » (Balacheff 1991).

La fin des années quatre-vingt-dix a été marquée par des propositions diverses visant à donner à l'enseignant un rôle plus important lors de la conception d'EIAH. Les propositions de standards concernant les langages d'indexation de données comme LOM (LOM 2001) visent un enseignant créateur, prospecteur, référenceur et capable d'agrèger des ressources en utilisant par exemple des modèles de mise œuvre comme SCORM (ADL 2004). D'autres propositions insistent sur la nécessité de prendre davantage en compte l'activité de

l'apprenant et replacent la notion de scénario d'apprentissage au centre des débats (Koper 2001), (Rawlings, Van Rosmalen et al. 2002), (Pernin 2003), (Paquette 2004a).

Dans les prochains paragraphes, j'examine les principes d'une approche à base de scénarios d'apprentissage tels qu'ils sont proposés par les langages de modélisation pédagogique. Un langage de modélisation pédagogique (ou EML) tel que le définit (Koper 2002; Rawlings, Van Rosmalen et al. 2002) est : « ... a semantic information model and binding, describing the content and process within a 'unit of learning' from a pedagogical perspective in order to support reuse and interoperability ». MISA (Paquette, Crevier et al. 1997) et IMS-LD (IMS 2003a) relèvent de cette définition (Paquette 2004b).

2.2.2.1 Présentation de MISA

Depuis 1992, de nombreux travaux de recherche au Centre de recherche CIRTA du LICEF (Québec) ont porté sur la mise en œuvre d'une approche de conception pédagogique fondée sur les sciences cognitives baptisée « Ingénierie pédagogique » (Instructional Engineering) : « A method that supports the analysis, the design and the delivery planning of a learning system, integrating the concepts, the processes and the principles of instructional design, software engineering and cognitive engineering » (Paquette 2004b). Ces travaux ont abouti à la proposition d'une méthode d'ingénierie pédagogique MISA (Paquette, Crevier et al. 1997), une plate-forme/LMS de formation à distance Explor@ et un environnement auteur pour les concepteurs ADISA (Atelier distribué d'ingénierie de systèmes d'apprentissage).

L'objectif de MISA est d'aider des concepteurs dans l'élaboration de la conception de systèmes pédagogiques. Cette conception pédagogique pourra alors être utilisée pour produire un système pédagogique exécutable (*runtime*) qui peut être délivré sur un système de diffusion comme Explor@ ou tout autre LMS. Ainsi, MISA se positionne (comme IMS-LD) dans un processus industriel de la e-formation pour lequel de nouveaux experts représentent le public visé pour l'application de ces méthodes d'ingénierie pédagogique.

Une approche de modélisation des connaissances a été suivie afin de définir la méthode MISA, ses concepts, ses processus et ses principes. Afin d'aboutir à la production d'un modèle d'un système d'apprentissage, la méthode propose de suivre six phases, chacune développant progressivement les quatre modèles de conception et les descriptions des objets. MISA 4.0 propose 35 sous-tâches, chacune produisant un élément de conception. Les quatre modèles sont :

- Le modèle de connaissances : c'est une représentation graphique du contenu du domaine du système d'apprentissage visé. Dans ce modèle, les compétences cibles ou de type prérequis sont associées aux unités de connaissance en fournissant les objectifs d'apprentissage du modèle d'apprentissage.
- Le modèle d'apprentissage : c'est essentiellement un réseau d'unités d'apprentissage (*Learning Unit ou LU*) et d'événements auxquels sont attachées les connaissances et compétences cibles. Chaque LU est décrit par un scénario d'apprentissage graphique décrivant les activités d'apprentissage et de support reliées à des ressources. Les ressources manipulant du contenu (appelées *instruments*) sont associées à un sous-ensemble du modèle de connaissances.
- Les modèles de matériaux d'apprentissage (optionnels). Chacun de ces modèles regroupe les instruments dans un *objet d'apprentissage*, en décrivant les composantes médiatiques, les documents sources et quelques principes de présentation et autres spécifications pour construire ou agréger les objets d'apprentissage.
- Le modèle de livraison : il rassemble les matériaux d'apprentissage et tous les autres types de ressources tels que les outils, les liens de communication, les services, etc., décrits dans le modèle d'apprentissage. Chacun de ces modèles de livraison représente un processus de flot de travail (*workflow*) multi-utilisateurs où les acteurs utilisent et produisent des ressources en réalisant leurs différents rôles.

Au coeur de la méthode se trouve la modélisation graphique. Toutes les représentations des modèles de la méthode sont réalisées grâce à la notation MOT (Paquette 1999). Tous

les modèles MOT et propriétés d'objets sont traduits vers XML pour être interprétables par la machine. La figure suivante décrit le modèle pédagogique comme un réseau d'événements d'apprentissage (*learning events*) :

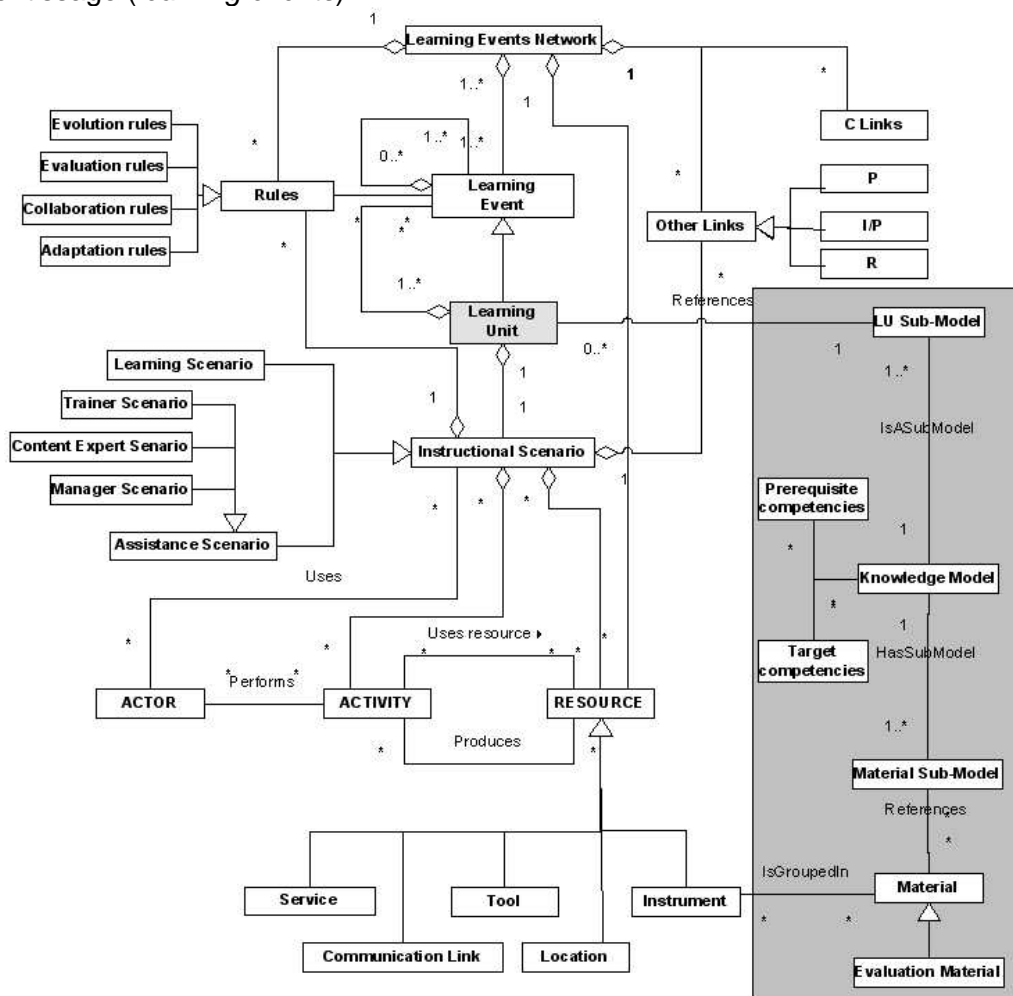


Figure 2.3 : Diagramme de classe du modèle pédagogique de MISA 4.0

Le concept d'événement est la base de toutes les représentations ; il permet de décrire des modules, des cours, etc. Tout réseau d'événements d'apprentissage est composé d'événements, de ressources, de liens et de règles. Les *C links* (Composition) permettent de spécifier la hiérarchie entre les événements d'apprentissage tandis que les *P links* (Précédence) décrivent leurs pré-requis. Les ressources peuvent être utilisées ou produites par les événements d'apprentissage en les reliant avec des *I/P links* (Input/Product). Les règles gouvernent l'utilisation des événements via l'utilisation des *R links* (*Rules*).

Il est possible de construire un scénario d'apprentissage pour chaque unité d'apprentissage. Ce scénario correspond alors à un événement d'apprentissage non décomposable en sous-réseau d'événements. Il a des objectifs et des pré-requis d'apprentissage, fournis par les *target* et *prerequisite competencies* définis dans le modèle des connaissances (cf partie droite de la Figure 2.3).

Une unité d'apprentissage fait référence à un unique scénario d'apprentissage regroupant les acteurs, les ressources, les activités et les liens. Les acteurs réalisent les activités (*R link*) qui utilisent ou produisent des ressources (*I/P links*). Il est possible d'exprimer la précédence entre activités (*P links*) mais la composition d'activités (*C links*) n'est pas permise (Paquette 2004b). Les règles d'exécution, d'évaluation, de collaboration, etc., sont reliées aux activités par les *R-links*. Le sous-ensemble regroupant les activités et leurs ressources réalisées respectivement par un apprenant / un facilitateur est appelé scénario d'apprentissage (*Learning Scenario*) / *Assistance Scenario*.

Cinq types de ressources peuvent être utilisés : *instruments*, *tools*, *services*, *locations*, et *communication links*. Les instruments sont les seuls représentant un contenu (ils sont reliés aux matériaux du modèle de connaissance). La Figure 2.4 présente un exemple de scénario d'apprentissage. Le modèle graphique utilise la notation MOT. Les ovales représentent des activités réalisées par les acteurs (L pour *learner* ou F pour *facilitator*). Les rectangles représentent des ressources (L pour *instruments*, T pour *tools*, S pour *services*, C pour *communication* ou L pour *location*). Les ressources non marquées sont des résultats produits pendant une activité. Les hexagones représentent les règles (X pour exécution, E pour évaluation, C pour collaboration et A pour adaptation).

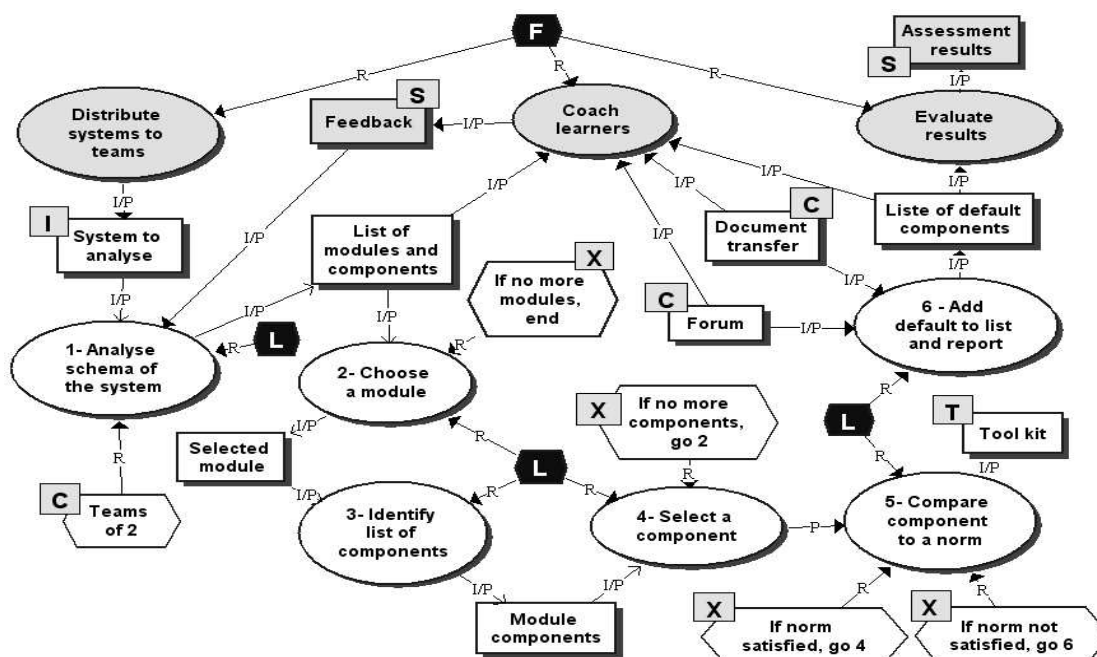


Figure 2.4 : Un exemple de scénario modélisé avec la notation MOT

Pour G. Paquette (Paquette 2004b), MISA est bien un EML au sens de la définition donnée au paragraphe 2.2.2. Le méta-modèle (modèle d'information sémantique) est constitué dans MISA de :

- l'ensemble des modèles MOT groupant le réseau d'événements d'apprentissage (*Network of learning events*) ;
- des scénarios d'apprentissage décrits pour chaque unité d'apprentissage (*learning unit*) ;
- et de leurs modèles de connaissances associés.

La transformation de ces modèles MOT en des fichiers XML constitue le « binding » d'informations dans l'objectif de favoriser la réutilisabilité et l'interopérabilité des modèles produits. En comparaison à un processus général de conception-développement-diffusion, MISA est une méthode similaire à celles de l'ingénierie logicielle qui vise à décrire de nombreux éléments de conception et pas uniquement le résultat de la spécification d'un scénario d'apprentissage. Notons que l'outil MOT a pas mal évolué lorsque l'équipe en charge de son développement a décidé d'en faire un outil appelé MOT+ permettant de produire des modèles compatibles IMS-LD.

La spécification IMS-LD qui sera présentée dans le prochain paragraphe s'intéresse davantage à la diffusion, au packaging des contenus, et aux moyens permettant l'implémentation/exécution des scénarios sur des plates-formes de FOAD.

2.2.2.2 Présentation du standard IMS-LD

Partant de différentes propositions de langages, tout particulièrement de la proposition EML-OUNL (Koper 2001), le consortium IMS a proposé la spécification Learning Design (IMS-LD). Je ne détaille pas ici la spécification mais j'en donne les principales

caractéristiques en présentant son méta-modèle (pour une analyse détaillée de la spécification, se reporter à (Laforcade 2004)). Le méta-modèle d'information d'IMS-LD se décompose en trois composants principaux :

1. Le modèle conceptuel : il représente le vocabulaire, les relations entre les concepts et fait le lien avec la spécification IMS Content Packaging (ressources exploitées).
2. Le modèle d'information : il décrit les éléments d'IMS-LD sur trois niveaux :
 - le niveau A décrit le vocabulaire au coeur de la spécification afin de concevoir des unités d'apprentissage sensées supporter diverses formes de pédagogie ;
 - le niveau B ajoute au niveau A des propriétés permettant, en combinaison avec l'expression de conditions, d'en personnaliser le déroulement (Hummel, Manderveld et al. 2004) ;
 - le niveau C ajoute un mécanisme de notification permettant d'étendre la dynamique des scénarios construits afin de proposer des scénarios adaptatifs (Hummel, Manderveld et al. 2004), (Koper et Olivier 2004).

La description de chacun des trois modèles d'informations s'appuie sur une version restrictive du modèle conceptuel précédent selon le niveau concerné.

3. Le modèle de comportement : il décrit un ensemble de comportements d'exécution (*runtime*) que des systèmes délivrant les unités d'apprentissage devront implémenter.

Le principe au coeur de cette spécification pour la conception d'apprentissage est qu'une personne (*person*) joue des rôles (*role*) dans un processus d'apprentissage/enseignement (typiquement, un rôle d'apprenant '*learner*' ou de personnel '*staff*'). En jouant ce rôle la personne cherche à aboutir à des résultats (*outcomes*) en réalisant des activités d'apprentissage (*learning activity*) ou de soutien (*support activity*) plus ou moins structurées (par le biais des structures d'activités '*activity-structure*'). Ces activités se réalisent dans le contexte d'un environnement (*environment*). Cet environnement est constitué d'objets d'apprentissage (*learning object*) et de services (*services*) qui seront utilisés pendant la réalisation des activités associées à cet environnement. La méthode (*method*) permet de déterminer quel rôle assurera quelles activités à tel moment du processus. Le mécanisme de notification (*notification*) permet également de préciser certaines structures du processus de l'apprentissage ; ce mécanisme appartient au niveau C. La méthode est conçue pour atteindre des objectifs d'apprentissage (*learning objectives*) qui correspondent à la spécification de résultats pour l'apprenant. La méthode est décrite sur l'hypothèse de certains prérequis (*prerequisites*) correspondant alors à la spécification des prérequis d'entrée pour les apprenants.

La méthode est modélisée selon la métaphore de la pièce de théâtre (*play*), plusieurs '*play*' pouvant s'exécuter en parallèle. Une '*play*' est constituée d'un ou plusieurs actes (*act*) séquentiels cette fois-ci. Chaque acte fait référence à un ou plusieurs '*role-part*' concurrents également ; ces '*role-part*' permettent d'associer un rôle avec une activité ou à une structure d'activité.

Au niveau B, une méthode peut contenir des conditions (*condition*) ou règles (par exemple du type *If-Then-Else*) qui permettent alors de raffiner la visibilité des activités et des environnements pour les personnes ou rôles. Ces conditions définissent alors des expressions booléennes sur les propriétés de ces personnes ou rôles. Une propriété peut être regroupée dans des '*property-groups*' qui sont un exemple de propriétés globales (*global elements*) pouvant être définies. Les propriétés peuvent être de différents types, locales comme globales pour les personnes comme pour les rôles.

Une '*notification*' peut être générée par un résultat et peut ainsi créer une nouvelle activité disponible pour un rôle donné (la personne recevant la notification n'est pas forcément celle qui l'a émise). Par exemple, si un apprenant termine une activité (ce qui correspond au fait d'avoir atteint un résultat), alors un autre apprenant ou bien un enseignant peut se voir attribuer une nouvelle activité en conséquence. Ce mécanisme peut aussi être utilisé pour concevoir des apprentissages pour lesquels la ressource d'une activité est dépendante du type de résultats obtenus lors d'activités précédentes.

Deux rôles explicites sont spécifiés dans IMS-LD : *learner* et *staff*. Ces rôles peuvent être spécialisés en sous-rôles d'après la spécification d'IMS-LD mais cela reste à la charge du concepteur d'apprentissage. Les activités peuvent être assemblées dans des structures. Celles-ci sont alors une agrégation d'un ensemble d'activités dans une unique structure ; les activités sont alors associables à un rôle grâce au '*role-part*'. Une structure peut modéliser une séquence d'activités qui devront toutes être réalisées dans l'ordre fourni, ou bien une sélection d'activités pour laquelle le rôle associé décidera du nombre d'activités qu'il voudra réaliser. Les structures d'activités peuvent également faire référence à d'autres structures comme à d'autres unités d'apprentissage externes. Les environnements peuvent contenir deux types de base :

- des objets d'apprentissage localisés (typiquement spécifiés avec une URL accompagnée parfois de métadonnées)
- des services génériques relatifs à des services concrets disponibles à l'exécution. Ces services n'ont pas d'URL associée à la conception. L'URL sera donnée quand la situation d'apprentissage conçue (*Learning Design*) sera instanciée à l'exécution.

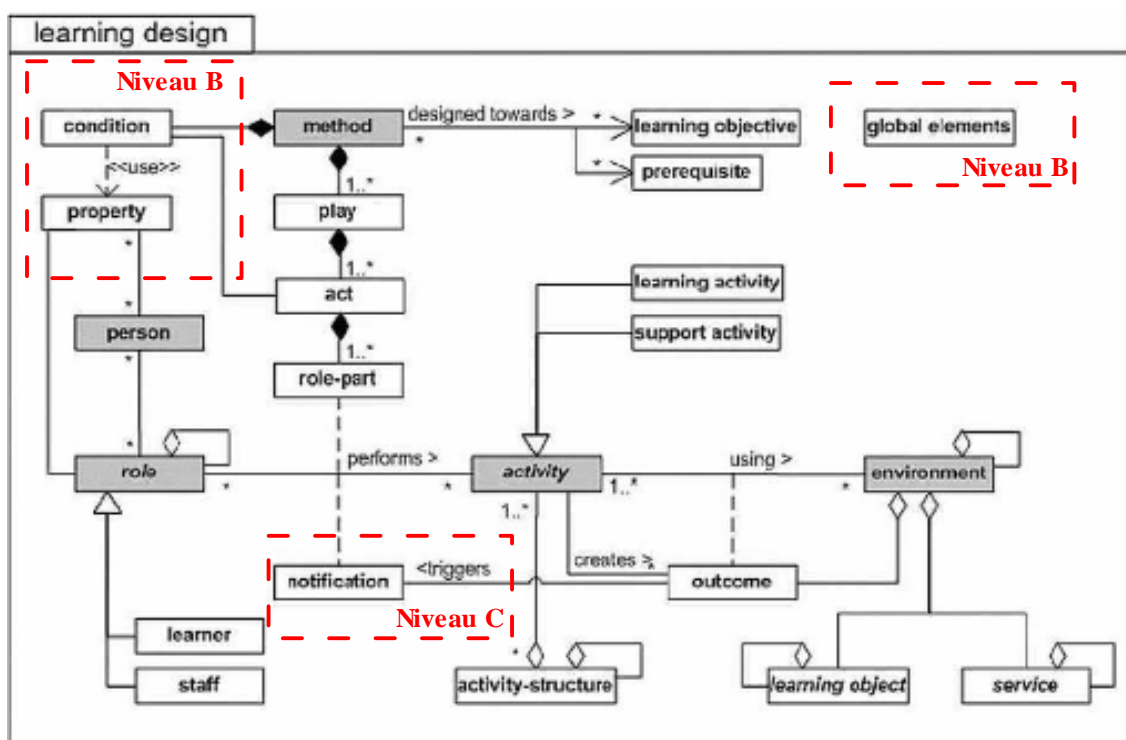


Figure 2.5 : Modèle conceptuel de la structure d'un Learning Design

Le consortium IMS fournit pour IMS-LD une implémentation (« binding ») du modèle d'information sous forme de schémas XML (XSD). Cette implémentation offre aux développeurs des possibilités de traitement automatique par des systèmes informatisés ; plusieurs prototypes d'interpréteurs capables d'exploiter les représentations XML associées à une situation d'apprentissage sont apparus récemment (Edubox, Reload, CopperCore), le projet étant de doter les plates-formes de formation à distance d'interpréteurs IMS-LD.

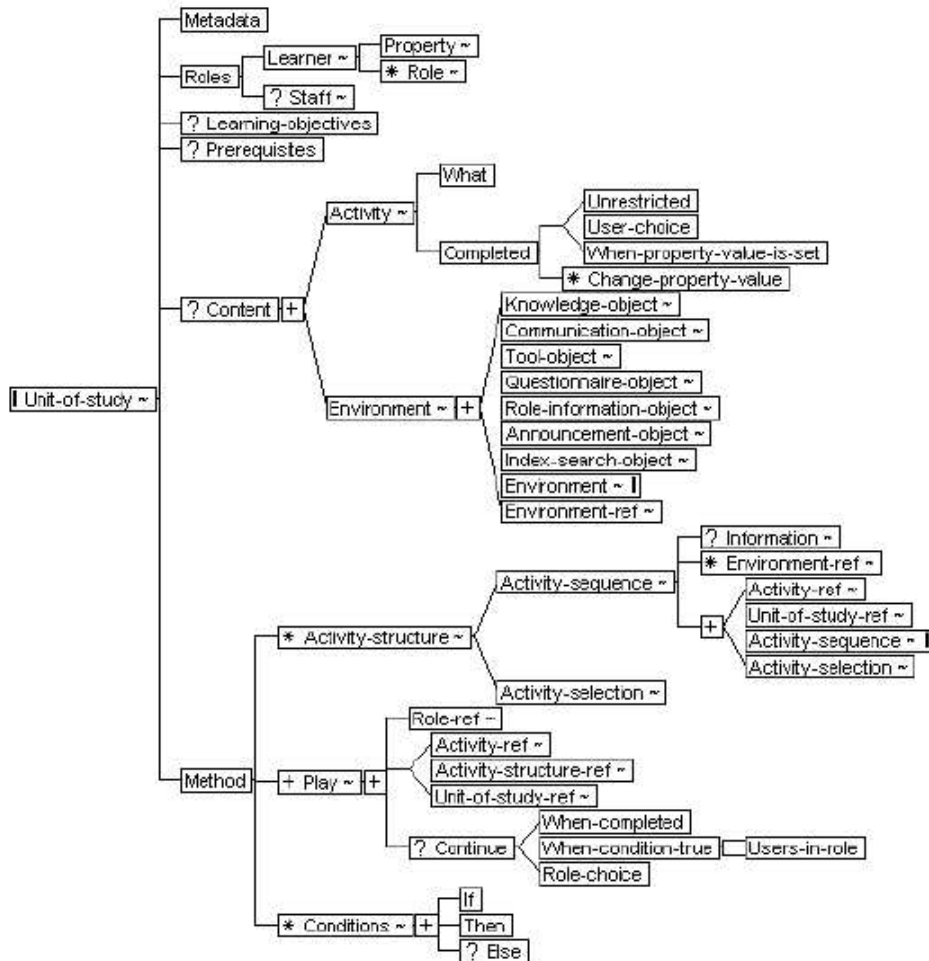


Figure 2.6 : Implémentation XML (binding) de la structure contenant les concepts IMS-LD

2.2.2.3 Apports des approches basées sur la scénarisation pédagogique

Bien qu'il y ait d'autres travaux sur les langages de scénarisation (Pernin 1996), mon choix dans le paragraphe précédent a été de limiter la présentation à deux EML (Educational Modelling languages) centrés scénario, IMS-LD et MISA, car ce sont des propositions mûres sur lesquelles s'appuient les travaux actuels de nombreuses équipes de recherche : travaux sur la structuration des scénarios (Crozat et Trigano 2002), travaux sur le cycle de vie des scénarios (Pernin et Lejeune 2004b), (Pernin et Lejeune 2004a), travaux proposant des extensions aux langages à base de scénario ...

Les deux propositions d'EML examinées s'appuient sur un vrai méta-modèle qui définit les caractéristiques de ce qui peut être exprimé par les modèles valides (respectant la relation « Conformea » cf 2.2.1.2). Ce méta-modèle définit les concepts du domaine (vocabulaire compréhensible par les concepteurs d'EIAH) et les relations qu'il est possible d'établir entre ces concepts : IMS-LD n'étant pas pour l'instant vraiment outillé, les assertions sont formulées en langue naturelle sous forme d'un document de bonnes pratiques annexé à la spécification (IMS 2003b) ; en ce qui concerne MISA, les assertions sont implémentées dans l'environnement auteur développé au laboratoire LICEF. Les théories sous-jacentes à ces méta-modèles sont explicitées : du point de vue du processus de création et de structuration des contenus, les EML proposés relèvent d'une approche « Instructional Design » même si MISA semble plus préoccupé par l'ingénierie que ne l'est IMS-LD : (Paquette 2004a) écrit par exemple à propos de MISA : « It is defined as a method that supports the analysis, the design and the delivery planning of a learning system, integrating the concepts, the processes and the principles of instructional design, software engineering and cognitive engineering ».

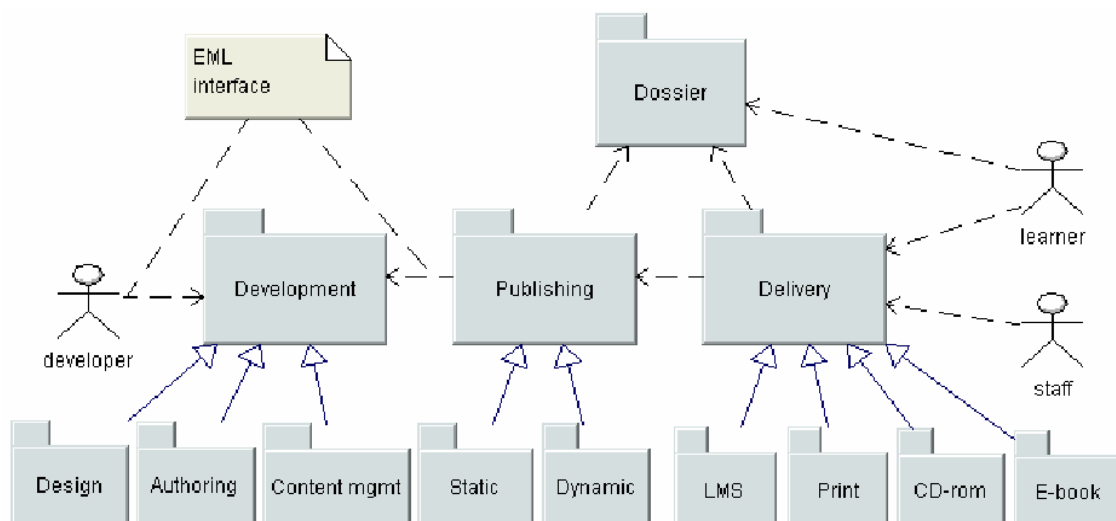


Figure 2.7 : Processus de développement proposé dans le cadre IMS-LD selon (Koper 2001)

Le processus proposé pour ces deux EML¹¹, de type systémique (Dick et Carey 1990), consiste à produire des spécifications pédagogiques à partir d'une analyse des besoins et des buts d'apprentissage (Gagné, Briggs et al. 1988) puis à développer un support de diffusion (*delivery*) répondant à ces besoins d'apprentissage. La notion de scénario pédagogique permet de guider l'analyse des besoins et la conception des contenus pédagogiques, facilitant l'appropriation du langage de modélisation par des enseignants.

Le modèle conceptuel des EML comme IMS-LD et MISA n'est cependant pas centré ressources mais plutôt centré activités : des acteurs (*learner* ou *staff*) conduisent des activités les amenant à manipuler des ressources et des outils (*environnement*). Ces caractéristiques permettent d'envisager l'usage de tels langages au service de différentes théories d'apprentissage.

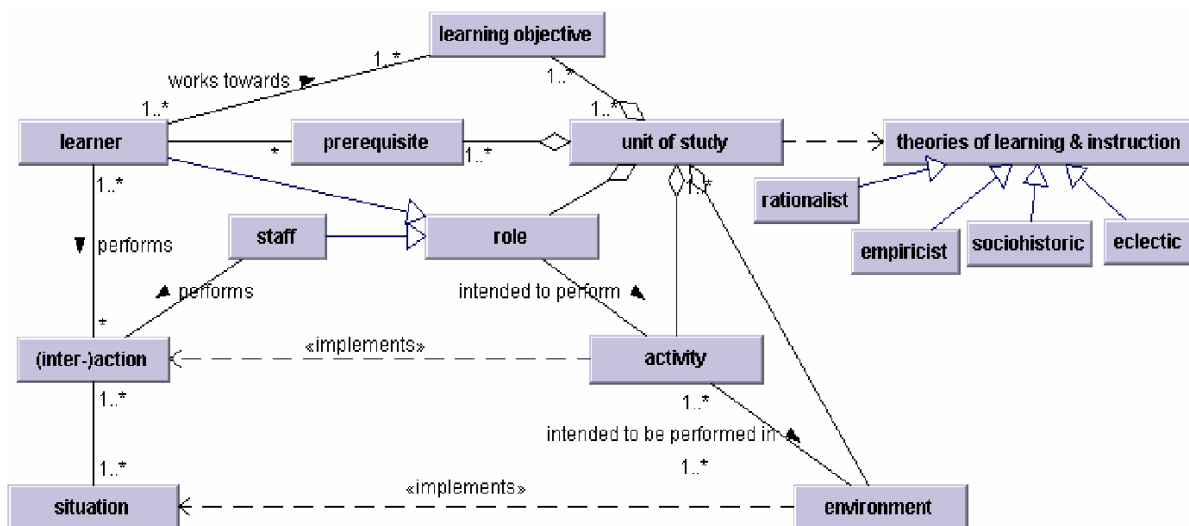


Figure 2.8 : Le modèle conceptuel sur lequel s'appuie IMS-LD (IMS 2003a)

Les concepts proposés par ces méta-modèles permettent de décrire :

- la structuration d'une situation d'apprentissage en unités d'apprentissage plus petites (concept d'*Activity-Structure* d'IMS-LD, concept de *Learning Events Network* de MISA),
- le scénario pédagogique reliant les différentes unités d'apprentissage (concept de *Method* en IMS-LD, concept de *Instructional Scenario* de MISA),

¹¹ Voir la présentation des processus de développement proposés dans (Paquette 2004a) pour MISA et dans (Koper 2001) pour IMS-LD.

- le détail d'une activité d'apprentissage (concept d'*Activity* d'IMS-LD, concepts de *Learning-Unit* et de *Learning-Event* de MISA) avec ses prérequis (concept de *Prerequisite* d'IMS-LD, concept de *Competency* de MISA), ses objectifs (concept de *Learning Objective* d'IMS-LD, concept de *Competency* de MISA) et ses règles de mise en œuvre (concept de *Rule* de MISA),
- le déroulement du scénario en fonction des caractéristiques statiques et dynamiques de l'apprenant (concepts de *Global Properties*, de *Condition* et de *Notification* d'IMS-LD, concepts de *Learning Event* et de *Learning Events Network* de MISA).
- Les ressources exploitées/à produire (intrants/extrants) par les acteurs dans le cadre de leurs activités.
- Les outils et services mis à disposition des acteurs (*Learner* et *Staff/Facilitator*) pour mener à bien les activités définies dans le scénario (concept d'*Environment* d'IMS-LD, concept de *Tool* et de *Service* de MISA). Ces concepts permettent de lier les activités décrites aux fonctionnalités d'outils offertes par les plates-formes.

Les méta-modèles proposés permettent donc d'envisager la conception d'EIAH diversifiés allant de systèmes centrés diffusion de médias (*media-driven designs*) à des systèmes centrés sur la prise en compte des processus distribués menés par les acteurs (*workflow-base designs*) (Morrison 2004). MOT me paraît cependant plus apte à favoriser une démarche d'ingénierie pédagogique qu'IMS-LD puisque les modèles reposent sur la mise en évidence d'un réseau d'événements pédagogiques associé à un modèle de connaissances alors que l'approche IMS-LD fait porter la conception sur la mise en évidence d'activités.

Du point de vue des formalismes de représentation offerts aux concepteurs, IMS-LD et MISA présentent des similitudes mais aussi quelques différences :

- Si pour la structuration (aspects statiques) des situations d'apprentissage, les formalismes se ressemblent (principes de décomposition d'activités en sous activités), il n'en est pas de même pour les aspects dynamiques : les formalismes offerts par IMS-LD pour décrire la dynamique des activités (prise en compte des caractéristiques du comportement de l'apprenant, workflow pédagogique) sont séparés du niveau structurel et regroupés aux niveaux B et C de la spécification (formalisme de type événement-condition-action). Pour sa part, le formalisme MOT offert par MISA mélange dans de mêmes modèles les aspects structurels et dynamiques (cf l'exemple présenté en Figure 2.4) ce qui va à l'encontre d'un principe général de modélisation consistant à séparer les points de vue et tend à complexifier les modèles.
- Le formalisme MOT offert par MISA est un formalisme graphique reposant sur un méta-modèle précis. Le caractère visuel des modèles produits permet d'abstraire la complexité des situations décrites (possibilité de choisir tel ou tel point de vue sur le modèle affiché) favorisant le dialogue, la compréhension et l'implication des différents concepteurs. Le formalisme IMS-LD n'est pas graphique, seule une implémentation XML existe. La documentation d'IMS-LD suggère cependant l'utilisation des diagrammes d'activités UML ou des diagrammes de Gantt pour décrire les scénarios d'apprentissage.
- IMS-LD tout comme le formalisme graphique MOT de MISA disposent d'une implémentation (binding) XML pour pouvoir exploiter les modèles produits. Les auteurs et les utilisateurs de ces propositions de langages ont compris l'importance de l'interopérabilité entre langages pour capitaliser les développements : ainsi (Paquette 2004a) propose une transformation entre MOT et IMS-LD, plusieurs propositions d'interpréteurs IMS-LD sont en cours de développement (cf projets Edubox, Reload, CopperCore) pour des plates-formes diverses notamment pour la plate-forme OpenUSS.
- Notons enfin que l'usage des méta-modèles d'IMS-LD ou MOT est souvent prescriptif et sert de guide pour la conception d'un EIAH. Des usages descriptifs sont également possibles pour rendre compte, dans un vocabulaire stabilisé, des scénarios réels se nouant entre apprenants impliqués dans une situation d'apprentissage. Certains travaux (Barré et Choquet 2005) portent donc sur la mise en relation entre les modèles prescriptifs correspondant au scénario imaginé par les concepteurs et les modèles descriptifs retraçant l'activité réelle des apprenants.

Plus généralement, la proposition du consortium IMS (et son langage d'expression de scénarios pédagogiques IMS-LD) me semble de nature à changer notre façon d'envisager l'implémentation des EIAH. Jusqu'ici, l'opérationnalisation d'un EIAH passait par des langages de programmation tels java, Prolog, ... Pour la première fois, les chercheurs et les développeurs ont pour perspective de pouvoir disposer d'un langage à vocation pédagogique qui est interprétable par des plateformes de FOAD et encapsule la complexité technologique sous-jacente ; Ceci explique que les propositions de langages de conception, tels MOT+ au LICEF suivent de près et essaient de rester compatibles avec la proposition IMS-LD (grâce notamment à des techniques de transformation de modèles).

Le prochain paragraphe 2.2.2.4 montrera que les propositions IMS-LD et MOT présentent cependant un certain nombre de limites puisque ces EML n'ont pas été pensés pour des EIAH orientés processus de résolution de problème ni pour des EIAH centrés sur la personnalisation des interactions.

2.2.2.4 Limites des approches basées sur la scénarisation pédagogique

Les EML étudiés affirment leur neutralité pédagogique et leur ouverture aux différentes théories d'apprentissage (voir par exemple la Figure 2.8). Le peu de recul que l'on a sur le standard IMS-LD, par exemple, ne permet ni d'infirmer totalement ces affirmations, ni de les confirmer (Hummel, Manderveld et al. 2004), (Santos, Boticario et al. 2004). Dans les prochains paragraphes, je décris les limites d'IMS-LD en m'appuyant successivement sur deux types d'analyse. J'examine tout d'abord les problèmes opérationnels que peut poser IMS-LD au concepteur utilisant ce langage pour concevoir des situations d'apprentissage. Dans un deuxième temps (cf paragraphe 2.2.2.4.2), je décris les biais que les langages visant la seule scénarisation pédagogique des situations d'apprentissage introduisent sur l'activité de conception.

2.2.2.4.1 Problèmes opérationnels pouvant se poser au concepteur qui utilise la spécification IMS-LD

La spécification IMS-LD est une proposition de standard très récente pour laquelle peu de retours d'expérience existent (lors de sa conférence à l'école d'été EIAH 2005, Chris Kew annonçait que, dans le cadre du programme UNFOLD, une dizaine d'études de cas avaient été menées en utilisant IMS-LD mais que ces études étaient partielles et n'avaient pas donné lieu, pour l'instant, à des environnements d'apprentissage opérationnels). Par ailleurs, une spécification faite en IMS-LD est un document XML de plusieurs dizaines de pages bien difficile à retranscrire et à analyser dans ce type de mémoire. Aussi, plutôt que de partir d'exemples (simplificateurs) que j'aurai moi même construits, j'ai choisi d'analyser les problèmes potentiels que peut poser IMS-LD aux concepteurs en me basant exclusivement sur les deux documents de référence aujourd'hui disponibles : la spécification IMS-LD version 1.0 (IMS 2003a) et le document Best Practice and Implementation Guide (IMS 2003b).

L'étude qui a été menée est retranscrite sous forme d'une série de tableaux présentée dans l'annexe A. Trois aspects de la spécification IMS-LD sont examinés : sa capacité à décrire des activités d'apprentissage (voir Figure Annexe A.1), sa capacité à décrire des activités collaboratives (voir Figure Annexe A.2), sa capacité à rendre compte de la structuration et de la dynamique des scénarios d'apprentissage conçus (voir Figure Annexe A.3). Chacun de ces tableaux liste une série de problèmes opérationnels découlant des orientations prises par la spécification (cf les extraits de la spécification qui sont donnés dans chacun des tableaux).

La synthèse de cette étude, qui est présentée dans la Figure 2.9, reprend les principaux problèmes opérationnels pour chacun des trois aspects étudiés.

Concepts étudiés du métamodèle IMS-LD	Caractéristique du concept dans la spécification IMS-LD pouvant poser problème	Problèmes soulevés quand on spécifie/conçoit des situations d'apprentissage avec IMS-LD
<p>Modèle d'information associable à une activité Voir détails de l'analyse en annexe A (figure A.1)</p>	<p>Elément non décomposable dont la description est un texte ou une référence à un objet externe</p> <p>Elément lié à 0 ou plusieurs références d'éléments de type Environment (un environnement constitué d'outils et/ou de ressources)</p> <p>Elément non associé à des éléments de type Outcome (Outcome apparaît dans le modèle conceptuel initial mais disparaît dans le modèle d'information). Les Outcomes sont pris en charge par la mise à jour de propriétés exploitables par le scénario et via des notifications sur condition de terminaison (fin d'activité sur choix de l'utilisateur, durée maximale d'activité atteinte, ...)</p> <p>Elément pouvant être associé à des références d'objectifs pédagogiques et de prérequis</p> <p>Le concept d'activité n'a aucune propriété particulière pour caractériser une activité collaborative</p> <p>Plusieurs play peuvent se dérouler en parallèle mais de manière totalement indépendante</p> <p>Le concept de role-part permet de représenter des activités se déroulant en parallèle</p> <p>La prise en compte des activités collaboratives se fait en associant des rôles à des environnements d'outils à vocation collaborative</p> <p>Dans IMS-LD les rôles sont globaux, statiques et non situés. Ils peuvent éventuellement être spécialisés lors de la conception. Les rôles sont associés aux activités par le concept de role-part</p> <p>Dans IMS-LD, c'est lors de l'étape d'instanciation d'un scénario que seront résolus les problèmes de droits d'accès des rôles aux fonctionnalités et services offerts par les outils mis à disposition</p> <p>Analyse descendante s'appuyant, si nécessaire, sur une adaptation des diagrammes d'activités UML</p>	<p>Une activité est une boîte noire du point de vue du concepteur du scénario IMS-LD</p> <p>Une activité est un conteneur pour associer dans un scénario des learning objects conçus par ailleurs impossible de spécifier les caractéristiques de l'environnement d'outils et des ressources à partir des besoins issus de la définition des activités à mener</p> <p>Impossible d'analyser une activité du point de vue des ressources produites / à produire par un apprenant ou un tuteur</p> <p>Rien n'est prévu pour faire apparaître, lors de la conception d'une activité pédagogique, les événements significatifs pour un concept (savoir / savoir faire) abordés dans le cadre de cette activité</p> <p>Rien n'est prévu pour faire apparaître, lors de la conception d'une activité pédagogique, les concepts, les prérequis et les objectifs significatifs du scénario conçu</p> <p>IMS-LD offre des moyens pour décrire des processus s'exécutant en parallèle (play, activity-structure, role-part) mais n'offre pas de moyens suffisants pour décrire leur synchronisation</p> <p>Impossible de spécifier les types d'accès de chaque rôle impliqué dans des activités collaboratives que l'on conçoit</p> <p>Une activité (collaborative) est un conteneur pour associer dans un scénario des rôles avec des environnements d'outils génériques conçus par ailleurs</p> <p>L'approche descendante utilisée pour analyser une situation d'apprentissage conduit à des scénarios favorisant la conception de micro-activités</p> <p>Les modèles obtenus multiplient les références au concept d'Activity-Structure. Or, ce concept prend des significations différentes selon le niveau d'analyse auquel il est utilisé. L'utilisation de deux concepts différents auraient facilité l'interprétation des scénarios</p> <p>Le modèle événementiel est limité à des événements prédéfinis (portant sur les concepts de description d'un scénario) et à la prise en compte des valeurs prises par des propriétés simples définies par le concepteur (mais non attachées à des objets significatifs que le concepteur envisage de faire manipuler aux acteurs). Il n'est donc pas possible de décrire l'évolution des propriétés attachées aux objets significatifs de conception puisque ces objets sont seulement référencés et pré-existants</p> <p>Pour combler les lacunes du modèle d'information, les noms de propriétés portent une partie de la sémantique de ces propriétés</p>
<p>Modèle d'information associable à une activité collaborative Voir détails de l'analyse en annexe A (figure A.2)</p>	<p>Activity, Play, Role-Part, Role, Environment</p>	<p>Principes méthodologiques proposés pour structurer le scénario, Method, Play, Act, Activity-Structure</p>
<p>Structure et dynamique des scénarios IMS-LD Voir détails de l'analyse en annexe A (figure A.3)</p>	<p>Principes méthodologiques proposés pour décrire la dynamique d'un scénario, Property, Notification, Condition</p>	<p>Activity-Structure sert à la fois pour organiser (séquence, sélection) des activités faites par un seul rôle d'acteur mais aussi pour grouper des unités d'apprentissage dans lesquelles sont impliqués plusieurs rôles d'acteurs.</p> <p>Les notifications portent sur des événements simples pouvant être associés à des concepts IMS-LD (play, act, ...) ou des valeurs prises par certaines propriétés</p> <p>Les diagrammes d'activités UML décrivent un workflow possible qui peut faire apparaître et/ou peut exploiter les valeurs prises par des éléments de type Property</p>

Figure 2.9 : Synthèse des problèmes que peut poser IMS-LD aux concepteurs

Les éléments qui résultent de cette étude sont les suivants :

- Les éléments du langage permettent de décomposer une situation d'apprentissage en blocs plus élémentaires (*play*, *activity-structure*, *activity*) dont l'enchaînement peut être défini (séquence, choix, ...). Mais aucun élément du langage n'est fourni pour décrire précisément comment se déroule une activité, ce que sont les produits issus d'une activité (*outcome*), ce qui caractérise l'environnement d'outils qui supporte l'activité menée.
- La spécification précise que l'environnement d'outils et les ressources utilisés durant une activité ont vocation à être réutilisables (donc décontextualisés) et interopérables (voir « Requirements » R6 et R8 du paragraphe 2.1 dans (IMS 2003a)). L'environnement d'outils associé à une activité décrite avec IMS-LD n'est donc pas constitutif de l'activité et aucun élément du langage ne permet de contextualiser (ou décrire) l'un par rapport à l'autre.
- Les rôles et les activités IMS-LD sont stéréotypés de manière statique (un rôle est soit de type *staff*, soit de type *learner* ; une activité est soit de type *learning*, soit de type *support*) alors que dans la réalité d'une situation d'apprentissage les acteurs doivent pouvoir changer de rôle de manière dynamique, chaque rôle joué par un acteur permettant d'entretenir des relations particulières avec d'autres rôles (un rôle dynamique est forcément contextualisé, ce qui peut amener un apprenant à jouer un rôle d'évaluateur pour une situation donnée). Par ailleurs, l'organisation des activités dans IMS-LD me paraît très statique, les affectations des rôles (concept *role-part*) aux activités étant prédéfinies par le scénario : pour des situations-problèmes, l'organisation des activités par les acteurs fait partie du scénario qui devient donc également dynamique et ouvert.
- Aucun élément du langage ne permet de décrire et synchroniser des activités collaboratives (en termes de responsabilités des acteurs, de ressources utilisées et produites, de services logiciels mis à disposition).
- L'approche qui est proposée pour faire apparaître un scénario pédagogique est descendante, ce qui conduit, en pratique, à des scénarios référençant un grand nombre de micro-activités (voir l'exemple Boeing dans (IMS 2003b)) sur lesquelles le concepteur n'a que peu d'emprise pédagogique (cf les événements pédagogiques prédéfinis associables aux activités et aux constituants d'un scénario ; cf les propriétés simples définissable par le concepteur mais non attachables à des objets significatifs que le concepteur envisage de faire manipuler aux acteurs).

De cette étude, je conclus que les concepteurs de la proposition IMS-LD ont privilégié la définition d'un langage complet (« requirement R1 »), formel (« requirement R4 »), interopérable (« requirement R6 ») et compatible avec les autres standards en cours de définition (« requirement R7 »), au détriment de la flexibilité par rapport aux différentes théories d'apprentissage (« requirement R2 ») mais surtout de la précision et de la reproductibilité¹² des modèles décrits (« requirement R5 »),.

2.2.2.4.2 Problèmes plus généraux posés par les approches centrées sur la scénarisation pédagogique

Ce paragraphe, qui s'appuie sur l'analyse des problèmes opérationnels posés au concepteur utilisant IMS-LD (voir paragraphe 2.2.2.4.1 et annexe A), aborde de manière plus globale les limites des langages centrés sur la scénarisation pédagogique. L'argumentaire développé dans les prochains paragraphes met en évidence les risques de planification stricte qui en résultent, les biais et les limites que ces langages font peser sur l'activité de conception.

¹² A la fin de ce chapitre (voir paragraphe 2.2.3 et Figure 2.11), je propose un certain nombre de pistes pour mieux satisfaire au critère de reproductibilité des modèles. Ces pistes seront ensuite détaillées dans le chapitre 3, ce qui conduira à la définition du méta-modèle CPM.

Les objectifs de complétude et d'interopérabilité amènent à une planification stricte

Des EML comme IMS-LD et MOT ont vocation à décrire de manière complète une situation d'apprentissage à partir des concepts portés par leur méta-modèle. Les objectifs fondateurs d'IMS-LD décrits dans (IMS 2003a) sont particulièrement clairs à ce sujet :

- le langage a pour ambition de décrire de manière complète le processus d'enseignement / d'apprentissage associé à une situation, en référençant tout objet d'apprentissage ou service logiciel nécessaire au bon déroulement de ce processus (voir « Requirement R1 » du paragraphe 2.1 dans (IMS 2003a)),
- une spécification IMS-LD se veut par ailleurs formelle afin que des « players » puissent exécuter une telle spécification (voir « Requirement » R4 du paragraphe 2.1 dans (IMS 2003a)).

La proposition IMS-LD vise donc la définition d'un langage formel et autosuffisant dont les insuffisances ont été mises en évidence au paragraphe 2.2.2.4.1. Concevoir des situations d'apprentissage avec IMS-LD n'est donc pas neutre, les situations privilégiées étant clairement des situations d'enseignement, et d'enseignement / évaluation (voir Figure 2.10) :

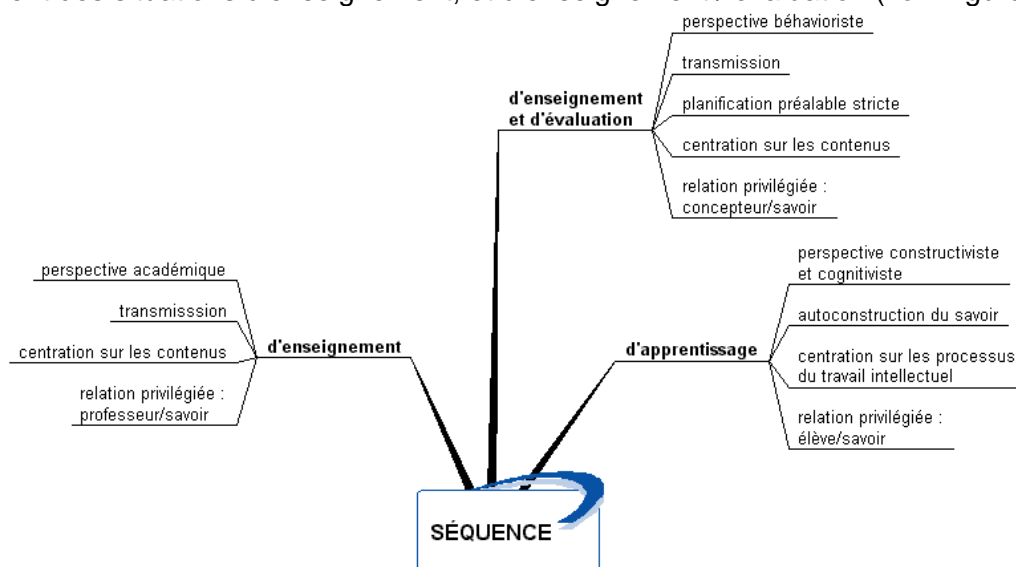


Figure 2.10 : Toutes les séquences d'apprentissage ne sont pas centrées sur une planification stricte au sens IMS-LD (d'après (Fritsch 2005))

Comme (Allert 2004), je considère, sur la base de l'analyse synthétisée en annexe A, qu'IMS-LD est conçu sur la métaphore de la transmission (et non sur la métaphore de la construction de savoirs), les outils (environnement) servant à mettre à disposition des contenus qui déterminent les activités que doivent conduire les acteurs (ce qui explique pourquoi le langage ne fournit pas d'éléments permettant de décrire ces activités). L'usage d'IMS-LD en tant que langage de conception me semble donc conduire à des systèmes basés sur une planification stricte (systèmes fermés) et non à des systèmes ouverts (Young 1993), dans lesquels les apprenants vont organiser les activités conduisant à une construction de leurs savoirs (Nodenot, Laforcade et al. 2003), (Allert 2004).

La scénarisation pédagogique peut biaiser l'activité de conception

Le rôle prépondérant donné au scénario pédagogique est justifié par la volonté de fournir un guide organisant la réflexion des concepteurs (Koper 2001). Si effectivement le scénario a quelque chose de rassurant pour guider le processus d'analyse, je ne suis pas sûr que son influence est réellement positive quand l'objectif est de créer des situations instrumentalisées engageant les apprenants dans des activités de construction de leurs connaissances :

- Le premier argument me vient par analogie au rôle que peut avoir un outil comme « Powerpoint » pour scénariser et maquetter des interfaces Homme-Machine. Le rôle structurant d'un tel outil (hiérarchisation des contenus, organisation des pages, liens

structurels entre contenus, ...) rassure souvent les étudiants mais limite largement leur créativité et les types d'interfaces qu'ils ont l'idée de produire. Powerpoint est peut être un outil (un langage) pour mettre en œuvre des principes de scénarisation identifiés au préalable mais certainement pas un outil (un langage) pour faire naître des idées originales d'interfaces Homme-Machine.

Utiliser IMS-LD comme base d'analyse d'une situation d'apprentissage peut présenter les mêmes inconvénients. Approcher une situation d'apprentissage par la création d'un scénario de type structuration en actes, scènes et activités a toutes les chances d'amener à une organisation hiérarchique de cette situation¹³ (approche par les contenus) et a donc peu de chance d'engager l'apprenant dans les interactions qui lui sont proposées (l'apprentissage à base de situations-problèmes coopératives relève plus d'un modèle d'apprentissage constructiviste que d'un modèle behavioriste).

- Le deuxième argument porte sur la relation à établir entre le scénario et l'objectif d'une situation d'apprentissage. Dans l'esprit d'IMS-LD, le scénario est souvent utilisé en tant que processus capable de faire apparaître des objectifs, des ressources, des activités, ... C'est un des rôles possibles du scénario mais sûrement pas celui qui me paraît le plus intéressant : le scénario devrait (du moins dans le cadre de la conception de situations-problèmes) être mis au service d'une idée pédagogique (qui est donc première) afin de la mettre en valeur et de la décliner dans une situation d'apprentissage. Ces deux orientations assez différentes peuvent se comprendre facilement en faisant une analogie tirée de (Morrison 2004) portant sur la notion de scénario au cinéma : à un critique qui avait du mal à comprendre les films dits de la Nouvelle vague et défendait des structures de narration classiques, Jean Luc Godard acquiesça dans un premier temps à ses propos puis lui répondit : « un film a effectivement besoin d'un début, d'un milieu et d'une fin ... Mais pas forcément dans cet ordre ... ». Ce qui prime, c'est donc l'ordonnement le plus adéquat pour accrocher et maintenir l'intérêt et la participation (intellectuelle) du spectateur ou de l'apprenant. L'émotion chez l'apprenant vient d'abord, elle précède l'apprentissage ; les modèles de scénarios restent importants mais leur usage ne doit pas être prépondérant sous peine d'aller vers une approche trop analytique qui va à l'encontre des idées de situations porteuses de sens pour les apprenants.
- Le troisième argument porte sur les finalités d'un scénario au sens IMS-LD ou MOT. Le scénario conduit à la prescription d'activités à conduire par les apprenants ou les tuteurs. Ces activités sont mises en liaison d'une part avec des ressources (à utiliser, à produire) et d'autre part avec des outils constituant l'environnement des acteurs engagés dans ces activités. En aucun cas, le scénario ne conduit les concepteurs à réellement étudier le contexte de mise en œuvre des activités en liaison avec les outils potentiels disponibles ; Des modèles spécifiques me semblent pourtant très importants pour que les outils mis au service des activités concrètes à mener aient une chance de devenir des instruments.

Comme (Wilson 2004b), je pense donc que la dimension « Instructional Design » (portée entre autres par la notion de scénario) peut apparaître à tout moment dans l'analyse d'une situation d'apprentissage (mais pas en premier lieu) ; qu'elle est compatible avec d'autres formes d'analyse (et de modèles) pour appuyer des idées de situations d'apprentissage imaginées par ailleurs, pour les décliner et les mettre en perspective de manière organisée et systématique (Gagné, Briggs et al. 1988), (Merrill 1996).

Un scénario est une forme limitée de modèle

Plus généralement, je crois important de revenir à une orientation centrée sur un ensemble de modèles faisant référence les uns aux autres plutôt que centrée sur les seuls scénarios pédagogiques : à l'époque de l'Enseignement Assisté par Ordinateur, le squelette d'un logiciel éducatif était un scénario, c'est-à-dire une séquence d'interactions pédagogiques ; sous l'influence des travaux en Intelligence Artificielle, l'élément central d'une application

¹³ Voir les neuf étapes de la théorie de R. Gagné : Obtenir l'attention des apprenants, Informer les apprenants des objectifs, ..., Renforcer les acquis, Evaluer les performances, Transférer les connaissances (Gagné, Briggs et al. 1988).

éducative n'a plus été le scénario mais un ensemble de modèles. Les raisons valables à l'époque me semblent toujours d'actualité et je m'appuie donc ici sur l'argumentaire proposé par Pierre Dillenbourg en 1999. Voici l'exemple qu'il prenait :

« Si je dois expliquer à un collègue comment se rendre de la gare à l'université, je peux soit lui fournir un scénario, soit un modèle. Le scénario est une séquence d'instructions : prends telle rue, tourne à droite, suis la rue jusqu'au deuxième feu, ... Le scénario comprend éventuellement des alternatives (par exemple un chemin plus court et un chemin plus agréable). Le modèle est un plan ou une carte de la ville, c'est-à-dire une représentation simplifiée de la ville réelle, permettant au lecteur de prendre des décisions sur son itinéraire ». Et voici quelques unes des conclusions qu'il en tirait :

- « Dans le second cas, on peut mettre en œuvre différents niveaux de guidage à partir du modèle de plan : il est possible de guider le collègue pas à pas, de le stopper dès qu'il s'éloigne du chemin correct, ou de lui laisser liberté totale dans son exploration de la ville.
- Dans le second cas, on peut modifier dynamiquement la « balance du contrôle », laquelle penchera soit du côté de l'apprenant, soit du côté du système selon celui des deux qui conduit les interactions (possibilité d'initiative mixte).
- Pour guider pas à pas l'apprenant (contrôle par le système), il suffit de disposer d'un itinéraire précis (scénario). Pour le laisser explorer librement la ville (contrôle par l'apprenant), le système ne doit pas disposer d'une représentation de la ville puisqu'il n'intervient pas ; pour un mode d'initiative mixte, le système doit disposer d'un modèle (la carte) permettant de raisonner sur la position de l'utilisateur vis-à-vis de sa destination (afin de déterminer un seuil de dérive à partir duquel il intervient et afin de pouvoir recommander une direction lorsqu'il intervient). Ce modèle explicite de la tâche est appelé modèle du domaine (ou modèle de l'expert).
- En réalité, pour développer tout didacticiel (avec ou sans IA), l'auteur doit disposer d'un modèle clair de la tâche. Les techniques d'analyse de contenu représentent la clé de voûte des techniques d'ingénierie éducative. L'innovation consiste ici à introduire explicitement ce modèle dans le système, au moyen des techniques de représentation des connaissances développées en IA. »

Un scénario pédagogique comprend à la fois des connaissances sur le contenu de l'enseignement et sur les interactions pédagogiques à réaliser. Là encore, on peut représenter de différentes manières ce scénario, sous forme d'une séquence d'actions ou d'un modèle, avec les mêmes avantages que précédemment dans le deuxième cas : le modèle pédagogique (ou modèle du tuteur) se définit par l'ensemble des connaissances permettant d'exploiter le modèle du domaine pour générer des interactions : la prise de décision pédagogique pouvant être liée au contenu, aux caractéristiques de l'élève (cf modèle de l'apprenant) ou à des facteurs externes (par exemple le temps disponible).

Le scénario pédagogique est donc un cas particulier de modèle. Dans le cas du scénario, la stratégie, bien que séparée du modèle du domaine, est implantée par une sorte de procédure informatique, les éléments de contenu sont simplement remplacés par des références à des objets (sortes de variables) qui seront instanciées avec les vrais objets d'apprentissage (ressources et outils) en cours d'exécution. Mais cette représentation ne permet pas au système de résoudre les problèmes posés à l'apprenant, d'avoir une connaissance explicite des actions de tutorat à entreprendre et des actions menées par l'apprenant. Le scénario intègre donc peu d'informations permettant d'évaluer les performances d'apprenants en phase de résolution de problème.

(Baker 2000) a souligné enfin l'importance des modèles mentaux comme base de la conception et réfute l'usage des seuls modèles computationnels. Il définit la recherche en Intelligence Artificielle appliquée à l'Education (AIED research) : « as the use of computers to model aspects of educational situations that themselves involve the use of computers as educational artefacts, some of which may incorporate computational models ». A côté des modèles scientifiques (modèles permettant de comprendre et prédire un événement particulier lié à une situation d'apprentissage) et des modèles ayant vocation à devenir des composants d'un système (cf modèles articulés du domaine, de l'apprenant, ...), il considère

que tout modèle ayant une cohérence interne et permettant de comprendre la complexité d'une situation d'apprentissage est digne d'intérêt même si au final ce modèle n'apparaît pas de manière explicite dans le scénario (ou l'ensemble des modèles opérationnels) exploité(s) par les apprenants en phase d'exploitation de la situation d'apprentissage. Le modèle qu'il propose pour comprendre les interactions et guider la conception de situations de résolution de problèmes me semble démontrer la justesse de cette approche. J'ajouterai que dans le domaine de la modélisation informatique, il n'est pas non plus rare d'utiliser des modèles pour comprendre et comme base de conception : les diagrammes de cas d'utilisation et les diagrammes de séquence d'UML ne servent qu'à délimiter, documenter et s'appropriier les contours d'une application informatique à développer.

A l'issue de cette analyse, IMS-LD et le formalisme MOT de MISA m'apparaissent toujours utiles (mais non suffisants) pour décrire le résultat opérationnel d'un processus d'analyse et de spécification des situations-problèmes. J'ai montré que ces mêmes formalismes sont en partie inadaptés pour transcrire les modèles mentaux des concepteurs : des modèles plus amont doivent pouvoir être produits par les concepteurs pour aboutir au scénario le plus approprié pour engager les apprenants dans la résolution collective d'une situation-problème. Le scénario (métaphore théâtrale, cinématographique ou autre) n'est donc qu'un modèle parmi d'autres qui ne me semble pas devoir être mis en avant comme élément de structuration principal des situations d'apprentissage.

2.2.3 Orientations de recherche au LIUPPA pour la modélisation des situations-problèmes coopératives

L'analyse qui vient d'être faite des forces et des limites des langages de modélisation pédagogique examinés, les interrogations nombreuses de la communauté EIAH sur ces standards, m'ont amené à considérer que les travaux n'étaient sans doute pas assez avancés pour conférer le statut de standard à IMS-LD notamment. Ces interrogations m'ont conduit à formuler des orientations de recherche dans le domaine de la modélisation qui sont présentées dans ce paragraphe. Ces orientations décidées dès le début de la thèse de Pierre Laforcade vont dans la même direction que celles formulées par G. Paquette dans un article consacré à MISA et IMS-LD (Paquette 2004a) : « The future of the major interoperability technical problems, through the implementation of international eLearning standards, will shift from media development to instructional engineering and pedagogical concerns. The greater availability of reusable digitized content, together with the larger set of instructional decisions to be made by instructional designers, will hopefully push forward the agenda for innovative engineering methods and tools. »

Le champ d'étude proposé concerne la description de situations-problèmes coopératives dans un contexte de formation instrumentalisée (via des plates-formes de FOAD à base de composants) : contrairement à IMS-LD ou MISA qui proposent des langages à spectre large, j'ai donc choisi de faire un travail en profondeur puisque je cible un type particulier d'apprentissage en articulation avec un dispositif informatique particulier.

Le projet est de rendre possible la description des situations-problèmes coopératives dans leur complexité :

- *cognitive* (modélisation des concepts que les apprenants vont devoir acquérir, des tâches qui leur sont assignées, des ressources et instruments permettant aux apprenants de manipuler les concepts à apprendre dans le cadre de ces tâches, ...),
- *structurelle* (modélisation de l'organisation générale structurant l'activité des apprenants et des tuteurs, décomposition d'activités générales en activités plus détaillées, description des événements pédagogiques significatifs permettant d'adapter les activités aux comportements constatés des apprenants, ...),
- *sociale* (modélisation des activités collectives, du rôle des acteurs dans ces activités, ...).

Dans ce but, je souhaite considérer les apports respectifs des approches de type « Instructional Design » et de type « Ingénierie de la connaissance ». Je propose d'effectuer

cette description à l'aide de modèles de la situation-problème considérée qui vont guider la conception sur deux niveaux :

- horizontalement : les modèles produits vont offrir différentes vues complémentaires de la situation : chaque vue, qui répond à un besoin d'explicitation que ressent le concepteur, a ses propres objectifs et capture des aspects particuliers de la situation (un même élément apparaissant dans plusieurs vues se voit doter de l'ensemble des propriétés spécifiées dans ces vues),
- verticalement : les modèles permettent de représenter la situation à différents niveaux de granularité/d'abstraction (des vues externes à la description fine du scénario envisagé et des activités proposées aux apprenants).

L'accent me paraît devoir être porté sur les phases amont de la description des situations-problèmes : l'enjeu majeur consiste à proposer des représentations permettant un dialogue entre les parties impliquées, notamment les membres de la communauté EIAH, les utilisateurs potentiels et les développeurs de composants et autres plates-formes de FOAD.

Compte tenu des éléments d'analyse présentés au paragraphe 2.2.2.3, le standard IMS-LD ne sera pas utilisé comme un langage de modélisation exploitable durant les phases amont de conception ; dans la suite des travaux présentés, IMS-LD sera considéré en tant que conteneur servant à déposer (donc à opérationnaliser) des références d'activités, de ressources et d'outils qui auront été conçus avec d'autres techniques de modélisation des situations-problèmes : Par opposition à IMS-LD qui considère les activités d'apprentissage comme des boîtes noires qu'il s'agit uniquement de référencer dans un scénario, l'approche proposée vise à décrire les activités dans le contexte précis de la situation d'apprentissage qu'elles servent.

De plus, comme les travaux concernent les phases amont, les propositions ne porteront pas sur des supports (méta-modèles et formalismes) pour la conception des interfaces Homme-Machine (IHM) à associer aux EIAH modélisés.

Afin d'être utiles, les travaux me semblent devoir être centrés sur la définition et l'usage de méta-modèles et formalismes acceptables à la fois par la communauté Génie Logiciel (en terme de syntaxe et de sémantique) et par la communauté EIAH (en terme de couverture du domaine, d'expressivité et d'utilisabilité). L'objectif étant de rendre possible le dialogue entre concepteurs, les informaticiens de la communauté EIAH ne sauraient donc être la seule cible des langages proposés. Les capacités d'expression des méta-modèles devront s'appuyer, quand cela est possible, sur les concepts proposés dans IMS-LD, mais devront aussi (voir Figure 2.11) :

- permettre de décrire les aspects cognitifs, structurels et sociaux d'une situation-problème coopérative ainsi que les processus d'apprentissage / de tutorat envisagés
- se baser sur les concepts des situations-problèmes, définissant les éléments d'une ontologie de ce domaine qui sera à la base des échanges entre concepteurs (Mizogushi et Bourdeau 2000), (Mizogushi 2002).

Les modèles produits avec de tels méta-modèles devront, au final, être exploitables par la communauté EIAH. Les usages classiques des méta-modèles sont envisagés afin de rendre certains modèles de la communauté EIAH opérationnalisables. Nous envisageons :

- la production d'outils à destination des concepteurs leur permettant de créer des modèles, de les réutiliser et de les échanger,
- la transformation des modèles produits par les concepteurs en d'autres modèles d'un niveau d'abstraction différent (par exemple, passage d'un espace du discours conceptuel à un espace technologique).

Du point de vue de l'ingénierie des modèles, la recherche proposée doit donc jouer un rôle important en établissant des ponts interdisciplinaires au sein de la communauté EIAH afin de profiter au mieux des avancées conceptuelles et technologiques de chaque discipline (propositions d'un cadre intégrateur solide).

Éléments de solutions à apporter	Problèmes soulevés quand on spécifie/conçoit des situations d'apprentissage avec IMS-LD			Concepts étudiés du métamodèle IMS-LD	Modèle d'information associable à une activité
<p>Rendre possible (et encourager) la description précise des activités (boîtes blanches) et de tout le contexte de leur mise en œuvre (objets manipulés, apprentissages escomptés, services logiciels attendus, ...)</p>	<p>Une activité est une boîte noire du point de vue du concepteur du scénario IMS-LD</p>	<p>Une activité est un conteneur pour associer dans un scénario des learning objects conçus par ailleurs</p>	<p>Impossible de spécifier les caractéristiques de l'environnement d'outils et des ressources à partir des besoins issus de la définition des activités à mener</p> <p>Impossible d'analyser une activité du point de vue des ressources produites / à produire par un apprenant ou un tuteur</p>	<p>Activity, Environment, Outcome, Notification, Property, Prerequisite, Learning-Objective</p>	<p>Voir détails de l'analyse en annexe A (figure A.1)</p>
<p>Rendre possible et encourager la description des responsabilités des acteurs impliqués dans les activités collaboratives identifiées : responsabilités sur les productions à effectuer, droits d'accès sur les ressources et les services logiciels, ...</p>	<p>IMS-LD offre des moyens pour décrire des processus s'exécutant en parallèle (play, activity-structure, role-part) mais n'offre pas de moyens suffisants pour décrire leur synchronisation</p> <p>Impossible de spécifier les types d'accès de chaque rôle impliqué dans des activités collaboratives que l'on conçoit</p>	<p>Une activité (collaborative) est un conteneur pour associer dans un scénario des rôles avec des environnements d'outils génériques conçus par ailleurs</p>	<p>Activity, Play, Role-Part, Role, Environment</p>	<p>Voir détails de l'analyse en annexe A (figure A.2)</p>	
<p>En relation avec la structuration du scénario, rendre possible et encourager la production de modèles décrivant les objets d'apprentissage à manipuler (propriétés et modalité d'évolution de l'état de ces propriétés), les compétences à acquérir dans le cadre des activités proposées, les principes d'adaptation du scénario aux comportements réels des acteurs</p>	<p>L'approche descendante utilisée pour analyser une situation d'apprentissage conduit à des scénarios favorisant la conception de micro-activités</p>	<p>Les modèles obtenus multiplient les références au concept d'Activity-Structure. Or, ce concept prend des significations différentes selon le niveau d'analyse auquel il est utilisé. L'utilisation de deux concepts différents auraient facilité l'interprétation des scénarios</p>	<p><u>Principes méthodologiques proposés pour structurer le scénario.</u> Method, Play, Act</p>	<p>Structuration et dynamique des scénarios IMS-LD</p> <p>Voir détails de l'analyse en annexe A (figure A.3)</p>	
<p>Encourager la production de diagrammes d'états attachés aux objets dynamiques apparaissant dans la conception, de diagrammes d'activités décrivant différents scénarios possibles pour un même macro-processus (représentée par un diagramme de use-case par exemple), ...</p>	<p>Le modèle événementiel est limité à des événements prédéfinis (portant sur les concepts de description d'un scénario) et à la prise en compte des valeurs prises par des propriétés simples définies par le concepteur (mais non attachées à des objets significatifs que le concepteur envisage de faire manipuler aux acteurs). Il n'est donc pas possible de décrire l'évolution des propriétés attachées aux objets significatifs de conception puisque ces objets sont seulement référencés et pré-existants</p> <p>Pour combler les lacunes du modèle d'information, les noms de propriétés portent une partie de la sémantique de ces propriétés</p>	<p><u>Principes méthodologiques proposés pour décrire la dynamique d'un scénario.</u> Property, Notification, Condition</p>			

Figure 2.11 : Éléments du cahier des charges d'un langage de modélisation pédagogique

2.3 Conclusion

L'activité de recherche sur le thème de la conception et de la mise en œuvre d'EIAH à base de situations-problèmes est riche et diversifiée. Dans ce chapitre, j'ai positionné les travaux menés au LIUPPA par rapport à ceux de la communauté EIAH tant du point de vue des plates-formes et environnements supports que du point de vue des modèles de description des situations-problèmes coopératives.

Du point de vue de la description de ces situations, l'approche proposée consiste à considérer les environnements supportant les situations-problèmes coopératives comme relevant des systèmes orientés support à la performance mais aussi des systèmes orientés Design Pédagogique (Tchounikine 2002a). Ceci explique pourquoi j'ai présenté les modèles de représentation orientés FOAD tels IMS-LD mais aussi d'autres modèles utilisés dans le domaine de l'apprentissage coopératif (CSCL), de l'apprentissage personnalisé (modèles de l'apprenant, modèle du domaine et modèle des interactions) et de l'ingénierie des connaissances.

Afin de concevoir, mettre en œuvre des environnements-supports pour ces situations-problèmes coopératives, j'ai expliqué en quoi les plates-formes à base de composants me paraissent constituer un point de départ prometteur ; j'ai alors ébauché une approche par les modèles visant à rapprocher et à contextualiser les modèles de composants offerts par ces plates-formes des modèles décrivant la situation d'apprentissage à mettre en œuvre. L'expérimentation de ces modèles et outils par des enseignants de terrain et des apprenants fait partie des éléments importants de la démarche qui s'appuie sur les retours d'usage pour guider la théorie.

Le chapitre suivant présentera les principaux résultats des travaux engagés et les perspectives de recherche qui découlent des résultats obtenus.

Chapitre 3

Résultats et Perspectives

Les deux premiers chapitres ont permis de situer mes travaux (objectifs et démarche) par rapport aux principales communautés et équipes de recherche nationales et internationales. Ce chapitre décrit de manière plus approfondie mes travaux, leur justification, les résultats obtenus et les propositions qui en découlent.

Dans une première partie, j'aborde les travaux que j'ai menés dans le domaine des modèles et méta-modèles de description des situations-problèmes coopératives : tout au long des projets que j'ai menés, j'ai pu expérimenter et proposer différents formalismes de représentation et des méta-modèles dont je décris les caractéristiques, les avantages et inconvénients du point de vue du concepteur des situations d'apprentissage.

Dans une deuxième partie, j'aborde le problème de l'exploitation des modèles décrivant une situation-problème. Les propositions qui s'inscrivent dans une perspective d'ingénierie dirigée par les modèles abordent : a/ le problème de la définition d'un processus de conception supportant les modèles et formalismes proposés, b/ le problème de la transformation des modèles de conception pour une mise en œuvre sur des plates-formes de formation à distance.

La dernière partie de ce chapitre est consacrée aux perspectives des travaux présentés : sont décrits successivement les pistes de recherche immédiates et le projet scientifique issu de ces travaux.

3.1 Travaux sur les modèles et méta-modèles utiles pour décrire des situations-problèmes

Mes travaux sur la modélisation des situations-problèmes ont été initiés dans (Nodenot 1995) ; les propositions figurant dans (Nodenot 2001) et (Nodenot, Laforcade et al. 2003) ont ensuite abordé respectivement les aspects coopératifs de ces situations d'apprentissage et leur mise en œuvre via des techniques de méta-modélisation. Les orientations proposées m'ont permis de mener / d'encadrer des travaux diversifiés de conception et mise en œuvre d'outils au service de l'apprentissage à base de situations-problèmes coopératives :

- dans le cadre de la thèse de Christophe Marquesuzaà (Marquesuzaà 1998), deux exemples de situations ont été étudiés : la situation GLOM (portant sur la modélisation des systèmes d'informations) et la situation BIO extraite de (Develay 1993) qui porte sur l'autorégulation par les êtres vivants de leurs fonctions respiratoires et circulatoires.
- dans le cadre de la thèse de Pierre Laforcade (Laforcade 2004), un travail important d'analyse et de conception d'outils a été effectué à partir d'une situation-problème coopérative : la situation Smash dont le point de départ est issu de (ACTIS 2001). Cette situation est présentée dans le paragraphe 3.1.2.1.

Ces expériences successives de développement ont nourri les réflexions qui ont conduit au sujet de thèse proposé à Pierre Laforcade : ces travaux conduits entre septembre 2001 et décembre 2004 ont eu pour résultat majeur la proposition d'un méta-modèle puis d'un profil UML dédiés à la conception des situations-problèmes coopératives.

Dans le paragraphe 3.1.1, je décris l'évolution de mes travaux en termes de formalismes et de modèles de description des situations-problèmes. Dans une deuxième partie (cf paragraphe 3.1.2), je présente les principales caractéristiques du profil UML appelé CPM (Cooperative Problem-based learning Metamodel) qui résulte de ces travaux.

3.1.1 Langages et formalismes pour la modélisation des situations-problèmes

Les travaux conduits depuis 1995 ont tenté de répondre à trois questions portant sur la conception d'EIAH à base de situation-problème :

1. Est-il possible d'exprimer des considérations pédagogiques suffisamment précises sur un EIAH à produire pour que des développeurs puissent ensuite exploiter les modélisations réalisées ?
2. Quels sont les formalismes et types de modèles les plus adéquats pour exprimer un point de vue d'ordre pédagogique sur un EIAH à produire ?
3. Est-il possible de proposer à des usagers non informaticiens des vues externes (langage de surface) pour qu'ils interprètent/créent des modélisations respectant les formalismes issus du point 2 ?

3.1.1.1 Travaux autour du langage « Spec »

Pour spécifier des logiciels éducatifs, j'ai utilisé durant ma thèse un langage proposé par (Berzins et Luqi 1990) pour la modélisation de systèmes complexes : le langage « Spec ». Ce langage basé sur les types abstraits de données et la logique des prédicats d'ordre 1 permet de spécifier formellement des systèmes à différents niveaux d'abstraction (y compris des systèmes fortement interactifs et temps-réel) via une notation mathématique : les variables qui apparaissent dans une spécification « Spec » expriment donc des modèles mathématiques d'un système d'informations ; ces modèles peuvent représenter l'environnement d'un système, ses buts, son comportement dynamique, les services qu'il offre ou qu'il utilise, son architecture.

<pre> DEFINITION sécurité_ascenseur INHERIT ascenseur -- le module "ascenseur" définit les concepts "ascenseur", "étage", "position", "porte_ouverte", et "arrêté". </pre>	<pre> CONCEPT contrainte_de_sécurité : boolean WHERE contrainte_de_sécurité <=> ALL(a : ascenseur, e : étage :: porte_ouverte(a, e) => position(a) = e and arrêté(a)) -- la porte ne peut être ouverte que si l'ascenseur est arrêté à un étage. END </pre>
---	--

Figure 3.1 : Un exemple de modélisation « Spec »

« Spec » a été expérimenté dans le cadre du développement d'un environnement d'apprentissage portant sur l'acquisition de techniques efficaces pour le calcul mental¹⁴. Le langage nous a permis de proposer des modèles couvrant les phases de spécification et de conception, tant du point de vue éducatif que du point de vue fonctionnel (ou architectural). Pour chacune de ces phases, les descriptions proposées couvrent le modèle du domaine, le modèle de l'apprenant, le modèle du tuteur :

¹⁴ Procéder de manière efficace pour calculer l'expression : $154+25+36-10$ suppose une maîtrise des opérations d'association et de commutation. Les solutions auxquelles on souhaite amener progressivement l'apprenant sont de type $154+25+36-10=190+15 = 205$.

<pre> DEFINITION Modèle-domaine INHERIT Vue-expression INHERIT Vue-opération INHERIT Opérations-sur-expression -- description des buts au niveau du modèle pédagogique CONCEPT trouver-suivants(e : expression-à-calculer ; op-ok : op-autorisées) VALUE (ens : ensemble-opération) WHERE (ALL o : opération :: o in ens => type(o) in op-ok and (o in commutations(e) or o in associations(e) or o in décomposer-à-dix-par-plus(e) or o in décomposer-à-dix- moins(e) or o in effectuer-calcul-simple(e)) par-) goal(trouver-suivants, Modèle-pédagogique) CONCEPT trouver-meilleur-suivant(e : expression-à-calculer ; op-ok : op-autorisées) VALUE (o : opération) WHERE est-meilleur-suivant(o, e, op-ok) goal(trouver-meilleur-suivant, Modèle-pédagogique) -- le concept suit permet de savoir si deux opérations se succèdent dans une liste d'opérations CONCEPT suit(o1 : opération, o2 : opération, s : liste-opération) VALUE (b : boolean) -- vrai si o1 suit o2 dans la liste s. </pre>	<pre> CONCEPT calculer-au-mieux(e : expression-à-calculer ; op-ok : op-autorisées) VALUE (s : liste-opération) WHERE (ALL s' : liste-opération SUCH THAT s' = calculer(e, op-ok) :: length(s) <= length(s')) goal(calculer-au-mieux, Modèle-pédagogique) CONCEPT calculer(e : expression-à-calculer ; op-ok : op-autorisées) VALUE (s : liste-opération) WHERE (ALL o1, o2 : opération SUCH THAT o1 in s and o2 in s and suit(o2, o1, s) :: appliquer(o2, jusqu'à(o1, s, e)) = jusqu'à(o2, s, e) and jusqu'à(s[length(s),s, e]) in opérande-simple goal(calculer, Modèle-pédagogique) CONCEPT jusqu'à(op : opération, s : liste-opération, e : expression-à-calculer) VALUE (e' : expression-à-calculer) WHERE op in s -- renvoie l'expression obtenue en appliquant la liste des opérateurs jusqu'à celui qui est spécifié dans les paramètres. CONCEPT est-meilleur-suivant (o : opération, e : expression-à-calculer ; op-ok : op-autorisées) VALUE (b : boolean) WHERE b <=> o in trouver-suivants(e, op-ok) and (ALL o' : opération SUCH THAT o' in trouver-suivants(e, op-ok) and o' <> o :: length(calculer-au-mieux(appliquer(o',e), op-ok)) >= length(calculer-au-mieux(appliquer(o,e), op-ok))) END </pre>
--	---

Figure 3.2 : Une description « Spec » pour le modèle du domaine de l'application Calcul Mental

Concernant le modèle du domaine, « Spec » a permis de spécifier :

1. ce qu'est une expression à calculer, un opérateur (+,-), un opérande ;
2. ce que sont les opérations privilégiées sur une expression (commutation/association de deux opérands, décomposition d'un opérande sous la forme d'une somme ou d'une différence faisant intervenir un opérande multiple de dix, calcul sur deux opérands) ;
3. les buts du système c'est-à-dire les fonctions que le système doit implémenter pour résoudre les calculs demandés par ailleurs aux apprenants : trouver toutes les expressions qui peuvent résulter de l'application d'une opération privilégiée, trouver l'opération qui, si on l'applique à une expression, amène vers une solution minimisant le nombre d'opérations, calculer une expression en appliquant une suite d'opérations, calculer une expression en utilisant une suite minimale d'opérations parmi les opérations privilégiées, ...

Concernant le modèle de l'apprenant, « Spec » a permis de modéliser : a/ les types d'opérations (commutation, association, calcul simple) que sait utiliser un apprenant, ceux qu'il sait utiliser à bon escient ; b/ le nombre d'opérandes à partir duquel il est en difficulté, l'ordre de grandeur des opérandes qu'il sait manipuler, le niveau d'aide dont il a besoin pour résoudre les problèmes qui lui sont posés.

Concernant le modèle du tuteur, « Spec » a permis de modéliser comment, en fonction des réussites et des échecs successifs des apprenants, le système va générer la prochaine expression à calculer soumise à l'apprenant, quel niveau d'aide on va lui fournir durant la résolution, ...

Les caractéristiques de « Spec », à savoir : 1/ son niveau d'abstraction qui permet de fabriquer les types abstraits utiles pour un domaine donné, 2/ la possibilité d'utiliser la logique des prédicats pour contraindre ou caractériser un type abstrait, 3/ le modèle événementiel proposé via la notion de *Machine* pour décrire le comportement d'un type abstrait (changement d'état) et ses relations avec d'autres types abstraits, ont largement facilité l'activité de modélisation puis de codage. Le modèle du domaine a d'abord été entièrement codé et testé, ce qui a permis de doter l'application de la capacité de résoudre les problèmes soumis à l'apprenant, quelles que soient les expressions à calculer. Le modèle du tuteur et l'interface-usager qui ont ensuite été codés s'appuyaient sur les fonctionnalités des modèles du domaine et de l'apprenant.

J'ai poursuivi mes travaux sur le langage « Spec » après ma thèse lorsque j'ai initié mes recherches sur les situations-problèmes. Partant de la description de situations-problèmes issues de (Develay 1993), j'ai formalisé dans (Nodenot 1995) des types abstraits pour spécialiser « Spec » en terme de vocabulaire et d'éléments de langage : les types abstraits permettaient de :

- formuler la connaissance déclarative à enseigner, formuler le registre de conceptualisation que les élèves doivent atteindre, formuler les objectifs d'apprentissage et prévoir les obstacles auxquels les apprenants risquent d'être confrontés, décrire la consigne et les ressources proposées aux apprenants,
- décrire les tâches proposées aux apprenants en termes de buts à atteindre, de langage de commande utilisable par l'apprenant pour accomplir la tâche qui lui est proposée, d'éléments de résolution de la tâche formulés à l'aide de ce même langage de commande.
- ...

<pre> DEFINITION Resolution-problème is IMPORT Subtype from TYPE IMPORT Problème from Taches CONCEPT Fixer-plan-resolution(P : Problème) VALUE(L : Liste-Etapes) WHERE ALL(E : Etape :: E in L => SOME(P1 : Problème SUCH THAT P1 in P AND Appliquer(E,P1)=Solutionner(P))) </pre>	<pre> CONCEPT Liste-Etapes : TYPE WHERE Subtype(Liste-Etapes : sequence(Etape)) CONCEPT Etape : type CONCEPT Appliquer(E : Etape, P : Probleme) VALUE(P1 : Probleme) CONCEPT Solutionner(P : Probleme) VALUE(P1 : Probleme) WHERE P1 = { } END </pre>
---	---

Figure 3.3 : Un type abstrait « Spec » définissant une procédure de résolution de problème

Mes travaux à partir du langage « Spec » comme ceux de (Miao 2000) qui sont basés sur l'exploitation du langage formel « Z » (Barden, Stepney et al. 1994) semblent démontrer que des langages de spécification capables d'exploiter des types abstraits, la logique des prédicats et un modèle événementiel permettent de spécifier de manière précise une application éducative.

3.1.1.2 Travaux visant à définir des modèles compréhensibles par des non informaticiens

Des modèles basés sur les langages « Spec » ou « Z » sont cependant difficiles à produire et ne facilitent pas les échanges entre enseignants et informaticiens. Durant ma thèse, j’ai donc exploré les possibilités d’offrir un ensemble de points de vue permettant à des non spécialistes de manipuler des modélisations « Spec ». Me basant sur les travaux de (Schoemaker, Nienhuis et al. 1990), j’ai essayé de proposer trois types de vues graphiques sur des modélisations « Spec » : la vue des données, la vue des comportements et la vue des activités. Pour chacune de ces vues, j’ai défini des règles de transformation (Nodenot 1992a) permettant de transformer les vues graphiques en équivalents « Spec » en me basant à l’époque sur le modèle Entité-Association pour la vue des données, les Statecharts pour la vue des comportements et les Diagrammes de Flots de Données pour la vue des activités. Les difficultés de nature sémantique rencontrées pour synchroniser¹⁵, sur des exemples réels, ces vues et le modèle « Spec » sous-jacent m’ont conduit à abandonner cette voie et à orienter ensuite les travaux vers des modèles basés sur des formalismes graphiques.

Le sujet de thèse proposé à Christophe Marquesuzaà a donc permis d’aborder le problème des formalismes de manière radicalement différente, l’idée étant de partir d’un langage minimal de type « graphe conceptuel » qui est facilement manipulable par un enseignant ; et de doter ce langage d’une sémantique particulière, en fonction de l’étape et des objectifs de conception de l’enseignant.

(Jonassen, Beissner et al. 1993) définissent les graphes conceptuels comme des « representations of concepts and their interrelationship that are intended to represent the knowledge structures that humans store in their minds » (Novak et Wandersee 1991), (Baron, Bruillard et al. 1999), (Novak 2002) ont fait des études très complètes sur ce mode de représentation de la connaissance et sur les applications exploitant les graphes conceptuels. Les travaux référencés ont montré que les graphes conceptuels sont des outils cognitifs qui permettent de décrire à la fois les connaissances déclaratives (Jacobi 1991) et les connaissances procédurales (Tochon 1990) d’un acteur.

La Figure 3.4 présente une description UML du méta-modèle d’un graphe conceptuel.

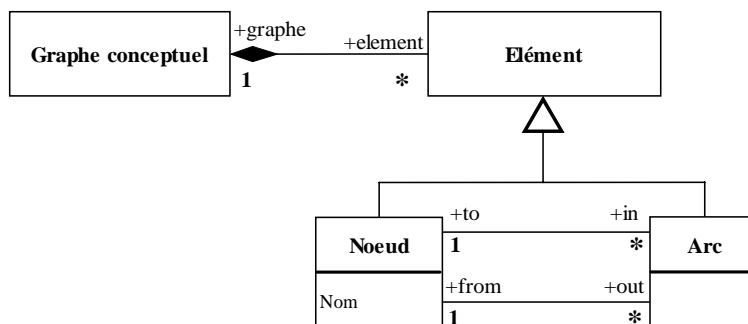


Figure 3.4 : Méta-modèle d’un graphe conceptuel

Dans le cadre de la thèse de Christophe Marquesuzaà, les graphes conceptuels ont été utilisés de deux manières. Tout d’abord pour doter les enseignants d’un langage leur permettant de concevoir des situations-problèmes : pour chacune des étapes de la conception, une sémantique particulière des nœuds et des arcs a été définie afin de permettre aux enseignants de représenter simplement la matrice disciplinaire qui définit le contexte de la situation-problème, les objets d’apprentissage, les représentations possibles

¹⁵ A cette fin, j’ai conçu un pattern de type MVC (Modèle-Vue-Contrôleur) synchronisant différentes vues d’une même spécification « Spec » qui a été repris ensuite dans la thèse de Christophe Marquesuzaà (Marquesuzaà et Nodenot 1996) et dans (Nodenot, Marquesuzaà et al. 2002).

des apprenants sur ces objets, les situations et les consignes permettant de lever les représentations erronées, ... Les enseignants créent donc des modèles (voir Figure 3.5 et Figure 3.6) exploitant des spécialisations du méta-modèle de la Figure 3.4 implantées dans l'environnement d'outils proposé. Les graphes conceptuels sont utilisés ici pour guider et rendre compte du processus de conception, ils n'ont pas de visée computationnelle.

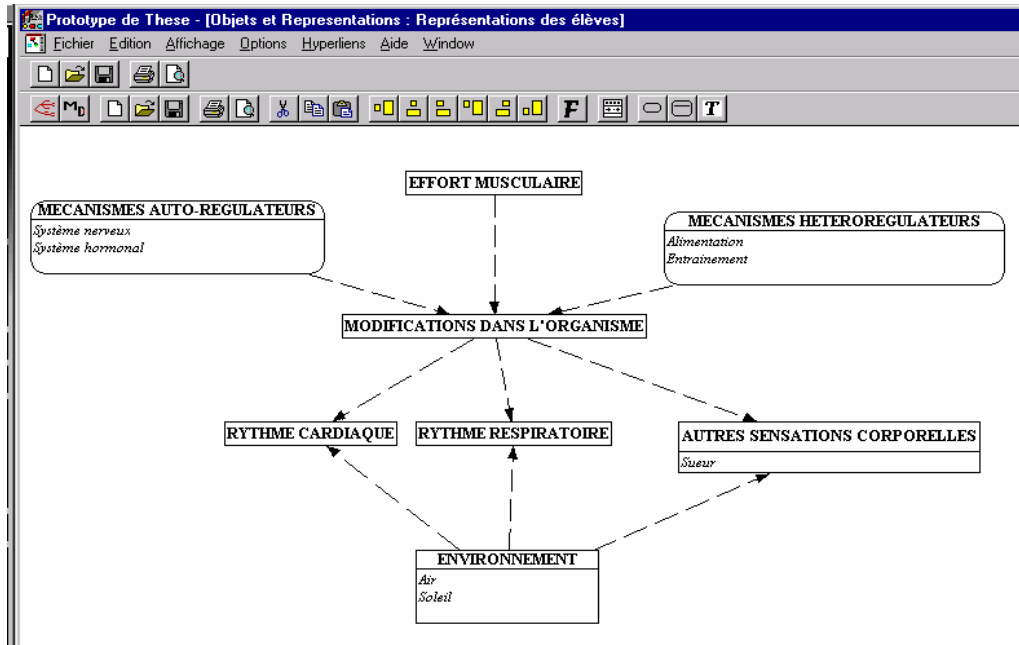


Figure 3.5 : Les objets d'apprentissage de la situation-problème BIO (Marquesuzaà 1998)

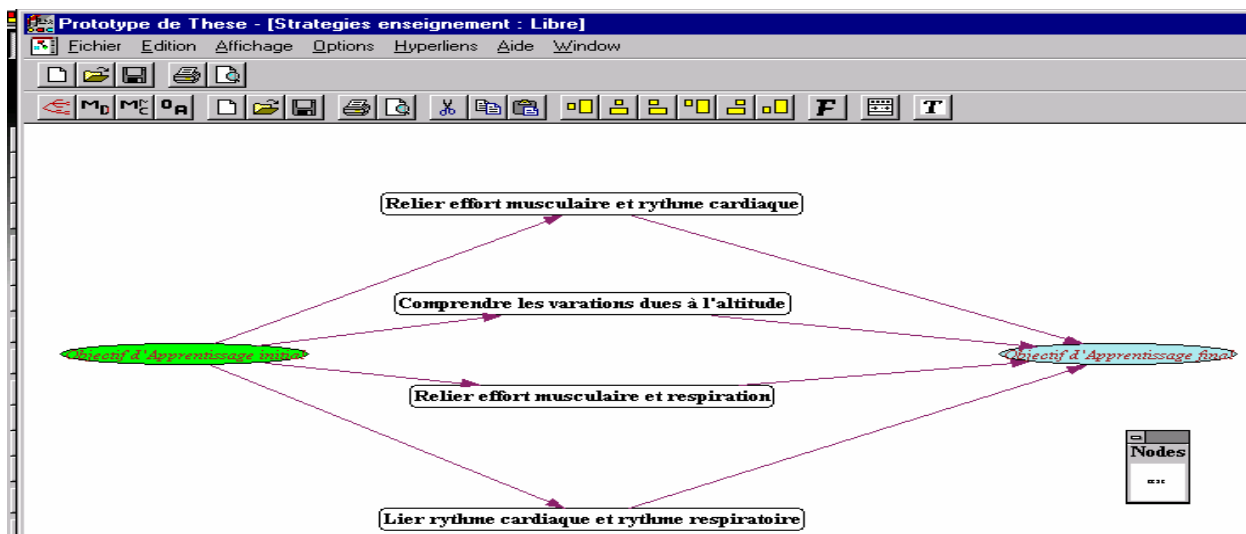


Figure 3.6 : Les objectifs intégrés dans la situation-problème BIO (Marquesuzaà 1998)

Les graphes conceptuels ont aussi été utilisés pour permettre aux enseignants de définir le langage de représentation qu'ils souhaitent offrir aux apprenants pour se représenter le domaine, pour formuler des hypothèses, pour organiser l'activité de résolution du problème. Les enseignants spécialisent donc eux-mêmes le méta-modèle des graphes conceptuels de la Figure 3.4 et les modèles produits par les apprenants permettent par exemple de rendre compte d'une hypothèse particulière portant sur un problème donné, de confirmer ou contredire cette hypothèse sur la base d'une analyse de ressources, ... La Figure 3.7 présente l'exemple d'une telle spécialisation du méta-modèle qui définit un langage que les apprenants pourront utiliser pour conduire la résolution d'une situation-problème (les spécialisations de la méta-classe Nœud sont représentées par des rectangles blancs et les spécialisations de la méta-classe Arc par des rectangles grisés).

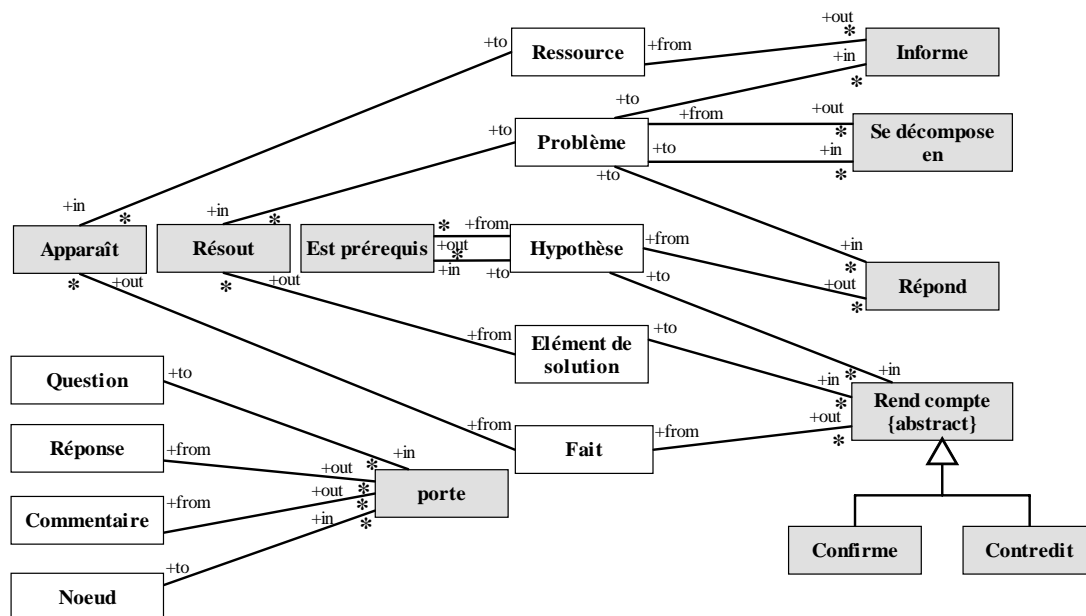


Figure 3.7 : Spécialisation du méta-modèle d'un graphe conceptuel définissant un langage que les apprenants pourront utiliser pour analyser une situation-problème

Ces travaux ont donné lieu au développement d'un environnement de conception réalisé durant la thèse de Christophe Marquesuzaà. L'implémentation de cet environnement s'appuyait sur un outil Meta-Case appelé Hardy (Smart et Rae 1995) paramétrable grâce à un gestionnaire de diagrammes qui est un outil permettant de définir les types de diagrammes autorisés pour une étape donnée. L'environnement était couplé à un générateur de systèmes experts (CLIPS) qui vérifiait au fur et à mesure la cohérence des diagrammes produits par les concepteurs en fonction des contraintes d'utilisation des outils et de la sémantique des nœuds et arcs proposés par la palette d'outils. L'outil permettait également de générer des documents rendant compte de la spécification effectuée (Marquesuzaà, Meyranx et al. 1997).

Cet environnement d'outils à base de graphes conceptuels a été expérimenté avec plusieurs enseignants qui n'ont pas vraiment utilisé l'outil pour concevoir des situations-problèmes nouvelles mais s'en sont servi pour documenter, stabiliser et garder trace de situations-problèmes imaginées préalablement. La très grande facilité de paramétrage de l'environnement Hardy nous a permis de faire évoluer rapidement le prototype en fonction des retours d'usage et des souhaits des enseignants. Nos propres expérimentations nous ont conduits à utiliser cet environnement en tant qu'outil de réflexion, les modèles réalisés étant des modèles très amont qui ont toute leur utilité pour guider la conception –modèles de type base du design au sens de (Baker 2000) – mais se révèlent ensuite souvent peu utiles pour opérationnaliser un EIAH exploitant de telles situations¹⁶. Ceci s'explique par le fait que le formalisme des graphes conceptuels :

- peut facilement être doté de stéréotypes et de contraintes (sur les nœuds et les arcs) qui permettent de spécialiser ce langage pour des usages pédagogiques (côté apprenant et enseignant-concepteur), sans pour autant changer le méta-modèle de la Figure 3.4. Les spécialisations faites dans le cadre de la thèse de Christophe Marquesuzaà ont permis de proposer un langage de modélisation proche des préoccupations des enseignants.
- est un formalisme qui se prête bien à la production de représentations graphiques qui permettent à un acteur d'illustrer ses idées et certains aspects d'une situation d'apprentissage qu'il veut mettre en évidence. L'illustration est forcément sélective ce qui en fait une activité assez informelle et dynamique mais l'illustration est cependant

¹⁶ Ceci ne remet pas en cause l'utilité et la facilité d'usage des graphes conceptuels en tant que langage de commande des apprenants ou des concepteurs (cf l'exemple de spécialisation du méta-modèle de graphe conceptuel proposé dans la Figure 3.7).

contrôlée puisqu'à chaque type de modèle (modèle de la matrice disciplinaire, modèle des objets d'apprentissage, modèle de la situation, ...) sont associées des règles de bon usage des arcs et des nœuds (que fait respecter l'environnement de conception développé par (Marquesuzaà 1998)).

- o est un formalisme qui se prête assez mal pour construire une spécification complète et détaillée d'une situation d'apprentissage. Notre expérience a montré que ce formalisme est commode pour produire des modèles statiques mais assez peu pour produire des modèles dynamiques (description du comportement d'un acteur ou d'un concept évoluant en fonction d'événements internes ou externes) : les modèles d'une situation-problème obtenus avec l'environnement Hardy sont donc proches des représentations ontologiques que d'autres auteurs (Desmoulin et Grandbastien 2002), (Bessagnet 2005) mettent en œuvre avec d'autres environnements, par exemple Protégé (Fridmann Noy, Ferguson et al. 2000).

3.1.1.3 Directions de recherche résultant de ces travaux

Les retours d'expérience sur : a/ l'utilisation du langage « Spec » (années 90-95) puis des graphes conceptuels (années 96-99) pour modéliser des situations d'apprentissage et b/ les langages de modélisation pédagogiques existants (cf IMS-LD et MOT présentés au paragraphe 2.2.2), ont nourri la réflexion qui a conduit, en 2001, au sujet de thèse proposé à Pierre Laforcade. En ce qui concerne les types de modèles que doit couvrir un langage de modélisation de situations-problèmes, plusieurs éléments me paraissent devoir être pris en compte (cf points 1 à 6) :

1. Les modèles de conception avancée que produisent les concepteurs via un langage comme IMS-LD masquent des prises de décisions plus amont qu'il est important de documenter via des modèles. Un langage de description de situations-problèmes doit permettre de relier les modèles amont et les modèles de conception avancée de façon à organiser et à guider le processus de conception (cf les modélisations « Spec » couvrant les phases de spécification et conception).
2. Quel que soit le niveau d'abstraction auquel on se situe, les modèles doivent pouvoir rendre compte des aspects statiques et dynamiques des éléments décrits. Nos expérimentations ont montré que la description des aspects dynamiques (modèles d'activité, modèles de comportement) permettait d'étudier des aspects d'une situation d'apprentissage (notamment les événements pédagogiques pertinents et les relations dynamiques entre rôles et acteurs) non pris en compte par un modèle statique à la IMS-LD (cf modélisations de niveau conception réalisées avec le langage « Spec », cf la préconisation IMS-LD consistant à utiliser les diagrammes d'activités UML pour préparer la modélisation IMS-LD au niveau B),
3. Les modèles de scénarios ne sont que des modèles parmi d'autres qu'il ne faut pas forcément privilégier lors de la conception amont. Il est important de pouvoir offrir différentes vues sur un même élément afin de prendre en compte ses différentes facettes et favoriser ainsi une ingénierie des connaissances portant sur une situation d'apprentissage.

En ce qui concerne les modalités de représentation que doit offrir un tel langage, trois considérations me paraissent ressortir des travaux menés ou étudiés :

4. Les représentations graphiques facilitent le dialogue, la compréhension et l'implication des intervenants (cf travaux sur les modélisations de type MOT et les modélisations à base de graphes conceptuels),
5. Il est important d'appuyer les modèles sur des éléments et un vocabulaire compréhensibles par les enseignants. Le domaine de l'apprentissage à base de situations-problèmes étant un domaine à part entière qui dispose de sa propre terminologie (cf paragraphe 1.2), le langage de modélisation doit s'appuyer sur cette terminologie (cf terminologies IMS-LD et MOT, cf terminologie proposée par C. Marquesuzaà pour spécialiser des graphes conceptuels),

6. Il est important de pouvoir ajouter de la sémantique aux représentations graphiques pour contraindre les éléments d'un modèle que la représentation graphique ne permet pas de modéliser. Ces assertions et autres prédicats permettent de raisonner sur les éléments d'un modèle (*cf usage des prédicats « Spec »*).

Ces considérations (1 à 6) m'ont amené à envisager un langage de modélisation basé sur UML. Les justifications d'UML sont multiples : a/ UML permet de produire des modèles à différents niveaux d'abstraction à partir d'un ensemble de diagrammes et assertions en OCL (Object Constraint Language). b/ Lors de la conception, UML ne privilégie pas un type de modèle au détriment des autres, les vues dynamiques (diagrammes d'activités, Statecharts) étant aussi significatives que les vues statiques (diagramme de classe). c/ UML me paraît également se justifier par le fait que c'est un standard ayant déjà fait ses preuves dans des domaines divers : la conception des EIAH paraît donc pouvoir bénéficier de l'expérience acquise autour de ce langage et de l'outillage mis au point par la communauté UML (Sallaberry, Nodenot et al. 2002), (Bessagnet, Marquesuzaà et al. 2002).

Toutefois, UML est un langage très général de conception de logiciels et les éléments de ce langage ne peuvent pas être simplement transposés pour d'autres usages sans une réflexion sur la sémantique des éléments du langage. L'approche préconisée dans le sujet de thèse proposé à Pierre Laforcade consiste donc à spécialiser UML via le mécanisme de méta-modélisation : le sujet de thèse porte sur la définition d'un profil UML¹⁷ pour la conception des situations-problèmes coopératives en vue de leur mise en œuvre sur des plates-formes à base de composants. Le public ciblé pour exploiter un tel profil n'est donc pas directement un enseignant-concepteur mais plutôt un ingénieur-logiciel spécialisé dans le domaine de l'ingénierie éducative (et travaillant en liaison étroite avec une équipe pluridisciplinaire de conception qui devra pouvoir comprendre et interpréter les modèles produits).

3.1.2 Un méta-modèle et un profil UML pour décrire des situations-problèmes coopératives

Dans le chapitre 2, j'ai décrit les modèles conceptuels d'IMS-LD et de MOT/MISA qui m'ont servi pour expliciter la sémantique des méta-modèles proposés. Les modèles conceptuels présentés dans les figures 2.3 et 2.5 sont donc des abstractions du cœur des méta-modèles de MISA (MOT) et IMS-LD. Ici, je présente tout d'abord les principes de la situation-problème Smash ; les éléments de cette situation sont ensuite utilisés dans le paragraphe 3.1.1.2 pour mettre en évidence les éléments du modèle conceptuel sur lequel s'appuie le profil UML proposé pour décrire les situations-problèmes coopératives.

3.1.2.1 La situation Smash

Smash est une situation-problème conçue pour initier les enfants (cycle 3 : 8-11 ans) au code de la route et leur faire appréhender la sécurité routière pour les voitures comme pour les cyclistes et les piétons. Les enfants sont encouragés à réfléchir sur leur sécurité et leurs responsabilités vis-à-vis d'eux-mêmes et des autres. Voici quelques éléments caractéristiques de la situation Smash :

- Une mise en situation authentique : un cycliste (même âge que les élèves) est renversé par une voiture dans le quartier du village Mille-Fleurs. Il s'agit de mener une enquête sur cet accident. Les objectifs donnés aux apprenants sont de reconstituer l'accident et de trouver le coupable. La tâche est authentique car elle se base sur les enquêtes policières et une analyse de témoignages : indices à extraire, informations essentielles ou superflues, recoupements difficiles entre témoignages incomplets, etc.). Les apprenants jouent le rôle d'inspecteurs de police, tandis que le tuteur joue le rôle du chef

¹⁷ La notion de profil UML qui sera présentée plus en détail dans le paragraphe 3.1.2.3. permet de spécialiser le langage UML pour un domaine particulier grâce à l'utilisation de stéréotypes, valeurs marquées et contraintes spécifiques.

de la police. Les rôles sont assignés pour la totalité de la tâche et correspondent à des métiers authentiques.

- Les ressources ou matériels : les apprenants ont à leur disposition des témoignages relevés après l'accident. Ces témoignages (9 au total) concernent des acteurs passifs (un piéton ou encore un conducteur arrêté à un stop) comme des acteurs impliqués dans l'accident (le conducteur de la voiture qui a percuté le vélo, le propriétaire de la voiture mal garée, ...). Les autres ressources mises à disposition des apprenants sont une carte complète du village des Mille-Fleurs, un plan du quartier concerné par l'accident ; une fiche bilan du code de la route, des feuilles de notes vierges, des personnages et des panneaux à positionner sur les cartes.
- La scénarisation (système de contraintes) va permettre de garantir que les apprenants rencontrent l'obstacle qui est à la base des apprentissages à réaliser. Ayant des témoignages partiels et différents, les élèves vont aboutir à des conclusions différentes (principes du jigsaw –cf <http://www.jigsaw.org>) qu'il va falloir comprendre, recouper afin d'arriver à une solution cohérente pour tous. Une organisation du groupe doit être mise en place pour surmonter l'obstacle, émettre des hypothèses et les vérifier, intégrer des éléments non signalés dans les témoignages comme le code de la route, l'évaluation des risques liés à la fatigue, à la vitesse, aux conditions météo, les règles de savoir-vivre au volant (respect d'autrui), ...

Cette situation-problème coopérative a été analysée, expérimentée dans une classe de CM2 dans le cadre du stage de DESS de Cécile Oudot (Oudot 2003) avant d'être insérée dans une organisation pédagogique plus globale pour approfondir, évaluer et transférer les connaissances acquises. Cette action a été menée avec l'enseignant qui a expérimenté la situation Smash dans sa classe, les documents produits ayant ensuite été repris dans le cadre d'un travail mené avec l'Inspection Départementale de l'Education Nationale (IDEN) de Lourdes (voir l'annexe B de ce manuscrit qui propose des extraits des documents utilisés pour exploiter cette situation d'apprentissage en classe).

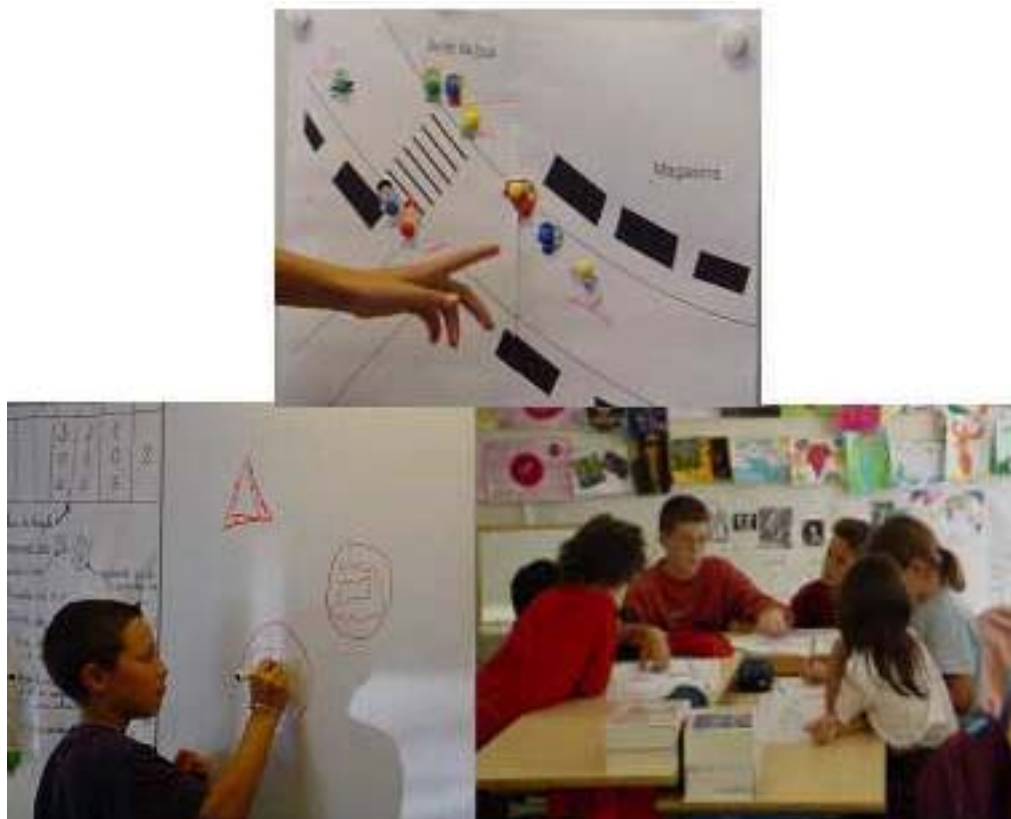


Figure 3.8 : Expérimentation en classe de la situation-problème Smash

3.1.2.2 Les éléments du modèle conceptuel

Comme le paragraphe 1.2 du premier chapitre l'a montré, l'apprentissage à base de situations-problèmes diffère par bien des points d'autres méthodes d'apprentissage. Dans la situation Smash comme dans d'autres situations-problèmes, l'attention est portée sur l'apprenant qui est confronté à des problèmes authentiques proposés par les concepteurs (dans le cas de la situation Smash, les apprenants sont mis en situation d'enquêteurs et essaient de comprendre les raisons d'un accident de vélo qui aurait pu arriver à l'un ou l'autre d'entre eux). Les activités pédagogiques (souvent coopératives) dans lesquelles les apprenants s'engagent sont donc organisées autour de ce problème à résoudre, des tuteurs pouvant éventuellement jouer le rôle de facilitateurs et donc mener également des activités. Ce processus de résolution d'une situation-problème conduit :

- à l'acquisition de connaissances et de compétences que chaque apprenant va pouvoir ensuite transférer pour résoudre des problèmes similaires (activités de renforcement et de transfert) : dans le cas de la situation Smash, il y a des connaissances et des compétences disciplinaires liées au code de la route, à la physique (énergie cinétique et distances de freinage, ...), à l'instruction civique (droits et devoirs des conducteurs, des piétons, règles de savoir-vivre). Il y a aussi des connaissances et compétences transdisciplinaires liées à la démarche de résolution de problème (mise en place et vérification d'hypothèses, recherche d'indices dans les témoignages, recoupement de témoignages, ...),
- à la construction de connaissances et de démarches partagées/acceptées par les apprenants : dans le cas de la situation Smash, plusieurs types d'*instruments* supportent l'enquête menée : des outils (outil de type tableau-blanc, outil permettant de mener des expérimentations sur les distances de freinage), des documents (témoignages mis à disposition des enquêteurs, procès verbaux créés par les enquêteurs, scénario plausible de l'accident créé par les enquêteurs, code de la route mis à disposition des enquêteurs, ...), un langage permettant d'échanger des idées sur la situation Smash, de mener une réflexion en commun pour s'entendre sur les activités à mener (répartition des tâches à conduire pour déterminer le niveau de responsabilité du cycliste dans l'accident) ou sur une connaissance particulière à partager (langage pour décrire la trajectoire d'un véhicule, sa vitesse, ...).

Le groupe (*communauté* au sens de la théorie de l'activité) est composé d'acteurs qui par leurs activités vont avoir une influence sur le processus de résolution du problème. Les acteurs n'ont pas tous les mêmes savoirs et savoir-faire, n'ont pas forcément les mêmes buts d'apprentissage (trouver le coupable de l'accident, connaître le code de la route, comprendre comme l'accident aurait pu être évité, ...), certains pouvant fournir une assistance aux autres et se sentir plus à l'aise dans telle ou telle activité utile à la résolution du problème. Les acteurs vont donc avoir un rôle (qu'ils s'assignent ou qu'on leur assigne) auquel sont attachées des activités. Les activités ne sont donc pas toutes conduites en même temps et une organisation du travail (*division du travail* au sens de la théorie de l'activité) est nécessaire. Cette organisation peut être à la charge du groupe qui résout le problème mais elle peut aussi incomber aux tuteurs (durant la phase de résolution du problème) voire aux concepteurs s'ils souhaitent privilégier (proposer/imposer) certains types d'organisation au sein des groupes d'apprenants. Une fois définies, les règles d'organisation vont venir contraindre les activités et les instruments assignés à un rôle.

Dans le cadre de la théorie empirique de l'apprentissage sur laquelle s'appuient ces travaux, l'organisation du travail est partagée entre les concepteurs d'une situation-problème et ses acteurs (apprenants, tuteurs). Contrairement à l'apprentissage à base de projets (project Based Learning) pour lequel l'organisation du projet par les apprenants est essentielle, l'apprentissage à base de situations-problèmes tel qu'il est envisagé ici vise autant des objectifs disciplinaires que des objectifs transdisciplinaires. Ceci explique que dans la situation Smash :

- une organisation de type jigsaw soit imposée aux apprenants par les concepteurs ; cette organisation conduit à une scénarisation partielle de la résolution de cette situation-

problème basée sur des conflits des représentations sur la position des acteurs au moment de l'accident : les apprenants n'ont qu'une partie des témoignages car l'approche pédagogique vise à éviter leur surcharge cognitive pour mieux approfondir le sens de chacun des témoignages. Ce scénario initial ne fige pas totalement l'organisation du travail, les apprenants ayant de nombreux degrés de liberté pour choisir qui va conduire et comment vont s'organiser les activités complexes s'inscrivant dans ce scénario de départ. Notons enfin que pour la situation Smash, les concepteurs ont prévu des organisations de travail différentes (approche analytique des témoignages, approche synthétique partant de l'accident, ...), les apprenants ayant à comprendre, choisir et adapter les scénarios proposés.

- des scénarios d'enseignement plus classiques soient prévus pour renforcer les connaissances de chaque apprenant et les transférer vers d'autres problèmes mettant en jeu la route et ses dangers.

Compte tenu de ces éléments, le modèle conceptuel proposé dans la Figure 3.9 fait apparaître trois blocs représentés par des numéros et des fonds colorés différents : le bloc n°1 porte sur la description des acteurs impliqués et de leurs rôles contextualisés par rapport à la situation-problème à résoudre ; le bloc n°2 porte sur l'organisation du travail (règles contraignant la réalisation d'activités par les différents rôles) liée à la résolution de la situation-problème coopérative ; le bloc n°3 porte sur les ressources (instruments) manipulées par les acteurs. Quel que soit le bloc considéré, une partie des éléments est fixée lors de la conception de la situation-problème, alors que d'autres éléments peuvent être définis par les acteurs eux-mêmes lors de la résolution de la situation-problème.

Les associations entre les concepts apparaissant dans ces différents blocs permettent d'exprimer l'idée selon laquelle : dans une situation-problème coopérative, un acteur joue différents rôles en s'impliquant dans des activités organisées qui l'amènent à utiliser et créer des ressources.

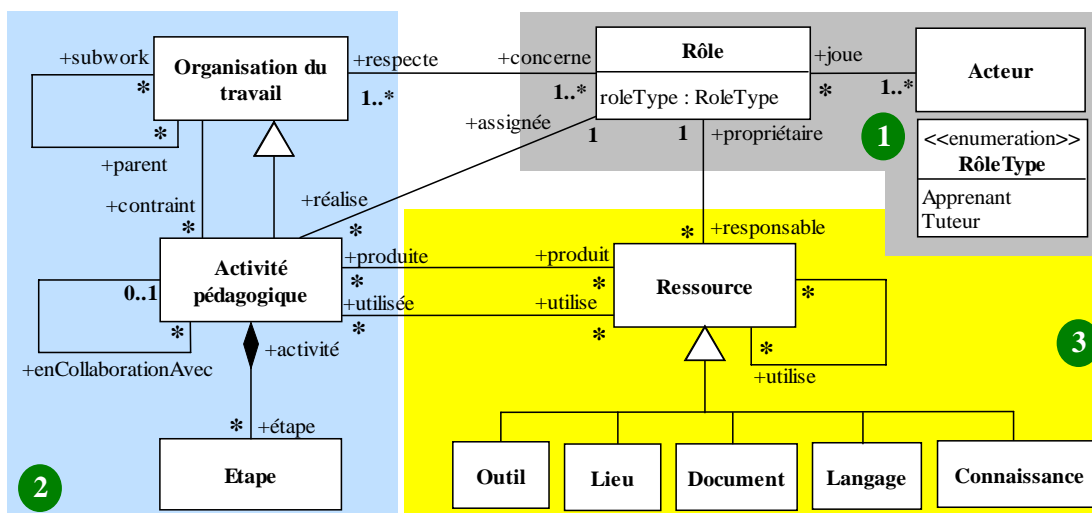


Figure 3.9 : Le modèle conceptuel de description des situations-problèmes coopératives

Dans ce modèle (à rapprocher des propositions de (Faerber 2004)),

- Une activité n'est réalisée que par un seul rôle. Ce choix privilégie la facilité de spécification de fiche de tâche individuelle. L'inconvénient réside dans la représentation des activités collaboratives qui nécessite alors de décomposer l'activité collaborative en autant d'activités qu'il y a de rôles impliqués dans la collaboration et de relier ces activités entre elles afin de spécifier leur lien de collaboration et aussi de les distinguer des activités individuelles.
- Le rôle représente aussi bien le rôle global joué dans la situation d'apprentissage qu'un rôle plus particulier joué dans une partie du scénario. Les rôles peuvent alors être spécialisés (relation d'héritage) et composés (relation d'agrégation) d'autres rôles.

- L'apprentissage coopératif s'effectue par le biais d'activités individuelles et collaboratives mais également par l'échange de ressources produites (associations « utilise » et « produit ») : une activité peut utiliser, voire modifier une ressource créée par une autre activité et donc un autre rôle.
- Les activités prennent place dans des lieux offrant des ressources et outils ayant une cohérence,
- Les ressources sont souvent liées les unes aux autres (association « utilise ») : un outil permet de manipuler un document, un langage permet de produire et comprendre un document, une connaissance s'exprime dans un certain langage, ...
- Le langage permet d'exprimer des représentations et des connaissances partagées sur la situation étudiée ; il décrit et guide le processus de résolution de problème.
- La connaissance porte sur l'acteur impliqué dans la résolution de problème ; les productions et les interactions de l'apprenant permettent de lever des événements qui peuvent être utilisés ensuite dans d'autres activités impliquant l'apprenant.

3.1.2.3 Du modèle conceptuel au profil CPM

Les éléments du modèle conceptuel proposé ont ensuite fait l'objet de différentes actions de recherche permettant d'aboutir à la proposition d'un profil UML pour la modélisation de situations-problèmes coopératives : le profil CPM.

La spécification UML 1.5 donne la définition suivante d'un profil : « a profile is a stereotyped package that contains model elements that have been customised for a specific domain by extending the metamodel using stereotypes, tagged definitions, and constraints. A profile may specify model libraries on which it depends and the metamodel subset that it extends. ». Ainsi, le profil CPM est mis en œuvre à partir des éléments suivants :

- Une syntaxe abstraite représentée par le méta-modèle CPM qui définit les concepts et relations spécialisant UML,
- Une syntaxe concrète qui définit la notation des concepts et de leurs relations ainsi que les principes d'utilisation de ces concepts dans des diagrammes UML,
- Une sémantique qui est définie au niveau de la terminologie (sous la forme de contraintes OCL et de règles en langage naturel) comme au niveau de la notation (sous la forme de propositions d'usage des diagrammes UML).

Par rapport à l'architecture en couches définie dans (OMG 2003), CPM est positionné de la manière suivante :

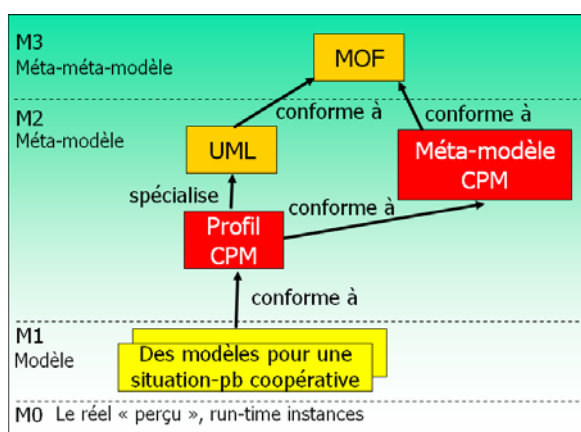


Figure 3.10 : Des modèles de situations-problèmes basés sur UML et le méta-modèle CPM

Cette figure fait apparaître une différence fondamentale entre l'approche que nous proposons et l'approche IMS-LD : le méta-modèle CPM ne se veut pas complet (contrairement à IMS-LD -voir paragraphe 2.2.2.4.1-) en tant que tel mais cette spécialisation du langage UML vise à pouvoir utiliser les différents diagrammes UML en leur faisant porter une sémantique adaptée à la description de situations-problèmes coopératives.

Les éléments du profil CPM sont donc susceptibles d'être utilisés dans des diagrammes UML de différents types, et ce afin de couvrir l'analyse d'une situation-problème depuis l'étape d'expression initiale des besoins jusqu'à l'étape de conception détaillée. Dans les travaux que nous avons menés, les résultats opérationnels de la modélisation d'une situation-problème donnent lieu à la spécification de composants logiciels pour mettre en œuvre les activités décrites (voir paragraphe 3.2.2.2 traitant du modèle de composant CPL). Dans un souci de compatibilité avec le modèle de conteneur offert par IMS-LD, les activités contextualisées par les ressources et composants décrits peuvent également être transformées (avec perte de sémantique) en modélisations IMS-LD (voir paragraphe 3.2.2.1) : l'orientation IMS-LD (préconisant l'interopérabilité et la décontextualisation des ressources et environnements) n'est du coup pas respectée par nos modèles, puisque nos composants et ressources n'ont pas vocation à être réutilisés ou à interopérer avec d'autres mais seulement à pouvoir être utilisés via une plate-forme de Formation à distance à base de composants (type OpenUss).

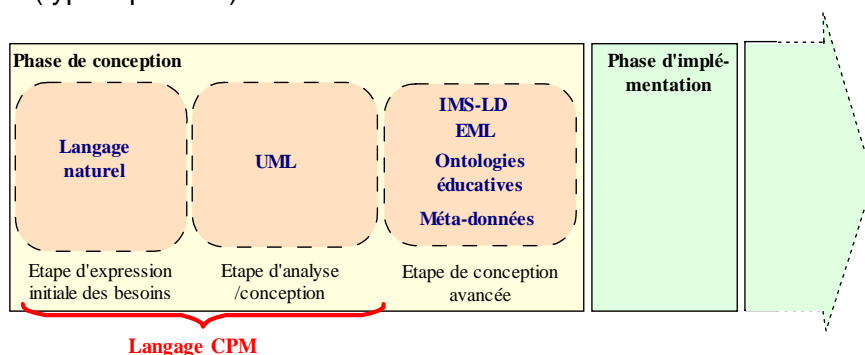


Figure 3.11 : Les étapes de conception couvertes par le profil CPM

Les prochains paragraphes présenteront les éléments fondamentaux de la syntaxe abstraite et de la syntaxe concrète du profil CPM ; je présenterai ensuite l'environnement de conception développé à partir du profil CPM, un exemple d'outil développé à partir des modèles de la situation Smash et nos expériences en terme de vérification / validation des modèles CPM.

3.1.2.3.1 La syntaxe abstraite

Le méta-modèle a été conçu de manière préliminaire à l'élaboration du profil afin de clairement séparer la terminologie de la notation. Le méta-modèle CPM repose sur deux principaux paquetages : le paquetage de fondation, qui représente un sous-ensemble du méta-modèle d'UML (ce sous-ensemble est en réalité le Core du méta-modèle d'UML), et le paquetage des extensions qui regroupe tous les concepts, relations et contraintes de notre langage. Le paquetage des extensions est décomposé à son tour en quatre sous-paquetages interdépendants afin de faciliter la présentation de la terminologie : paquetage des éléments de base, paquetage pédagogique, paquetage structurel, paquetage social (voir Figure 3.12).

Concrètement, les concepts et les relations pour le langage CPM sont définis dans le second paquetage CPM_Extensions, le méta-modèle CPM étant construit par extension d'un sous-ensemble du méta-modèle physique d'UML 1.4. Ce sous-ensemble d'UML correspond au paquetage CPM_Foundation. Ce découplage a été réalisé pour les raisons suivantes :

- l'alignement d'UML sur le MOF n'a été réalisé que récemment par l'OMG avec la spécification d'UML 2.0 alors que les travaux de Pierre Laforcade avaient déjà commencé sur la base des spécifications UML 1.X ;
- de nombreux travaux analogues de spécification de méta-modèles puis de profils suivent la même démarche (OMG 2002a), (OMG 2004) ;
- cette démarche permet d'identifier les éléments du méta-modèle d'UML qui servent de base à la définition des stéréotypes dans le profil CPM ; ainsi, la spécification du profil est facilitée ;

- certains éléments de méta-modélisation nécessaires pour le point précédent n'existent pas dans le MOF (par exemple, les éléments nécessaires pour créer des diagrammes d'activités qui sont des éléments de description importants de la dynamique des situations d'apprentissage).

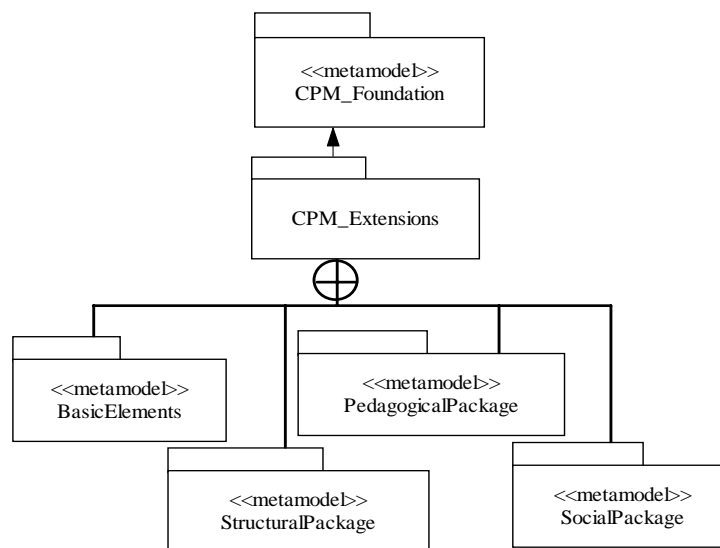


Figure 3.12 : Hiérarchie des paquetages du méta-modèle CPM

Tous les éléments du méta-modèle CPM sont dérivés directement ou indirectement de *ModelElement* (qui appartient au méta-modèle d'UML et fait donc partie du paquetage *CPM_Foundation*). Les spécialisations de *ModelElement* exploitées par UML (*Parameter*, *ActionState*, *Operation*, *Dependency*, *Classifier*, *Package*, *Constraint*, ...) ont servi pour définir les types de base des éléments du méta-modèle CPM. Lorsque les concepts que nous souhaitons utiliser existaient dans IMS-LD, nous avons simplement repris les noms des concepts tels que définis par IMS-LD et nous les avons associés à l'une ou l'autre des spécialisations de l'élément UML *ModelElement*. Tous les concepts n'ayant pas de correspondance dans IMS-LD ont été définis puis rattachés à *ModelElement* ou à l'une de ses spécialisations.

Le paquetage ***CPM_BasicElements*** (voir Figure Annexe C.1) définit les éléments de modélisation que l'on peut attacher à n'importe quel concept du langage CPM (relevant donc du type *ModelElement*). Deux points sont à noter :

- En dehors des liens définis explicitement dans le méta-modèle CPM (voir le détail des paquetages *CPM_StructuralPackage*, *CPM_PedagogicalPackage*, *CPM_SocialPackage* dans (Laforcade et Barbier 2004)), tout concept CPM peut être relié à d'autres concepts CPM par le biais de liens (représentés par le concept UML *Dependency*). Dans le cadre de la modélisation d'une situation d'apprentissage, les concepteurs peuvent donc personnaliser le langage de description proposé pour par exemple établir des liens comme « équivalence/contradiction » entre deux ressources, comme « décrit /commente » entre un langage et une ressource, comme « traduit » entre un langage et une connaissance, comme « précède/suit » entre deux activités pédagogiques, ... (voir les éléments du modèle conceptuel de la Figure 3.9). Ces liens peuvent facilement être ajoutés au profil (comme nous l'avons fait pour le lien *precedes*) si des liens récurrents sont mis en évidence par les utilisateurs du profil CPM.
- Tout concept CPM peut être associé à une description externe via l'élément *External Description*. Cette description permet de faire une correspondance entre l'élément modélisé et l'objet d'apprentissage concret qu'il représente (notion de « binding » proposé dans IMS-LD). Nous utilisons par exemple ce concept pour décrire, dans un fichier externe, la sémantique et les connaissances associées à un élément d'un modèle (voir prédicats décrits dans le paragraphe 3.1.1.1).

Le paquetage **CPM_PedagogicalPackage** (voir Figure Annexe C.2) fournit au concepteur les éléments du langage permettant de décrire les éléments cognitifs d'une situation-problème. Ceci inclut la description des conceptions (justes ou erronées) sur lesquelles s'appuie la situation étudiée, les obstacles auxquels le concepteur veut confronter les apprenants, les critères de succès et les ressources mises à disposition des apprenants, ... Les éléments importants de cette partie du méta-modèle sont les suivants :

- Le méta-modèle différencie les éléments qui vont être représentés de manière statique (*Task*, *Obstacle*, *Objective*, *SuccessCriterion*, ...) des éléments dynamiques : les ressources identifiées dans le modèle conceptuel de la Figure 3.9 (document, outil, langage, connaissance) sont des éléments dynamiques susceptibles de changer d'état durant la résolution d'une situation-problème (en fonction d'événements pédagogiques devant être mis en évidence). Le comportement de ces éléments dynamiques est décrit dans nos modèles à l'aide d'une *StateMachine*.
- Le concept UML *Constraint* a été spécialisé afin de pouvoir poser des contraintes de type précondition ou postcondition sur les éléments de nos modèles de situations-problèmes (conditions d'utilisation des ressources, pré-requis et post-requis des activités, contraintes de temps sur les activités, ...).

Le paquetage **CPM_StructuralPackage** (voir Figure Annexe C.3) fournit au concepteur les éléments du langage lui permettant de décrire la manière dont les activités proposées aux apprenants vont être organisées. Rappelons qu'une organisation est prévue par le concepteur lorsqu'il analyse la situation d'apprentissage mais que cette organisation des activités peut également être, en partie, à la charge des apprenants dans le cadre d'activités spécifiques d'organisation du processus de résolution (ce type d'activité étant souvent coopératif afin d'arriver à un agrément entre acteurs et une répartition acceptée des tâches – voir les éléments du paquetage CPM-SocialPackage). Cette partie du méta-modèle CPM rappelle la notion de scénario IMS-LD : *ActivityConcept* représente le concept abstrait d'activité (représenté par le concept UML *Operation*) et permet de définir la relation de décomposition hiérarchique pour tous les types d'activités. *LearningPhase* est une spécialisation d'*ActivityConcept* qui permet de définir la décomposition externe du scénario de la situation d'apprentissage, c'est-à-dire le découpage en séquences du scénario indépendamment des activités jouées par des rôles particuliers. En comparaison avec la spécification IMS-LD, *LearningPhase* correspond au concept d'acte. Toutefois, alors que pour IMS-LD il n'y a qu'un seul niveau de décomposition du scénario, *LearningPhase* permet de définir autant de niveaux que nécessaire ; il hérite en effet de l'association *subStructure* d'*ActivityConcept* et le méta-attribut *phaseKind* permet d'indiquer le type de la phase. *ActivityStructure* et *Activity* sont également des spécialisations d'*ActivityConcept* et représentent respectivement un ensemble d'activités / une activité précise réalisée par un rôle particulier. Les structures d'activités peuvent être de différents types (précisé via le méta-attribut *structureKind*) afin d'indiquer l'ordonnancement souhaité (ordonnancement imposé ou libre, choix d'une certain nombre d'activités parmi celles proposées, ...).

Enfin le paquetage **CPM_SocialPackage** (voir Figure Annexe C.4) fournit au concepteur les éléments du langage permettant de décrire les activités coopératives que vont pouvoir mener les apprenants impliqués dans la résolution d'une situation-problème. Cet aspect de la description d'une situation d'apprentissage qui n'est pas traité par IMS-LD fait l'objet de nombreux travaux (Ellis et Wainer 1994), (Odell, Van Dyke Parunak et al. 2000), (Guareis de Farias, Ferreira Pires et al. 2000), (Bourguin 2000), (David 2001), (Rubart et Dawabi 2002), (Delium 2003) travaux analysés dans la thèse de P. Laforcade et qui ont inspiré le paquetage social proposé. La description des aspects sociaux est basée sur les notions de *Role*, de *Resource* et de *CollaborativeActivity* :

- La notion de *Role* permet de spécifier les rôles situés proposés par la situation-problème et liés à la situation authentique qu'elle met en scène ; le rôle plus global (apprenant, tuteur, etc.) est spécifié via le méta-attribut *roleKind*.
- Une *CollaborativeActivity* est une spécialisation d'*Activity* (décrite dans le paquetage structurel) pour désigner les activités réalisées de manière collective (George 2001). Dans notre méta-modèle, ces activités apparaissent comme jouées par un seul rôle (voir

la relation assignée entre *Activité Pédagogique* et *Rôle* de la Figure 3.9). Afin de préciser ce lien de collaboration entre plusieurs activités, on leur associe un même *CollaborationMode* qui est une spécialisation de l'élément *Constraint* d'UML et permet de préciser les formes de coordination entre ces activités : supervision directe, adaptation mutuelle, ... (Etcheverry 2002).

- Les caractéristiques liant une activité aux ressources utiles pour la conduire sont décrites via la méta-classe *ActivityResourceParameter* : le concepteur peut indiquer si une ressource associée à une activité est une entrée, une sortie ou une entrée modifiable.
- Enfin, le concept de *useRights* permet de spécifier les droits d'utilisation que peuvent avoir les différents rôles sur les ressources / activités auxquelles ils ont accès :
 - partage éventuel de ressources sur lesquelles des droits peuvent être spécifiés (exemples : une ressource « lockable » sur laquelle un rôle peut se définir des droits exclusifs le temps d'une transaction ; une ressource « contrôlable » de telle sorte que les acteurs puissent gérer eux-mêmes les droits en écriture/en lecture sur les ressources durant les activités conduites),
 - partage éventuel d'informations sur les activités conduites par un rôle donné (activité observable / awareness).

Les 35 concepts du langage CPM peuvent se classer en fonction de leurs usages dans les différents modèles produits durant la phase de conception : certains concepts représentent le vocabulaire nécessaire à l'étape d'expression initiale des besoins (la terminologie employée est celle des situations-problèmes : *Subject*, *Objective*, *Resource*, *Obstacle*, *SuccessCriterion*, etc.). D'autres concepts, à l'opposé, conviennent davantage à un usage de conception détaillée pour lequel le vocabulaire est proche de la terminologie d'autres langages existants comme IMS-LD. La sémantique des différents concepts et relations proposés est exprimée via des contraintes OCL et des explications en langage naturel ; pour chacun des paquetages, la terminologie et des exemples d'utilisation peuvent être trouvés au chapitre 7 de (Laforcade 2004). L'annexe C de ce manuscrit donne la représentation UML de chacun des paquetages venant d'être décrits.

3.1.2.3.2 La syntaxe concrète

Comme tout profil, le profil CPM est composé entre autres :

- de stéréotypes : défini pour une méta-classe spécifique du méta-modèle d'UML, un stéréotype crée une nouvelle méta-classe basée sur la méta-classe UML existante. Il fournit ainsi un moyen de classer les instances de cette méta-classe de base et peut également aussi spécifier des contraintes additionnelles ou des valeurs marquées requises.
- de définitions de valeurs marquées : définie pour une méta-classe spécifique du méta-modèle d'UML comme associée à un stéréotype, la définition d'une valeur marquée agit comme un attribut d'une méta-classe UML, permettant ainsi l'attachement arbitraire d'informations à une instance.
- d'une notation : la notation d'UML est personnalisée via la définition d'icônes associées aux stéréotypes.

La définition de stéréotypes est dirigée également par les diagrammes UML que l'on veut privilégier au niveau de la notation du profil. Ainsi, sur la base d'une analyse préliminaire des usages traditionnels des diagrammes UML en génie logiciel comme en génie éducatif, nous avons retenu l'utilisation des diagrammes de classe, de cas d'utilisation, d'états et d'activités. D'autres diagrammes sont utilisables (diagrammes d'objets, de séquence et de collaboration) mais aucun de leurs éléments de modélisation UML n'a été étendu (pas de correspondance ou sémantique définie directement en relation avec les concepts CPM). Le paragraphe 3.1.2.4 permettra de décrire les usages faits de ces diagrammes.

La Figure 3.13 présente un extrait des définitions des stéréotypes pour le profil CPM. A chaque stéréotype correspondent des méta-classes UML de référence qui vont être « étendues ». Une première correspondance entre le méta-modèle CPM et le profil CPM suit la règle suivante : à tout concept défini dans le paquetage *CPM_Extensions* correspond un

stéréotype défini sur la base de la méta-classe UML dont il hérite directement ou indirectement dans le méta-modèle CPM (méta-classes de référence en gras dans la figure ci-dessous). D'autres classes du sous-ensemble du méta-modèle d'UML sont proposées toutefois comme alternatives à la classe de base afin de pouvoir utiliser les concepts du langage CPM dans des diagrammes UML différents et ainsi augmenter les possibilités de description visuelle des modèles pour la conception de situations-problèmes. Ainsi, par exemple, *LearningPhase* est un stéréotype défini pour la méta-classe UML de référence *Operation* (les opérations sont utilisées dans les diagrammes de classe pour représenter des comportements ou méthodes d'un classifieur) mais également pour les méta-classes *ActionState* et *SubActivityState* (pour être utilisé dans les diagrammes d'activités), *UseCase* (pour les diagrammes de cas d'utilisation) et *Classifier* (pour une utilisation dans les diagrammes de classe). Coopératives. Cette idée, déjà utilisée dans (OMG 2002a), signifie que les éléments de notation *ActionState* et *SubActivityState* sont des *proxy* pour l'*Operation* stéréotypée qui est associée avec le *CallAction* de l'*ActionState*.




Stéréotype	Méta-classe	Icône
LearningPhase	Core::Operation ActivityGraphs::ActionState ActivityGraphs::SubactivityState UseCases::UseCase Core::Classifier	
Activity	Core::Operation ActivityGraphs::ActionState ActivityGraphs::SubactivityState UseCases::UseCase Core::Classifier	
Role	UseCases::Actor ActivityGraphs::Partition	

Figure 3.13 : Extraits de la table de définitions des stéréotypes du profil CPM

Des icônes sont associées aux stéréotypes les plus utilisés afin de simplifier leur interprétation sémantique dans les diagrammes où ils apparaissent.

De manière similaire aux stéréotypes, la définition des valeurs marquées fait l'objet d'une correspondance directe avec le méta-modèle CPM : tout méta-attribut rattaché à un concept du paquetage *CPM_Extensions* fait l'objet d'une définition de valeur marquée rattachée au stéréotype de même nom et prenant un paramètre de type *String*.

Toutes les associations entre les classes du méta-modèle CPM trouvent une correspondance avec les différentes possibilités de relation proposées par le méta-modèle d'UML. Par exemple, l'association entre *Role* et *ActivityStructure* du méta-modèle CPM s'exprime via l'association entre *Classifier* et *Feature* définie dans le méta-modèle d'UML pour la sémantique des diagrammes de classe (cf les figures de l'Annexe D). Les détails sur le profil CPM peuvent être trouvés au chapitre 8 de (Laforcade 2004).

3.1.2.4 Examen de quelques modèles produits avec le profil CPM

Les éléments du langage étant présentés, j'examine ici quelques exemples de modèles produits à l'aide du langage CPM pour la situation-problème Smash¹⁸; Les exemples issus de l'annexe E ont été choisis pour rendre compte des possibilités de modélisation offertes par le langage et expliquer en quoi il apporte une plus value par rapport à IMS-LD sur les trois aspects de l'analyse détaillée en annexe A (et synthétisée dans la Figure 2.11).

3.1.2.4.1 Modèle d'information associable à une activité

¹⁸ L'annexe E, dont sont tirés ces exemples, présente de manière concrète les modèles produits pour décrire les aspects pédagogiques de l'acte 2 de la situation Smash (voir la Figure Annexe B.3 de l'annexe B). Les exemples proposés complètent la mise à l'essai du langage que P. Laforcade a menée à partir de l'analyse de l'acte 3 de cette même situation d'apprentissage.

Les diagrammes de classe exploitant les stéréotypes CPM permettent :

- *La définition du rôle impliqué dans l'activité* (cf Figure Annexe E.4 en page 163) : précision du type de rôle, apprenant ou tuteur (avec la valeur marquée *roleKind*), mise en relation des rôles globaux et des rôles locaux, décomposition des rôles en sous-rôles (par spécialisation), regroupement de rôles pour pouvoir les manipuler comme un seul élément dans les modèles.
- *L'analyse détaillée des aspects pédagogiques de chaque activité* en termes d'objectifs, de sujets d'apprentissage (cf Figure Annexe E.9 en page 165) et de critères de réussite.
- *L'analyse détaillée des ressources utilisées, modifiées, créées* (cf Figure Annexe E.13 en page 167). Cette représentation est empruntée à celle des diagrammes de processus (Eriksson et Penker 2000), (DeCesare, Lycett et al. 2002) : les ressources utilisées sont placées à gauche (*input objects*), les ressources produites à droite (*output objects*), les ressources/concepts contrôlant l'activité au dessus (*controlling objects*).

Les diagrammes de cas d'utilisation exploitant les stéréotypes CPM permettent *la description des relations entre les rôles et les activités qu'ils réalisent* : la Figure Annexe E.3 en page 162 est un exemple montrant des participations différenciées à la réalisation d'une activité donnée (association stéréotypée <<performs>> / association stéréotypée <<assists>>). Les diagrammes d'activités permettent également de préciser les activités (raffinement via le stéréotype <<Step>>) et les structures d'activités dans le contexte du rôle qui les réalise (cf Figure Annexe E.17 à la page 169).

Ces éléments permettent de dégager les premières conclusions suivantes (qui seront reprises dans la Figure 3.14) :

1. Le langage CPM propose des concepts pour décrire les détails du déroulement d'une activité (le concept d'activité devient une boîte blanche).
2. Le langage CPM propose des concepts pour décrire les objets d'apprentissage, les services et outils, ... qui seront manipulés dans le cadre d'une activité.

3.1.2.4.2 Modèle d'information associable à une activité collaborative CPM

Les diagrammes d'activités permettent de décrire des activités se déroulant en parallèle. A partir des diagrammes d'activités, le langage CPM permet :

- *La description de plusieurs activités qu'un même rôle va être amené à effectuer en parallèle.* La partie gauche de la Figure Annexe E.16 en page 169 montre par exemple deux activités que le rôle *RôleEnquêteurs 1 à 3* doit mener en parallèle : l'activité collaborative *répond aux demandes* et la structure d'activités *Analyse les témoignages fournis*. Cette même figure montre trois activités devant être menées en parallèle par le rôle *RôleEnquêteur 4*. Les barres horizontales Fork et Join marquent respectivement le début et la fin du parallélisme.
- *La description d'activités menées en parallèle par différents rôles* : c'est le cas de tous les diagrammes d'activités dans lesquels interviennent plusieurs partitions. Ces activités peuvent être totalement indépendantes (dans la Figure Annexe E.16, l'activité *répond aux demandes* du rôle *RôleEnquêteurs 1 à 3* est indépendante de l'activité *propose des orientations de recherche* du rôle *RôleEnquêteur 4*). Par contre, certaines activités menées par différents rôles peuvent être collaboratives, il faut alors décrire la façon dont le concepteur souhaite les synchroniser (cf les deux exemples donnés dans la Figure Annexe E.18 et la Figure Annexe E.19) : le langage CPM permet alors de prescrire la forme de collaboration souhaitée (stéréotype <<CollaborationMode>>), le rôle de chaque activité participant à la collaboration (stéréotype <<RoleParameter>>) et les ressources échangées entre les activités. Ces éléments concourent à la définition des services logiciels à associer à une activité (cf le composant *CPL* présenté au paragraphe 3.2.2.2 et les exemples donnés en Figure Annexe E 22 et en Figure Annexe E 23, à partir de la page 173).

Ces possibilités du langage permettent de rajouter aux conclusions précédentes les éléments suivants (qui seront repris dans la Figure 3.14) :

3. Le langage CPM propose des concepts pour différencier les activités individuelles des activités collaboratives (et décrire les synchronisations prévues entre ces activités).
4. Le langage CPM permet de décrire les détails des services et des droits des usagers sur les services et ressources affectées.

3.1.2.4.3 Structuration et dynamique des scénarios

Les diagrammes de cas d'utilisation exploitant les stéréotypes CPM permettent d'identifier, de manière globale, les activités impliquant un ou plusieurs rôles d'acteurs (cf Figure Annexe E.3 en page 162). A l'exception de la relation d'inclusion entre deux activités, il n'est pas possible de décrire avec ce type de diagramme l'enchaînement des activités.

Les diagrammes d'activités permettent de décrire l'organisation des activités pédagogiques sur un niveau uniquement : par exemple, le niveau des actes (cf Figure Annexe E.1 en page 161), ou le niveau des scènes (cf Figure Annexe E.2 en page 162) : dans les deux cas, le stéréotype utilisé est <<LearningPhase>>. Les transitions entre les phases ou les activités indiquent le sens de leur réalisation dans le temps. Ainsi, aucun rôle n'apparaît dans ce type de diagramme. Pour définir un niveau hiérarchique inférieur, par exemple les étapes d'une activité (stéréotype <<Step>>), ou les activités d'une structure d'activités, un nouveau diagramme d'activités doit être créé (voir les exemples montrant comment la structure d'activité *Analyse les témoignages fournis* qui apparaît en Figure Annexe E.16 est ensuite raffinée dans la Figure Annexe E.17 en page 170).

Ces mêmes figures montrent que les diagrammes d'activités permettent de décrire différentes possibilités dans la réalisation des scènes et des activités : l'exemple de la Figure Annexe E.16 en page 169 montre deux scénarios possibles de la scène *Analyse des témoignages* ; l'exemple de la Figure Annexe E.17 en page 170 montre une branche conditionnelle montrant des alternatives en fonction de ce qui se passe lors d'une activité (ici l'activité *Effectue une analyse*).

Les diagrammes d'états permettent d'illustrer le comportement dynamique des éléments de modélisation. La sémantique du méta-modèle CPM précise que seuls les éléments *DynamicPBLElement* (comme les ressources ou les critères de succès) peuvent avoir leur comportement précisé sous la forme d'un diagramme décrivant les états qu'ils traversent pendant leur cycle de vie. Les exemples du paragraphe E.7 en page 171 décrivent les diagrammes d'états de la ressource *Graphe de croyances* et de la ressource *Question*. Ces états sont alors utilisables dans les diagrammes d'activités produits (cf par exemple la Figure Annexe E.16 en page 169).

Ces possibilités du langage permettent de rajouter aux conclusions précédentes les éléments suivants (qui seront repris dans la Figure 3.14) :

5. Le langage CPM s'appuie sur les diagrammes de *Use-Case*, les diagrammes de classes, les diagrammes d'états et les diagrammes d'activités ; l'analyse n'est donc pas seulement descendante mais guidée par les événements.
6. Le langage CPM différencie le concept d'*Activity-Structure* (associé à un rôle) du concept de *LearningPhase* (impliquant plusieurs rôles).
7. Le langage CPM propose le concept d'élément dynamique de modélisation afin de pouvoir associer des diagrammes d'états-transitions à des concepts significatifs (représentés par des diagrammes de classe ou des diagrammes de *Use-Case*) des situations d'apprentissage décrites. Les modèles d'activités décrivant un scénario possible peuvent exploiter les états de ces objets significatifs.

Au final, le langage CPM, qui n'a pas vocation à être complet en tant que tel (comme IMS-LD) mais s'appuie et spécialise les diagrammes qu'UML permet de produire, offre aux concepteurs des possibilités de modélisation qui me paraissent aller bien au-delà d'IMS-LD. La Figure 3.14 situe les apports du langage CPM par rapport aux problèmes d'IMS-LD que j'ai précédemment soulevés :

Concepts étudiés du métamodèle IMS-LD		Problèmes soulevés quand on spécifie/conçoit des situations d'apprentissage avec IMS-LD		Éléments de solution proposés par le langage CPM	
Modèle d'information associable à une activité Voir détails de l'analyse en annexe A (figure A.1)	Activity. Environment, Outcome, Notification, Property, Prerequisite, Learning-Objective	Une activité est une boîte noire du point de vue du concepteur du scénario IMS-LD	CPM propose des concepts pour décrire les détails du déroulement d'une activité (le concept d'activité devient une boîte blanche)	P é d a P q a q u g e t o t a g a g o g e q u e	P é d a P q a q u g e t o t a g a g o g e q u e
		Une activité est un conteneur pour associer dans un scénario des learning objects conçus par ailleurs Impossible de spécifier les caractéristiques de l'environnement d'outils et des ressources à partir des besoins issus de la définition des activités à mener Impossible d'analyser une activité du point de vue des ressources produites / à produire par un apprenant ou un tuteur Rien n'est prévu pour faire apparaître, lors de la conception d'une activité pédagogique, les événements significatifs pour un concept (savoir / savoir faire) abordés dans le cadre de cette activité Rien n'est prévu pour faire apparaître, lors de la conception d'une activité pédagogique, les concepts, les prérequis et les objectifs significatifs du scénario conçu	CPM propose des concepts pour décrire les objets d'apprentissage / les services et outils / les objectifs qui seront manipulés dans le cadre d'une activité		
Modèle d'information associable à une activité collaborative Voir détails de l'analyse en annexe A (figure A.2)	Activity. Play, Role-Part, Role, Environment	IMS-LD offre des moyens pour décrire des processus s'exécutant en parallèle (play, activity-structure, role-part) mais n'offre pas de moyens suffisants pour décrire leur synchronisation	CPM propose des concepts pour différencier les activités individuelles des activités collaboratives (et décrire les synchronisations prévues entre ces activités)	P a q u e t e a g e e C P L	P a q u e t e a g e e C P L
		Impossible de spécifier les types d'accès de chaque rôle impliqué dans des activités collaboratives que l'on conçoit Une activité (collaborative) est un conteneur pour associer dans un scénario des rôles avec des environnements d'outils génériques conçus par ailleurs	CPM permet de décrire les détails des services et des droits des usagers sur les services et ressources affectées		
Structuration et dynamique des scénarios IMS-LD Voir détails de l'analyse en annexe A (figure A.3)	Principes méthodologiques proposés pour structurer le scénario. Method, Play, Act, Activity-Structure	L'approche descendante utilisée pour analyser une situation d'apprentissage conduit à des scénarios favorisant la conception de micro-activités	CPM s'appuie sur les diagrammes de use-case, les diagrammes de classe, les diagrammes d'états et les diagrammes d'activités	S t r u c t u r e P a q u e t e a g e e	S t r u c t u r e P a q u e t e a g e e
		Les modèles obtenus multiplient les références au concept d'Activity-Structure. Or, ce concept prend des significations différentes selon le niveau d'analyse auquel il est utilisé. L'utilisation de deux concepts différents aurait facilité l'interprétation des scénarios Le modèle événementiel est limité à des événements prédéfinis (portant sur les concepts de description d'un scénario) et à la prise en compte des valeurs prises par des propriétés simples définies par le concepteur (mais non attachées à des objets significatifs que le concepteur envisage de faire manipuler aux acteurs). Il n'est donc pas possible de décrire l'évolution des propriétés attachées aux objets significatifs de conception puisque ces objets sont seulement référencés et pré-existants Pour combler les lacunes du modèle d'information, les noms de propriétés portent une partie de la sémantique de ces propriétés	CPM différencie le concept d'Activity-Structure (associé à un rôle) du concept de LearningPhase (impliquant plusieurs rôles)		
			CPM propose le concept d'élément dynamique de modélisation afin de pouvoir associer des diagrammes d'états / transitions à des concepts significatifs (représentés par des diagrammes de classes ou des use-cases) des situations d'apprentissage décrites		
			Les modèles d'activités décrivant un scénario possible peuvent exploiter les états des objets significatifs		

Figure 3.14 : Les solutions apportées par le profil CPM aux problèmes posés par IMS-LD

3.1.2.5 Outillage du profil et bilan des expérimentations menées

Le profil CPM a été outillé à partir d'un Atelier de Génie Logiciel UML existant : Objecteering (le seul outil supportant toutes les propriétés et caractéristiques des profils UML). Ce travail mené par Pierre Laforcade a permis :

- o de proposer une implémentation du langage CPM (via l'outil *Objecteering Profile Builder*) et de la mettre à disposition de la communauté EIAH (le profil CPM qui résulte de ce travail d'implémentation est exploitable à partir de l'outil *Objecteering Modeler* qui est gratuit et téléchargeable via le Web),
- o de prototyper un environnement-auteur adapté à des ingénieurs pédagogiques connaissant les bases de la modélisation UML mais non-experts (voir Figure 3.15) ; grâce au langage propriétaire J (langage objet interprété dédié à la manipulation des éléments de modélisation et du méta-modèle d'UML) nous avons pu développer un ensemble de fonctionnalités personnalisant et adaptant l'interface de l'outil pour faciliter la création, le suivi et la manipulation de modèles CPM : recherche guidée d'éléments, ajout d'éléments dirigée par des boîtes de dialogue, etc. Les développements menés depuis janvier 2005 par des étudiants de Master1 de l'UPPA ont permis de faire évoluer l'outil afin de guider l'activité de conception et faciliter la mise en œuvre de modèles conformes à CPM (Blanco et De Zambotti 2005).
- o d'expérimenter la plus-value des modèles graphiques et d'examiner les possibilités d'interprétation de la sémantique des modèles par la machine :
 1. étude de la cohérence entre la sémantique du langage CPM et la notation choisie,
 2. vérification de la conformité des modèles que nous produisons par rapport au langage CPM (en fonction des contraintes et assertions OCL associées à chaque concept) : la vérification qui s'effectue pour l'instant sur demande du concepteur permet d'afficher les erreurs détectées,
 3. transformation de modèles (projection de modèles produits avec CPM vers IMS-LD – voir paragraphe 3.2.2.1).

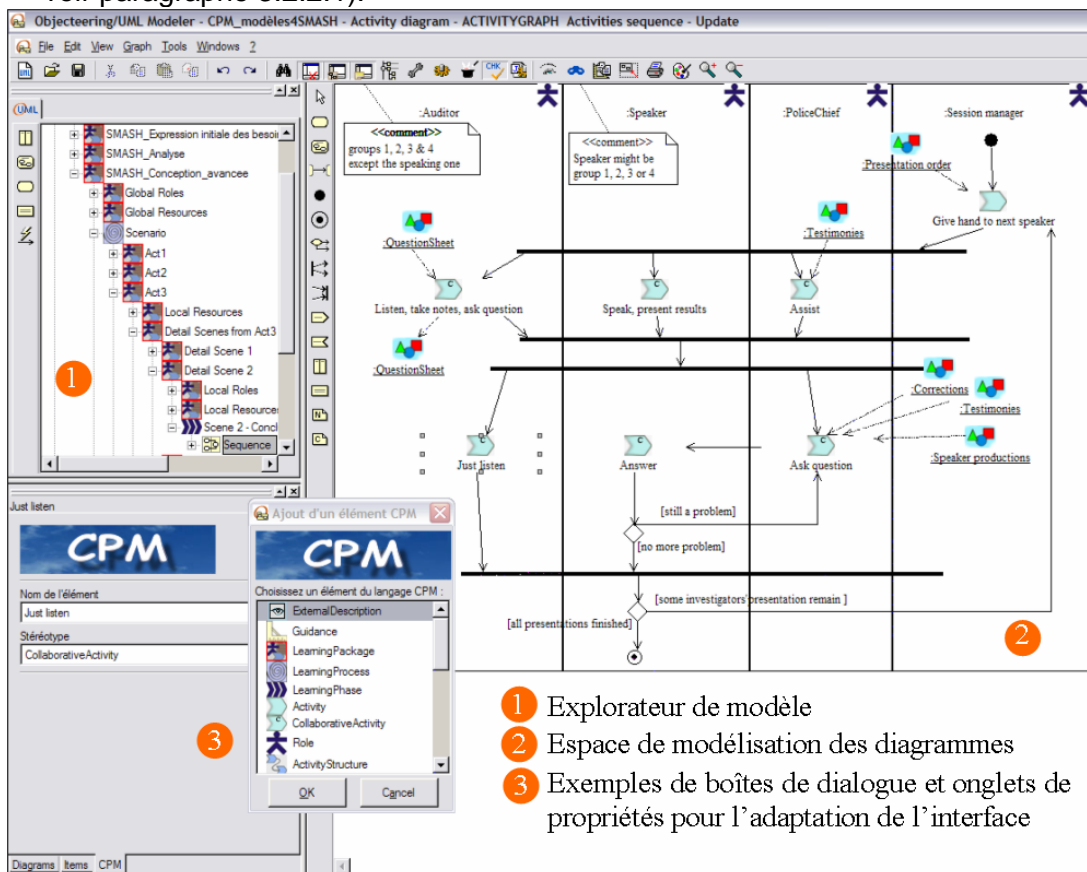


Figure 3.15 : L'environnement d'outil offert aux concepteurs

L'expérimentation du langage CPM a été menée principalement sur la situation problème Smash. Partant des spécifications textuelles dont certains extraits sont fournis en annexe B, nous avons mené un gros travail d'analyse qui a conduit à modéliser l'ensemble des éléments de la situation-problème avec le langage CPM outillé via Objectteering. Certains de ces modèles ont été présentés dans le paragraphe 3.1.2.4, les exemples choisis permettant de comprendre en quoi le langage CPM apporte une plus value par rapport à IMS-LD sur les trois aspects de l'analyse proposée dans la Figure 2.11 et la Figure 3.14. Le paragraphe 9.2 de la thèse de Pierre Laforcade (pages 211-240) retranscrit un éventail plus important de modèles de la situation Smash qui ont conçus sur la base d'échanges avec des enseignants de terrain de Pau et Lourdes.

Cette mise à l'essai du langage CPM a permis d'illustrer et d'explorer la richesse d'expression que l'on retire du couplage du méta-modèle CPM avec UML : la grande diversité des diagrammes à niveaux d'abstraction variables qu'offre UML nous a permis de décrire les activités demandées aux apprenants, d'examiner et faire apparaître les événements pédagogiques significatifs associés aux interactions prévues entre apprenants. Les spécifications que nous avons produites ont facilité ensuite le développement d'outils associés aux activités d'apprentissage décrites via CPM. La Figure 3.16 est un exemple d'outil de type tableau blanc que nous avons développé pour supporter l'activité de mise en commun des résultats d'enquêtes relatives à l'accident de vélo qui est à la base de la situation Smash. Cet outil implémente les principes de gestion des droits des utilisateurs sur les ressources (chacun n'a accès qu'à certains témoignages), permet aux acteurs de ne manipuler que certains éléments référencés (positionnement des véhicules et des acteurs, orientation dans l'espace) dans les témoignages disponibles (Sallaberry, Nodenot et al. 2005). L'outil permet également de rendre compte des événements pédagogiques qui surviennent dans les interactions entre acteurs (déplacements successifs et contradictoires d'un même personnage ou d'un même véhicule, inactivité, objets non déplacés, témoignages consultés, ...).

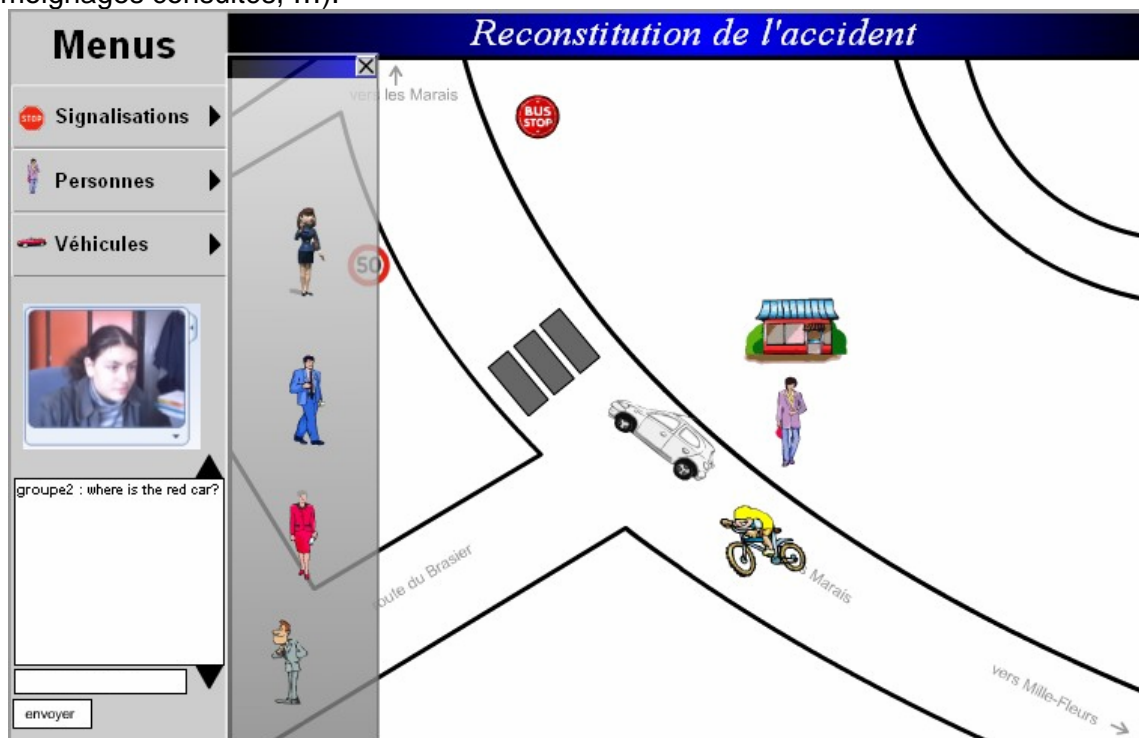


Figure 3.16 : Un outil de type tableau blanc contextualisé pour la situation Smash

L'outil a fait l'objet de différents développements qui nous ont conduits à l'implémentation actuelle effectuée avec l'environnement Laszlo que des stagiaires sont en train de coupler avec la plate-forme de FOAD OpenUss. Les témoignages sont ainsi directement déposés sur cette plate-forme qui gère les accès que font les apprenants aux documents mis à

disposition. Par ailleurs, l'outil de type Tableau Blanc issu de ce travail sur la situation Smash a été repris par deux stagiaires en avril 2005 afin d'en faire un outil générique supportant la conception interactive et l'exploitation de toutes sortes d'activités d'apprentissage de type « construction de représentations partagées » à partir de ressources documentaires à base de textes (Dubourdieu 2005), (Voisin 2005).

3.1.2.6 La vérification et la validation des modèles produits

Comme expliqué dans le paragraphe consacré au bilan de ces travaux (cf paragraphe 3.3.1), ces premiers résultats ne prouvent pas la complétude du langage ni son acceptabilité par les concepteurs et la communauté EIAH : nous avons simplement montré que les éléments du langage permettent de produire des modèles pour décrire de manière précise (semi-formelle) des considérations pédagogiques relatives à des situations-problèmes coopératives : le langage est outillé (via Objectteering) mais il n'est pas associé à une méthode de conception comme l'est MOT (via la méthode MISA) ce qui limite forcément son utilisabilité actuelle.

Au cours des derniers mois de la thèse de Pierre Laforcade, nous avons abordé le travail de vérification / de validation des modèles que le langage CPM permet de produire : la vérification d'un modèle consiste à examiner si ce modèle est bien fait (consistance et complétude), alors que la validation de ce même modèle consiste à déterminer si le modèle produit est le plus adéquat pour une situation donnée (Malak et Paredis 2004). Mon approche de ce problème a évolué dans le temps. En 2003, suite aux réflexions de Nicolas Balacheff lors d'une présentation de nos travaux devant le GDR I3 (SA 4.1), nous avons tout d'abord essayé d'appliquer aux modèles de la situation Smash certains principes de vérification / validation des modèles issus des sciences expérimentales préconisés notamment dans (Balacheff 2001) : nous avons rapidement pu développer un outil vérifiant la conformité des modèles produits par rapport à la syntaxe du méta-modèle CPM. Nous avons ensuite réfléchi à la vérification de la sémantique des modèles de la situation Smash en nous centrant sur les modèles dits « essentiels » (Gluch, Comella-Dorda et al. 2002) que nous avons produits¹⁹ : dans le cas de la situation Smash, nous avons entamé ce travail de vérification sur un cas particulier, celui des modèles de comportement décrivant en UML comment initier et entretenir des échanges constructifs entre apprenants lorsque ceux-ci entrent en conflit (objet placé par un apprenant et déplacé par un autre) quant au positionnement d'un véhicule ou d'un acteur de la situation Smash. Afin de vérifier la bonne synchronisation des états des deux diagrammes concernés (diagramme d'états UML du gestionnaire de conflits et diagramme d'états UML de l'outil de Chat utilisé par ce gestionnaire de conflits comme vecteur d'échanges entre acteurs), nous avons d'abord essayé d'utiliser une approche mathématique, ce qui nous a amené à utiliser l'heuristique de transformation proposée par (Varro 2004) pour passer des diagrammes UML à leur équivalent sous forme de prédicats exploitables par des règles de production. Cette approche utilisée en Génie Logiciel pour mettre ensuite en œuvre des outils automatisés de vérification des modèles (Bensalem, Ganesh et al. 2000), (Muro 2003), s'est avérée peu adaptée à nos besoins et nos domaines de compétence. Nous lui préférons aujourd'hui des techniques plus appliquées qui consistent à encoder (nous l'avons fait manuellement mais des outils de transformation automatisés peuvent également être envisagés) les diagrammes à vérifier pour les exploiter avec des moteurs d'exécution capables d'exécuter (transition par transition) n'importe quel diagramme d'état/de séquence UML bien formé (Harel, Kugler et al. 2002), (Barbier et Constant 2005) ; cette approche permet d'examiner sur des cas d'utilisation (jeux de tests) la qualité / les défauts des modèles produits en détectant les interblocages entre états de divers objets modélisés devant se synchroniser : en fin de thèse,

¹⁹ Un modèle essentiel est un modèle issu de la conception détaillée qui capture l'essence d'un système (par opposition à des modèles ayant pour finalité de fournir une vue exhaustive de ce système). Un modèle essentiel a souvent un caractère dynamique (diagrammes d'états, diagrammes de séquence, ...) qu'il s'agit d'étudier précisément afin de vérifier le respect des propriétés essentielles de ce modèle (invariants, respect des post-conditions, non présence d'interblocages, ...).

Pierre Laforcade a effectué ce type de vérification sur quelques modèles CPM en utilisant la librairie java d'exécution de statecharts développée par l'équipe AOC du LIUPPA (Barbier, Briand et al. 1998).

Plus généralement, l'examen critique que nous avons fait des modèles produits pour la situation Smash m'amène à considérer que la validation de modèles prescriptifs (ceux qui sont produits en ingénierie et conception – cf 2.2.1.1) est une activité qui présente des aspects subjectifs : si comme le souligne (Balacheff 2001), les principes de transparence et de sincérité sont fondateurs pour évaluer les modèles que nous proposons, la notion de preuve en ingénierie (des EIAH) me paraît difficile à mettre en œuvre : la preuve, si elle existe, est donnée par le comportement « valide » de l'EIAH issu du travail de modélisation. Il me semble qu'en ingénierie, ce n'est pas tant la complétude et le niveau de détail d'un modèle qui est important mais sa capacité à nous permettre :

- d'échanger sur les informations portées par le modèle, les conséquences de son usage, les limites à son utilisation,
- de prévoir le comportement futur du système représenté par ce modèle.

Le modèle porte donc de l'information (forcément discutable) pour que ceux qui vont utiliser / interpréter ce modèle puissent prendre les bonnes décisions de conception (en tenant compte de leurs préférences personnelles). Des travaux en Génie Logiciel que j'ai consultés dans ce domaine, je retiens ceux de (Hazelrigg 2003) et (Malak et Paredis 2004) qui tentent de modéliser mathématiquement (modèles probabilistes) les risques attachés aux modèles prédictifs mais tiennent compte des préférences des concepteurs. Je retiens également la notion de contrat (Cariou, Marvie et al. 2004) que l'on peut attacher à un modèle ou à une transformation (les transcriptions successives de ce modèle devant maintenir les propriétés spécifiées dans le contrat en terme d'éléments du modèle et de son méta-modèle). Ce contrat peut servir de base pour examiner le comportement d'un système (Mottu, Baudry et al. 2005) et faire le pont entre modèles prescriptifs et modèles descriptifs.

3.2 Travaux en ingénierie dirigée par les modèles

Les travaux présentés jusqu'ici portent sur la modélisation de l'objet de la conception, à savoir une situation-problème coopérative, en appréhendant ses dimensions fonctionnelles, structurelles, cognitives et sociales. Dans cette partie du chapitre, je présente les travaux que j'ai réalisés ou encadrés en méthodologies de la conception (modèles de processus) dans le but d'exploiter les modèles de la situation d'apprentissage décrite dans des environnements logiciels.

Les travaux décrits au paragraphe 3.2.1 portent sur la formalisation du processus de développement et les outils susceptibles d'être mis à disposition des concepteurs. La présentation fait la synthèse d'un continuum de recherche initié durant ma thèse (Nodenot 1992a), poursuivi dans le cadre des thèses de Christophe Marquesuzaà (Marquesuzaà 1998) et Pierre Laforcade (Laforcade 2004).

Les travaux présentés au paragraphe 3.2.2 sont des travaux plus récents qui ont pu être mis en œuvre grâce à l'orientation Génie Logiciel de la thèse de Pierre Laforcade, ce qui a permis de prototyper des outils s'insérant effectivement dans une perspective Ingénierie des Modèles en EIAH.

3.2.1 Des modèles de situations d'apprentissage aux modèles de processus

Un environnement d'apprentissage accepté par ses utilisateurs et ses concepteurs est le fruit d'un ensemble / d'une succession de compromis et d'agrément. Les travaux présentés jusqu'ici ont porté sur l'objet à modéliser, ce qui m'a permis d'aborder : a/ les langages de représentation à partir desquels nous avons travaillé et b/ les modèles de représentation utiles lors des phases amont de la conception d'une situation-problème. La figure ci-dessous

fait apparaître une troisième dimension d'analyse qui permet de rendre compte du processus de spécification / conception :

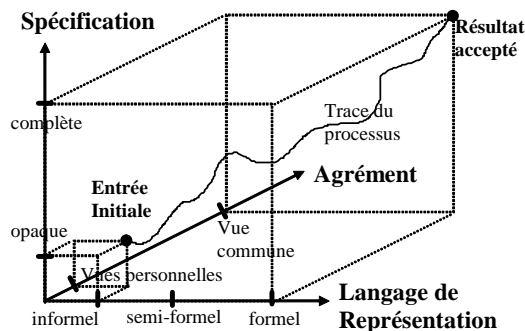


Figure 3.17 : Les trois dimensions de l'ingénierie des besoins d'après (Pohl 1994)

Ce schéma est basé sur trois axes :

- l'axe représentation va d'un niveau totalement informel (langue naturelle) à des représentations très formelles (spécification en Z,...) en passant par des spécifications semi-formelles et graphiques comme UML et CPM.
- l'axe spécification traite de la clarté et de la complétude de la spécification produite. Il s'agit d'un axe orthogonal au premier axe car on peut très bien disposer d'un formalisme très complexe et aboutir à des spécifications très pauvres,
- enfin l'axe agrément stipule que l'activité de spécification / conception doit se penser comme un processus de négociation entre partenaires ayant des compétences, des buts et des contraintes complémentaires.

Mes travaux de thèse (Nodenot 1992a) ont permis d'établir et valider un certain nombre de principes relatifs à l'axe « Agrément » pour renforcer et catalyser les dialogues entre pédagogues et informaticiens. Ces principes sont rapidement présentés ci-dessous.

3.2.1.1 La formalisation de chaque étape du processus de conception

Reprenant les travaux de (Khammaci et El Ayeb 1991), j'ai proposé un modèle de description décrivant le processus de développement d'un logiciel éducatif à l'aide d'un quintuplé constitué des éléments suivants :

- un modèle de données qui précise les types d'objets qui sont concernés par le processus considéré,
- un ensemble de types d'opérateurs (ou outils). Un type d'opérateur correspond à une phase du développement. Il est caractérisé par son nom, sa signature, ainsi que par une pré-condition et une post-condition. La signature de l'opérateur définit les types de paramètres en entrée et sortie, la pré-condition et la post-condition sont des prédicats qui doivent être vérifiés lorsque l'opérateur est activé et après son activation,
- un ensemble de règles d'activation des opérateurs qui décrivent sous quelles conditions un opérateur est activé. Une telle règle est composée de trois éléments : un événement, un prédicat portant sur la valeur des données stockées dans le modèle de données, et le nom d'un opérateur à activer,
- un ensemble de règles d'ordonnancement des opérateurs qui spécifient la façon dont les opérateurs activés sont autorisés à réagir. Certains d'entre eux pourront s'exécuter de façon concurrente ou séquentielle, simultanée, synchronisée ou optionnelle, ...
- un ensemble de caractéristiques que doit respecter le modèle de processus qui a été décrit précédemment. Ces caractéristiques sont soit des contraintes d'intégrité, soit des buts assignés au modèle de processus implémenté.

La figure ci-dessous décrit par exemple les caractéristiques de l'étape de spécification éducative centrée sur la description du modèle du tuteur. Le langage de description utilisé est le langage « Spec » (présenté au paragraphe 3.1.1.1) :

Opérateur	Spécifier-Pédagogue-MD
Signature	p : projet-eao -> spec-MD : {Objet-MD} -- {} signifie ensemble de ...
Données utilisées	<p>ph : phase SUCH THAT spécification(ph,p) and éducative(ph, p) -- on considère la phase de spécification éducative du logiciel éducatif</p> <p>ob : Objet-MD SUCH THAT dida(ob, ph) -- on considère un Objet-MD de cette phase</p> <p>ob1 : Objet-MP SUCH THAT péda(ob1, ph) and utilise-p(ob, ob1) -- on considère le objets du MP (modèle du domaine) utilisés par l'objet ob -- du MD (modèle du tuteur)</p>
Pré-condition	correspond à la post-condition de l'opérateur "spécifier-pédagogue-MP"
Post-condition	<p>(ALL o : Objet-Monde-consideré SUCH THAT MD(o, ob) (SOME c : Concept-monde :: a-caractéristique(o, c)))</p> <p>-- un objet lié à un Objet-MD n'a de sens que s'il est utilisé par au moins un concept monde, c'est-à-dire possède des attributs utilisés dans la spécification éducative du projet</p> <p>and then</p> <p>(ALL o : Objet-Monde-consideré SUCH THAT MD(o, ob), ALL c : Concept-monde :: a-caractéristique(o,c) => (SOME b : Concept-but ::utilise(b,c)))</p> <p>-- Un concept-monde ne doit pas être décrit au niveau de la spécif du MD s'il n'est utilisé par aucun concept-but du MD du logiciel éducatif considéré</p> <p>and then</p> <p>(ALL o : Objet-Monde-consideré SUCH THAT MD(o, ob), ALL b : Concept-but, a-pour-but(o,b)=> (ALL o1 : Objet-Monde-consideré SUCH THAT fait-ref(b,o1) :: MD(o1, ob) or MP(o1, ob1)))</p> <p>-- Un concept-but du MD d'un logiciel éducatif utilise soit des objets de ce MD, soit des objets du MP (modèle du domaine) correspondant</p>
Commentaires	Ce type d'opérateur permet la spécification éducative du MD (modèle du tuteur) d'un logiciel éducatif. La post-condition exprime les contraintes que doit vérifier la modélisation effectuée par le pédagogue pour que celle-ci soit considérée comme complète

Figure 3.18 : La spécification d'une étape à l'aide du langage « Spec » (Nodenot 1992a)

3.2.1.2 Différents processus possibles

Les modèles obtenus ne sont pas entièrement déterministes (il y a plusieurs façons d'atteindre un but logiciel), ce qui permet d'exploiter les formalisations proposées quel que soit le processus finalement choisi par l'équipe pluridisciplinaire en charge du développement (Nodenot 1992c). Dans le cadre de ma thèse, j'ai cependant proposé un modèle de processus favorisant les cycles auteur-lecteur entre pédagogues et informaticiens de façon à identifier, au plus tôt, les incohérences éventuelles du développement. Selon ce modèle,

- o Le développement d'un logiciel éducatif se décompose en cinq étapes (spécification, conception, réalisation, validation, évaluation) qui doivent, elles-mêmes, être prises en compte en considérant tout d'abord leurs aspects éducatifs puis leurs aspects techniques.
- o Quelle que soit l'étape considérée et le type d'acteur impliqué, l'analyse d'un logiciel éducatif se fait selon trois niveaux en différenciant les modèles du domaine (ou de l'expert), les modèles pédagogiques (ou du tuteur) et les modèles d'interface (appelé aussi modèle médiatique). Les décisions relatives à chacun de ces aspects sont formalisées sous forme de modules (DEFINITION, MACHINE, TYPE, FUNCTION) exploitant le langage "Spec".
- o Les techniques de prototypage sont préconisées dans ce modèle de processus pour évaluer les modélisations effectuées quel que soit le point de vue adopté (éducatif ou technique).

Ces différentes considérations m'ont amené à décrire sous forme d'heuristique, le processus que nous avons utilisé ensuite pour développer deux applications éducatives : a/ Un logiciel centré sur l'apprentissage de techniques efficaces pour le calcul mental qui était totalement spécifié à l'aide du langage « Spec » (Nodenot 1992b) puis implémenté en Ada. b/ Une situation-problème permettant à des élèves de CM2 de comprendre les principes

régissant un état démocratique (Nodenot 1995). L'heuristique régissant ce processus est donnée dans la Figure 3.19 :

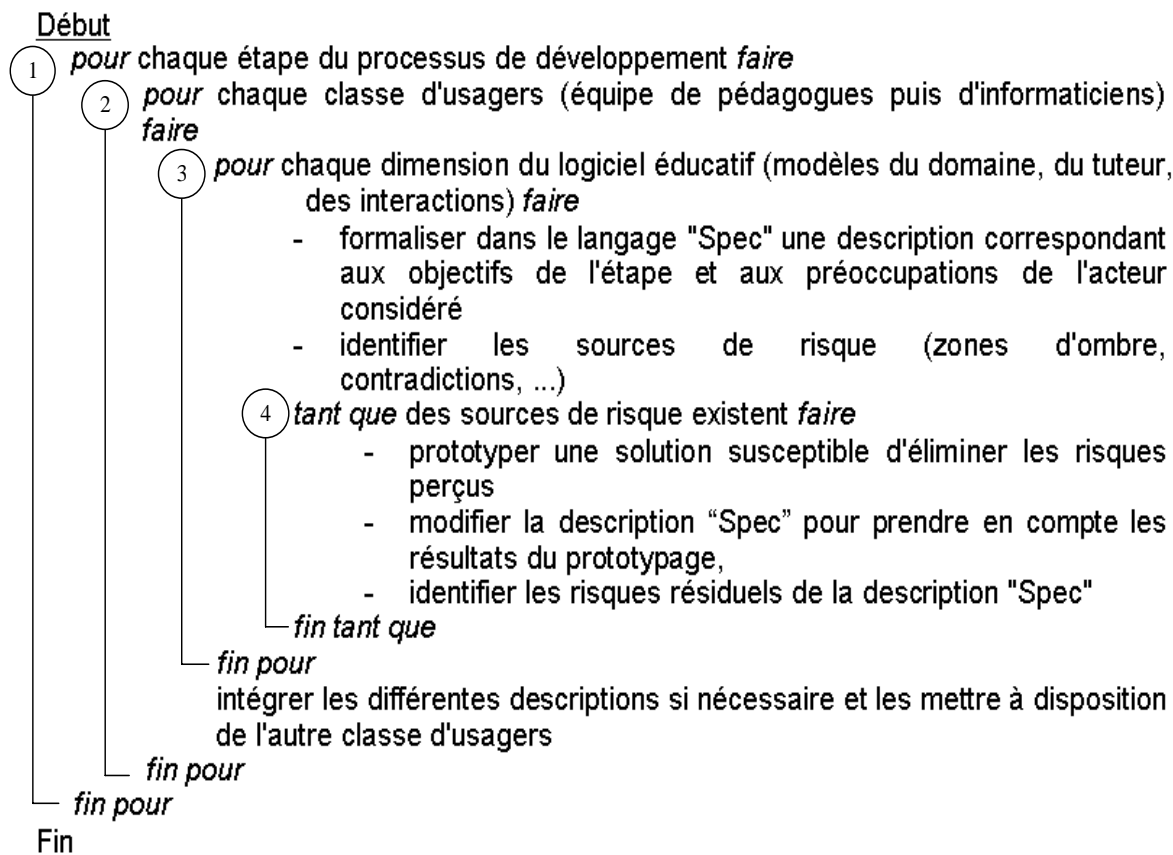


Figure 3.19 : un processus favorisant les échanges entre concepteurs (Nodenot 1992b)

Les travaux de thèse de Christophe Marquesuzaà ont permis d'approfondir sur un domaine particulier, les situations-problèmes, le problème de la conception tant du point de vue de la description du processus de développement que du point de vue de l'outillage pouvant être mis à disposition des concepteurs (Marquesuzaà 1998).

3.2.1.3 Le processus de conception des situations-problèmes vu par les enseignants

Les travaux ont porté principalement sur les points et de la Figure 3.19, à savoir l'analyse des modèles et processus utiles aux enseignants spécifiant une situation-problème. S'appuyant sur la théorie empirique de l'apprentissage (Develay 1993), les propositions (Marquesuzaà et Nodenot 1996) décrivent le processus de conception utilisé reliant des modèles de la situation-problème spécifiée (objets constituant la matrice disciplinaire, objectifs d'apprentissage, représentations des apprenants, stratégies d'enseignement) au modèle de chaque étape d'analyse (prérequis, post-requis, mécanismes de représentation offerts aux enseignants, conditions de validité des modèles proposés par les concepteurs).

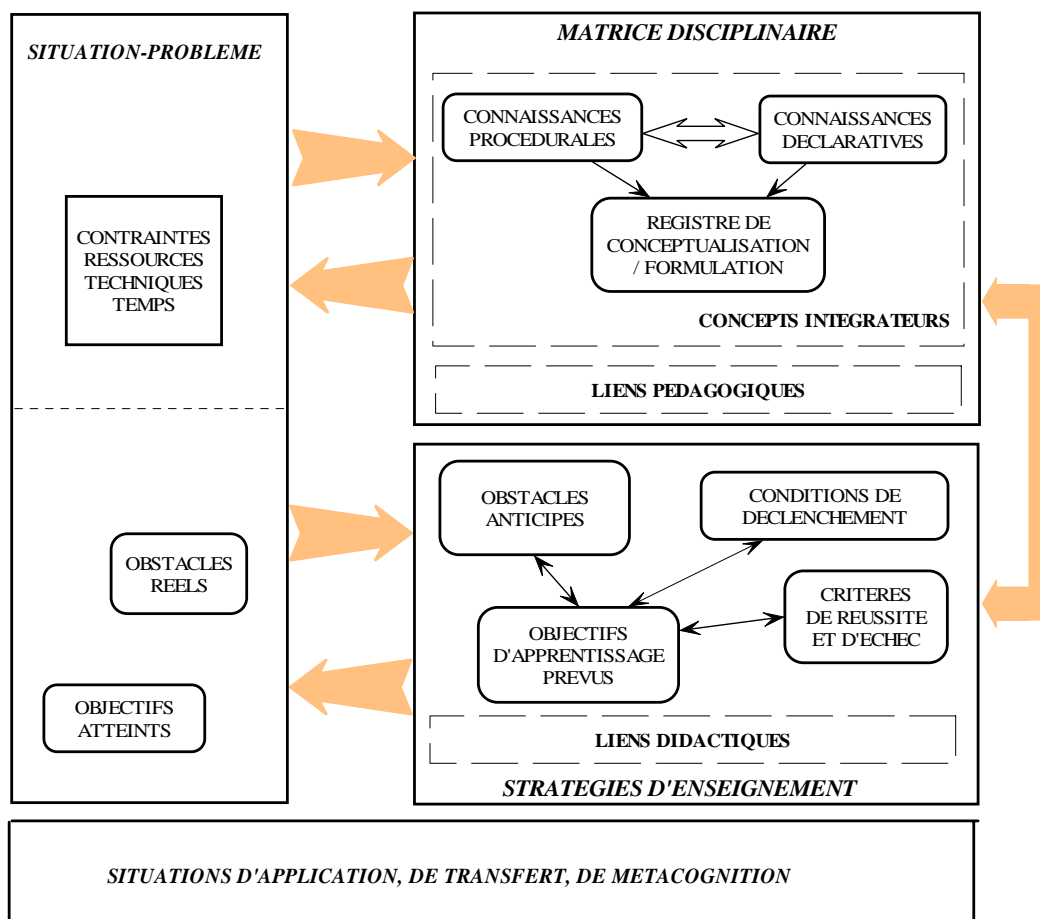


Figure 3.20 : L'activité de spécification d'une situation-problème (Marquesuzaà 1998)

Le processus global de spécification d'une situation-problème est décrit par un ensemble de graphes de contextes, chaque graphe correspondant à un ensemble d'activités hiérarchisées conduisant à un but (Marquesuzaà, Meyranx et al. 1996). L'approche choisie pour représenter le processus est donc orientée décisions, une décision représentant un choix qu'un acteur va pouvoir effectuer lorsqu'il est dans une certaine situation du processus de spécification considéré (notion de contexte). Le modèle proposé dans (Marquesuzaà, Meyranx et al. 1997) reprend le modèle MACT développé par (Péninou 1993) et (Mengelle 1995) par opposition à des modèles de processus orientés activités tels les modèles GOMS ou ETAG (De Haan, Van der Veer et al. 1991).

Afin d'être exploitables par des outils informatiques, les connaissances sur le processus de spécification des situations-problèmes ont été implantées dans le système de représentation des connaissances utilisé par l'outil de production de systèmes experts CLIPS (Giaratano et Riley 1994). CLIPS a été choisi car il permet de représenter des connaissances en s'appuyant sur trois paradigmes de programmation : la logique des prédicats, l'orienté objet et le procédural. Ainsi, les faits traités par les règles de production peuvent être des objets au sens de la programmation orientée objets (via le langage de description COOL -Clips Object Oriented Language-). Ces objets (et classes d'objets) sont exploitables dans les règles de production, notamment au travers de constructeurs procéduraux sur le côté droit de chacune des règles (constructeurs de type `if ... then ... else, while`). Cette richesse fonctionnelle de CLIPS (qui le rend proche du langage « Spec » utilisé durant ma thèse) a permis de proposer des outils pour assister les concepteurs en phase de spécification d'une situation-problème.

3.2.1.4 L'environnement d'aide à la spécification des situations-problèmes proposé aux concepteurs

Différents types de fonctionnalités ont ainsi pu être expérimentés par spécialisation d'un outil meta-CASE : Hardy (Smart et Rae 1995). Développé dans un laboratoire d'intelligence artificielle à l'Université d'Edimbourg, cet outil permet d'associer une sémantique précise aux graphes conceptuels produits par un utilisateur via une interface diagrammatique (voir les exemples de diagrammes présentés au paragraphe 3.1.1.2). Hardy est orienté événements et écrit en wxCLIPS (Smart 1995) qui étend CLIPS pour interpréter des interactions Homme-Machine.

Partant de cet environnement, nous avons défini les types de diagrammes utilisables selon que le concepteur cherchait à décrire la matrice disciplinaire, les stratégies d'enseignement ou des éléments de la situation-problème. (cf les copies d'écran de la Figure 3.6 et de la Figure 3.7). Nous avons ensuite adapté l'interface proposée par l'environnement HARDY en la dotant de fonctionnalités à vocation cognitive qui sont accessibles à l'initiative du concepteur : menus permettant de connaître les préconditions, post-conditions d'une étape de la spécification, menus permettant de valider un ensemble de diagrammes produits par le concepteur, outils de recherche utilisant des critères portant sur le processus de spécification, outils de recherche et de sélection d'objectifs d'apprentissage / d'activités extraits de bibliothèques reprenant les instructions officielles de l'Education Nationale, ... D'autres fonctionnalités à l'initiative de l'environnement de spécification mis à disposition des acteurs ont également été développées (Marquesuzaà, Meyranx et al. 1997) :

- contrôle des actions que l'utilisateur peut activer (masquage des items de menu en dehors du contexte de spécification courant),
- rappel des étapes déjà menées, des étapes commencées mais non terminées, des étapes restant à réaliser,
- informations sur les incohérences potentielles d'un modèle produit par le concepteur,
- proposition de rappels sur la sémantique des arcs et nœuds d'un graphe conceptuel à telle étape du processus de spécification,
- proposition de modèles déjà produits se rapprochant (ressemblance au niveau du graphe conceptuel de description, similarité des termes utilisés) du modèle en cours de production par un concepteur et aide à la réutilisation.

La quasi totalité de ce qui est modélisé grâce aux diagrammes conceptuels faits avec HARDY peut être également représentée en CLIPS et par conséquent ajoutée dans la base de faits. L'exécution des règles portant sur le processus de spécification nous a permis d'assurer non seulement un guidage par la méthode, mais aussi par les actions de l'utilisateur tout en offrant une souplesse au système. Le prototype développé par C. Marquesuzaà a ensuite été utilisé par des enseignants pour décrire a posteriori des situations-problèmes qu'ils avaient mis en œuvre dans leur classe.

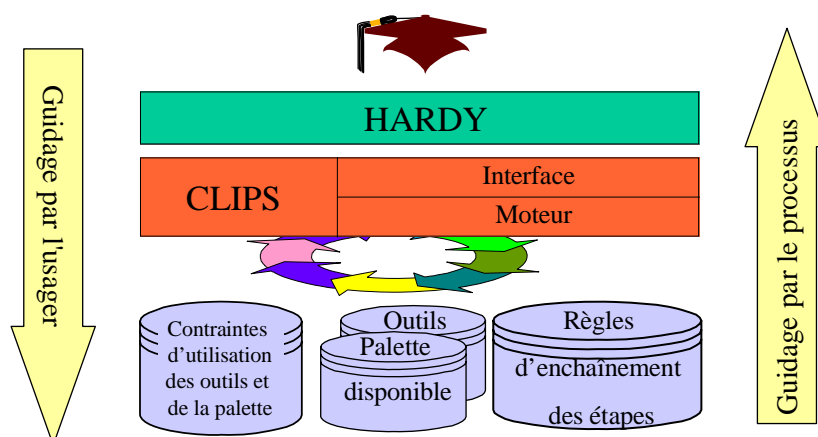


Figure 3.21 : L'environnement de spécification des situations-problèmes (Marquesuzaà 1998)

L'évaluation de ce prototype a montré que les concepteurs et les enseignants ont besoin de pouvoir accéder (pour se les approprier et les paramétrer) aux spécifications des outils mis à leur disposition et qu'ils avaient également tendance à détourner de leurs usages certains des outils mis à leur disposition ; avoir une représentation séparée du code des outils pour tout ce qui porte sur le processus de spécification ainsi que sur les activités qu'un outil permet d'accomplir facilite grandement l'adaptation d'un environnement à ses utilisateurs (flexibilité vis-à-vis de l'utilisateur et guidage par les processus).

3.2.1.5 La prise en compte de l'environnement d'exploitation des situations-problèmes coopératives

L'analyse des usages de l'environnement de spécification développé par C. Marquesuzaà a montré que cet outil permettait aux enseignants de décrire et améliorer des spécifications de situations-problèmes qu'ils avaient préalablement imaginées sur papier. L'utilité de cet environnement s'est révélée plus discutable pour préparer le développement d'applications informatiques supportant de telles situations-problèmes. Partant de ces constatations du terrain, j'ai proposé, dans un article publié en 2001, d'orienter les recherches vers des modèles de spécification des situations-problèmes tenant compte du contexte de diffusion envisagé par les concepteurs, donc de l'environnement d'outils susceptible d'être utilisé par les apprenants (Nodenot 2001).

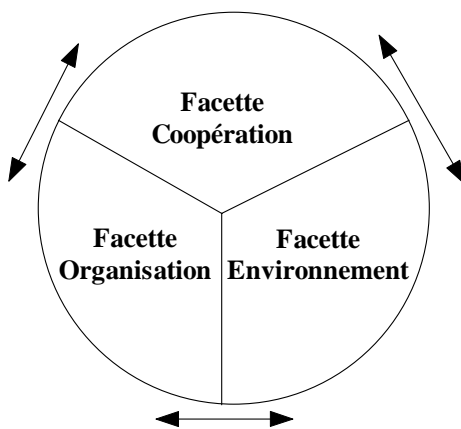


Figure 3.22 : Différentes facettes d'une application éducative (Nodenot 2001)

Le sujet de thèse proposé à Pierre Laforcade visait donc la conception de situations-problèmes coopératives pour des plates-formes de formation à distance à base de composants. Cette prise en compte de l'environnement-support a permis de repenser une partie du processus de conception proposé dans la Figure 3.19, tout particulièrement les points et de l'heuristique proposée²⁰.

Partant des travaux de (Schneemayer 2002) traitant du problème de la contextualisation de composants logiciels au service d'une situation d'apprentissage donnée, nous avons proposé dans (Nodenot, Marquesuzaà et al. 2004) un processus de conception exploitant le profil CPM (Laforcade, Barbier et al. 2003) : ce processus relie des modèles structurels, cognitifs et sociaux d'une situation-problème avec des abstractions de composants logiciels disponibles sur une plate-forme (ou une classe de plates-formes). Le modèle de processus proposé est présenté dans la figure suivante :

²⁰ cf la tâche de l'heuristique intitulée : « intégrer les différentes descriptions, si nécessaire, et les mettre à disposition de l'autre classe d'utilisateurs ».

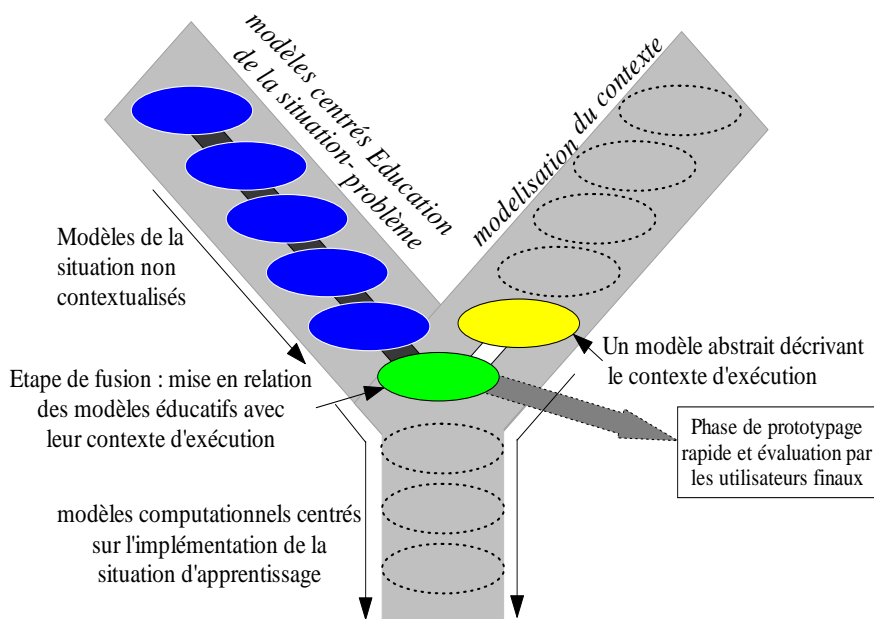


Figure 3.23 : Prise en compte du contexte par les modèles de conception

Cette figure montre les différents types de modèles pris en compte par le profil CPM (ovales de couleur bleue, jaune et verte) ; les ovales en pointillés représentent des modèles non pris en charge par le profil CPM. Dans cette figure :

- La partie gauche du « Y » est centrée Education et exploite les éléments du méta-modèle CPM permettant de décrire la situation-problème à différents niveaux d'abstraction (vues externes, vues permettant d'organiser le scénario proposé aux acteurs, vues centrées sur les interactions, ...).
- La partie droite du « Y » est pour une bonne part en dehors de préoccupations du méta-modèle CPM. Dans l'architecture MDA (OMG 2001) dont s'inspire le processus proposé, les modèles apparaissant dans cette partie décrivent l'environnement technologique d'une application à mettre en place (cf Platform Description Models du MDA) : plus les modèles sont bas dans la partie droite du « Y » et plus la spécification de l'environnement d'exploitation est abstraite et stable –cf les abstractions des plateformes de FOAD proposées dans (Dewanto 2005). Au niveau du profil CPM, seules sont prises en compte et représentées les spécifications abstraites des composants qui constituent le contexte d'exécution d'une situation-problème (cf ovale coloré en jaune). La partie du profil CPM permettant de décrire ces composants abstraits est présentée dans le paragraphe 3.2.2.2.
- L'étape de fusion (voir ovale vert) consiste, via des éléments du méta-modèle CPM (voir Figure 3.26) à mettre en relation les modèles de la situation avec les modèles de composants.
- La partie basse du « Y » n'est pas réellement dans le champ du profil CPM (ovales en pointillés). Le paragraphe 3.2.2 présente cependant des exemples de transformations permettant de passer des modèles abstraits contextualisés d'une situation-problème à des modèles implantant / opérationnalisant cette situation pour un environnement-support particulier (transformations CPM / IMS-LD, transcription des spécifications de composants éducatifs en spécifications de composants logiciels pour une plate-forme donnée).

Le processus que suggèrent ces travaux repose sur trois niveaux pour intégrer les descriptions des différents acteurs participant à la conception de situations d'apprentissage : une dimension horizontale qui permet d'analyser une situation-problème selon différents points de vue (cf modèles structurels, cognitifs et sociaux présentés au paragraphe 3.1.2.3.1) ; une dimension verticale s'appuyant sur le modèle en « Y » pour aborder l'analyse d'une situation d'apprentissage à différents niveaux d'abstraction (des vues

externes à la description détaillée des activités des apprenants exploitant les fonctionnalités d'outils mis à leur disposition) ; une dimension en profondeur qui s'appuie sur les techniques de métamodélisation pour relier : a/ les modèles centrés « éducation » de la situation d'apprentissage (modèles issus de CPM par exemple) avec des modèles centrés « technologie » (modèle EJB, modèle EJOSA, modèle ADL SCORM, ...), et b/ les modèles de la situation (ou modèles du produit) avec les modèles décrivant le processus de modélisation.

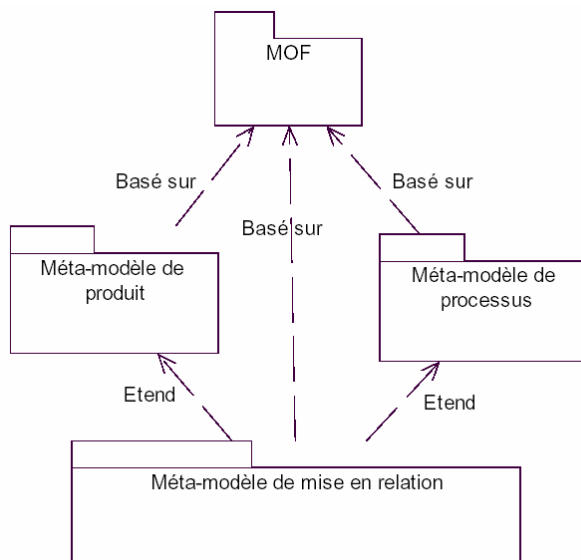


Figure 3.24 : Mise en relation des méta-modèles de produit et de processus (Breton 2002)

La description formelle d'un tel processus ne faisait pas partie des objectifs de la thèse de Pierre Laforcade et ces premières propositions n'ont donc pas été pour le moment évaluées en vraie grandeur. Cette proposition de processus est cependant actuellement étudiée par un groupe de recherche constitué de chercheurs de l'équipe IDEE du LIUPPA, des chercheurs du groupe GHYM à la faculté de San Sebastian (Espagne) et des entreprises ou institutions locales ayant mené récemment des projets en E-Learning. Le travail d'analyse est mené dans le cadre d'un projet Aquitaine-Euskadi-Navarre de 3 ans piloté par P. Lopistéguy sur le thème « Développer une compétence universitaire trans-frontalière sur le processus E-learning » (Lopistéguy 2004), (Lopistéguy 2005) : considérant une dizaine de projets E-learning ayant abouti à des applications exploitées par des apprenants tant côté espagnol que français, le travail consiste à décrire le processus de développement utilisé pour chacune de ces applications, à examiner les problèmes qui apparaissent de manière récurrente pour proposer un modèle de processus rendant compte des niveaux de décision et structurant donc les retours d'expérience proposés par ce groupe de travail : les différentes versions de processus proposées dans le paragraphe 3.2.1 sont ainsi confrontées à celles proposées par les autres participants de ce projet.

3.2.2 Le problème de l'exploitation des modèles de conception

Cet aspect n'a réellement été abordé que dans la deuxième partie de la thèse de Pierre Laforcade. Les techniques utilisées consistent à traiter de manière automatique les modèles de conception produits avec CPM de façon à les « mapper » avec des formalismes de représentation cibles. Les premiers développements ont porté :

- Sur un outil opérationnel permettant de transformer des modèles CPM en représentations IMS-LD (Laforcade 2005a) afin de pouvoir les exploiter sur des plateformes de formation à distance disposant d'un interpréteur IMS-LD (cf paragraphe 3.2.2.1),
- Sur des règles de transcription permettant de transformer des spécifications de composants éducatifs tels que définis dans (Laforcade 2004) en spécifications tenant

compte de la plate-forme d'exploitation choisie (Nodenot, Gaio et al. 2005b). Les développements en cours sont présentés dans le paragraphe 3.2.2.2.

3.2.2.1 Transformation des modèles CPM en représentations IMS-LD

La technique utilisée a pour but de transformer un modèle UML vers un autre type de modèle. La transformation est réalisable grâce aux éléments spécifiques définis dans les profils UML : stéréotypes et définitions de valeurs marquées. Dans le cas du profil CPM, ces éléments informent et dirigent la collecte d'informations contenues dans les modèles UML créés à partir de CPM (Laforcade, Nodenot et al. 2005). Cette technique permet de transformer des diagrammes d'activités modélisant des scénarios d'apprentissage à l'aide du langage CPM vers des modèles XML conformes au standard de la spécification actuelle d'IMS-LD ; l'outil accessible à partir de l'interface Objecteering est capable de générer sans problème le niveau A de la spécification IMS-LD :

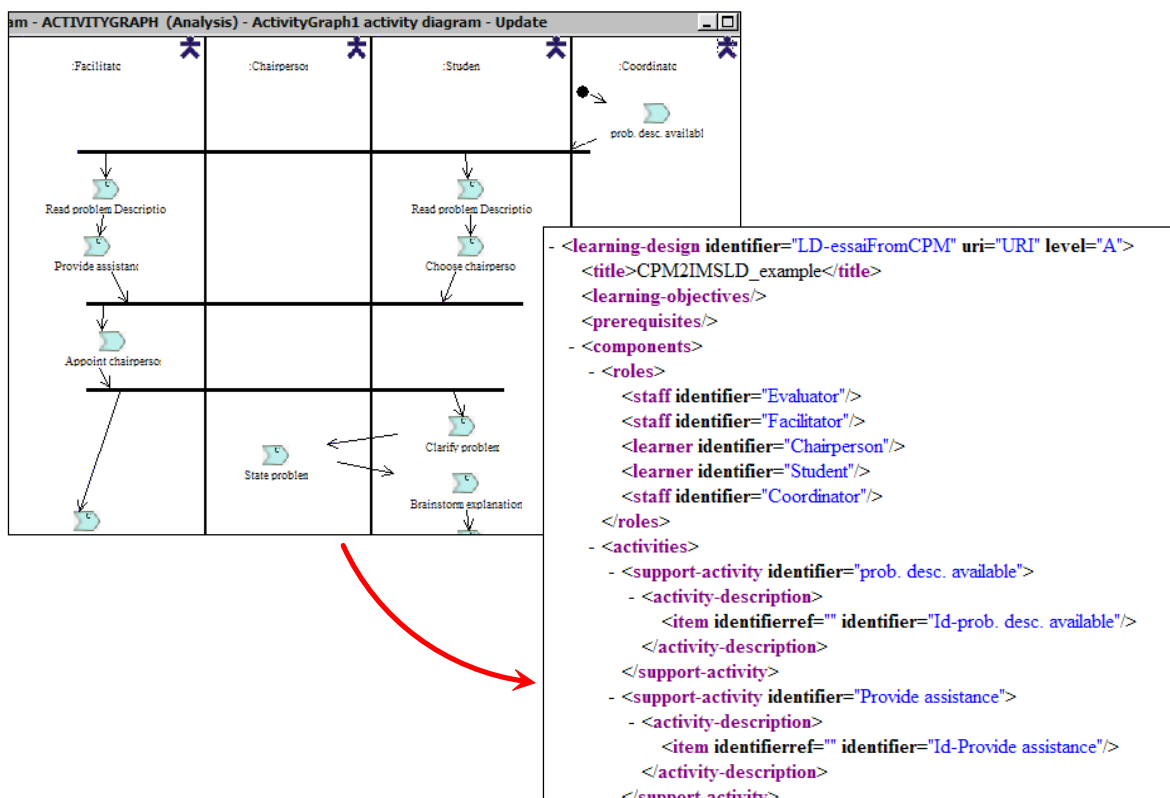


Figure 3.25 : génération automatique d'un fichier XML conforme à la spécification IMS-LD à partir d'un diagramme d'activités CPM

Pour mener à bien ce travail, nous avons exploité une caractéristique de l'AGL Objecteering qui permet de gérer / interroger les modèles comme un véritable système d'informations : ainsi, si le nom d'une classe est modifié, le changement est répercuté dans tous les autres diagrammes dans lesquels cette classe est utilisée ; de même, tout élément stocké dans un modèle peut être analysé et exploité via le langage de programmation propriétaire « J ». Les éléments de terminologie du langage CPM sont traités par Objecteering comme des instances de stéréotypes et de valeurs marquées (voir paragraphe 3.1.2.3.2), ce qui permet de connaître pour tout élément de modélisation le stéréotype et les valeurs marquées qui lui sont appliqués.

La démarche technique employée pour extraire les informations relatives au modèle cible conforme à IMS-LD consiste à déterminer pour chaque concept d'IMS-LD (Laforcade 2005b) :

- o La méta-classe UML et le stéréotype utilisés dans CPM correspondant à un concept IMS-LD donné,

- Optionnellement (si la méta-classe et le stéréotype ne suffisent pas pour déterminer le concept IMS-LD correspondant) la définition de valeur marquée apporte une information supplémentaire significative pour l'identification du concept.

Le modèle cible conforme à IMS-LD est alors construit dans l'ordre, chaque composante étant déduite d'informations extraites du modèle CPM. Les stéréotypes (représentés pour la plupart dans la partie gauche de la Figure 3.25 par une icône spécifique) sont ainsi facilement identifiables par des algorithmes de recherche : voir les stéréotypes *Role*, *CollaborativeActivity* et *ActivityStructure*. Le découpage en *actes IMS-LD* pour un *Play* donné nécessite un algorithme spécifique dans la mesure où ces notions n'apparaissent pas dans les diagrammes d'activités ; cet algorithme est basé sur les principes suivants :

- L'utilisation des barres de synchronisation (*fork* et *join* des diagrammes d'activités UML) permet de modéliser le parallélisme des activités,
- Les flèches traversant les couloirs expriment un besoin de précédence entre les deux activités reliées,
- Les cadres entourant des activités qui se succèdent pour un même rôle font référence aux *Activity-Structure* spécifiées dans le document IMS-LD.

A l'issue de sa thèse, Pierre Laforcade a entrepris un travail portant sur la prise en compte des niveaux B et C (cf 2.2.2.2) de la spécification IMS-LD (il y a peu d'études de cas suffisamment précises sur les niveaux B et C d'IMS-LD, ce qui rend ce travail difficile à mener). Nous envisageons aussi d'utiliser d'autres diagrammes, par exemple les diagrammes de classes décrivant les objectifs et les pré-requis d'une situation-problème. Ainsi, d'autres éléments terminologiques de la spécification IMS-LD pourront être générés et manipulés à partir de nos modèles CPM.

En revanche, ces travaux d'expérimentation ont montré que de nombreuses informations capturées dans les modèles CPM n'ont pas de correspondance avec IMS-LD (toutes les notions relatives à la terminologie des situations-problèmes, tous les concepts décrivant le détail d'une activité pédagogique donnée, toutes les notions relatives aux aspects sociaux tels la gestion des droits des usagers sur les ressources et les activités, ...) ; ceci est en partie dû au positionnement plus amont de CPM dans le processus de conception. Pour opérationnaliser une spécification faite avec CPM, cette perte d'information peut sans doute être contournée puisque les *players* IMS-LD s'appuient sur des références d'activités (dont le comportement peut très bien respecter les spécifications faites avec CPM). De plus, le modèle événementiel proposé par IMS-LD est suffisamment général (événement – condition – action) pour exécuter tout type d'activité selon un modèle de workflow donné. En ce qui concerne les traces sur les activités menées réellement par les apprenants, la perte de sémantique serait beaucoup plus problématique (la perte du lien entre le modèle de conception et le modèle d'implémentation me semble aller à l'encontre des pratiques en matière de maintenance logicielle et industrielle).

Le travail portant sur les transformations vers IMS-LD nous a permis de démontrer les possibilités offertes par l'environnement d'outils bâti à partir de CPM. Les démonstrations de l'outil que nous avons pu faire lors de l'école d'été 2005 du RTP39 ont montré l'intérêt de la communauté pour ce type de transformations (aspects ingénierie et rétro-ingénierie) : les participants à l'école ont par ailleurs rapidement su exploiter/interpréter le formalisme graphique issu d'UML enrichi par les concepts CPM (permettant de décrire les ressources, le détail des activités, les collaborations, ...). Nous considérons cependant que transformer des modèles CPM en modèles compatibles IMS-LD n'est pas une finalité sur laquelle il faut centrer les travaux (voir notamment le paragraphe 2.2.2.4) : l'enjeu consistant à se servir des modèles CPM pour spécifier des composants éducatifs nous paraît plus pertinent pour la communauté EIAH.

3.2.2.2 Du profil CPM aux modèles de composants éducatifs

Nous avons proposé un modèle pour la spécification d'un composant éducatif appelé CPL (Composant Pédagogique Logiciel) afin de rapprocher les modèles des situations des

modèles des plates-formes de formation à distance. Du point de vue des concepteurs d'unités d'apprentissage, le composant CPL représente une activité pédagogique élémentaire et non un outil. Cette activité est une brique de base réutilisable pour la conception d'activités pédagogiques plus complexes pour les apprenants comme pour les tuteurs. Les services pédagogiques fournis par ces composants sont dynamiques et dépendent du comportement de l'utilisateur réalisant l'activité. En ce sens, les services ne sont plus transversaux aux activités d'apprentissage, comme ils le sont actuellement dans les LMS, mais intégrés à l'activité. Du point de vue du développeur de composant, le composant CPL représente un composant logiciel « métier » construit sur la base des composants fonctionnels traditionnels des LMS ; le principe étant de contraindre les outils existants pour en bâtir de nouveaux adaptés aux besoins apparus lors de la conception.

Le modèle adapté aux composants CPL proposé dans (Laforcade et Barbier 2004), (Barbier et Laforcade 2005) est basé sur le concept de composant d'UML 2.0. Ce modèle permet d'aider les développeurs de composants à décrire, spécifier, puis développer ces composants CPL mais il permet surtout aux concepteurs de situations-problèmes de spécifier de nouveaux modèles de conception avancée pour leurs scénarios pédagogiques en réutilisant des activités de base proposées dans une bibliothèque de CPL. Le langage CPM a également été étendu afin de prendre en compte le concept d'activité élémentaire réutilisable et permettre ainsi l'élaboration des modèles de conception avancée (voir Figure 3.26) :

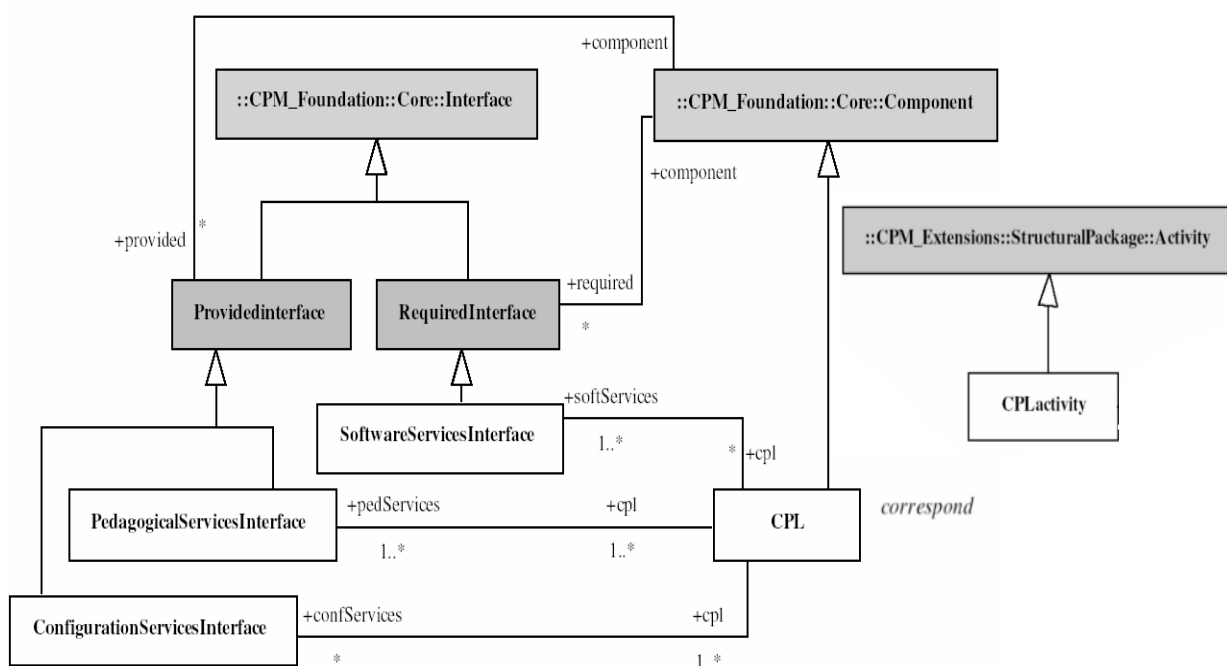


Figure 3.26 : Le paquetage des composants CPL d’après (Laforcade 2004)

Les stéréotypes et valeurs marquées permettant d’exploiter ces concepts dans des modèles CPM ont été définis dans la thèse de Pierre Laforcade et l’approche a ensuite été mise à l’essai sur le cas d’étude Smash²¹ : l’exemple proposé est celui d’une activité élémentaire de gestion de conflits qui a pour but d’établir et piloter une conversation synchrone entre acteurs (rôles d’apprenants et de tuteur spécifiques) leur permettant de s’expliquer sur des divergences d’opinion (interprétations différentes d’un même témoignage, prise en compte de témoignages complémentaires). Ce composant de gestion de conflit utilise et contextualise un composant traditionnel de communication synchrone : le Chat.

²¹ Pour plus de détail, voir (Laforcade 2004) : page 254 et suivantes.

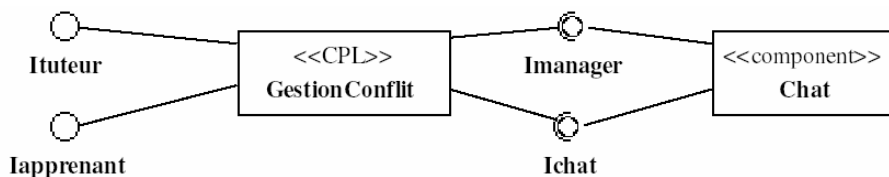


Figure 3.27 : Modélisation « boîte noire » du composant CPL de gestion des conflits

En utilisant les concepts de CPM, le composant de Gestion de Conflit a été spécifié (modélisation boîte noire et description comportementale pour les interfaces *Ituteur* et *Iapprenant*) ainsi que son contexte d'utilisation du composant de Chat (préalablement modélisé). Pierre Laforcade a ensuite développé un composant Java sous forme d'une servlet (voir chapitre 10 de (Laforcade 2004)) en utilisant une librairie Java développée au LIUPPA pour exploiter les statecharts ainsi spécifiés (Barbier 2003), (Barbier et Constant 2005). Ce composant a été couplé au tableau blanc présenté dans la Figure 3.16 pour initier automatiquement des dialogues (via le Chat) portant sur le conflit d'interprétation détecté.

Nous prolongeons ici cette proposition en décrivant des travaux que nous venons d'initier pour décrire les composants logiciels requis pour une activité donnée. L'approche consiste à utiliser une approche de type MDA (Model Driven Architecture) appliquée à la formation à distance qui vise à automatiser la transformation de modèles de plates-formes abstraits (de type PIM ou Platform Independent Model dans le jargon MDA) en modèles (de type PSM ou Platform Specific Model dans le jargon MDA) tenant compte des spécificités de telle ou telle plate-forme concrète (Dewanto 2005) :

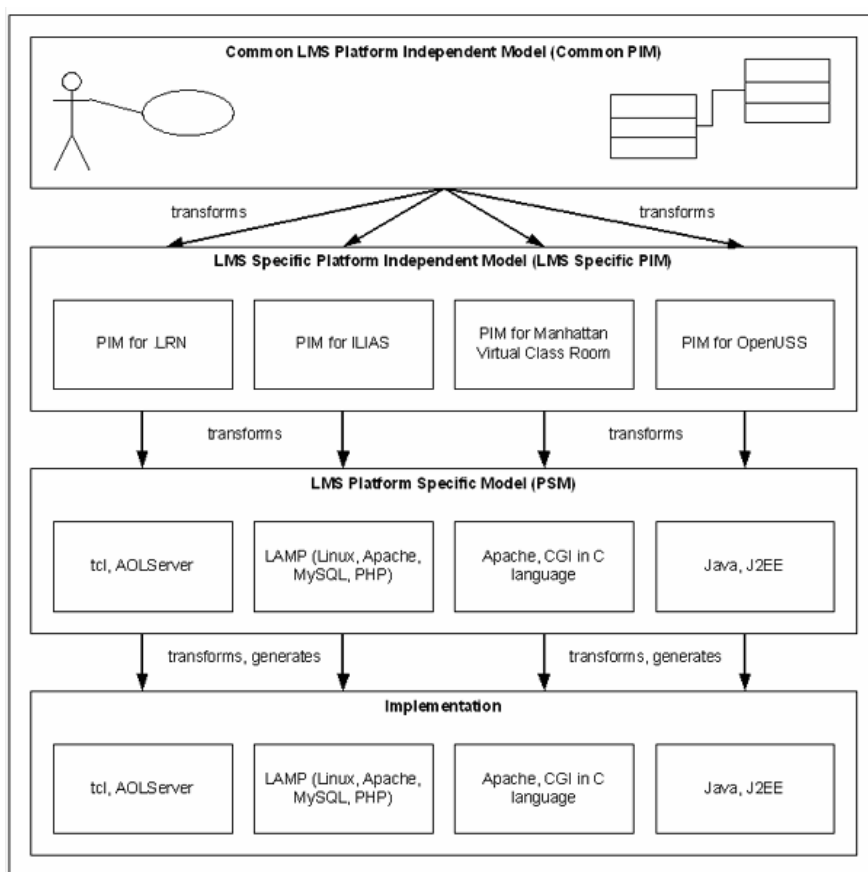


Figure 3.28 : Plates-formes de FOAD et transformation de modèles d'après (Dewanto 2005)

Partant d'une codification XMI d'un modèle UML décrivant des entités significatives d'un système de FOAD (l'exemple utilisé étant un diagramme de classe modélisant un étudiant tel qu'il est perçu par les systèmes de FOAD classiques), (Dewanto 2005) propose en effet des

cartouches MDA (l'outil utilisé est AndroMDA²²) capables, quand on les exécute, de transformer cette représentation abstraite en une version spécialisée tenant compte de l'infrastructure support (modèle EJB, modèle EJOSA compatible avec la plate-forme OpenUSS, ...). Une cartouche incorpore un ensemble de règles de transformation pour passer d'un modèle à un autre en exploitant les stéréotypes et valeurs marquées du modèle à transformer. Il s'agit d'un procédé totalement opérationnel qui est un mécanisme de base utilisé pour compiler les sources de la plate-forme OpenUSS constituées de sources java, XML, ... mais aussi de fichiers (modèles) XMI exploitables par des éditeurs UML standards.

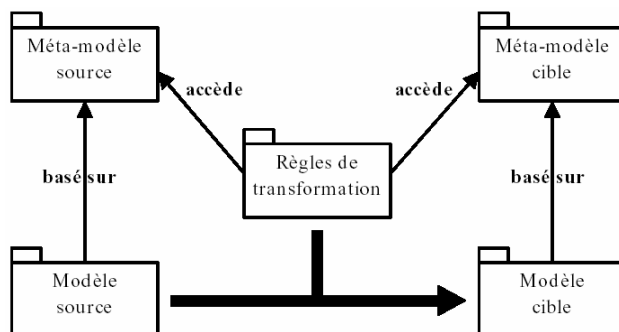


Figure 3.29 : Principe de transformation des modèles

Nous avons repris les cartouches AndroMDA développées par L. Dewanto et nous tentons de les étendre (Nodenot, Gaio et al. 2005b) pour mettre en correspondance les modèles de composants éducatifs issus de CPM (services fournis et services requis par les composants éducatifs) avec des modèles de composants logiciels respectant le cahier des charges de l'infrastructure EJOSA utilisée par OpenUSS. Des travaux de développement sont en cours (Pohu, Louberry et al. 2006) mais ils ne sont cependant pas encore suffisamment avancés pour présenter des résultats confirmant la plus-value de cette approche pour étendre les fonctionnalités de la plate-forme OpenUSS en fonction des activités à conduire. Signalons les travaux assez comparables menés actuellement au laboratoire Trigone dans le cadre du projet BRICOLES pour « mapper » des modèles IMS-LD avec les modèles non standards (de type PSM dans le jargon MDA) utilisés par des plates-formes de formation à distance comme Ganesha ou Claroline (Renaux, Caron et al. 2005), (Caron, Derycke et al. 2005).

3.3 Perspectives de recherche

La dernière partie de ce chapitre a pour but de présenter les perspectives de recherche que je compte explorer et approfondir au cours des années à venir. Certaines s'inscrivent dans le prolongement des travaux déjà réalisés, d'autres relèvent de problématiques déjà soulevées mais pas encore suffisamment explorées. Dans les sections qui suivent, je proposerai tout d'abord un rapide bilan et quelques perspectives immédiates des travaux que j'ai menés jusqu'ici puis je présenterai de manière globale mon projet scientifique. Ceci m'amènera à préciser la problématique des actions retenues pour ce projet ainsi que quelques orientations pour conduire ces actions de recherche.

3.3.1 Bilan et perspectives immédiates de ces travaux

Les travaux menés ont porté sur les langages et outils pour la conception d'EIAH dans le but de faciliter l'activité des acteurs impliqués dans la conception mais aussi dans l'implémentation des EIAH. Depuis que je suis enseignant-chercheur à l'UPPA, j'ai conduit différents projets dans ce domaine :

1. Le premier projet qui a été valorisé par la thèse de doctorat de Christophe Marquesuza en 1998 a permis de proposer des outils et une méthode pour la spécification de situations-problèmes par des enseignants :

²² Voir <http://www.andromda.org/>

- Les modèles de conception proposés couvraient la phase d'ingénierie des connaissances relatives à l'élaboration d'une situation-problème, ces modèles s'appuyant d'une part sur une représentation ontologique des situations basée sur la théorie empirique de l'enseignement (cf paragraphe 1.2.1), d'autre part sur les capacités de représentation offertes par le formalisme des graphes conceptuels.
- La méthode était basée sur le prototypage rapide et permettait à l'enseignant de construire ces modèles de situation en s'appuyant sur des bibliothèques d'objets (notions, objectifs d'apprentissage, stratégies d'enseignement) et des règles de construction de modèles.
- Le prototype informatique a été développé à partir d'outils méta-CASE ; ces outils étaient couplés à un générateur de systèmes experts pour vérifier les modèles semi-formels proposés par les enseignants et pour leur fournir une assistance méthodologique.

Ces premières propositions ont permis de montrer l'aptitude (mais aussi certaines limites) des modèles proposés pour supporter la conception de situations-problèmes du point de vue de l'enseignant. Le prototype réalisé a été utilisé par des collègues de l'Inspection Départementale de Lourdes qui ont ainsi exploité l'interface conviviale de cet outil pour mémoriser et améliorer des séquences de situations-problèmes préalablement dispensées dans des classes. Une validation plus complète des propositions implémentées dans ce prototype n'a cependant pas été réalisée car nous n'avons pas engagé les développements nécessaires à l'amélioration de l'interface et des fonctions d'assistance à l'enseignant ; ces développements permettraient des retours d'usage à plus grande échelle qui contribueraient à une meilleure compréhension des façons de procéder des enseignants ; ils contribueraient ainsi à la mise en œuvre de mécanismes d'assistance permettant aux enseignants de raccourcir les cycles de spécification-évaluation des situations d'apprentissage.

Au terme de ce travail, nous avons cependant noté que le caractère statique des représentations ontologiques proposées dans cette thèse n'était pas totalement adapté à la modélisation de la dynamique des apprentissages. Plus généralement, ce travail m'amène également à réfléchir au positionnement des langages de modélisation : proches du langage de l'enseignant, ils permettent de développer facilement des interfaces Homme-Machine répondant au critère d'acceptabilité mais n'offrent pas toutes les garanties en matière d'exploitation computationnelle des modélisations ; comme le bilan de la thèse de Pierre Laforcade le démontre, s'ils sont plus formels, ces langages permettent d'effectuer vérifications et transformations mais sont plus difficiles à outiller via des interfaces Homme-Machine conviviales. Ce travail de positionnement fin des outils et langages de conception d'enseignements me paraît donc devoir être poursuivi.

2. Le deuxième projet, valorisé par la thèse de doctorat de Pierre Laforcade en 2004, a été initié pour prolonger les travaux précédents. Ces travaux en ingénierie des EIAH avaient deux ambitions : a/ une conception de situations d'apprentissage dirigée par des modèles pour pouvoir réutiliser et mieux exploiter les spécifications produites et ainsi faciliter le travail de mise en œuvre ; b/ une approche visant à ne pas développer de nouveaux environnements informatiques mais plutôt se diriger vers une réutilisation de l'existant, ce qui passe par une étape d'adaptation entre les besoins de l'enseignant et l'environnement informatique dont il dispose.

Ce travail a permis de proposer un langage implémenté sous forme d'un profil UML qui permet à l'équipe de conception de spécifier, via un environnement de type AGL, les différentes vues de l'apprentissage visé dans une perspective de mise en œuvre de ces situations sur des plates-formes de FOAD. Ces vues s'organisent selon trois dimensions :

- une dimension verticale pour représenter des situations d'apprentissage à différents niveaux d'abstraction (des vues externes décrivant le scénario et les activités proposées aux apprenants à des descriptions très concrètes exploitables par des plates-formes de Formation à Distance par exemple),
- une dimension horizontale pour tenir compte des différents points de vue sur de telles

situations : nous avons en particulier proposé des éléments de langage pour décrire les aspects structurels, cognitifs et sociaux des situations-problèmes coopératives, une dimension en profondeur s'appuyant sur le méta-modèle CPM pour intégrer et corrélérer les différentes vues horizontales et verticales (modèles et instances de modèles) proposées par les concepteurs.

La validation des propositions est partielle puisque les travaux menés ont permis de montrer, d'une part que les modèles produits par les usagers restent cohérents vis-à-vis d'UML et du langage CPM, d'autre part que l'outillage informatique (construit à partir de CPM et de l'AGL Objecteering) permet : a/ à des concepteurs connaissant CPM et UML de scénariser la situation-problème Smash et d'en décrire sa dynamique. b/ à des développeurs d'exploiter les modèles produits pour aller vers l'implémentation de composants éducatifs ou pour transformer les modèles CPM en modèles opérationnels au format IMS-LD.

La validation expérimentale du langage CPM suppose cependant d'aller au devant de la communauté EIAH pour mettre à sa disposition le modèle théorique CPM et sa version outillée (le profil CPM exploitable par l'AGL Objecteering). C'est ce que nous avons entrepris tout particulièrement dans le cadre des travaux de l'AS « Conception d'une Plate-forme pour la recherche en EIAH » mais aussi en participant à la préparation de l'atelier "Outils de Modélisation" pour l'école d'été 2005 du RTP39 ; au cours de cet atelier, les participants à cette école ont pu exploiter divers outils (dont le profil CPM) pour spécifier une même situation d'apprentissage et comparer in-situ la puissance d'expression du langage CPM par rapport à des langages comme MOT+ et IMS-LD.

Du point de vue de l'outillage proposé pour le langage CPM, je pense que l'implémentation du profil sur un AGL propriétaire comme Objecteering n'est qu'une étape. Ce choix a permis de rapidement produire de premiers outils aidant les concepteurs de situations d'apprentissage coopératives mais la contrepartie est que les modèles produits restent pour l'instant des modèles patrimoniaux c'est-à-dire liés à l'AGL Objecteering ; le fait d'étendre les fonctionnalités de cet AGL pour transformer des modèles CPM en modèles compatibles IMS-LD est encore une étape provisoire et l'approche serait beaucoup plus générale si des outils de transformation de modèles étaient développés en dehors d'Objecteering sur la base de la représentation XML des modèles produits par les concepteurs. CPM deviendrait donc un DSL (Domain Specific Language) de petite taille facilement manipulable par des outils externes (éditeurs généraux, générateurs, vérificateurs, ...) pour produire des modèles divers réclamés par la communauté EIAH et pouvoir intégrer ainsi différents espaces technologiques (modèles de transformation pour la génération de code IMS-LD, modèles de transformation pour la génération de composants logiciels type EJB, ...).

Un bilan d'étape, proposé dans la Figure 3.30, situe ma démarche et les résultats obtenus à ce jour. La figure met également en évidence deux points :

- le peu de recul actuel sur le langage CPM dans la mesure où les modélisations ont porté sur une situation d'apprentissage unique, la situation Smash, ces modélisations ayant été produites par les seuls promoteurs de ce langage. La modélisation que j'ai effectuée très récemment sur l'acte 2 de Smash (voir annexe E) a cependant confirmé les orientations prises dans la thèse de Pierre Laforcade pour modéliser l'acte 3 de cette même situation.
- les difficultés pour opérationnaliser les modélisations dans des EIAH. Les développements d'outils pour les apprenants ont été mis en œuvre pour vérifier la faisabilité de l'approche : ils ont donc abouti à la mise en œuvre d'activités ciblées (cf la scène de « debriefing » de l'enquête exploitant un composant de type tableau blanc) et non à un EIAH abouti. Le nombre très limité de développeurs au service de ce projet est un élément d'explication qui est amplifié par la faible mutualisation d'outils / composants créés par la communauté EIAH.

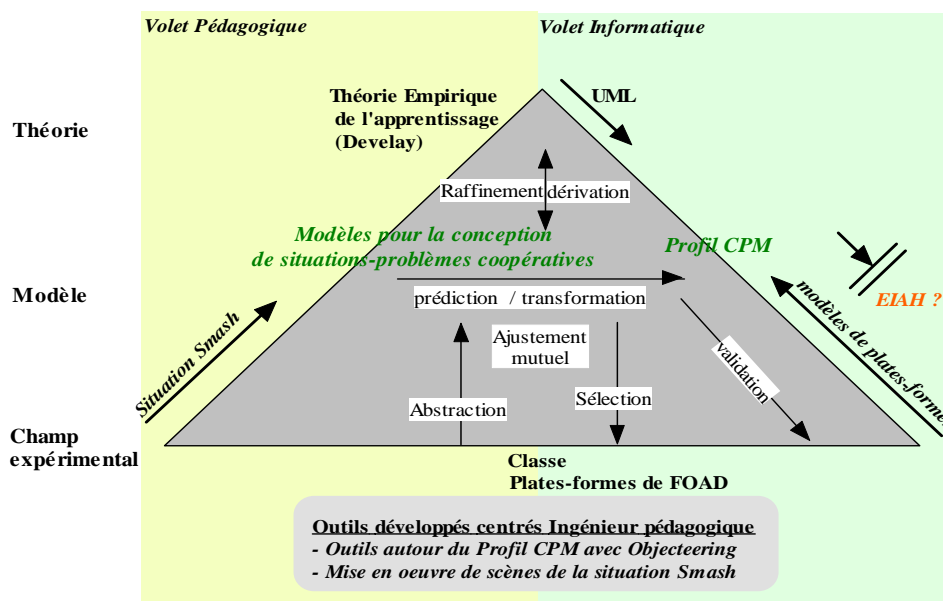


Figure 3.30 : Bilan d'étape des travaux menés

La Figure 3.30 montre clairement que ces travaux n'ont pas atteint, pour l'instant, le stade qui permettrait de mettre en oeuvre, via une plate-forme de FOAD, les modèles issus de CPM et donc de confronter l'EIAH obtenu au comportement réel d'apprenants (champ expérimental). Les travaux en ingénierie des EIAH relèvent d'un domaine qu'il faut piloter par les applicatifs ; les résultats obtenus jusqu'ici, s'ils contribuent à réduire le fossé entre les théories d'apprentissage et les environnements-supports aux apprentissages ne sont qu'une contribution qui vient compléter :

- les résultats d'autres équipes françaises (cf le projet REDIM et l'environnement OSCAR au laboratoire LIUM de l'Université du Maine, cf les travaux menés sur les scénarios pédagogiques au laboratoire CLIPS de Grenoble, cf les outils développés autour du cartable électronique par le laboratoire SYSCOM de Chambéry, cf les travaux du laboratoire TRIGONE, notamment le projet BRICOLES, ...),
- les résultats d'autres équipes internationales (travaux de l'Open University sur les EML, travaux sur les plates-formes à base de composants OpenUSS et SAKAI, travaux menés au laboratoire TECFA sur les portails communautaires, travaux menés à l'université de Pittsburgh sur les LMS adaptatifs, ...).

Le projet de recherche qui sera présenté dans le paragraphe 3.3.2 propose différentes initiatives pour coordonner et agréger ces différentes contributions, ce qui aura pour conséquence de renforcer la visibilité internationale de la communauté de recherche française en EIAH.

A l'issue de ce travail, il est aussi nécessaire d'examiner la problématique des méthodes d'ingénierie pédagogique qui pourraient exploiter un langage comme CPM. Nous avons volontairement situé ce langage à un niveau amont par rapport à des langages comme IMS-LD et nous savons bien que pour impliquer des acteurs tels que les enseignants dans ces phases amont, un support méthodologique et des environnements dédiés bien différents de l'AGL Objecteering devront être proposés. Plusieurs pistes me paraissent possibles :

- s'appuyer sur CPM pour proposer une méthode inspirée des travaux autour des méthodes UML. Comme UML, le langage CPM est un langage de modélisation orienté objet mais indépendant d'une quelconque méthode objet. Ceci m'amènerait à proposer un processus de modélisation itératif et incrémental dirigé par les cas d'utilisation et centré architecture.
- s'orienter vers une méthode inspirée des travaux autour du design pédagogique qui consisterait, comme pour la méthode MISA, à appliquer des techniques de modélisation cognitive, à la fois pour la représentation des connaissances, des traitements pédagogiques et médiatiques. Il est clair que le travail de Pierre Laforcade, assez

marqué par l'approche Génie Logiciel, n'est pas allé au bout des capacités du langage CPM pour l'ingénierie des connaissances en EIAH alors que c'était un des aspects intéressants de la thèse de Christophe Marquesuzaà. Je pense en particulier que l'ingénierie des connaissances associée à la phase de régulation pédagogique pourrait être un travail particulièrement intéressant à mener à partir des concepts du langage CPM (voir notamment le sujet de thèse proposé dans le paragraphe 3.3.2.2).

- s'orienter vers des méthodes de type Agile basées sur le prototypage rapide, la réutilisation et la transformation de modèles. Peu de travaux sont menés dans cette direction par la communauté EIAH mais je pense que le caractère assez formel du langage CPM se prêterait bien à des méthodes donnant une grande importance aux modèles comme support et guide de la conception. Comme première application, il serait particulièrement intéressant de modéliser via CPM des patterns de conception relatifs aux aspects sociaux (modèles généraux proposant des formes générales de coordination entre acteurs), ces patterns constituant des blocs de construction d'une méthode de conception et prototypage rapide centrée sur la réutilisation de modèles.

Le projet scientifique présenté ci-dessous reprend les propositions d'actions de recherche énoncées au paragraphe 3.3.1 en les mettant en perspective par rapport aux problématiques de la communauté EIAH mais aussi par rapport aux travaux actuels de l'équipe IDEE.

3.3.2 Projet scientifique : modèles et outils pour la co-conception des EIAH

3.3.2.1 Problématique et pistes d'investigation

La communauté EIAH s'accorde aujourd'hui pour insister sur le caractère transdisciplinaire de la conception en EIAH. Mais cette caractéristique fondamentale est difficile à assumer totalement dans notre activité quotidienne de recherche :

- L'hypothèse selon laquelle il y aurait neutralité des technologies dans l'expression des spécifications d'un EIAH sous-tend nombre de développements et conduit souvent, par facilité, à séparer chronologiquement la conception du point de vue Sciences Humaines et Sociales de l'analyse informatique. Or, la conception détaillée d'un EIAH (objectifs, notions à acquérir, stratégies d'apprentissage et de tutorat, ...) est souvent influencée par l'analyse et la compréhension des phénomènes liés à la dimension informatique (Tchounikine 2002b), ce qui devrait conduire à intégrer ces différentes dimensions depuis l'analyse des besoins jusqu'à la mise en œuvre et l'évaluation d'un EIAH.
- La communauté de recherche en EIAH a également du mal à intégrer les utilisateurs finaux (enseignants et apprenants) dans la boucle d'analyse – conception - mise en œuvre des EIAH. L'hypothèse selon laquelle les enseignants seraient de simples consommateurs des produits que la communauté met au point me paraît artificielle et illusoire. Ceci devrait conduire réfléchir davantage à la définition d'outils qui leur donneraient une plus grande capacité d'action (« notion d'enactment ») tout au long du processus d'analyse – conception - mise en œuvre d'un EIAH.

Les deux difficultés énoncées ci-dessus me paraissent largement contribuer au problème de fond souligné dans (Tchounikine, Baker et al. 2004) à savoir que pour un EIAH, le schéma basé sur le fait qu'une "théorie permet de construire un modèle qui spécifie entièrement l'artefact est illusoire". Si effectivement l'ingénierie d'un EIAH doit passer par un processus itératif ne dissociant pas les aspects compréhension / modélisation du travail de conception informatique, il faut que des travaux de recherche portent spécifiquement sur la définition d'un cadre théorique rendant possible une conception transdisciplinaire des EIAH. C'est le sens des orientations que je propose sous le titre : « modèles et outils pour la co-conception des EIAH ». Ces propositions ne sont pas des recherches en Génie Logiciel mais s'appuient sur les avancées actuelles dans ce domaine pour faire progresser notre compréhension des systèmes de type EIAH et améliorer notre approche pour la conception de ces systèmes.

Au cours des prochains mois, je souhaite m'engager dans des travaux de fond sur la sémantique des modèles. Jusqu'ici, j'ai essayé de mener un tel travail sur le cas particulier d'apprentissages construits sur la notion de contrat didactique -cf la théorie empirique de l'apprentissage défendue par des auteurs comme (Develay 1993), (Meirieu 1994) et la théorie des situations didactiques de (Brousseau 1998), (Rogalski 2004)-. Ainsi, dans le cadre de la thèse de P. Laforcade, nous avons pu proposer une première série d'éléments de langage (les paquetages du méta-modèle CPM) positionnés clairement du point de vue de leur sémantique, facilitant ainsi l'usage des modèles et leur évaluation (Laforcade 2004). Un travail comparable me paraît pouvoir être mené pour d'autres théories porteuses en EIAH telles que la théorie connexionniste (Laks 1996), la théorie de la cognition située (Clancey 1997), la théorie de l'activité (Engeström, Mietinen et al. 1998), ... Ceci permettrait d'expliciter les spécificités de ces théories mais aussi sans doute certains de leurs points communs (aspects structurels, rôle des connaissances préalables, ...). Cette explicitation des théories qui me paraît très importante pour comprendre les effets réciproques des théories / des technologies de communication sur les modèles de conception, doit s'appuyer sur *des exemples concrets, sortes de scénarios de référence* pour lesquels au moins deux questions de fond doivent être approfondies :

- étant donnée une situation d'apprentissage clairement définie (la situation Smash ou une autre), quels modèles sont affectés par le fait de se référer à telle ou telle théorie ?
- en quoi faire référence à telle ou telle théorie influe-t-il sur le langage d'expression des modèles ?

Pour être signifiant, ce travail doit s'appuyer sur des formalismes suffisamment riches et expressifs, les modèles n'étant pas neutres par rapport au méta-modèle utilisé pour les décrire (Bézivin, Blay et al. 2005). L'enjeu est donc de proposer pour les EIAH des langages de type DSL (Domain Specific Language) séparant la sémantique du langage de sa syntaxe et offrant des mécanismes de validation et de raisonnement en adéquation avec les objectifs de conception et mise en œuvre d'EIAH.

Ceci m'amène à énoncer un deuxième objectif pour mon projet de recherche qui vise à examiner comment les modèles produits avec de tels langages vont pouvoir être utilisés à des fins cognitives et computationnelles. La conception d'un EIAH repose sur des modélisations multiples couvrant différents points de vue sur un tel système et différents niveaux d'abstraction. Nous avons montré dans le cadre des travaux de thèse de Pierre Laforcade que les modèles structurels, cognitifs et sociaux définis par les concepteurs pouvaient être transformés sur la base de la sémantique de leur méta-modèle, ce qui nous a permis de dériver de ces modèles des vues plus opérationnelles de type modélisations IMS-LD et modèles de composants éducatifs. Depuis la fin du doctorat de Pierre Laforcade, nous étudions des langages de transformation de modèles de type QVT (Queries – Views – Transformations), tout particulièrement l'outil AndroMDA déjà cité (cf paragraphe 3.2.2.2) que l'équipe de développement du projet OpenUSS utilise systématiquement maintenant pour dériver des modèles de composants logiciels EJB à partir de modèles UML décrivant les spécifications fonctionnelles du Learning Management System qu'est OpenUSS (Dewanto 2005). Bien qu'impliqué dans l'équipe de développement d'OpenUSS, mon projet n'est pas d'aller prioritairement vers des transformations technologiques mais plutôt d'exploiter les fonctionnalités de tels langages de transformation pour relier les modèles cognitifs et computationnels des EIAH. Une application très directe de ces technologies permettrait par exemple de tisser un modèle d'ingénierie des connaissances et le méta-modèle CPM : les chercheurs de l'équipe MIS à l'Université de York (UK) ont en effet développé un profil UML pour la conception de systèmes à base de connaissances en partant du principe que des méthodes comme CommonKads (Schreiber, Akkermans et al. 1999) ou Moka (Stokes 2001) s'appuyaient déjà sur des extensions non formalisées d'UML pour structurer les connaissances et les mécanismes d'inférence exploitant ces connaissances. Le profil défini dans (Abdullah, Evans et al. 2004), (Abdullah, Paige et al. 2005) propose différents paquetages spécialisant la sémantique d'UML pour de tels systèmes à base de connaissances, spécialisations d'UML que je propose de relier avec les modèles CPM, notamment au travers du concept de « External Description » des modèles

CPM. Ceci m'amènerait vers des travaux sans doute peu lisibles du point de vue des concepteurs finaux mais importants dans une perspective de mise en place de techniques et d'outils permettant de comparer, d'évaluer et mieux utiliser les modèles produits par la communauté et les concepteurs d'EIAH, voire appliquer des transformations automatiques à ces modèles.

On ne peut en effet préjuger du caractère immédiatement opérationnel des travaux proposés dans une perspective de diffusion d'outils à destination des concepteurs d'EIAH car l'ingénierie des EIAH dirigée par les modèles telle qu'elle est proposée peut sembler éloignée de la problématique initiale liée au caractère transdisciplinaire des EIAH (cf début du paragraphe 3.3.2). Je crois cependant que les deux actions de recherche proposées (travail sur la sémantique des modèles et des méta-modèles d'EIAH, travail sur le tissage de modèles via leurs méta-modèles) vont permettre : a/ de guider la conception et b/ de produire des outils qui faciliteront la coopération entre chercheurs de disciplines différentes en masquant aux différents acteurs la complexité des techniques sous-jacentes. Plusieurs éléments m'amènent à ce point de vue :

- Le processus de développement d'un EIAH amène les concepteurs à s'engager dans des activités diverses couvrant l'analyse du domaine et des connaissances supposées des apprenants, l'analyse des interactions souhaitées, la perception par le système des actions de l'apprenant, la médiatisation des contenus, ... Aujourd'hui, chacune de ces activités conduit à la production d'artefacts de type documentation, diagrammes, code-source, rapports d'expérimentation, ... qui offrent des vues multiples sur l'EIAH en cours de développement mais se révèlent indépendantes car aucune cohérence entre ces différentes vues n'existe. Je pense que l'ingénierie dirigée par les modèles peut contribuer à améliorer les choses : il n'est bien sûr pas question de trouver un formalisme unique qui s'imposerait à tous pour modéliser l'artefact à toute étape du cycle de vie d'un EIAH mais plutôt d'utiliser les possibilités de transformation de modèles pour unifier différents formalismes (graphes conceptuels, XML, UML, ...) dans un processus homogène.
- Les EIAH exigent la plupart du temps un processus de conception itératif pour lequel théories (d'apprentissage, de transmission des connaissances...) et expérimentations vont de pair. Aussi, l'une des difficultés en conception d'EIAH est la mouvance des spécifications. Un des objectifs de l'ingénierie dirigée par les modèles est de capitaliser sur les modèles et non plus sur le code-source, ce permet de capitaliser une partie du développement grâce aux modèles. Les concepteurs vont ainsi pouvoir faire évoluer leurs modèles (à un niveau abstrait), l'idée étant de "mécaniser" la production d'un modèle donné (par exemple du code pour simuler un aspect de l'EIAH) à partir d'un certain nombre de modèles d'entrée (des modèles statiques et dynamiques) et d'autre part d'un savoir-faire réifié sous forme d'une transformation de modèles (INRIA 2004).

Bien qu'ambitieux, le projet de recherche présenté dans ce paragraphe me semble cohérent avec l'orientation donnée à la thèse de P. Laforcade. Ce projet me paraît important pour la communauté EIAH dans une perspective de facilitation des échanges entre acteurs participant à la conception des EIAH. L'ingénierie dirigée par les modèles est un champ d'investigation très actif en Génie Logiciel qui ne doit pas être confondu avec les travaux proposés ici : notre propos vise principalement la recherche de solutions pour favoriser l'interdisciplinarité nécessaire à la conception des EIAH (ce qui passe par l'explicitation des théories cognitives et des décisions de conception), l'ingénierie dirigée par les modèles fournissant les bases théoriques et l'outillage rendant possible un tel projet.

Il est bien clair que le projet qui vient d'être ébauché ne peut pas être celui d'un unique chercheur, ni même d'une équipe : l'école d'été 2005 du RTP39 (STIC CNRS) qui a eu lieu très récemment a été l'occasion de présenter les principes et les enjeux de l'ingénierie des modèles en EIAH et d'échanger avec les responsables des équipes françaises en EIAH ; je tire du travail de préparation, et de l'école elle-même, quelques pistes permettant de poursuivre et concrétiser le projet global qui vient d'être décrit. Ces actions sont présentées dans le prochain paragraphe.

3.3.2.2 Propositions d'actions de recherche à court terme

Afin d'engager la communauté EIAH dans une démarche d'ingénierie dirigée par les modèles, je propose de contribuer avec l'équipe IDEE à deux types de coopérations entre laboratoires :

- La mise en place d'un groupe de travail réunissant les différentes sensibilités de la communauté française en EIAH pour étudier, conforter et capitaliser quelques modèles - à choisir) de la communauté SHS (Sciences Humaines et Sociales) : des modèles tels le modèle d'interactions entre pairs proposé dans (Baker 2002) ou le modèle de l'apprenant basé sur cKç (Balacheff 1995), (Webber 2003), (Balacheff 2004) proposé par le laboratoire Leibniz me semblent pouvoir donner lieu à des interactions très riches entre chercheurs que ceux-ci se préoccupent de la diffusion des connaissances autour de ces modèles, de leur formalisation et de leurs usages, de modèles d'implémentation, ... Idéalement, un tel groupe de travail devrait comporter un nombre limité de participants s'engageant à étudier ces objets scientifiques très ciblés en agrégeant les compétences diverses réunies. La production d'un rapport d'activités rendant compte des résultats de ce projet transdisciplinaire témoignerait de l'engagement de la communauté mais aussi de la spécificité des recherches en EIAH.
- La mise en place d'un groupe de travail réunissant les chercheurs (informaticiens) qui abordent l'ingénierie des EIAH par le prisme des modèles. L'objectif est de comparer les langages de description de modèles utilisés (grâce à des techniques de Benchmarking), d'identifier des mécanismes de transformation entre langages et de proposer à la communauté des outils de transformation implémentant ces mécanismes. Un tel projet me paraît devoir être mené à l'échelle internationale pour que le groupe de travail agrège suffisamment de force et de diversité ; compte tenu des développements à réaliser pour promouvoir et évaluer des langages et outils de transformation de modèles, ce projet me paraît également nécessiter l'engagement d'industriels. La communauté EIAH française a pas mal d'atouts pour piloter un tel projet, le laboratoire Trigone (Renau, Caron et al. 2005) et le LIUM (El-Kechai et Choquet 2005), (Choquet 2005) ayant déjà orienté une partie de leurs activités vers ce type de problématique. Par ailleurs, les résultats des travaux de l'AS « Fondements théoriques et méthodologiques de la conception des EIAH », de l'AS « Conception d'une plate-forme pour la recherche en EIAH »²³, l'intérêt grandissant de la communauté pour les langages de scénarisation pédagogique (Pernin, Choquet et al. 2005), ... me semblent justifier l'opportunité à constituer rapidement un tel groupe de travail. A l'échelle internationale, ce rôle moteur nous permettra d'avoir une influence sur les types de modèles et langages à privilégier pour décrire et mettre en œuvre des apprentissages basés sur la construction des savoirs par l'apprenant.

Pour ma part, je souhaite approfondir l'ingénierie dirigée par les modèles sur des points très précis au travers du sujet de thèse que j'expose ci-après. Ce sujet a été soumis en avril 2005 au laboratoire LIUPPA dans le cadre de la procédure d'attribution des bourses du MENRT par l'Université de Pau et des Pays de l'Adour. Le sujet porte sur la mise en œuvre de modèles de régulation des situations d'apprentissage humain. L'objet scientifique qui sert de point de départ à ce travail est constitué d'un corpus documentaire limité à partir duquel un concepteur pédagogue souhaite construire une situation d'apprentissage : le corpus documentaire sur lequel s'appuie le travail proposé est géocentré (prise en compte de l'espace) et situé dans le temps : l'exemple sur lequel nous avons commencé à travailler (Nodenot, Gaio et al. 2005a) concerne la retranscription d'un ensemble de témoignages sur un accident qui sert de point de départ pour concevoir une situation-problème comme Smash (voir documents et cartes en annexe B). L'approche dirigée par les modèles est envisagée à deux niveaux :

1. Pour analyser l'information contenue dans les documents initiaux et guider l'activité de conception en enrichissant ces documents par des techniques de transformation : nos

²³ Il est clair que les travaux menés sur les ontologies éducatives dans le cadre de l'AS « Web sémantique et Elearning » entrent également dans le champ des travaux qui sont proposés.

travaux récents (Gaio et Szmurlo 2002), (Bilhaut, Charnois et al. 2003), (Lesbegueries 2005), (Etcheverry, Marquesuzaà et al. 2005) ont permis de mettre en place un prototype d'environnement d'outils pour analyser ces documents géocentrés et ajouter ainsi de la sémantique aux documents (Loustau 2005) : connaissance sur le positionnement des objets dans l'espace, inférences permettant de reconstruire un parcours, ... Les modèles de documents issus de ce travail de traitement semi-automatique du langage sont donc des modèles computationnels que l'on peut interroger / exploiter et je fais l'hypothèse qu'ils permettent de guider l'activité de conception d'une situation d'apprentissage (la conception devenant, en partie, un processus interactif guidé par les contenus),

2. Pour intégrer dans les modèles de documents résultant du point 1 des informations rendant possible une régulation pédagogique par un système informatique lorsque ces documents sont exploités au service d'une situation d'apprentissage donnée. « On vit dans une boucle de régulation où l'on fait des propositions et où l'on recueille des informations qui régulent ces propositions. On propose et on régule. C'est ainsi que cela fonctionne » (Meirieu 1994). Il s'agit donc d'utiliser les documents enrichis (Garlatti et Iksal 2004), (Yoo, Vité et al. 2004) au service des attendus des concepteurs et de les doter des connaissances rendant possible une différenciation des usages.

Dans ce cadre particulier, le doctorant doit proposer : a/ une rhétorique de régulation adaptée aux enseignants-concepteurs (voir la proposition de (Perrenoud 1997) basée sur la métaphore du voyageur présentée au paragraphe 1.2.1.1), b/ des modèles (représentation des connaissances relatives à la régulation, ...) s'appuyant sur les modèles de contenus documentaires et les attendus du concepteur, c/ des outils (outils d'aide à la conception pédagogique, composant logiciel offrant un service de régulation intégrable à des plateformes de formation à distance) permettant au concepteur d'évaluer et d'améliorer les éléments de différenciation intégrés dans les modèles (outil de simulation).

3.4 Conclusion

Dans ce chapitre, j'ai présenté les différents travaux que j'ai pu mener ou encadrer depuis une dizaine d'années dans le domaine de l'ingénierie d'environnements à base de situations-problèmes. Mes propositions ont été organisées en deux volets : j'ai présenté tout d'abord mes travaux dans le domaine des modèles et méta-modèles pour décrire des situations-problèmes ; j'ai ensuite abordé le problème de l'usage des modèles sous le double éclairage des méthodes de conception et des mécanismes de transformation entre modèles (transformations CPM / IMS-LD par exemple).

Ces travaux sur les situations-problèmes ont structuré l'activité de recherche de plusieurs étudiants de troisième cycle et ont donné lieu à deux doctorats en informatique de l'UPPA. Ils ont aussi permis de mettre en place / développer de nombreux échanges internationaux (Europe et Amérique Latine) ainsi que plusieurs coopérations industrielles (Projets Canal+ et Télévisions numériques) et académiques (RTP39, Université de San Sebastian, ...).

Un enjeu important des prochaines années se situe dans la formalisation des interactions entre modèles représentant différents points de vue d'un même EIAH. Ce travail peut être mené pour passer des vues abstraites à des vues concrètes opérationnalisables par un ordinateur; nous croyons que ce travail est aussi utile à un niveau d'abstraction donné afin de confronter, renforcer les points de vue multiples de la communauté SHS et de la communauté informatique sur des EIAH à concevoir.

Ce travail de recherche en ingénierie des EIAH doit forcément s'appuyer sur des bases sémantiques solides à savoir sur la formalisation des méta-modèles dont sont issus les modèles (Seidwitz 2003). Les perspectives présentées m'ont permis d'étayer ces idées par des propositions concrètes visant, d'une part à rendre possible la comparaison et la transformation de modèles, d'autre part à outiller les concepteurs et les chercheurs.

Le dernier chapitre de ce manuscrit me permettra de dresser un bilan de mon activité d'enseignant-chercheur à l'UPPA.

Bilan de mon activité d'enseignant-chercheur à l'UPPA

Les travaux présentés dans ce manuscrit m'ont permis de décrire et mettre en perspective les différentes actions de recherche et développement conduites depuis que je suis enseignant-chercheur à l'IUT de Bayonne. L'exposé de ces actions structuré selon les principes exposés dans le document Platon-1 (Tchounikine, Baker et al. 2004) a volontairement masqué la chronologie des travaux et le contexte dans lequel elles ont été menées. Dans cette conclusion, je propose donc un bilan plus personnel sur mon activité en abordant mes motivations de recherche mais aussi mon parcours professionnel à l'Université de Pau et des Pays de l'Adour.

La recherche sur les EIAH présente pour l'informaticien un caractère exaltant lorsque les applicatifs conçus/développés parviennent à *articuler les capacités de l'apprenant et celles de la machine au service de savoirs à transmettre*. Cet équilibre n'est jamais facile à trouver, la recherche en EIAH s'appuyant sur des *théories qui progressent par expérimentation des applicatifs développés*. Prescription / codification informatique des interactions souhaitées et processus d'expérimentation / analyse des usages sont des *activités largement pluridisciplinaires* qui suscitent des échanges toujours enrichissants avec les enseignants et les usagers des EIAH développés. Ce sont aussi des activités *exigeantes* car les enseignants, chercheurs dans le domaine des SHS et informaticiens abordent les problèmes de manière souvent différente, ce qui impose à tous pas mal d'humilité et d'ouverture aux autres. Par ailleurs, les nombreuses évolutions requises pour rendre acceptable un EIAH imposent de longues heures de développement qui souvent, faute de moyens humains, sont à la charge du chercheur en informatique. Ces éléments (en italique) mis bout à bout font de l'EIAH un objet de recherche original et passionnant.

Mon approche de cet objet de recherche consiste à examiner les avancées de l'Informatique et à approfondir celles pouvant être utiles et mises au service de la recherche en EIAH :

- les techniques de modélisation et métamodélisation issues du Génie Logiciel ont ainsi été mises au service du dialogue entre acteurs pour proposer des solutions au problème de l'interdisciplinarité en ingénierie des EIAH.
- les techniques et outils issus de l'ingénierie documentaire ont été proposés dans la partie « Perspectives » de ce manuscrit afin de pouvoir exploiter des corpus documentaires comme base de la conception d'un EIAH.

Les travaux conduits sur la base des propositions formulées dans ce mémoire s'appuient sur les compétences d'une équipe qui s'est développée depuis 2000 autour d'un projet scientifique et humain : c'est à cette époque que Franck Barbier, nouveau professeur recruté par l'UPPA a structuré une nouvelle équipe d'accueil du Ministère de la Recherche : le

LIUPPA (Laboratoire d'Informatique de l'Université de Pau et des Pays de l'Adour, EA 3000). Après examen d'un dossier scientifique soumis à cette époque à Didier Arquès (responsable de la MSU), Franck Barbier a soutenu notre initiative visant à constituer un groupe de recherche autonome au sein de l'Université de Pau et des Pays de l'Adour. L'expertise de notre laboratoire par le Ministère en février 2003 a ensuite confirmé les orientations prises en 2001.

De 2001 à fin 2003, l'équipe SIA (Systèmes d'Information et Apprentissage) dont j'avais la responsabilité était constituée de 4 permanents et un doctorant financé par une allocation de recherche du Ministère. La phase de démarrage a mobilisé toute mon énergie puisqu'il a fallu à la fois définir (et conforter) les orientations scientifiques choisies, trouver des financements à moyen terme (contrats et projets) pour le fonctionnement de l'équipe et participer à la vie du laboratoire.

En septembre 2003, l'équipe s'est fortement renforcée puisque trois Maîtres de Conférences en Informatique et un professeur des Universités ont rejoint l'équipe. L'année 2004 a donc été une année de transition particulièrement importante puisqu'il a fallu tout à la fois :

- Poursuivre et conforter les travaux engagés autour de la thèse de Pierre Laforcade, en l'aidant²⁴ dans la publication de ses résultats scientifiques et la relecture de son mémoire. Fin décembre 2004 soit 3 ans après sa première inscription en thèse, ce dernier a soutenu son mémoire et il est vient d'être recruté Maître de Conférences en Informatique à l'IUT de Laval.
- Elargir le spectre des problématiques scientifiques de l'équipe SIA pour intégrer au mieux les compétences des quatre enseignants-chercheurs qui nous ont rejoints.

Courant 2004, j'ai donc proposé au conseil du laboratoire de créer l'équipe IDEE pour « Interaction, Document Electronique, Education » (cf <http://idee.iutbayonne.univ-pau.fr>) en remplacement de l'équipe SIA. Je suis l'actuel responsable de cette équipe IDEE qui est constituée de 8 enseignants-chercheurs permanents (7 MCF et 1 PR en informatique), 3 doctorants (2 allocations MESR, 1 allocation Communauté d'Agglomération de Pau). L'équipe que nous avons constituée est *multi-sites* (comme l'UPPA) puisque cinq membres permanents de l'équipe sont en poste sur Bayonne et trois sont en poste à Pau. Cette équipe est aussi *multi-compétences* puisque nos savoir-faire couvrent l'Ingénierie des EIAH (compétence SIA), le traitement sémantique de documents électroniques (compétences de Mauro Gaio, nouveau PR) et l'Interaction Homme-Machine (compétences des 3 MCF nous ayant rejoints). Les nombreuses réunions organisées tout au long de l'année 2004 nous ont permis de définir un cadre commun pour notre recherche. De manière générale, nos travaux visent à faciliter une approche intégrée des situations d'interaction et des systèmes documentaires, les situations privilégiées étant toujours les situations d'apprentissage humain. Notre approche est largement *multidisciplinaire* puisque nous travaillons sur des corpus documentaires géocentrés issus du patrimoine culturel local dans le cadre d'une collaboration continue au sein de l'UPPA avec le laboratoire IKER (UMR n°5478). L'équipe IDEE participe donc activement à l'effort de structuration engagé au sein du laboratoire LIUPPA pour *agréger les compétences* des chercheurs autour de pôles thématiques scientifiquement cohérents mais qui ont une masse *critique suffisante* pour asseoir la visibilité du laboratoire.

Je pense avoir saisi, avec mes collègues de l'équipe IDEE, le coup de pouce du laboratoire qui nous a fait confiance en nous permettant de créer et structurer une équipe de recherche. Depuis 2001, ces travaux ont donné lieu à des résultats scientifiques reconnus par la communauté nationale en EIAH, de nombreuses publications internationales (IEEE, ACM, IFIP). J'ai contribué à la formation par la recherche de deux thésards et je participe à la diffusion des connaissances liées à mon domaine de recherche (cf mon implication dans le comité scientifique et pédagogique de l'école d'été sur les EIAH qui a eu lieu à Autrans en

²⁴ Le Directeur de cette thèse était Franck Barbier, les deux co-encadrants étant moi-même et C. Sallaberry.

juillet 2005 - voir <http://www-clips.imag.fr/ecoleEIAH2005/>). Ces activités scientifiques s'ajoutent à ma responsabilité de l'équipe IDEE qui est aujourd'hui l'équipe la plus importante en nombre de chercheurs au laboratoire LIUPPA.

Mes principales charges administratives ne concernent pas seulement la recherche puisqu'elles sont dues aussi à mon statut de responsable du service Formation Continue de l'IUT de Bayonne. Cette fonction m'a amené, entre 1999 et 2003, à porter pour l'IUT un projet de Licence Professionnelle dans le domaine de la communication multimédia, à assurer la responsabilité de cette formation durant 3 ans en m'appuyant sur mes collègues de l'IUT, du BTS audiovisuel du lycée Cassin et d'une quinzaine de professionnels du multimédia issus des PME locales. En tant que chef de service, je suis aussi membre du conseil de Direction de l'IUT qui assiste le Directeur dans ses diverses prises de décision. Dans ce cadre, je participe notamment aux réunions hebdomadaires qui nous conduisent à discuter de tous les problèmes liés à la vie de l'IUT (orientations budgétaires, problèmes de personnel, problèmes de fonctionnement quotidiens).

Ces charges d'administration (responsabilité de formation continue, responsabilité de l'équipe IDEE) et d'organisation des enseignements (licence professionnelle) qui s'ajoutent à mes activités de recherche se révèlent parfois pesantes mais me permettent en même temps de bien connaître le fonctionnement opérationnel de la structure universitaire dans laquelle j'ai été nommé enseignant-chercheur.

Derrière le statut d'enseignant-chercheur se cachent différentes façons d'aborder un métier passionnant, pour peu que l'on accepte sa diversité et que l'on arrive, sur des périodes données, à se fixer des priorités. Le LIUPPA m'a donné ma chance en 2000 de structurer une équipe de recherche, mes collègues d'IDEE et de l'IUT m'ont permis, durant quelques mois, de me consacrer à la rédaction de ce manuscrit en prenant en charge certaines des tâches que je ne pouvais pas assumer correctement. Je les remercie à tous de leur confiance, leur disponibilité et leur relecture de ce mémoire.

Curriculum Vitae

Thierry Nodenot

Né le 7 mai 1963 à Vic Fezensac (Gers)
Nationalité française, 2 enfants

Etat civil

Adresse personnelle

Thierry, Cathy, Bastien et Pablo
86, rue de la Semie
40130 Capbreton – France

Adresse professionnelle

Laboratoire LIUPPA – IUT de Bayonne
3 avenue Jean Darrigrand
64115 Bayonne – France
Tel : 05 59 57 43 53 Fax : 05 59 57 43 09

Email : Thierry.Nodenot@iutbayonne.univ-pau.fr

Ancienneté professionnelle hors Université

- 1981-1984 Ecole Normale d'Instituteurs Tarbes (Hautes-Pyrénées)
- 1985-1986 Instituteur à mi-temps Bernac-Dessus (Hautes-Pyrénées)

Expérience professionnelle

Ancienneté générale à l'Université

- Maître de Conférences en Informatique à l'IUT de Bayonne (Université de Pau et des Pays de l'Adour) depuis le 1^{er} septembre 1993. Classe normale 6^{ème} échelon en janvier 2005.

Titres universitaires

Docteur en Informatique de l'Université Paul Sabatier de Toulouse
Mention très honorable et félicitations du jury le 20 novembre 1992

Titulaire d'un Deug en Sciences de l'Education obtenu à Pau en juin 1984

Responsabilités scientifiques

Depuis novembre 2003 : Responsable de l'équipe IDEE « Interaction, Document Electronique, Education » (voir <http://idee.iutbayonne.univ-pau.fr>) au sein du laboratoire LIUPPA (Equipe d'Accueil n°3000). 8 enseignants-chercheurs en Informatique de l'UPPA, 3 doctorants

De 2001 à 2003 : Responsable de l'équipe SIA « Systèmes d'Informations et Apprentissage » au sein du laboratoire LIUPPA (Equipe d'Accueil n°3000). 4 enseignants-chercheurs en informatique de l'UPPA ; 1 doctorant

Bilan des publications scientifiques

Au total :

- 4 articles dans des revues internationales avec comité de lecture / ouvrages collectifs internationaux
- 3 articles en revue nationale avec comité de lecture / ouvrage collectif national
- 22 articles dans des conférences internationales avec actes et comité de lecture
- 2 articles dans des conférences nationales avec actes et comité de lecture
- 2 workshops internationaux avec actes et comité de lecture

16 publications scientifiques depuis 2001 (année de création du LIUPPA)

Encadrement de thèses à l'Université de Pau et des Pays de l'Adour

- P. Laforcade, « Méta-modélisation UML pour la conception et la mise en œuvre de situations problèmes coopératives ». Thèse préparée à l'Université de Pau et des Pays de l'Adour sous la Direction de Franck Barbier. Thèse soutenue le 15 décembre 2004 à l'Université de Pau et des pays de l'Adour. Situation actuelle : Maître de Conférences en Informatique à l'IUT de Laval (Université du Maine).
- C. Marquesuzaa, « OMAGE : Outils et Méthodes pour la spécification des connaissances au sein d'un Atelier de Génie Educatif ». Thèse soutenue le 12 février 1998 à l'Université de Pau et des Pays de l'Adour. Situation actuelle : Maître de Conférences en Informatique à l'IUT de Bayonne (Université de Pau et des Pays de l'Adour).

Encadrement scientifique de doctorants en partenariat avec une université étrangère

Dans le cadre des relations internationales de l'IUT de Bayonne, nous accueillons chaque année depuis 1998 deux collègues informaticiens de l'Instituto Varona (Université de la Havane, Cuba). Ces enseignants sont accueillis durant 3 mois par an à l'IUT pour conduire leurs travaux de thèse ; ils effectuent également quelques enseignements dans le domaine du Multimédia. Depuis 2001, je m'occupe de l'accompagnement de leurs travaux de recherche qui portent sur la définition et l'évaluation de patrons de conception pour la production d'applications éducatives. Leur soutenance est prévue fin 2005.

Encadrement de stagiaires 3^{ème} cycle (DEA, DESS, Master)

Animation scientifique

- Encadrement scientifique avec P. Lopisteguy du stage de Master 2 (spécialité Sciences cognitives appliquées) de Audrey Rivière à l'IUT de Bayonne (Projet piloté par le GRETA Pays Basque dans le domaine des Télévisions numériques pour France Télévision l'Union Européenne de Radiodiffusion, ...), avril à septembre 2004
- Encadrement scientifique avec P. Lopisteguy du stage de DESS de Emmanuel Vieux à l'IUT de Bayonne (Projet Campus+, Société Canal+), avril à septembre 2003
- Encadrement scientifique avec C. Marquesuzaa du stage de Ivan Paz à l'IUT de Bayonne entre avril 2003 et août 2003 sur le thème : Evaluation et déploiement de plates-formes de Formation à Distance à destination des Instituts de Formation des Enseignants de la République de Cuba ».
- Encadrement en 2000 de 3 élèves Ingénieurs de l'ESTIA (Technopole Izarbel) chargés de poursuivre les développements initiés par I. Ochoa (cf ci-dessous) dans le cadre du projet européen OLE (Online Localisation in Europe). Ces développements ont consisté à réaliser et évaluer l'ergonomie d'une interface multilingue construite au dessus d'une application éducative.
- Encadrement scientifique de I. Ochoa (Université de Gran Canaria – Espagne) dans le cadre de son Master d'Informatique entre octobre 1998 et avril 1999. Cet étudiant a travaillé particulièrement sur le développement d'outils logiciels dans le cadre du projet européen OLE (Online Localisation in Europe)
- Encadrement du stage de DEA de C. Marquesuzaa (sous la responsabilité de G. Gouardères) au laboratoire API (UPS Toulouse), novembre 92 à juillet 93
- Encadrement du stage de DEA de D. Duboscq (sous la responsabilité de G. Gouardères) au laboratoire API (UPS Toulouse), novembre 91 à juillet 92
- Encadrement du stage de DEA de J.B. Onagoity (sous la responsabilité de G. Gouardères) au laboratoire API (UPS Toulouse), novembre 90 à juillet 91
- Encadrement du stage de DEA de P. Esquer (sous la responsabilité de G. Gouardères) au laboratoire API (UPS Toulouse), novembre 90 à juillet 91

Comités et conseils scientifiques

- Membre du comité scientifique et pédagogique chargé de l'organisation de l'Ecole d'été du RTP39 à Autrans en juillet 2005 sur le thème « Modèles, Architectures Logicielles et Normes pour le Développement et l'Intégration des EIAH » (<http://www-clips.imag.fr/ecoleEIAH2005/>)
- Membre du comité de programme de la conférence EIAH2005 patronnée par l'ATIEF – Montpellier (25 au 27 mai 2005). Président du comité de Programme : Pierre Tchounikine du Laboratoire d'Informatique de l'Université du Maine (<http://www.lirmm.fr/eiah2005/index2.htm>)
- Membre du comité d'Organisation de la XXII^{ème} conférence nationale INFORSID'2004 – Biarritz (<http://inforsid2004.univ-pau.fr>). Plus particulièrement chargé de l'organisation du Forum Jeunes Chercheurs
- Membre du comité d'organisation du 3^{ème} Basque International Workshop on Information Technology : (<http://csdl.computer.org/comp/proceedings/biwit/1997/8049/00/8049toc.htm>) BIWIT'97 – Biarritz, France
- Membre du comité d'Organisation de la conférence CALISCE'96 : Third International Conference on Computer Aided Learning and Instruction in Science and Engineering (<http://www.informatik.uni-trier.de/~lev/db/conf/calisce/calisce1996.html>) – San Sebastian (Espagne)
- Membre des comités de relecture pour les conférences et journaux suivants :
 - Revue STICEF (<http://www.sticef.org>) éditée par Hermès (jusqu'en 2002) puis l'ATIEF
 - Conférence internationale ICALT 2005 (<http://www.ask.iti.gr/icalt/2005/>) éditée par IEEE
 - Conférence internationale UML'2004 (<http://ctp.di.fct.unl.pt/UML2004/>) éditée par Springer
 - Conférence internationale CIDE.7 (<http://infodoc.unicaen.fr/cide/cide.7/pages/accueil.php>)
 - Conférence internationale ICCI 2003 (<http://www.computer.org/cspress/CATALOG/pr01986.htm>) éditée par IEEE
 - Journal SoSyM (<http://www.sosym.org>) édité par Springer Verlag
 - Conférence internationale BIWIT'97 (<http://csdl.computer.org/comp/proceedings/biwit/1997/8049/00/8049toc.htm>)
 - Conférence internationale CALISCE'96 éditée par Springer (<http://www.informatik.uni-trier.de/~lev/db/conf/calisce/calisce1996.html>)
- Membre de la commission de Spécialistes en Informatique de l'Université de Pau et des Pays de l'Adour de 1994 à 2000

Animation
scientifique
(suite)

Groupes de travail

- Membre du groupe de travail en charge de l'Action Spécifique (AS) « Plates-formes pour la recherche en EIAH » mené par le Département STIC du CNRS dans le domaine « Apprentissage, Education, Formation ». Porteur de cette AS :
A. Derycke (Laboratoire Trigone), Université de Lille1
- Membre du groupe EIAH (Environnements Informatiques pour l'Apprentissage Humain) du GDR I3 (un des 5 GDR du département STIC du CNRS). Ce groupe fait partie du Pôle « Interaction et Coopération (Axe 4) » du GDR I3. Responsable du groupe EIAH :
J.F. Nicaud (Laboratoire IMAG-Leibniz-MeTAH), Grenoble
- Membre du groupe de travail ayant proposé une équipe Projet « Hippocampe » du RTP 33 (Documents et contenu : création, indexation, navigation) dans le cadre du département STIC CNRS. Le domaine de travail porte sur « Interactions Humaines et Cognition », le projet s'intitulant : « (re)construction et scénarisation de documents géographiques ». Porteurs de l'équipe Projet :
P. Buléon (Laboratoire ESO – UMR CNRS 6590), Université de Caen
M. Gaio (Laboratoire LIUPPA – EA 3000), Université de Pau
J. Madelaine (Laboratoire GREYC – UMR CNRS 6072), Université de Caen
- Membre du groupe inter-laboratoires travaillant sur le thème « Ingénierie des scénarios pédagogiques et des traces d'usage ». A ce jour, ce groupe s'appuie sur les laboratoires CLIPS (Grenoble), LIUM (Le Mans, Laval), LIUPPA (Pau), SYSCOM (Chambéry) et TRIGONE (Lille). Porteur du projet :
C. Choquet (Laboratoire LIUM), Université du Maine

Jurys de thèse

Animation
scientifique
(suite)

- Examineur de la thèse de Pierre Laforcade intitulée « Méta-modélisation UML pour la conception et la mise en œuvre de situations problèmes coopératives ». Soutenue le 15 décembre 2004 à l'Université de Pau et des pays de l'Adour (présidente : Danièle Héryn, rapporteurs : Alain Derycke et Chantal Soulé-Dupuy)
- Examineur de la thèse de Christophe Marquesuzaa intitulée « Outils et Méthodes pour la spécification des connaissances dans un Atelier de Génie Educatif » présentée le 12 février 1998 à l'Université de Pau et des Pays de l'Adour (président du jury : B. Causse, rapporteurs : Violaine Prince et Michel Lamure).

Coopérations inter-laboratoires au sein de l'Université

- Une coopération avec le laboratoire IKER (UMR 5478) et notre équipe est mise en place depuis septembre 2004. Nous avons répondu à deux appels d'offres en commun dans le cadre d'un projet interdisciplinaire visant la (Re)Construction et la scénarisation orientées usages de ressources numériques liées au patrimoine.

Coopérations régionales et nationales

Conduite et
Participation à
des
programmes
de recherche

- Action Spécifique « Conception d'une Plate-forme pour la recherche en EIAH » du RTP 39 (STIC CNRS). J'ai participé aux discussions ayant précédé le montage de cette action spécifique, aux réunions de travail de la dizaine de laboratoires français impliqués dans cette action spécifique ainsi qu'à la rédaction des rapports d'activités. Cette action a permis de spécifier les services d'une plate-forme de mutualisation des travaux menés par les chercheurs en EIAH, d'autre part d'amorcer la mutualisation en proposant des contenus qui recensent les bonnes pratiques en ingénierie des EIAH.
- Sous l'impulsion de Mauro Gaio, nouveau professeur ayant intégré l'équipe LIUPPA-Idee en septembre 2003, nous participons à l'expérimentation, la valorisation de la plate-forme de TAL créée par le laboratoire GREYC (Caen) : nous sommes partenaires du projet Geosem2 dans le cadre du programme « Société de l'Information » du STIC-CNRS en partenariat avec le GREYC (UMR CNRS 6072), ERSS (UMR CNRS 5610), ESO (UMR CNRS 6590), IKER (UMR CNRS 5478), LATTICE (UMR CNRS 8094), IRIT (UMR CNRS 5505), l'Ecole Polytechnique Fédérale de Lausanne. Notre contribution consiste en la (re)construction et la scénarisation de documents géographiques pour des applications finalisées de type Touristique et ELearning.

Actions de valorisation de la recherche et transfert technologique

Actuellement :

- Projet européen « Cactus » dont le maître d'œuvre est le GRETA Pays Basque (responsable : C. Latreille). Plus particulièrement chargé (avec P. Lopisteguy) de mettre en place et de réaliser l'évaluation ergonomique et pédagogique des modules d'enseignement en ligne développés dans le cadre de ce projet.
- Projet Aquitaine Euskadi sur le thème « Développer une compétence universitaire trans-frontalière sur le processus e-learning ». Responsable du projet : P. Lopisteguy. Projet financé sur 3 ans en partenariat notamment avec la Faculté d'Informatique de San Sebastian (Espagne). Budget annuel : 58000 euros.
- Groupe de travail chargé du développement de la plate-forme de Formation à distance « OpenUSS » implémentée à base de composants logiciels respectant les spécifications J2EE. Cette plate-forme a obtenu le prix spécial du jury lors de la dernière conférence ObjectWeb 2005 à Lyon.
- Projet financé par le Conseil Général des Pyrénées Atlantiques sur le thème : « Interactions avec des systèmes documentaires hétérogènes : initiation au patrimoine culturel local » Responsables du projet : T. Nodenot et JM Bruel. Projet sur 2 ans en partenariat avec l'équipe IKER (UMR CNRS 6072) et deux autres équipes du laboratoire LIUPPA (équipes AOC et ALM). Budget annuel : 30000 euros par an.

Conduite et
Participation à
des
programmes
de recherche
(suite)

Auparavant :

- Membre de l'équipe projet chargée en 2003 du développement de modules pédagogiques centrés sur les technologies numériques en Audiovisuel. Plus particulièrement impliqué avec P. Lopistéguy dans la mise en œuvre d'une méthode d'évaluation des modules exploités sur la plate-forme de FOAD Academynet. Projet de deux ans piloté par le GRETA Pays Basque. Ces modules de formation ont été développés pour la plupart par une équipe d'enseignants, de scénaristes et développeurs recrutés par le GRETA Pays Basque. Ces modules qui représentent 180 heures d'enseignement du BTS audiovisuel ont été produits et mis en exploitation pour les grandes chaînes de télévision française (Canal+, France Télévision, TF1) ainsi que l'Union Européenne de Radiodiffusion (UER). Budget : 233000 euros sur 2 ans
- Projet de 2 ans (2002-2004) financé par le Conseil Régional d'Aquitaine sur le thème : Modélisation et mise en œuvre de composants d'apprentissage coopératif pour la plate-forme J2EE (Java2 Enterprise Edition) ». Responsable du projet : T. Nodenot. Budget annuel : 27200 euros par an sur 2 ans
- Projet d'un an (2002) financé par le Conseil Général des Pyrénées Atlantiques sur le thème : « Modèles, Composants et Architectures logicielles pour applications coopératives multimédia ». Responsable du projet : F. Barbier. Budget annuel : 17300 euros
- Projet européen OLE 28922-IC-2-97-1-GB-ERASMUS-CDI-1 On-Line Localisation in Europe (projet de 2 ans terminé en décembre 2000)
- Projet européen KAMP ET1013 : Knowledge Assurance in Multimedia Publishing (terminé en décembre 1997)

Responsabilités assurées actuellement

Animation
pédagogique

- *Responsable du Service Formation Continue de l'IUT de Bayonne depuis septembre 1999.* Chiffre d'Affaires du Service en 2003 : 145000 € ; Personnel associé à ce service : une secrétaire administrative (F. Michelin), un responsable de service (T. Nodenot). Le service assure, pour l'IUT de Bayonne et en relation avec les différents départements d'Enseignement de l'IUT, la responsabilité administrative/pédagogique de plusieurs actions de Formation (Diplôme d'Université de niveau 2 intitulé « Technologies de l'Information et de la Communication » - 1350h de formation dispensées, Licence Professionnelle « Systèmes Informatiques et Logiciels » option « Communication Multimédia » - 900 h de formation dispensées). Dans ce cadre, nous travaillons en partenariat avec le Conseil régional d'Aquitaine et les différentes ANPEs locales. Nous mettons également en place des formations courtes, des audits sur demande des entreprises du secteur Bayonne-Anglet-Biarritz. Depuis 2003, nous assurons, en partenariat avec le service Forco de l'UPPA, le suivi des dossiers de VAE (Validation des Acquis et de l'Expérience).
- *Chargé de la Communication et des relations institutionnelles entre l'IUT de Bayonne et ses partenaires (BTS Audiovisuel du Lycée Cassin, Entreprises du secteur Multimédia, dans le cadre de la Licence Professionnelle « Systèmes Informatiques et Logiciels » Option « Communication Multimédia »*
- *Chargé de la coordination des propositions de parcours différenciés visant à offrir différentes spécialisations aux étudiants de la Licence Professionnelle « Systèmes Informatiques et Logiciels » Option « Communication Multimédia »*

Responsabilités assurées précédemment

- *Responsable de la Licence Professionnelle « Systèmes Informatiques et Logiciels » Option « Communication Multimédia » de septembre 2000 à juin 2003 (trois premières promotions d'étudiants)*
 - *Porteur pour l'IUT de Bayonne du projet de Licence Professionnelle « Systèmes Informatiques et Logiciels » Option « Communication Multimédia » durant l'année universitaire 1999-2000*
 - *Responsable des Etudes au Département GEA de l'IUT de Bayonne de 1995 à 1998*
-

Revue avec comité de lecture et ouvrages collectifs internationaux

Towards an Open Source Framework for Collaborative Problem-Based Learning Environments

T. Nodenot, M. Gaio, C. Sallaberry, P. Lopistéguy
"e-learning and education (elearn)" Journal, n°1, ISSN 1860-7470, février 2005

Designing Multilingual, Multidisciplinary Distance Education

T. Hartley, T. Nodenot, L. Pemberton, R. Salkie
New Directions in Professional Higher Education, Open University Press ed., ISBN 0-335-20614-X, pp. 105-111, 2000.

The European Intensive Course Experience

M. Daniels, L. Pemberton, R. Griffiths, T. Nodenot, E. Bengoetxea
New Directions in Professional Higher Education, Open University Press, ISBN 0-335-20614-X, pp. 149-154, 2000.

Instructional Design and Formal Specification : a step through Courseware Engineering

C. Marquesuzaa, T. Nodenot
Journal of Educational Multimedia and Hypermedia (JEMH), ISSN 1055-8896, AACE Publications, 1996

Revue avec comité de lecture et ouvrages collectifs nationaux

Un langage de modélisation pédagogique basé sur UML

P. Laforcade, T. Nodenot, Sallaberry
Revue Sciences et Techniques de l'Information et de la Communication pour l'Éducation et la Formation (STICEF), ISSN : 1764-7223, Numéro spécial : Conceptions et usages des plates-formes de formation, volume 12, 2005

Contributions de l'Action Spécifique " Conception d'une Plateforme pour la recherche en EIAH " à l'ingénierie des Environnements Informatiques pour l'Apprentissage Humain

Collectif AS Plateforme pour la recherche en EIAH
Revue Sciences et Techniques de l'Information et de la Communication pour l'Éducation et la Formation (STICEF), ISSN : 1764-7223, Numéro spécial : Conceptions et usages des plates-formes de formation, volume 12, 2005

L'AGDI : des solutions pour rationaliser la production des logiciels éducatifs

M.N. Bessagnet, T. Nodenot
Revue Génie Éducatif, Editions EC2, n°3, mars 1992

Conférences Internationales avec actes et comité de lecture

Creating rich collaborative learning scenarios: a model driven approach for contextualizing software components and documents

Thierry Nodenot, Mauro Gaio, Marie-Noëlle Bessagnet, Christian Sallaberry
8th IFIP World Conference on Computers in Education (WCCE 2005), Cape Town (SA)

Model driven development of cooperative Problem-Based Learning Situations.

Implementing tools for teachers and learners from pedagogical models

Christian Sallaberry, Thierry Nodenot, Pierre Laforcade, Christophe Marquesuzaa
38th annual Hawaii International Conference on System Sciences, IEEE Computer Society, ISBN : 0-7695-2268-8, January 3-6, 2005, Hawaii (USA)

Model based Engineering of Learning Situations for Adaptive Web Based Educational Systems

Thierry Nodenot, Christophe Marquesuzaa, Pierre Laforcade, Christian Sallaberry
ACM Thirteenth International World Wide Web Conference (IW3C2 Conference), ISBN : 1-58113-912-8, 17-22 may 2004, New York (USA)

Détail des
publications
scientifiques

Détail des
publications
scientifiques
(suite)

Profiling Co-operative Problem-Based Learning Situations

Pierre Laforcade, Franck Barbier, Christian Sallaberry, [Thierry Nodenot](#)
Second IEEE International Conference on Cognitive Informatics (ICCI 2003), ISBN 0-7695-1986-5, 2003. London (UK)

A UML Profile incorporating separate viewpoints when modeling Co-operative Learning Situations

[Thierry Nodenot](#), Pierre Laforcade, Christian Sallaberry, Christophe Marquesuzaa
IEEE International Conference on Information Technology : Research and Education on IT for enabling the next generation technology development, ISBN 0-7803-7878-4, August 10-13, 2003, Newark (USA)

Knowledge Modelling of Co-operative Learning Situations: Towards a UML profile

[Thierry Nodenot](#), Pierre Laforcade, Christophe Marquesuzaa, Christian Sallaberry
11th International Conference on Artificial Intelligence in Education, International AI-ED Society, ISBN 1-58603-356-5, IOS Press, volume n°97, 2003 Sydney (Australia)

Spécifications d'un environnement Web supportant des activités coopératives d'apprentissage

[Thierry Nodenot](#), Christophe Marquesuzaa, Pierre Laforcade, Marie-Noëlle Bessagnet, Christian Sallaberry

Technologies de l'Information et de la Communication dans les Enseignements d'ingénieurs et dans l'industrie, pp. 253-262, 13 au 15 novembre 2002, Lyon (France)

Information systems and educational engineering: Bridging two concepts through meta modelling

Marie-Noëlle Bessagnet, Christophe Marquesuzaa, [Thierry Nodenot](#), Christian Sallaberry, Pierre Laforcade

IFIP WCC (World Computer Congress) 2002, ISBN 1-4020-7219-8, Springer Editeur, 25 au 30 août 2002, Montréal (Canada)

An attempt to design an Information System supporting Collaborative Problem-Based Learning Situations

Christian Sallaberry, [Thierry Nodenot](#), Christophe Marquesuzaa, Marie-Noëlle Bessagnet, Pierre Laforcade

6th World Multiconference on Systemics, Cybernetics and Informatics, July 2002, Orlando (USA)

Information modelling within a Net-Learning Environment

Christian Sallaberry, [Thierry Nodenot](#), Christophe Marquesuzaa, Marie-Noëlle Bessagnet, Pierre Laforcade

12th Conference On Information Modelling and Knowledge Bases, ISBN 1-58603-318-2, IOS Press, 27 - 30 May 2002, Krippen (Germany)

A case for an adult educational technology

[Thierry Nodenot](#)

7th IFIP World Conference on Computers in Education, Kluwer Academic Publishers (Springer), pp. 115-124, ISBN 1-4020-7132-9, volume 94, July/August 27-3, 2001, Copenhagen (Denmark)

The role of CASE tools in the process of specifying educational software

Christophe Marquesuzaa, Jacques Meyranx, [Thierry Nodenot](#)

5th Annual Conference on Methodologies, Specialist Group on Information System Methodologies, British Computer Society, ISBN 1 85233 068 6, Août 1997, Preston (UK)

Architecture and Principles of a Specification Environment Dedicated to Pedagogues

Christophe Marquesuzaa, Jacques Meyranx, [Thierry Nodenot](#)

Actes de la conférence SITE 97, 8th International Conference Society for Information technology and Teacher Education , Avril 1997, Orlando (USA)

Towards pedagogically-sound Learning Environments : the specification process.

Christophe Marquesuzaa, Jacques Meyranx, [Thierry Nodenot](#)

Third International Conference on Computer Aided Learning and Instruction in Science and Engineering, ISBN 3-540-61491-5, Springer, juillet 96, San Sebastian (Espagne)

*Specifying Educational Software : Goals and Process*Christophe Marquesuzaa, [Thierry Nodenot](#)

Conference Proceedings ED-MEDIA96 : World AACE Conference on Educational Multimedia and Hypermedia , Juin 1996, Boston (USA)

What if the pedagogues specified Educational Software ?[Thierry Nodenot](#)

Conference Proceedings WCCE'95 : sixth IFIP World Conference on Computers in Education, Elsevier (North Holland Publications), ISBN 0-412-62670-5, Juillet 1995, Birmingham (UK)

*Intégration des données et des processus dans un Atelier de Génie Educatif*Frédéric Blanc, [Thierry Nodenot](#)

Actes des 6èmes journées internationales : "Le Génie logiciel et ses applications", novembre 1993, Paris (France)

MAGE : a methodological framework for the cooperative development of educational software[Thierry Nodenot](#), Guy Gouardères

Conference Proceedings, IFIP Congress'92 for Research and Practice, pp.59-63, ISBN 0-444-89750-X, Elsevier (North Holland Publications), septembre 1992, Madrid (Espagne)

A methodology for cooperative developments of educational software[Thierry Nodenot](#)

Conference proceedings ICCAL'92 : Fourth International Conference on Computers and Learning, p. 529-541, ISBN 3-540-55578-1, Springer, juin 1992, Wolfville (Canada)

*Adapting Authoring/Learning environments to the users needs*Thierry Mengelle, [Thierry Nodenot](#), Guy GouardèresConference proceedings ICTE'92 : The 9th International Conference on Technology and Education, 1992 : Education "Sans Frontières", volume 3, p. 1235-1237, mars 1992, Paris*Dialogue dans un Atelier de Génie Educatif*[Thierry Nodenot](#)

Actes des 4èmes journées internationales : "Le Génie Logiciel et ses applications", p. 223-236, ISBN 2-906 899-68-2, Décembre 1991, Toulouse (France)

*L'AGDI : un environnement pour la production d'applications EAO*Marie-Noëlle Bessagnet, [Thierry Nodenot](#), Guy Gouardères

Actes des 3èmes journées internationales : "Le Génie logiciel et ses applications", p. 441-461, ISBN 2-906 899-48-8, Décembre 1990, Toulouse (France)

*A new approach : Courseware Engineering*Marie-Noëlle Bessagnet, [Thierry Nodenot](#), Guy Gouardères, Jean-Jacques Rigal

Conference Proceedings WCCE'90 : Fifth World Conference on Computers in Education, p. 339-344, ISBN 0-444-88750-4, Elsevier (North Holland Publications), Juillet 1990, Sydney (Australie)

*Workshops internationaux et Conférences nationales avec actes et comité de lecture**Modelling scenarios of cooperation promoting Human Learning: Added value of the CPM Profile*Christian Sallaberry, [Thierry Nodenot](#), Pierre Laforcade, Christophe Marquesuzaa

7th International Conference on Tutoring Systems - Modeling Human Teaching Tactics and Strategies WorkShop, August 31 2004, Maceio Alagoas (Brasil)

*Making is better than reading and making together is best: Web-based activities for promoting interaction between young Europeans*Lyn Pemberton, [Thierry Nodenot](#)

Troisième conférence des Usages des Nouvelles Technologies dans l'Enseignement des Langues Etrangères, 23-25 mars 2000, Université de Technologie de Compiègne (France)

Détail des
publications
scientifiques
(suite)

Autres communications

Responsable et orateur de deux cours de l'école d'été 2005 (Autrans) du RTP39 portant sur « l'Ingénierie dirigée par les modèles en EIAH » et « les questions vives de recherche » Troisième école thématique EIAH du CNRS, Autrans, juillet 2005 (cf http://www-rtp39.imag.fr/Autrans_2005/ProgrammeEcole/ProgrammeAutrans2005.htm)

*Autres formes
de diffusion des
savoirs*

Participant à la table ronde organisée durant la conférence EIAH2005 (Montpellier) sur le thème : « Scénarisation pédagogique : que faire de la proposition IMS Learning Design? », Animateur : Jean-Philippe PERNIN (INRP-LYON & Laboratoire CLIPS-IMAG-GRENOBLE),

Participants : C. Choquet (Laboratoire LIUM, Laval), C. Depover (Université de Mons, Belgique), C. Martel (Université de Savoie, Chambéry), T. Nodenot (laboratoire LIUPPA, Bayonne), JP. Pernin (Laboratoire Clips-IMAG, Grenoble),
1h17mn, 26 mai 2005 (cf <http://www.canalc2.tv/video.asp?idvideo=3648>)

Présentation des activités de transfert de technologie de l'équipe IDEE dans le cadre de la manifestation Pau Cité Multimédia (CITE-M 2001 et 2003). Thème des exposés :
« formation à distance pour la (re)qualification des personnes »

Références bibliographiques

- Abdullah, M. S., A. Evans, I. Benest et C. Kimble (2004). Developing a UML Profile for Modelling Knowledge-Based Systems. Model Driven Architecture®: Foundations and Applications (MDAFA 2004), Linköping University, Sweden.
- Abdullah, M. S., R. Paige, I. Benest, C. Kimble et A. Evans (2005). Designing Knowledge-Based Systems Using UML Profile (Metamodel Based Extension). Second International Conference on Intelligent Computing and Information Systems (ICICIS 2005), Le Caire (Egypte).
- ACTIS (2001). Smash Scenario. <http://www.projectboxes.co.uk/catalogue/scenarios.html#smash>
- ADL (2004). Advanced Distributed Learning (ADL), Sharable Content Object Reference Model (SCORM). 2nd Edition. www.adlnet.org.
- Albion, P. (2000). Evaluating the implementation of Problem-based Learning in Interactive Multimedia. ASCILITE 2000 : The Australasian Society for Computers in Learning in Tertiary Education, Coffs Harbour (Australia).
- Aleven, V. et K. Koedinger (2000). Limitations of Student Control: Do Students Know When They Need Help? Proceedings of the 5th International Conference on Intelligent Tutoring Systems, p.292-303, Biarritz (France).
- Aleven, V. et K. Koedinger (2001). Investigations into Help Seeking and Learning with a Cognitive Tutor. AIED-2001 Workshop on Help Provision and Help Seeking in Interactive Learning Environments, San Antonio (USA).
- Allert, H. (2004). "Coherent Social Systems for Learning: an Approach for Contextualized and Community-Centred Metadata." Journal of Interactive Media in Education (2).
- Anderson, J., A. Corbett, K. Koedinger et R. Pelletier (1995). "Cognitive Tutors: Lessons Learned." Journal of the Learning Sciences volume 4(2).
- Asencio, J. I., Y. A. Dimitriadis, M. Heredia, A. Martínez, F. J. Álvarez et M. T. Blasco (2004). Collaborative Learning Patterns: Assisting the Development of Component-Based CACL Applications. 12th Euromicro Conference on Parallel, Distributed and Network-Based Processing (PDP'04), Coruña (Espagne).
- ASPF (2005). "Contribution de l'AS "Conception d'une plateforme pour la recherche en EIAH" à l'ingénierie des EIAH." Numéro spécial "Conceptions et usages des plates-formes de formation" de la revue STICEF (Sciences et Technologies de l'Information et de la Communication pour l'Education et la Formation), volume 12.

-
- Atkinson, C. et T. Kühne (2003). "Model-Driven Development: a metamodeling foundation." IEEE Software.
- Bachelard, G. (1938). La formation de l'esprit scientifique, Editions Vrin, Paris.
- Baker, M. et C. Lund (1997). "Promoting reflective interactions in a computer-supported collaborative learning environment." Journal of Computer Aided Learning volume 13.
- Baker, M. (2000). "The roles of models in Artificial Intelligence and Education research: a prospective view." International Journal of Artificial Intelligence in Education vol.11: p. 122-143.
- Baker, M., E. De Vries, C. Lund et M. Quignard (2001). "Interactions épistémiques médiatisées par ordinateur pour la co-élaboration des notions scientifiques." Revue Sciences et techniques Educatives volume 8.
- Balacheff, N. (1991). Contribution de la didactique et de l'épistémologie aux recherches en EIAO. 13^{ème} journée francophones sur la formation intelligemment assistée par ordinateur (FIAO), Genève.
- Balacheff, N. (1995). "Connaissance, Concept, Conception." Didactique et Technologies Cognitives en Mathématiques: pp. 219-244.
- Balacheff, N. (2001). A propos de la recherche sur les environnements informatiques pour l'apprentissage humain. Note rédigée en préparation du séminaire Cognitique sur les "Technologies de l'apprentissage".
<http://www-leibniz.imag.fr/DIDACTIQUE/Balacheff/TextesDivers/CognitiqueEIAH.html>
- Balacheff, N. (2004). Conceptions de l'apprenant et design des EIAH. Conférence invitée de la Seconde Ecole thématique sur les EIAH "Processus cognitifs et technologies pour l'apprentissage". Autrans (France).
- Baloian, N., H. Breuer, K. Hoeksema, U. Hoppe et M. Milrad (2004). Implementing the Challenge Based Learning in Classroom Scenarios. Proceedings of the symposium on Advanced Technologies in Education, Argostoli (Greece).
- Bandura, A. (1986). Social Foundations of Thought and Action. A Social Cognitive Theory, Englewood Cliffs: Prentice-Hall.
- Barbier, F., H. Briand, B. Dano et S. Rideau (1998). "The executability of object-oriented finite state machines." Journal of Object-Oriented Programming volume 11(4).
- Barbier, F. (2003). Built-in Test Vademecum part V. Rapport du Projet européen Component+ (Version 1.0 du 11 Mars 03). 5th Framework Project IST-1999-20162. <http://www.component-plus.org/sidor/reports.htm>
- Barbier, F. et O. Constant (2005). Executable Component-Based Scenarios. proceedings of the 4th International Workshop on Scenarios and State Machines: Models, Algorithms and Tools. 27th International Conference on Software Engineering, Saint-Louis (USA).
- Barbier, F. et P. Laforcade (2005). Concepts, Techniques and Tools for an Educational Modeling Language. International Conference on Cognitive Informatics (ICCI2005), University of California, Irvine (USA).
- Barden, R., S. Stepney et D. Cooper (1994). Z in practice, BCS practitioner Series, Prentice-Hall.
- Baron, G. L., E. Bruillard et C. Dansac (1999). Représentations, modèles et modélisations. Implications sur les stratégies éducatives et sur les processus d'apprentissage : synthèse bibliographique, Rapport final rédigé par le Representation Consortium, Educational Multimedia Task Force.
- Barros, B. et M. F. Verdejo (2000). "Analysing student interaction processes in order to improve collaboration. The DEGREE approach." International Journal of Artificial Intelligence and Education.

- Baulac, Y. (1990). Un micromonde de géométrie, Cabri-géomètre, Thèse de l'université Joseph Fourier, Grenoble I.
- Bensalem, S., V. Ganesh, Y. Lakhnech, C. Munoz et S. Owre (2000). An overview of SAL. Fifth NASA Langley Formal Methods Workshop.
- Bertrand, Y. (1993). Théories contemporaines de l'Education, Collection "Synthèse".
- Berzins, V. et Luqi (1990). Software Engineering with Abstractions, Addison-Wesley Publishing Company.
- Bessagnet, M.-N., C. Marquesuzaà, T. Nodenot, C. Sallaberry et P. Laforcade (2002). Information systems and educational engineering: bridging two concepts through meta modelling. 17e Congrès international des technologies de l'information, IFIP WCC (World Computer Congress), Montréal (Canada), International Federation of Information Processing (IFIP).
- Bessagnet, M.-N. (2005). Ontology for Problem-Based Learning Situation. SW-WL'05 - Second International Workshop on Semantic Web for Web-based Learning. In conjunction with CAISE'05 (The 17th Conference on Advanced Information Systems Engineering), Porto (Portugal).
- Betbeder, M.-L. et P. Tchounikine (2004). Modélisation et perception de l'activité dans l'environnement Symba. Actes de la conférence RFIA'04, Toulouse (France).
- Bézivin, J. et O. Gerbé (2001). Towards a precise Definition of the OMG/MDA framework. IEEE Conference Automated Software Engineering (ASE'2001), San Diego (USA).
- Bézivin, J., M. Blay, M. Bouzeghoub, J. Estublier et J.-M. Favre (2005). Rapport de synthèse de l'Action Spécifique CNRS sur l'Ingénierie Dirigée par les Modèles, Action Spécifique MDA du CNRS.
- Bilhaut, F., T. Charnois, P. Enjalbert et Y. Mathet (2003). Geographic reference analysis for geographic document querying. Proceedings of New Trends in Intelligent Information Processing and Web Mining (IIPWM), Zakopane, Poland.
- Blanco, S. et A. De Zambotti (2005). Analyse des besoins pour un environnement-auteur de modélisation de situations d'apprentissage et extension d'un prototype basé sur l'utilisation d'un Atelier de Génie Logiciel existant, Rapport de projet du Master 1 (Technologies de l'Internet), Université de Pau et des Pays de l'Adour.
- Bourguin, G. (2000). Un support informatique à l'activité coopérative fondé sur la Théorie de l'Activité : le projet DARE. Lille, Doctorat de l'Université des Sciences et Technologie de Lille.
- Bouzeghoub, M. (2005). Espace Technologique "Base de Données" et l'IDM, Action Spécifique MDA du CNRS.
- Brassard, C. et A. Daele (2003). Un outil réflexif pour concevoir un scénario pédagogique intégrant les TIC. Conférence "Environnements Informatiques pour l'Apprentissage Humain (EIAH2003), Strasbourg, Hermès.
- Brethes, T., F. Hisquin et P. Pezziardi (2000). Serveur d'Applications.
- Breton, E. et J. Bézivin (2001). Towards an understanding of model executability. Proceedings of the second International Conference on Formal Ontology in Information Systems (FOIS'2001), Ogunquit (USA).
- Breton, E. (2002). Contribution à la représentation de processus par des techniques de méta-modélisation, Thèse de doctorat de l'Université de Nantes.
- Brousseau, G. (1998). Théories des situations didactiques, La pensée sauvage Editions.
- Brown, G. et M. Atkins (1988). Effective Teaching in Higher Education, London, Routledge.
- Bruillard, E. (1997). Les machines à enseigner. Editions Hermès. Paris (France)

-
- Brusilovsky, P. (1999). "Adaptive and Intelligent technologies for Web-based Education." Special Issue on Intelligent Systems and Teleteaching, C. Rollinger and C. Peylo (eds.).
- Brusilovsky, P. et H. Nijhawan (2002). A framework for Adaptive E-Learning Based on Distributed Re-usable Learning Activities. AACE World Conference on E-Learning, Elearn 2002, Montreal (Canada).
- Brusilovsky, P. (2003). A Distributed Architecture for Adaptive and Intelligent Learning Management Systems. Workshop "Towards Intelligent Learning Management Systems" , 11th International Conference on Artificial Intelligence in Education, Sydney (Australia).
- Brusilovsky, P. (2004). KnowledgeTree: a distributed architecture for adaptive e-learning. Proceedings of the 13th International World Wide Web Conference on Alternate Track Papers, New York (USA).
- Cariou, E., R. Marvie, L. Seinturier et L. Duchien (2004). OCL for the Specification of Model Transformation Contracts. Workshop OCL and Model Driven Engineering of the Seventh International Conference on UML Modeling Languages and Applications (UML 2004), Lisbonne (Portugal).
- Caron, P.-A. (2004). Une proposition de composants MDA simples pour le eLearning. Communication présentée dans le cadre de l'Action Spécifique "Plateforme pour la recherche en EIAH". Paris.
- Caron, P.-A., A. Derycke et X. Le Pallec (2005). Bricolage and Model Driven Approach to design distant course. Conférence E-Learn 2005, Association for the Advancement of Computing in Education (AACE), Vancouver, Canada.
- Caron, P.-A., X. Le Pallec et A. Derycke (2005). Bricolage and Model Driven Approach to design distant course. E-learn 2005, Vancouver (Canada).
- Chakroun, M. (2003). Conception et mise en place d'un module pédagogique pour portails communautaires PostNuke. Projet de fin d'études mené à l'Université de Genève pour l'obtention du diplôme national d'ingénieur en sciences appliquées et en technologie, Université de Tunis.
- Choquet, C. (2005). Design patterns for recording and analysing usage of learning systems. Kaleidoscope: Jointly Executed Integrating Research Projects (JEIRP) - work package 32.
- Clancey, W. J. (1997). Situated Cognition: on human knowledge and computer representations, Cambridge University Press.
- Cook, S. (2002). Model-driven approaches to software development. Présentation au colloque OCM 2002, Nantes.
- Counterman, C., G. Golden, R. Gollub, M. Norton, C. Severance et L. Speelmon (2004). Sakai Java Framework. <http://cvs.sakaiproject.org/release/1.0.0/docs/Sakai-Java-Framework-1.0.0.pdf>
- Crozat, S. et P. Trigano (2002). "Structuration et scénarisation de documents pédagogiques numériques dans une logique de massification." Revue Sciences et Techniques Educatives volume 9, n°3, Editions Hermès.
- David, B. (2001). "IHM pour les collecticiels." Les télé-applications (13).
- De Haan, G., G. C. Van der Veer et J. C. Van Vliet (1991). "Formal modeling techniques in Human Computer Interaction." Acta Psychologica n°78.
- De los Angeles, M. et D. Suthers (2003). Automated coaching of collaboration based on Workspace analysis: evaluation and implications for future learning environments. Proceedings of the 36th Hawaiï International Conference on System Sciences, Hawaiï (USA).
- De Rosnay, J. (1975). Le Macroscopie, Editions Seuil, Paris.
- DeCesare, S., M. Lycett et D. Patel (2002). Business modelling with UML. Fourth International Conference on Enterprise Information Systems, Ciudad real (Espagne).
-

- Delium, C. (2003). OSCAR, un environnement de communication médiatisée structurée par les actes de langage. Actes de la conférence EIAH'2003, Strasbourg (France).
- Denos, N. (1997). Modélisation de la pertinence en recherche d'information : modèle conceptuel, formalisation et application. Doctorat de l'Université Joseph Fourier, Université Grenoble 1.
- Desmoulins, C. et M. Grandbastien (2002). "Des ontologies pour la conception de manuels de formation à partir de documents techniques." *Revue Sciences et Techniques Educatives* (9).
- Després, C. (2001). Modélisation et conception d'un environnement de suivi pédagogique synchrone d'activités d'apprentissage à distance. Doctorat de l'Université du Maine (LIUM). Le Mans (France).
- Dessus, P. et J. J. Maurice (1998). "Les décisions de l'enseignant à l'aune de valeurs rationnelles." *Spirale*(n°21): pages 47-56.
- Dessus, P. (2002). "Description et prescription dans les méthodes de recherche en éducation." *Les sciences de l'Education, des recherches, une discipline*, Paris : L'Harmattan.
- Develay, M., Ed. (1993). *De l'apprentissage à l'enseignement*, Collection Pédagogies, ESF Edition.
- Dewanto, B. L. (2002). ObjectWeb and OpenUSS: a win-win situation ? Conference objectweb 2002, INRIA Rocquencourt.
- Dewanto, B. L. (2004). Enterprise Java Open Source Architecture.
<http://prdownloads.sourceforge.net/ejosa/ejosa-revo-doc.pdf>
- Dewanto, B. L. (2005). "Model Driven Architecture (MDA): Integration and Model Reuse for Open Source eLearning Platforms." *Eleed Journal (E-learning and Education)* volume 1, ISSN: 1860-7470.
- Dick, W. et L. Carey (1990). *The systematic Design of Instruction*, Glenview: Scott and Foresman, 2nd Edition.
- DiGiano, C., L. Yarnall, C. Patton, J. Roschelle, D. Tatar et M. Manley (2002). Collaborative Design Patterns: conceptual tools for planning the wireless classroom. International Workshop on Wireless and Mobile Technologies in Education (WMTE2002), Växjö (Suède).
- Dillenbourg, P. (1999). Comprendre les implications pédagogiques de choix techniques, Laboratoire TECFA, Université de Genève.
- Dillenbourg, P., D. Schneider et P. Synteta (2002). Virtual Learning Environments. Proceedings of the Third congress on Information and Communication Technologies in Education, Rhodes (Grèce).
- Downes, S. (2002). "Problems and issues in Online Learning." *LearningPlace : Online Learning Magazine*.
- Dubourdieu, S. (2005). Conception d'un tableau blanc générique avec l'environnement Laszlo. Rapport du stage de fin d'études de l'IUP d'Anglet.
- Edelson, D., K. O'Neill, L. Gomez et L. D'Amico (1995). A design for effective support of inquiry and collaboration. Proceedings of CSCL'95, Mahwah (USA).
- El-Kechaï, H. et C. Choquet (2005). Approche pragmatique de conception d'un EIAH : réingénierie pédagogique dirigée par les modèles. Conférence EIAH 2005 (Environnements Informatiques pour l'Apprentissage Humain), Montpellier (France).
- Ellis, C. et J. Wainer (1994). A conceptual model of Groupware. ACM Conference on Computer Supported Cooperative Work (CSCW'94), Chapel Hill, North Carolina (USA).
- Engeström, Y., R. Miettinen et R. L. Punamäki (1998). "Perspectives on activity theory." Cambridge University Press.

-
- Eriksson, H. et M. Penker (2000). Business Modelling with UML : Business Patterns at Work, Robert Ipsen Editeur.
- Etcheverry, P. (2002). SAPIC : Modélisation et Spécification de la Coordination Interne aux Processus d'Activités. Mémoire de Doctorat en Informatique de l'Université de Pau et des Pays de l'Adour (Laboratoire LIUPPA). Bayonne (France).
- Etcheverry, P., C. Marquesuzaà et J. Lesbegueries (2005). Revitalisation de documents territorialisés : Principes, outils et premiers résultats. Atelier MetSI, Congrès INFORSID, 24 - 27 avril 2005, Grenoble (France).
- Faerber, R. (2004). "Caractérisation des situations d'apprentissage en groupe." Revue STICEF : Sciences et Technologies de l'Information et de la Communication pour l'Education et la Formation, Volume 11.
- FFFOD (2002). Le B.A.BA de la FOAD. Fédération Française de la Formation Ouverte et à Distance. www.fffod.org
- Flavell, J. (1976). Metacognition Aspects of Problem Solving, The Nature of Intelligence. Resnick Editors. Hilldale, NJ: Lawrence Erlbaum.
- Fridmann Noy, N., R. W. Ferguson et M. Musen (2000). The knowledge model of Protégé-2000: combining interoperability and flexibility. Proceedings of 12th International Conference on Knowledge Engineering and Knowledge Management (EKAW 2000), Juan-Les-Pins.
- Fritsch, G. (2005). La séquence d'apprentissage. Revue Rézo. http://sites.estvideo.net/gfritsch/doc/rezo-cfa-306_bis.htm
- Gagné, R., L. Briggs et W. Wager (1988). Principles of Instructional Design, New York: Holt, Rinehart and Winston.
- Gaio, M. et M. Szmurlo (2002). "L'apport de la bimodalité dans l'élaboration d'interfaces adaptées à la recherche d'informations géographiques." chapitre de la Revue d'Interaction Homme-Machine RHIM (31), Europa Editions.
- Garanderie, A. (1980). Les profils pédagogiques, Editions Centurion (Paris).
- Garlatti, S. et S. Iksal (2004). A Flexible Composition Engine for Adaptive WebSites. Third International Conference AH 2004 (Adaptive Hypermedia and Adaptive Web-Based Systems), Eindhoven (Netherlands).
- George, S. (2001). Apprentissage collectif à distance. SPLASH : un environnement informatique support d'une pédagogie de projet, Thèse de doctorat de l'Université du Maine.
- Giaratano, J. et G. Riley, Eds. (1994). Expert Systems: Principes et Programming. PWS publishing Company, 2^{nde} édition. Boston (USA).
- Gluch, D. P., S. Comella-Dorda, J. Hudak, G. Lewis, J. Walker, C. B. Weinstock et D. Zubrow (2002). Model-Based Verification: an Engineering Practice, Carengie Mellon Software Engineering Institute, Pittsburgh (USA).
- Goyhenex, N. et M. Capliez (2004). Apprentissage à base de situation problèmes et composant tableau blanc, rapport de TER, Maîtrise Informatique de l'Université de Pau.
- Grabe, M. et C. Grabe (1996). Integrating Technology for Meaningful Learning, Houghton Mifflin Company, Boston.
- Graesser, A., K. VanLehn, C. Rosé, P. Jordan et D. Harter (2001). "Intelligent Tutoring Systems with Conversational Dialogue." American Association for Artificial Intelligence (AI Magazine).
- Graesser, A., K. Moreno, J. Marineau, A. Adcock, A. Olney et N. Person (2003). AutoTutor Improves Learning of Computerr Literacy: Is it the Dialog or the Talking ahead ? 11th International Conference on Artificial Intelligence in Education (AIED 2003): Shaping the future of Learning through Intelligent Technologies, Sydney (Australia), IOS Press (volume n°97).
-

- Grison, B. (2002). La cognition située : courants et filiation. XXXVII^{ème} congrès SELF, Aix en Provence.
- Grob, H. L., F. Bensberg et B. L. Dewanto (2004). Developing, Deploying, Using and Evaluating an Open Source Learning Management System.
http://www.campussource.de/aktuelles/docs/Bensberg_Paper_ITI04_1.pdf
- Grosso, W. (2005). Laszlo: An Open Source Framework for Rich Internet Applications. java.net: the source for Java technology Collaboration.
<http://today.java.net/pub/a/today/2005/03/22/laszlo.html>
- Guareis de Farias, C. R., L. Ferreira Pires et M. Van Sinderen (2000). A conceptual model for the development of CSCW systems. Fifth International Conference of the Design of Cooperative Systems (COOP 2000), Sophia Antipolis (France).
- Guéraud, V., J.-M. Adam, J.-P. Pernin, G. Calvary et J.-P. David (2004). "L'exploitation d'Objets Pédagogiques Interactifs à distance : le projet FORMID." Revue STICEF : Sciences et Technologies de l'Information et de la Communication pour l'Education et la Formation.
- Guzdial, M., J. L. Kolodner, C. Hmelo, H. Narayanan, D. Carlon, N. Rappin, R. Huebscher, J. Turns et W. Newstetter (1996). "Computer Support for learning through complex problem solving." Communication of the ACM.
- Guzdial, M., C. Hmelo, R. Huebscher, K. Nagel, W. Newstetter, S. Puntembakar, S. Shabo, J. Turns et J. L. Kolodner (1997). Integrating and guiding Collaboration: lessons learned in computer-supported Collaborative Learning Research at Georgia Tech. Proceedings of CSCL'97, Toronto (USA).
- Hacker, T. (1985). "On some fundamentals of action regulation." Discovery Strategies in the psychology of action London: academic press: p. 63-84.
- Harel, D., H. Kugler et M. Rami (2002). The Play-in / Play-out Approach and Tool: Specifying and Executing Behavioral Requirements. The Israeli Workshop on Programming Languages & Development Environments (PLE'02).
- Harrer, A., L. Bollen et U. Hoppe (2004). Processing and Transforming Collaborative Learning Protocols for Learner's reflections and Tutor's Evaluation. Workshop on Artificial Intelligence in Computer Supported Collaborative Learning (in conjunction with ECAI2004), Valencia (Espagne).
- Hazelrigg, G. A. (2003). Thoughts on model validation for engineering design. Proceedings of DETC'03. ASME 2003 Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Chicago (USA).
- Houssaye, J. (1988). "Le triangle pédagogique." Berne, Peter Lang.
- Howard, D. L. (1994). "Pertinence as reflected in personal constructs." Journal of the American Society for Information Science volume 45(3): 172-185.
- Hummel, H., J. Manderveld, C. Tattersall et R. Koper (2004). "Educational modelling language and learning design: new opportunities for instructional reusability and personalised learning." International Journal of Learning Technology Vol. 1, No.1 pp. 111 - 126.
- IMS (2001). Instructional management systems project website. <http://imsproject.org/>
- IMS (2003a). IMS Learning Design Information Model, IMS Global Learning Consortium.
- IMS (2003b). IMS Learning Design Best Practice and Implementation Guide, Technical report.
- INRIA (2004). Model Transformation at INRIA. <http://modelware.inria.fr/>
- Jacobi, D. (1991). Dresser une carte de concepts : effets linguistiques et sémiotiques. Troisième rencontre européenne de didactiques de la biologie, Rome (Italie).
- Jermann, P. et D. Schneider (1997). Semi-Structured interface in collaborative problem-solving. 1st Swiss Workshop on distributed and parallel Systems.

-
- Jermann, P. et P. Dillenbourg (1999). An analysis of learner arguments in a collective learning environment. Proceedings of the third CSCL Conference, Stanford (USA).
- Jermann, P. (2002). "Task and Interaction Regulation in controlling a traffic Simulation." Computer Support for collaborative learning: Foundations for a CSCL community, Hillsdale, NJ: Erlbaum.
- Johnson, L. (2001). "Pedagogical Agent research at CARTE." American Association for Artificial Intelligence (AI Magazine) Winter 2001.
- Johnson, L. et P. Rizzo (2004). Politeness in tutoring dialogues: "Run the factory, That's what I'd do." 7th International Conference on Intelligent Tutoring Systems (ITS2004), Maceio (Brasil), Springer Verlag (LNCS 3220).
- Jonassen, D., K. Beissner et M. A. Yacci (1993). "Structural knowledge: techniques for representing, conveying and acquiring knowledge." Hillsdale (N.J): Erlbaum.
- Khammaci, T. et B. El Ayeb (1991). Software Development Assistance by Reasoning and Planning. 4^{ème} conférence internationale "Le Génie Logiciel et ses applications", Toulouse.
- Kleppe, A., S. Warmer et W. Bast (2003). MDA explained. The Model Driven Architecture: Practice and Promise, Addison-Wesley.
- Kolodner, J. L., C. Hmelo et H. Narayanan (1996). Problem-based learning meets case based reasoning. International conference on the Learning Sciences, Northwestern University.
- Koper, R. (2001). Modeling Units of Study from a Pedagogical Perspective : the pedagogical meta-model behind EML, Educational Expertise Technology Centre, Open University of the Netherlands.
- Koper, R. (2002). Educational Modelling Language : adding instructional design to existing specifications. Open University of the Netherlands.
- Koper, R., K. Giesbers, P. Van Rosmalen, C. Tattersall, P. B. Sloep, J. Van Bruggen, C. Tattersall, H. Vogten et F. Brouns, Eds. (2003). A Design Model for Lifelong Learning Networks. Interactive Learning Environments (Preprint).
- Koper, R. et B. Olivier (2004). "Representing the learning design of Units of Learning." Educational Technology and Society volume 7, n°3.
- Koschmann, T. D., P. S. Feltovich, A. M. Myers et H. S. Barrows (1990). Designing communication protocols for a computer-mediated tutorial laboratory for problem based learning. Proceedings of the 14th Annual Symposium on Computer Applications in Medical Care, Los Alamitos (USA), IEEE Computer Society Press.
- Koschmann, T. D., P. S. Feltovich, A. M. Myers et H. S. Barrows (1992). Implications of CSCL for Problem-Based Learning. Proceedings of the Spring'92 ACM Conference on Computer Supported Collaborative Learning, Mahwah (USA).
- Kozman, R. B. (1991). "Learning with media." Review of Educational Research, 61(2), 179-212.
- Laborde, C. et B. Capponi (1991). "Cabri géomètre constituant d'un milieu pour l'apprentissage de la notion de figure géométrique." Recherche en didactique des mathématiques, Volume 14/1.2, 1994, n° 40/41 Grenoble : La pensée sauvage, p. 167-209.
- Laforcade, P., F. Barbier, C. Sallaberry et T. Nodenot (2003). Profiling Co-operative Problem-Based Learning Situations. IEEE International Conference on Cognitive Informatics (ICCI 2003), London (UK).
- Laforcade, P. (2004). Méta-modélisation UML pour la mise en oeuvre de situations problèmes coopératives. LIUPPA. Doctorat en informatique de l'Université de Pau et des Pays de l'Adour.
- Laforcade, P. et F. Barbier (2004). "UML Modeling for Cooperative Problem-Based Learning Situations: Towards Educational Components." Instructional Technologies: Cognitive Aspects of Online Programs, P. Darbyshire Editor, IDEA-Group Publishing.
-

- Laforcade, P. (2005a). Towards an UML-based Educational Language. The 5th IEEE International Conference on Advanced Learning Technologies (ICALT), Kaohsiung (Taiwan).
- Laforcade, P. (2005b). Approche par transformation de modèles pour la conception d'EIAH : illustration entre les langages CPM et IMS-LD. Conférence EIAH 2005 (Environnements Informatiques pour l'Apprentissage Humain), Montpellier (France).
- Laforcade, P., T. Nodenot et C. Sallaberry (2005). "Résultats et perspectives d'un travail exploratoire mené en modélisation et méta-modélisation UML pour la conception de situations d'apprentissage." Numéro spécial "Conceptions et usages des plates-formes de formation" de la revue STICEF (Sciences et Technologies de l'Information et de la Communication pour l'Education et la Formation), volume 12.
- Lahaye, P. (2004). Les systèmes de gestion de contenus. Mémoire de fin d'études d'Ingénieur Informatique du CNAM.
- Laks, B., Ed. (1996). Langage et cognition - L'approche connexionniste, Editions Hermès.
- Lebrun, M. (2002). Des technologies pour enseigner et apprendre. Louvain (Belgique), Editions De Boeck.
- Leroux, P. (1996). Intégration du contrôle d'objets réels dans un hypermédia. Un exemple d'implantation dans le système ROBOTTEACH. Troisième colloque Hypermédias et Apprentissages, Bruillard, E., Baldner, J.-M., Baron, G.-L. éditeurs, Châtenay-Malabry.
- Leroux, P. (2002). Machines partenaires des apprenants et des enseignants - Étude dans le cadre d'environnements supports de projets pédagogiques. Mémoire d'Habilitation à Diriger les Recherches de l'Université du Mans.
- Lesbegueries, J. (2005). Des services web destinés à l'indexation de la recherche spatio-temporelle dans un corpus territorialisé. Rapport interne du Laboratoire LIUPPA.
- Lin, X., C. Hmelo, C. K. Kinzer et T. J. Secules (1999). "Designing technology to support reflection." Educational Technology Research & Development Journal.
- Linard, M. (2003). "Autoformation, éthique et technologies : enjeux et paradoxes de l'autonomie." Editions Hermès, pp 241-263.
- LOM (2001). LOM working draft v6.1. <http://ltsc.ieee.org/doc/#LOM%20Draft>
- Lopistéguy, P. (2004). Développer une compétence universitaire trans-frontalière sur le processus e-learning. Année 1 : 'Echange et mise en commun de compétences relatives au processus e-learning', Rapport de synthèse de l'équipe IDEE remis au Conseil Régional d'Aquitaine.
- Lopistéguy, P. (2005). Développer une compétence universitaire trans-frontalière sur le processus e-learning. Année 2 : 'Organisation de la mémoire commune', Rapport de synthèse de l'équipe IDEE remis au Conseil Régional d'Aquitaine.
- Loustau, P. (2005). Traitement sémantique de documents dans leur composante spatiale. Rapport de stage Recherche (Master 2) du laboratoire LIUPPA.
- Luckin, R. et B. du Boulay (1999). "Designing a Zone of Proximal Adjustment." International Journal of Artificial Intelligence in Education (10)(2).
- Mahling, D. E., B. B. Sorrows et I. Skogseid (1995). A collaborative environment for semi-structured medical problem based learning. Proceedings of CSCL'95, Bloomington (USA).
- Malak, R. et C. Paredis (2004). On characterizing and assessing the validity of behavioral models and their predictions. Proceedings of DETC'04. ASME 2004 Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Salt Lake City (USA).
- Manapragada, S. (2004). Developing Scalable Flash Communication Server MX Applications. Flash Communication Server Article. <http://www.macromedia.com/devnet/mx/flashcom/articles/scalable.html>

-
- Marquesuzaà, C., J. Meyranx et T. Nodenot (1996). Towards pedagogically-sound Learning Environments : the specification process. Third International Conference on Computer Aided Learning and Instruction in Science and Engineering (CALISCE'96), San Sebastian (Espagne).
- Marquesuzaà, C. et T. Nodenot (1996). Specifying Educational Software : Goals and Process. Conference Proceedings ED-MEDIA96 : World AACE Conference on Educational Multimedia and Hypermedia, Boston (USA).
- Marquesuzaà, C., J. Meyranx et T. Nodenot (1997). The role of CASE tools in the process of specifying educational software. 5th Annual Conference on' Methodologies, British Computer Society, Specialist Group on Information System Methodologies, Preston (UK).
- Marquesuzaà, C. (1998). OMAGE : Outils et Méthode pour la spécification des connaissances au sein d'un Atelier de Génie Educatif, Doctorat en Informatique de l'Université de Pau et des Pays de l'Adour.
- MASIE (2003). Making sense of learning specifications and standards; a decision maker's guide to their adoption, Technical report of the MASIE Center, 2nd edition.
- McGilly, K. (1994). Classroom lessons: Integrating Cognitive Theory and Classroom Practice, Cambridge Press, MIT Press.
- McManus, M. et R. Aiken (1995). "Monitoring Computer-Based Collaborative Learning." Journal of Artificial Intelligence in Education volume 6(4): pages 307-336.
- Meirieu, P., Ed. (1994). Apprendre... Oui mais comment? Collection Pédagogies, ESF Edition.
- Mengelle, T. (1995). Etude d'une architecture d'environnements d'apprentissage basés sur le concept de préceptorat avisé. Doctorat de l'Université Paul Sabatier. Toulouse.
- Merrill, D. (1996). Reclaiming Instructional Design. Educational Technology, 36(5). www.ittheory.com/reclaim.htm
- Miao, Y. (2000). Design and implementation of a Collaborative Virtual Problem-Based Learning Environment. University of Darmstadt (Germany).
- Miao, Y., S. L. Holst, J. M. Haake et R. Steinmetz (2000). PBL protocols: guiding and controlling problem based learning processes in virtual learning environments. Fourth international conference of the learning sciences, Mahwah (USA).
- Mizogushi, R. et J. Bourdeau (2000). "Using Ontological Engineering to Overcome Common AI-ED Problems." International Journal of Artificial Intelligence in Education volume 11, n°2.
- Mizogushi, R. (2002). Ontology-based systematization of functional knowledge. TMCE2002:Tools and methods of competitive engineering, Wuhan (China).
- Morin, E. (1977). La méthode. La nature de la nature, Editions Seuil, Paris.
- Morrison, D. (2004). What do Instructional Designers design ? The eLearning Network Instructional Design Conference. <http://www.morrisonco.com/speaking.htm>
- Mottu, J.-M., B. Baudry, E. Brottier et Y. Le Traon (2005). Génération automatique de tests pour les transformations de modèles. 1^{ères} journées sur l'Ingénierie Dirigée par les Modèles, Paris (France).
- Muro (2003). The Muro Model Checker. <http://verify.stanford.edu/dill/murphi.html>
- Murray, T. (1999). "Authoring Intelligent Tutoring Systems: an analysis of the State of the Art." International Journal of Artificial Intelligence and Education volume 10, pages 98-129.
- Mwanza, D. et Y. Engeström (2003). Pedagogical Adeptness in the design of E-Learning Environments: experiences from the Lab@Future project. International conference on E-Learning in Corporate, Government, Healthcare, and Higher Education, Phoenix (USA).

- Naidu, S. et M. Olliver (1996). "Computer-supported collaborative Problem-based Learning : an instructional design Architecture for virtual learning in nursing education." *Journal of Distance Education/Revue de l'enseignement à distance*, number ISSN : 0830-0445.
- Ngomo, M., L. Oubahssi et H. Abdulrab (2005). La fourniture de services : une approche novatrice pour l'ouverture des systèmes de formation et du marché de la FOAD. Conférence EIAH 2005 (Environnements Informatiques pour l'Apprentissage Humain), Montpellier (France).
- Nicaud, J.-F. (1987). APLUSIX : un système expert de résolution pédagogique d'exercices d'algèbre, Thèse d'université de Paris-Sud (Orsay).
- Nodenot, T. (1992a). M.A.G.E. : un Méta-Atelier de Génie Educatif. Doctorat en Informatique de l'Université Paul Sabatier. Laboratoire API (Apprentissage, Pédagogie, Intelligence Artificielle). Toulouse.
- Nodenot, T. (1992b). A methodology for cooperative developments of educational software. Conference proceedings ICCAL'92 : Fourth International Conference on Computers and Learning, Wolfville (Canada).
- Nodenot, T. (1992c). MAGE : a methodological framework for the cooperative development of educational software. Conference Proceedings, IFIP Congress'92 for Research and Practice, Madrid (Espagne).
- Nodenot, T. (1995). What if the pedagogues specified Educational Software ? IFIP 6th World Conference on Computers in Education, Birmingham (UK).
- Nodenot, T. (2001). A case for an Adult Educational Technology. IFIP 7th World Conference on Computers in Education, Copenhagen (Denmark).
- Nodenot, T., C. Marquesuzaà, P. Laforcade, M.-N. Bessagnet et C. Sallaberry (2002). Spécifications et Architecture d'un Environnement Web supportant des Activités Coopératives d'Apprentissage. Conférence Internationale TICE 2002 (Technologies de L'information et de la Communication dans les Enseignements d'Ingénieurs et dans l'Industrie), INSA Lyon.
- Nodenot, T., P. Laforcade, C. Marquesuzaà et C. Sallaberry (2003). Knowledge Modelling of Cooperative Learning Situations: Towards a UML profile. 11th International Conference on Artificial Intelligence in Education (AIED 2003): Shaping the future of Learning through Intelligent Technologies, Sydney (Australia), IOS Press (volume n°97).
- Nodenot, T., C. Marquesuzaà, P. Laforcade et C. Sallaberry (2004). Model based Engineering of Learning Situations for Adaptive Web Based Educational Systems. ACM Thirteenth International World Wide Web Conference (IW3C2 Conference), ISBN : ACM 1-58113-912-8, New-York (USA).
- Nodenot, T., M. Gaio, M.-N. Bessagnet et C. Sallaberry (2005a). Creating rich collaborative learning scenarios : a model-driven approach for contextualizing software components and electronic documents. 8th IFIP World Conference on Computers in Education, Cape Town (South-Africa).
- Nodenot, T., M. Gaio, C. Sallaberry et P. Lopistéguy (2005b). "Towards an Open Source Framework for Collaborative Problem-Based Learning Environnements." *Eleed Journal (E-learning and Education)*, ISSN:1860-7470.
- Novak, J. D. et J. Wandersee (1991). "Special Issue on Concept Mapping." *Journal of Research in Science Teaching*.
- Novak, J. D. (2002). *The Theory Underlying Concept Maps and How To Construct Them*. Cornell University. <http://cmap.coginst.uwf.edu/info/printer.html>
- Odell, J., H. Van Dyke Parunak et B. Bauer (2000). Extending UML for Agents. AOIS Workshop (Agent Oriented Information Systems) at AAAI Conference (American Association for Artificial Intelligence), Austin (USA).
- OMG (2001). *Model Driven Architecture (MDA)*.

-
- OMG (2002a). The Software Process Engineering Management Metamodel (SPEM), Technical Report formal/2002-11-14.
- OMG (2002b). Meta-Object Facility (MOF) Specification Version 1.4.
- OMG (2003). Unified Modeling Language v1.5 Specification, Report formal/03-03-01.
- OMG (2004). UML Profile for Enterprise Distributed Object Computing (EDOC), OMG final adopted specification version 1.0.
- Oubahssi, L., M. Grandbastien et G. Claës (2004). Ré-ingénierie d'une plate-forme fondée sur la modélisation d'un processus global de FOAD. Conférence TICE 2004, "Technologies de l'Information et de la Connaissance dans l'Enseignement Supérieur et l'Industrie", Compiègne (France).
- Oudot, C. (2003). Modélisation UML d'une situation d'apprentissage coopérative, Rapport de stage du DESS "Sciences Cognitives Appliquées", Université de Bordeaux 2.
- Papert, S. (1981). Le jaillissement de l'esprit, Editions Flammarion, Paris.
- Paquette, G., F. Crevier et C. Aubin (1997). "Méthode d'ingénierie d'un système d'apprentissage." Revue Informations In Cognito n°8.
- Paquette, G. (1999). Meta-knowledge representation for Learning Scenarios Engineering. Proceedings of the 10th International Conference on Artificial Intelligence in Education (AIED'99), Le Mans (France).
- Paquette, G., I. De la Teja et A. Dufresne (2000). Explor@ : An Open Virtual Campus. EDMEDIA'2000, Montréal (Québec).
- Paquette, G. (2002). L'ingénierie du télé-apprentissage, pour construire l'apprentissage en réseaux, Presses de l'Université du Québec.
- Paquette, G. (2004a). Instructional engineering for learning objects repositories networks. 2nd International Conference on Computer Aided learning in Engineering Education, Grenoble.
- Paquette, G. (2004b). "Educational Modeling Language, from an Instructional Engineering Perspective." R. McGreal éditeur, Online education using learning objects. London: Routledge/Falmer.
- Paraskevi, S. (2002). Project-Based e-Learning in higher education: the model and the method, the practice and the portal. PhD proposal, Université de Genève (Suisse).
- Pastré, P. (1999). "La conceptualisation dans l'action : bilan et nouvelles perspectives." Education Permanente, n°2, p. 63-72.
- Péninou, A. (1993). MACT, un modèle d'agents centrés tâche pour la production de tuteurs intelligents par l'AGDI. Doctorat en Informatique de l'Université Paul Sabatier. Toulouse.
- Pernin, J.-P. (1996). M.A.R.S. Un modèle opérationnel de conception de simulations pédagogiques. Doctorat de l'Université Joseph Fourier, Laboratoire CLIPS-IMAG. Grenoble.
- Pernin, J.-P. (2003). "Objets pédagogiques : unités d'apprentissage, activités ou ressources ?" revue Sciences et Techniques Educatives (Numéro spécial "Ressources numériques, XML et Education").
- Pernin, J.-P. et A. Lejeune (2004a). Modèles pour la réutilisation de scénarios d'apprentissage. Colloque TICE Méditerranée (L'humain dans l'enseignement en ligne), Nice (France).
- Pernin, J.-P. et A. Lejeune (2004b). Dispositifs d'apprentissage instrumentés par les technologies : vers une ingénierie centrée sur les scénarios. Conférence internationale TICE 2004 (Technologies de L'information et de la Communication dans les Enseignements d'Ingénieurs et dans l'Industrie), Compiègne.

- Pernin, J.-P., C. Choquet, C. Martel et T. Nodenot (2005). Scénarisation pédagogique : que faire de la proposition IMS Learning Design ? ». Débat-Panel de la conférence EIAH 2005 (Environnements Informatiques pour l'Apprentissage Humain). Montpellier (France).
- Perrenoud, P. (1997). Pédagogie différenciée : des intentions à l'action, Collection Pédagogies, ESF éditeur.
- Pesty, S., C. Webber et N. Balacheff, Eds. (2003). Baghera : une architecture multi-agents pour l'apprentissage humain. Cognitive : Vers une informatique plus cognitive et sociale, Editions Cepadues.
- Piaget, J. (1967). Biologie et connaissance. Essai sur les relations entre les régulations organiques et les processus cognitifs, Gallimard, Paris.
- Piaget, J. (1974). Réussir et comprendre. Paris, PUF.
- Pinkwart, N., A. Harrer, S. Lohmann et S. Vetter (2004). Integrating Portal-Based Support Tools to foster Learning Communities in University Courses. Proceedings of the 4th International Conference on Web Based Education, Anaheim (Canada).
- Pohl, M. C. (1994). "The three dimensions of requirements engineering: a framework and its application." Informations Systems volume 19 (n°2).
- Pohu, A., C. Louberry et G. Mroz (2006). Le formalisme XMI : importation, exportation et transformation de modèles conformes à un profil UML. Rapport de projet de Master 2 (Technologies de l'Internet). Université de Pau et des Pays de l'Adour. A paraître.
- Py, D. (1996). "Aide à la démonstration en géométrie : le projet Mentoniezsh." Revue Sciences et Techniques Educatives volume 3 n°2, pages 227-256.
- Rawlings, A., P. Van Rosmalen, R. Koper, M. Rodriguez-Artacho et P. Lefrere (2002). Survey of Educational Modelling Languages (EMLs).
- Renaux, E., P.-A. Caron et X. Le Pallec (2005). Learning Management System component-based design: a model driven approach. Montreal Conference on e-Technologies (Mccetech), Montréal (Canada).
- Repenning, A., A. Ioannidou, M. Payton, W. Ye et J. Roschelle (2001). "Using Components for Rapid Distributed Software Development." IEEE Software.
- Rogalski, J. (2004). "La didactique professionnelle : une alternative aux approches de cognition située et cognitiviste en psychologie des acquisitions." Revue Activités volume 1, n°2.
- Roschelle, J., C. DiGiano et a. all (1999). "Developing Educational Software Components." IEEE Computer Cover Feature 32(9): 50-58.
- Roschelle, J., R. Pea, C. Hoadley, D. Gordin et B. Means (2000). "Changing how and what children learn in school with computer-based technologies." The future of children, children and computer-technology, volume 10.
- Rubart, J. et P. Dawabi (2002). A UML Profile for Modeling Groupware. 8th International Workshop , CRIWG'2002, La Serena (Chile), Published in J. M. Haake & J. A. Pino (Ed.): Groupware: Design, Implementation, and Use.
- Rumelhart, D. et J. L. Mc Clelland (1986). Parallel Distributed Processing: Explorations in the Microstructure of Cognition, MIT Press/ Bradford Books.
- Sallaberry, C., T. Nodenot, C. Marquesuzaà, M.-N. Bessagnet et P. Laforcade (2002). Information modelling within a Net-Learning Environment. 12th Conference on Information Modelling and Knowledge Bases, Krippen, Swiss Saxony (Deutschland), IOS Press.
- Sallaberry, C., T. Nodenot, P. Laforcade et C. Marquesuzaà (2005). Model driven development of cooperative Problem-Based Learning Situations. Implementing tools for teachers and learners from pedagogical models. Hawaii International Conference on System Sciences, IEEE Computer Society, ISBN : 0-7695-2268-8, Hawaii (USA).

-
- Santos, O. C., J. G. Boticario et C. Barrera (2004). Authoring a collaborative Task Extending the IMS LD to be performed in a standard-based Adaptive Learning Management System called ALFANET. International Conference on Web Engineering, Munich (Allemagne).
- Scardamalia, M., C. Bereiter et M. Lamon (1994). "The CSILE project: trying to bring the classroom into World 3." Classroom Lessons - Integrating cognitive theory and classroom practice Cambridge, MA : MIT Press.
- Scardamalia, M. et C. Bereiter (2003). "Knowledge building environments: extending the limits of the possible in education and knowledge work." Encyclopedia of distributed learning, Thousand Oaks, CA: Sage Publications.
- Schneemayer, G. (2002). Contextual Web Services for Teaching. Institut für Informatik. München (Allemagne), Ludwig Maximilians Universität.
- Schneider, D. (2003). Conception et implémentation de scénarios pédagogiques riches avec des portails communautaires. second colloque de Guéret, Les communautés virtuelles éducatives.
- Schoenmaker, J., E. Nienhuis, J. Scholten et J. Titulaer (1990). A methodology for Educational Software Engineering. IFIP Fifth World Conference on Computers in Education, Sydney (Australie).
- Schreiber, G., H. Akkermans, A. Anjewierden, R. de Hoog, N. Shadbolt, W. V. de Velde et B. Wielinga (1999). Knowledge Engineering and Management: the CommonKADS methodology, MIT press, Massachusetts.
- SCORM (2001). The Department of Defense Advanced Distributed Learning -- ADL -- Initiative Releases Version 1.2 of the Sharable Courseware Object Reference Model -- SCORM 1.2.
- SEC (1996). Rapport de la Task Force "Logiciels éducatifs et multimédia", Document de travail des services de la commission. Document SEC(96), 1426.
- Seidwitz, E. (2003). "What models Mean." IEEE Software: pp26-32.
- Séjourné, A., M. Baker, K. Lund et G. Molinari (2004). Schématisation argumentative et co-élaboration de connaissances : le cas des interactions médiatisées par ordinateur. Actes du colloque international "Faut-il parler pour apprendre ?" Arras (France).
- Selic, B. (2003). "The pragmatics of Model-Driven development." Revue IEEE Software(n°0740-7459/03): p.19-25.
- Sikkel, K., L. Gommer et J. Van der Veen (2001). A cross-case comparison of BSCW in different educational settings. First European Conference on COmputer-Supported Collaborative Learning, Maastricht (Pays Bas).
- Skinner, B. (1968). The Technology of Teaching, Des Moines: Meredith.
- Slavin, R. (1987). "Cooperative Learning and the Cooperative School." Educational Leadership, 45 n°3.
- Smart, J. (1995). User manual for WxCLIPS 1.43. University of Edinburgh (UK), Artificial Intelligence Applications Institute.
- Smart, J. et R. Rae (1995). Hardy user Guide version 1.3. University of Edinburgh (UK), Artificial Intelligence Applications Institute.
- Soller, A. (2002). Computational Analysis of Knowledge Sharing in Collaborative Learning System. Thesis of the University of Pittsburgh (USA).
- Soller, A. et A. Lesgold (2003). A Computational Approach to Analyzing Online Knowledge Sharing Interaction. 11th International Conference on Artificial Intelligence in Education (AIED 2003): Shaping the future of Learning through Intelligent Technologies, Sydney (Australia), IOS Press (volume n°97).
-

- Steinkuehler, C. A., S. J. Derry, D. K. Woods et C. E. Hmelo (2002). "The STEP environment for distributed Problem-Based Learning on the World Wide Web." Computer Support for collaborative learning: Foundations for a CSCL community, Hillsdale, NJ: Erlbaum.
- Steinkuehler, C. A., S. J. Derry, H. C. E et M. Delmarcelle (2002). "Cracking the resource nut with distributed problem based learning in secondary teacher education." Distance Education, volume 23: pages 23-39.
- Stokes, M. (2001). "Management Engineering Knowledge: MOKA methodology for Knowledge Based Engineering Applications." Professional Engineering and Publishing Limited, London (UK).
- Suchman, L. (1987). Plans and situated actions - The problem of human-machine communication, Cambridge University Press.
- Suthers, D. et A. Weiner (1995). Groupware for developing critical discussion skills. Proceedings of CSCL'95, Bloomington (USA).
- Suthers, D. (1997). An integrated approach to implementing collaborative inquiry in the classroom. Proceedings of CSCL'97, Toronto (USA).
- Suthers, D. (1999). Effects of Alternate Representations of Evidential Relations on Collaborative Learning Discourse. 3rd Conference on Computer Supported Collaborative Learning (CSCL99), Stanford (USA).
- Suthers, D. (2003). Representation Guidance for Collaborative Learning. 11th International Conference on Artificial Intelligence in Education (AIED 2003): Shaping the future of Learning through Intelligent Technologies, Sydney (Australia), IOS Press (volume n°97).
- Synteta, P. (2002). Project-Based e-Learning in higher education: the model and the method, the practice and the portal. PhD proposal, Université de Genève (Suisse).
- Tchounikine, P. (2002a). "Pour une Ingénierie des Environnements Informatiques pour l'Apprentissage Humain." Revue Information Interaction Intelligence (www.revue-i3.org) volume 2(n°1): 59-93.
- Tchounikine, P. (2002b). "Conception des environnements informatiques d'apprentissage : mieux articuler informatique et sciences humaines et sociales." Les technologies en Education : Perspectives de recherche et questions vives Baron G.L. et Bruillard E. éditeurs INRP-MSH-IUFM Basse normandie: p203-210.
- Tchounikine, P., M. Baker, N. Balacheff, M. Baron, A. Derycke, D. Guin, J.-F. Nicaud et P. Rabardel (2004). Platon-1 : quelques dimensions pour l'analyse des travaux de recherche en conception d'EIAH, Rapport de l'AS 51 du RTP39 intitulée "Fondements théoriques et méthodologiques de la conception des EIAH", STIC-CNRS.
- Thorne, S., C. Shubert et J. Merriman (2002). OKI Architecture overview. Massachusetts Institute of Technology. http://web.mit.edu/oki/product/whtpapers/arch_overview.html
- Tochon, F. (1990). "Les cartes de concepts dans la recherche cognitive sur l'apprentissage et l'enseignement." Perspectives documentaires en éducation n°21.
- Tricot, A., F. Plégat-Soutjis, J.-F. Camps, A. Amiel, G. Lutz et A. Morcillo (2003). Utilité, utilisabilité, acceptabilité : interpréter les relations entre trois dimensions de l'évaluation des EIAH. EIAH 2003 : Environnements Informatiques pour l'Apprentissage Humain, Strasbourg (France).
- Vantroys, T. (2003). Du langage métier au langage technique, une plateforme flexible pour l'exécution de scénarios pédagogiques. Doctorat en informatique de l'Université des Sciences et techniques. Lille.
- Varro, D. (2004). "Automated Formal Verification of Visual Modeling Languages by Model Checking." Journal of Software and Systems Modeling volume 3(2): pp 85-113.
- Voisin, R. (2005). Etude des usages d'un outil de type Tableau Blanc en contexte pédagogique. Exemples et mise en oeuvre avec Laszlo. Rapport de stage de Licence Professionnelle, Université de Limoges.

-
- Vygotsky, L. (1985). *Pensée et langage* (Traduction de *Thought and Language* publié en 1934, Cambridge, MIT press), Editions sociales (Paris).
- Webber, C. (2003). *Modélisation informatique de l'apprenant. Une approche basée sur le modèle cKc et la théorie de l'émergence*. Thèse en informatique de l'Université Joseph Fourier, Laboratoire LEIBNIZ-IMAG. Grenoble.
- Wenger, E. (1987). *Artificial Intelligence and Tutoring Systems*, Los Altos, CA : Morgan Kaufmann.
- Wenger, E. (1998). *Communities of practice: Learning, Meaning and Identity*, Cambridge University Press.
- Wiley, D. (2003). *Learning Objects : Difficulties and Opportunities*. Retrieved from the Internet http://wiley.ed.usu.edu/docs/lo_do.pdf
- Wilson, B. et M. Lowry (2000). *Constructivist Learning on the Web. Learning Technologies : Reflective and Strategic Thinking. New Directions for Adult and Continuing Education*. San Francisco.
- Wilson, B. (2004a). *Bounded Community: Designing and facilitating Communities in Formal Courses*. Meeting of the American Educational Research Association. San Diego.
- Wilson, B. (2004b). "Foundations for Instructional design: reclaiming the conversation." *Innovations in Instructional design: Essays in Honor of M. David Merrill* Mahwah NJ: Erlbaum.
- Yoo, M., B. Vité, J. Trincherini, V. Badan et M. Wentland Forte (2004). *Phoenix: An authoring tool for hybrid elearning documents*. World Conference on Educational Multimedia, Hypermedia & Telecommunications (EdMedia 2004), Lugano (Suisse).
- Young, M. F. (1993). "Instructional design for situated learning." *Educational Technology Research & Development Journal* (41): pp. 43-58.
- Zumbach, J. et P. Reimann (2003). "Influence of feedback on distributed problem based learning. Enhancing online communities in group functioning and well-being." *Designing for Change in Networked Learning Environments*, B. Wasson, S. Ludvigsen & U. Hoppe (eds.); Kluwer.

Annexe A

Tableaux synthétiques analysant le méta-modèle IMS-LD du point de vue de la conception des situations d'apprentissage

Cette annexe est structurée de la façon suivante :

- la figure A.1 présente l'étude menée pour analyser le modèle d'information IMS-LD associable à une activité,
- la figure A.2 présente l'étude menée pour analyser le modèle d'information IMS-LD associable à une activité collaborative,
- la figure A.3 présente l'étude menée pour analyser les propositions IMS-LD en termes de structuration et de dynamique des scénarios
- la figure A.4 présente les solutions apportées par le méta-modèle CPM aux différents problèmes soulevés.

Concepts étudiés du métamodèle IMS-LD	Autres concepts concernés	Caractéristique du concept dans la spécification IMS-LD pouvant poser problème	Référence au document de spécification 'IMS-LD version 1.0 Final Specification'	Raisons pouvant expliquer les choix effectués dans la spécification IMS-LD	Problèmes soulevés quand on spécifie/conçoit des situations d'apprentissage avec IMS-LD
<p style="text-align: center;">Modèle d'information associable à une activité</p> <p style="text-align: center;"><i>Activity</i></p>		<p>Elément non décomposable dont la description est un texte ou une référence à un objet externe</p>	<p>Pages 15-16 et 30 : "The activity-description is the actual cue given to the user (rendered in the user-interface) to describe the activity to be performed by the user. In most cases, the activity-description is text (of type webcontent or imsidcontent). In other cases it can be audio-file, video file, ... Whatever forms it takes, the activity-description is references via an <item>-element derived from Content Packaging, referencing a resource element in the content package"</p> <p>Pages 15 et 30 : "An activity references the environment in which the activity must be executed ..."</p> <p>Page 16 : "Activities take place in a so-called 'environment', which is a structured collection of learning objects, services and sub-environments. This relationship between an activity and an environment can be derived from the linguistic description of activities. Most nouns in the activity imply the availability of learning objects in the environment; references to other persons imply the availability of communication services; some verbs imply the availability of supportive services or tools."</p> <p>Page 17 : "Learning Objects are defined here as any reproducible and addressable digital or non-digital resource used to perform learning activities or support activities. In IMS Content Packaging, they are represented with the element 'Resources'. A classification of different types of learning elements can be found in the LOW Specification"</p>	<p>Priorité à l'interopérabilité sur la reproductibilité des scénarios décrits</p> <p>Priorité à la réutilisabilité d'environnements préexistants (et génériques pour ce qui est des services et outils)</p>	<p>Une activité est une boîte noire du point de vue du concepteur du scénario IMS-LD</p> <p>Une activité est un conteneur pour associer dans un scénario des learning objects conçus par ailleurs</p> <p>Impossible de spécifier les caractéristiques de l'environnement d'outils et des ressources à partir des besoins issus de la définition des activités à mener</p> <p>Impossible d'analyser une activité du point de vue des ressources produites / à produire par un apprenant ou un tuteur</p>
	<p>Environment</p>	<p>Elément non associé à des éléments de type Outcome (Outcome apparaît dans le modèle conceptuel initial mais disparaît dans le modèle d'information). Les Outcomes sont pris en charge par la mise à jour de propriétés exploitables par le scénario et via des notifications sur condition de terminaison (fin d'activité sur choix de l'utilisateur, durée maximale d'activité atteinte, ...)</p>	<p>Page 9 : "A component can be one of seven different types: role, property group, property, activity structure, activity, environment, or outcome. With the exception of outcome, these are all elements in the LD Information Model."</p> <p>Page 11 : "A notification is triggered by an outcome ... For instance, when one student completes an activity (= an outcome), then another student or the teacher may be notified and set another activity as a consequence."</p> <p>Page 44 : "Properties are defined and or declared (for already defined global properties) under learning-design/components/properties and operated upon with property-operation elements (view-property, set-property, conditions, change-property-value, etc.) ..."</p> <p>Pages 77-78 : "The results or outcomes of each learner's activities in the Unit of Learning will typically be recorded and maintained by the runtime service. The learning design presupposes some form of 'dossier' or learner record that is used to hold and maintain the various personal properties that are part of Learning Design Level B ..."</p>	<p>Comme une activité est une forme de conteneur dans IMS-LD qui n'a pas de connaissance précise sur les learning-objects et outils qui sont activés, seuls des outcomes (événements pédagogiques) très généraux peuvent être pris en compte</p> <p>Le niveau B (où sont gérées les propriétés) n'a que peu de lien avec le niveau A : il s'agit d'un rajout qui n'est pas intégré dans la description des activités puisque c'est au niveau des Learning Objects qu'est connue la sémantique de l'activité menée</p>	<p>Rien n'est prévu pour faire apparaître, lors de la conception d'une activité pédagogique, les événements significatifs pour un concept (savoir / savoir faire) abordés dans le cadre de cette activité</p> <p>Rien n'est prévu pour faire apparaître, lors de la conception d'une activité pédagogique, les concepts, les prérequis et les objectifs significatifs du scénario conçu</p>
	<p>Prerequisite, Learning Objective</p>	<p>Elément pouvant être associé à des références d'objectifs pédagogiques et de prérequis</p>	<p>Page 27 : "Learning objectives describe the intended outcome for learners. Learning-objectives and prerequisites contain a standard organization of items, referring to resources or sub manifests. Resource types connected to learning objectives and prerequisites can be webcontent, imsidcontent or it can point to an IMS RDCEO Schema ..."</p>	<p>Comme une activité est une forme de conteneur qui n'a pas de connaissance précise sur les learning-objects manipulés, les prérequis sont définis par ailleurs et apparaissent à titre d'information pour les apprenants</p>	<p>Rien n'est prévu pour faire apparaître, lors de la conception d'une activité pédagogique, les prérequis et les objectifs significatifs du scénario conçu</p>

Figure Annexe A.1 : Etude du modèle d'information associable à une activité

Concepts étudiés du méta-modèle IMS-LD	Autres concepts concernés	Caractéristique du concept dans la spécification IMS-LD pouvant poser problème	Référence au document de spécification "IMS-LD version 1.0 Final Specification"	Raisons pouvant expliquer les choix effectués dans la spécification IMS-LD	Problèmes soulevés quand on spécifie/conçoit des situations d'apprentissage avec IMS-LD
Modèle d'information associable à une activité collaborative		Le concept d'activité n'a aucune propriété particulière pour caractériser une activité collaborative	Page 4 : The designs which can be described by this meta-language might involve a single user or multiple users; the learning and instructional designers and providers might take a behaviorist, constructivist, or some other approach; they might require learners to work separately or collaboratively, but the OUNL studies found these could all be captured in terms of a Method containing Roles, Activity-structures, and Environments and a number of other concepts elaborated around these"		IMS-LD offre des moyens pour décrire des processus s'exécutant en parallèle (play, activity-structure, role-part) mais n'offre pas de moyens suffisants pour décrire leur synchronisation
		Plusieurs play peuvent se dérouler en parallèle mais de manière totalement indépendante	Page 38 : "There is always at least one play in every learning design (and every unit-of-learning). In runtime the play is interpreted to show and hide activities, (other)units-of-learning, environments and resources to the users. When there is more than one play, these are interpreted concurrently and independent of each other. The same user can see the results of more than one play in the user-interface. Practical experience has shown that a lot of designs use multiple plays, to represent the flows of activities per role, e.g. a play for the learners and a play for staff. However this can only be done when the activities are independent of each other."		
	Play, Role-part	Le concept de role-part permet de représenter des activités se déroulant en parallèle	Page 11 : "A play has acts, and in each act has one or more role-parts. The acts in a play follow each other in a sequence (although more complex sequencing behavior can take place within an act). The role-parts within an act associate each role with an activity. The activity in turn describes what that role is to do and what environment is available to it within the act. In the analogy, the assigned activity is the equivalent of the script for the part that the role plays in the act, although less prescriptive. Where there is more than one role-part within an act, these are run in parallel."	Priorité à l'interopérabilité et à la réutilisation de services logiciels sur la reproductibilité des scénarios collaboratifs imaginés par le concepteur	Impossible de spécifier les types d'accès de chaque rôle impliqué dans des activités collaboratives que l'on conçoit
		La prise en compte des activités collaboratives se fait en associant des rôles à des environnements d'outils à vocation collaborative	Page 17 : voir notions de service, send-mail service, conference-service. Example : "The conference service, in addition to a title and metadata, specifies four conference system roles: participant, observer, conference-manager, and moderator. These contain references to roles in the learning design. When the learning design roles have been assigned players, this information can be used to automatically set up the dedicated conference space. It is not defined in this Learning Design Specification what permissions the conference roles should have, so this is left up to the implementer."		
	Role, Environment	Dans IMS-LD les rôles sont globaux, statiques et non situés. Ils peuvent éventuellement être spécialisés lors de la conception. Les rôles sont associés aux activités par le concept de role-part	Page 15 : "Roles allow the type of participant in a unit of learning to be specified. There are two basic Role types: Learner and Staff. These however can be sub-typed to allow learners to play different roles in certain types of learning activity such as task-based, role-play and simulations. Similarly support staff can be sub-typed and given more specialized roles, such as Tutor, Teaching Assistant, Mentor, etc. Roles thus lay the basis for multi-user models of learning."		
		Dans IMS-LD, c'est lors de l'étape d'instanciation d'un scénario que seront résolus les problèmes de droits d'accès des rôles aux fonctionnalités et services offerts par les outils mis à disposition	Page 17 : "Service facilities are resources that cannot be given a URL at design time. They have to be instantiated by a local runtime service... For this to work, it requires a well defined set of service types, which are known to the runtime service, such as chat, discussion forum, announcement channel etc. These are now commonly found in learning management systems. In a learning design, the use and setup of such a service is declared at an abstract level, so that a runtime facility (or a human) can setup the necessary facility according to the requirements. In the learning design specification, the abstract declaration of a service facility is called a 'service'. The instantiation of a service is called a 'service facility'."		Une activité (collaborative) est un conteneur pour associer dans un scénario des rôles avec des environnements d'outils génériques conçus par ailleurs

Figure Annexe A.2 : Etude du modèle d'information associable à une activité collaborative

Concepts étudiés du métamodèle IMS-LD	Autres concepts concernés	Caractéristique du concept dans la spécification IMS-LD pouvant poser problème	Référence au document de spécification "IMS-LD version 1.0 Final Specification"	Raisons pouvant expliquer les choix effectués dans la spécification IMS-LD	Problèmes soulevés quand on spécifie/conçoit des situations d'apprentissage avec IMS-LD
<p>Structuration et dynamique des scénarios IMS-LD</p> <p><i>Principes méthodologiques proposés pour structurer le scénario</i></p>	<p>Method, Play, Act</p>	<p>Analyse descendante s'appuyant, si nécessaire, sur une adaptation des diagrammes d'activités UML</p>	<p>Page 25 (du document IMS LD Best Practice and Implementation Guide) : "The above advice on the sequence of steps to be taken is valid for creating XML document instances from narratives. When it comes to 'reading' existing document instances, one had better start with the method and work one's way down starting with the play element and then on to act, role-part, activity-structure elements all the way to the individual learning-activity or support-activity elements."</p> <p>Page 32 (du document IMS LD Best Practice and Implementation Guide) : "The third step concerns itself with the didactic scenario proper (i.e., the method element). Here, one may most conveniently follow a top-down approach, starting with plays and then working your way down via acts and role-parts to finally the activity structures and activities that were identified as components."</p> <p>Page 25 (du document IMS LD Best Practice and Implementation Guide) : "... The UML diagrams capture the workflow aspects of the narrative of a Unit of Learning. The UML diagram is an intermediary step, a kind of semi-formalization if you like. A UML diagram is much more rigorous than narrative, but contains significantly less detail than an XML document instance. Activity diagrams are used as they are well-suited to depict a workflow and parallel processes. Parallel processes are likely to occur when a variety of roles are distinguished with different responsibilities. In such cases swimlanes will be used to describe which role is responsible for which activities. The diagrams are of a composite nature in order to reflect the hierarchy of activities, activity structures, role parts, acts, and plays. Acts and role parts will be drawn in a single diagram, if needed the various activities that constitute a role part may be drawn in separate diagram for readability reasons." Voir aussi l'exemple Boeing</p> <p>Page 11 : "Activities can be assembled into activity-structures. An activity-structure aggregates a set of related activities into a single structure, which can be associated to a role in a role-part. A structure can model a sequence or a selection of activities. In a sequence, a role has to complete the different activities in the structure in the order provided. In a selection, a role may select a given number of activities from the set provided in the activity-structure. This can, for instance, be used to model situations where students have to complete two activities, which they may freely select from a collection of e.g., five activities contained in the activity-structure. Activity-structures can also reference other Activity-structures and reference external Units of Learning, enabling elaborate structures to be defined if required."</p> <p>Page 16 : "An Activity-structure in turn consists of references to one or more of: A Learning Activity, A Support Activity, An (sub) Activity-structure, Another (separate) Unit of Learning"</p> <p>Page 19 : "Notifications are only available at Level C of the Learning Design Specification. With notifications it is possible to send a message to a role or to assign new learning or support activities to roles based on certain events. These events are: the completion of a certain activity, the completion of a certain act, the completion of a play, the completion of the unit of learning, when an expression in a certain condition is true, when a certain property-value has been changed."</p> <p>Page 44 : "Properties are defined and/or declared (for already defined global properties) under learning-design/components/properties and operated upon with property-operation elements (view-property, set-property, conditions, change-property-value, etc)."</p> <p>Pages 18-19 : "Conditions are only available in Levels B and C of the IMS Learning Design. They are used in conjunction with properties to further refinement and to add personalization facilities in the learning design. Conditions have the basic format: IF [expression] THEN [show, hide, or change something or notify someone]. The expressions are mostly defined on the properties of the dossier of a learner (e.g., IF pre-knowledge-english=4). The effects of a condition are mostly different for individual users, although they can be assigned to the same role. Conditions work in the context of the current active act. In practice, conditions are mostly useful within activity-structures of the type 'selection.'"</p> <p>Page 27 : "UML activity diagrams place activities in a sequential or parallel order. Choices are allowed and activities may be nested. Also, responsibilities for activities may be indicated by the use of swim lanes."</p> <p>Pages 36 à 47 de l'exemple Boeing (du document IMS LD Best Practice and Implementation Guide) montrent les propriétés non attachées à des objets dans le diagramme d'activité UML et la multiplication des noms de propriétés (exemple : P-Hazardous-Lesson-Couner, P-Components-Test-Result, ...)</p>	<p>IMS-LD s'insère dans une perspective industrielle de la formation</p>	<p>L'approche descendante utilisée pour analyser une situation d'apprentissage conduit à des scénarios favorisant la conception de micro-activités</p>
	<p>Activity-Structure</p>	<p>Activity-Structure sert à la fois pour organiser (séquence, sélection) des activités faites par un seul rôle d'acteur mais aussi pour grouper des unités d'apprentissage dans lesquelles sont impliqués plusieurs rôles d'acteurs.</p>	<p>Les modèles obtenus multiplient les références au concept d'Activity-Structure. Or, ce concept prend des significations différentes selon le niveau d'analyse auquel il est utilisé. L'utilisation de deux concepts différents aurait facilité l'interprétation des scénarios</p>	<p>Priorité à la réutilisation plutôt qu'à la précision des concepts</p>	<p>Le modèle événementiel est limité à des événements prédéfinis (portant sur les concepts de description d'un scénario) et à la prise en compte des valeurs prises par des propriétés simples définies par le concepteur (mais non attachées à des objets significatifs que le concepteur envisage de faire manipuler aux acteurs). Il n'est donc pas possible de décrire l'évolution des propriétés attachées aux objets significatifs de conception puisque ces objets sont seulement référencés et pré-existants</p>
<p><i>Principes méthodologiques proposés pour décrire la dynamique d'un scénario</i></p>	<p>Property, Notification, Condition</p>	<p>Les notifications portent sur des événements simples pouvant être associés à des concepts IMS-LD (play, act, ...) ou des valeurs prises par certaines propriétés</p>	<p>Les niveaux B et C de la spécification ont été ajoutés au méta-modèle du niveau A en ayant le souci de limiter au maximum les intrusions entre niveaux</p>	<p>Pour combler les lacunes du modèle d'information, les noms de propriétés portent une partie de la sémantique de ces propriétés</p>	

Figure Annexe A.3 : Etude portant sur la structuration et la dynamique des scénarios

Concepts étudiés du méta-modèle IMS-LD	Caractéristique du concept dans la spécification IMS-LD pouvant poser problème	Problèmes soulevés quand on spécifie/conçoit des situations d'apprentissage avec IMS-LD	Éléments de solution proposés par le langage CPM
<p>Modèle d'information associable à une activité</p> <p>Voir détails de l'analyse en annexe A (figure A.1)</p>	<p>Élément non décomposable dont la description est un texte ou une référence à un objet externe</p> <p>Élément lié à 0 ou plusieurs références d'éléments de type Environment (un environnement constitué d'outils et/ou de ressources)</p> <p>Élément non associé à des éléments de type Outcome (Outcome apparaît dans le modèle conceptuel initial mais disparaît dans le modèle d'information). Les Outcomes sont pris en charge par la mise à jour de propriétés exploitables par le scénario et via des notifications sur condition de terminaison (fin d'activité sur choix de l'utilisateur, durée maximale d'activité atteinte, ...)</p> <p>Élément pouvant être associé à des références d'objectifs pédagogiques et de prérequis</p> <p>Le concept d'activité n'a aucune propriété particulière pour caractériser une activité collaborative</p> <p>Plusieurs play peuvent se dérouler en parallèle mais de manière totalement indépendante</p> <p>Le concept de rôle-part permet de représenter des activités se déroulant en parallèle</p> <p>La prise en compte des activités collaboratives se fait en associant des rôles à des environnements d'outils à vocation collaborative</p> <p>Dans IMS-LD les rôles sont globaux, statiques et non situés. Ils peuvent éventuellement être spécialisés lors de la conception. Les rôles sont associés aux activités par le concept de rôle-part</p> <p>Dans IMS-LD, c'est lors de l'étape d'instanciation d'un scénario que seront résolus les problèmes de droits d'accès des rôles aux fonctionnalités et services offerts par les outils mis à disposition</p>	<p>Une activité est une boîte noire du point de vue du concepteur du scénario IMS-LD</p> <p>Une activité est un conteneur pour associer dans un scénario des learning objects conçus par ailleurs</p> <p>Impossible de spécifier les caractéristiques de l'environnement d'outils et des ressources à partir des besoins issus de la définition des activités à mener</p> <p>Impossible d'analyser une activité du point de vue des ressources produites / à produire par un apprenant ou un tuteur</p> <p>Rien n'est prévu pour faire apparaître, lors de la conception d'une activité pédagogique, les événements significatifs pour un concept (savoir / savoir faire) abordés dans le cadre de cette activité</p> <p>Rien n'est prévu pour faire apparaître, lors de la conception d'une activité pédagogique, les concepts, les prérequis et les objectifs significatifs du scénario conçu</p>	<p>CPM propose des concepts pour décrire les détails du déroulement d'une activité (le concept d'activité devient une boîte blanche)</p> <p>CPM propose des concepts pour décrire les objets d'apprentissage / les services et outils / les objectifs qui seront manipulés dans le cadre d'une activité</p>
<p>Modèle d'information associable à une activité collaborative</p> <p>Voir détails de l'analyse en annexe A (figure A.2)</p>	<p>Le concept d'activité n'a aucune propriété particulière pour caractériser une activité collaborative</p> <p>Plusieurs play peuvent se dérouler en parallèle mais de manière totalement indépendante</p> <p>Le concept de rôle-part permet de représenter des activités se déroulant en parallèle</p> <p>La prise en compte des activités collaboratives se fait en associant des rôles à des environnements d'outils à vocation collaborative</p> <p>Dans IMS-LD les rôles sont globaux, statiques et non situés. Ils peuvent éventuellement être spécialisés lors de la conception. Les rôles sont associés aux activités par le concept de rôle-part</p> <p>Dans IMS-LD, c'est lors de l'étape d'instanciation d'un scénario que seront résolus les problèmes de droits d'accès des rôles aux fonctionnalités et services offerts par les outils mis à disposition</p>	<p>IMS-LD offre des moyens pour décrire des processus s'exécutant en parallèle (play, activity-structure, role-part) mais n'offre pas de moyens suffisants pour décrire leur synchronisation</p> <p>Impossible de spécifier les types d'accès de chaque rôle impliqué dans des activités collaboratives que l'on conçoit</p> <p>Une activité (collaborative) est un conteneur pour associer dans un scénario des rôles avec des environnements d'outils génériques conçus par ailleurs</p>	<p>CPM propose des concepts pour différencier les activités individuelles des activités collaboratives (et décrire les synchronisations prévues entre ces activités)</p> <p>CPM permet de décrire les détails des services et des droits des usagers sur les services et ressources affectées</p>
<p>Structuration et dynamique des scénarios IMS-LD</p> <p>Voir détails de l'analyse en annexe A (figure A.3)</p>	<p>Analyse descendante s'appuyant, si nécessaire, sur une adaptation des diagrammes d'activités UML</p> <p>Activity-Structure sert à la fois pour organiser (séquence, sélection) des activités faites par un seul rôle d'acteur mais aussi pour grouper des unités d'apprentissage dans lesquelles sont impliqués plusieurs rôles d'acteurs.</p> <p>Les notifications portent sur des événements simples pouvant être associés à des concepts IMS-LD (play, act, ...) ou des valeurs prises par certaines propriétés</p> <p>Les diagrammes d'activités UML décrivent un learning possible qui peut faire apparaître et/ou peut exploiter les valeurs prises par des éléments de type Property</p>	<p>L'approche descendante utilisée pour analyser une situation d'apprentissage conduit à des scénarios favorisant la conception de micro-activités</p> <p>Les modèles obtenus multiplient les références au concept d'Activity-Structure. Or, ce concept prend des significations différentes selon le niveau d'analyse auquel il est utilisé. L'utilisation de deux concepts différents aurait facilité l'interprétation des scénarios</p> <p>Le modèle événementiel est limité à des événements prédéfinis (portant sur les concepts de description d'un scénario) et à la prise en compte des valeurs prises par des propriétés simples définies par le concepteur (mais non attachées à des objets significatifs que le concepteur envisage de faire manipuler aux acteurs). Il n'est donc pas possible de décrire l'évolution des propriétés attachées aux objets significatifs de conception puisque ces objets sont seulement référencés et pré-existants</p>	<p>CPM s'appuie sur les diagrammes de use-case, les diagrammes de classe, les diagrammes d'états et les diagrammes d'activités</p> <p>CPM différencie le concept d'Activity-Structure (associé à un rôle) du concept de LearningPhase (impliquant plusieurs rôles)</p> <p>CPM propose le concept d'élément dynamique de modélisation afin de pouvoir associer des diagrammes d'états / transitions à des concepts significatifs (représentés par des diagrammes de classes ou des use-cases) des situations d'apprentissage décrites</p> <p>Les modèles d'activités décrivant un scénario possible peuvent exploiter les états des objets significatifs</p>

Figure Annexe A.4 : Synthèse de l'étude et propositions portées par le langage CPM

Annexe B

Extraits des documents relatifs à la situation-problème Smash



Témoignage n°1

Transcription de l'interview

Témoignage interrogé sur les lieux de l'accident par l'inspecteur Geoffrey Morante et son assistante Carine Gregor.

Nom: Sophie Lebrun

Adresse: rue des figuiers, les Marais

Age: 32 ans

Sophie Lebrun : J'attendais à l'arrêt de bus. J'ai vu une voiture blanche arriver vers le village. Elle venait des Marais. C'est vrai qu'elle allait à plus de 50 km/h mais cette route n'est pas dangereuse. Et puis ce vélo est arrivé comme un fou, il a fait un écart sur la route, sans prévenir avec le bras. En tous cas, la voiture blanche n'a pas pu le voir : c'est allé trop vite. Elle l'a percuté... le conducteur était décomposé...

Figure Annexe B.1 : Exemple du témoignage 1 de Smash



QCM n°1, groupe 1

- 1) Où se situe Simon Dupond au moment de l'accident ?
 - Sur la route, il traverse le passage piéton
 - Il se prépare à traverser le passage piéton
 - Il marche à côté du passage piéton

- 2) La voiture qui sort du village des Mille-fleurs :
 - S'est arrêtée au passage piéton
 - Ne s'est pas arrêtée au passage piéton
 - On ne sait pas

- 3) La voiture blanche
 - S'est arrêtée au passage piéton
 - Ne s'est pas arrêtée au passage piéton
 - On ne sait pas

- 4) Qui a percuté le vélo ?
 - La voiture qui sort du village des Mille-fleurs
 - La voiture blanche
 - Il est tombé tout seul

Figure Annexe B.2 : Exemple de QCM associé au premier témoignage

		Question de départ	Activités conduites par les élèves	Activités langagières	Organisation de la classe	Connaissances, savoirs et savoir-faire en jeu
Séance 1	Présentation de la situation-problème	Un accident de vélo a eu lieu. Comment trouver le coupable ?	Organisation en groupes de travail. Réflexion sur les tâches à accomplir pour réussir	Oral	classe entière	Communiquer, proposer des modes d'organisation
Séance 2	Phase de réalisation des enquêtes	Un accident de vélo a eu lieu. Qui est le coupable ?	Recherche d'information, positionnement d'objets sur un plan géographique, rédaction d'une synthèse	Oral et écrit à l'intérieur de chaque groupe	4 groupes	S'organiser à l'intérieur de chaque groupe. Communiquer par schémas et textes
Séance 3	Phase de mise en commun des résultats	Idem phase 2	Faire la synthèse du travail mené en séance 2, évaluer et corriger	Oral (compte-rendu de l'enquête) et écrit (modification de la carte)	4 groupes, classe entière	Argumenter, questionner
Séance 4	Phase de renforcement	Un accident de vélo a eu lieu. Quels sont les niveaux de responsabilité ?	positionnement d'objets sur un plan géographique, rédaction d'une synthèse argumentée	écrit	individuel	Réinvestir les connaissances acquises lors des séances précédentes
Séance 5	Phase de prise de recul	Nous avons enquêté sur un accident de vélo. Cela va-t-il changer notre comportement sur la route ?	Reformulation des conclusions à tirer, recontextualisation des conclusions de la séquence, Activités métacognitives	oral	4 groupes, classe entière	Exprimer des points de vue personnel et argumenter une position prise. Accepter le point de vue de l'autre ; juger sa propre implication au sein d'un groupe

Figure Annexe B.3 : Découpage en séances (actes) de la situation Smash

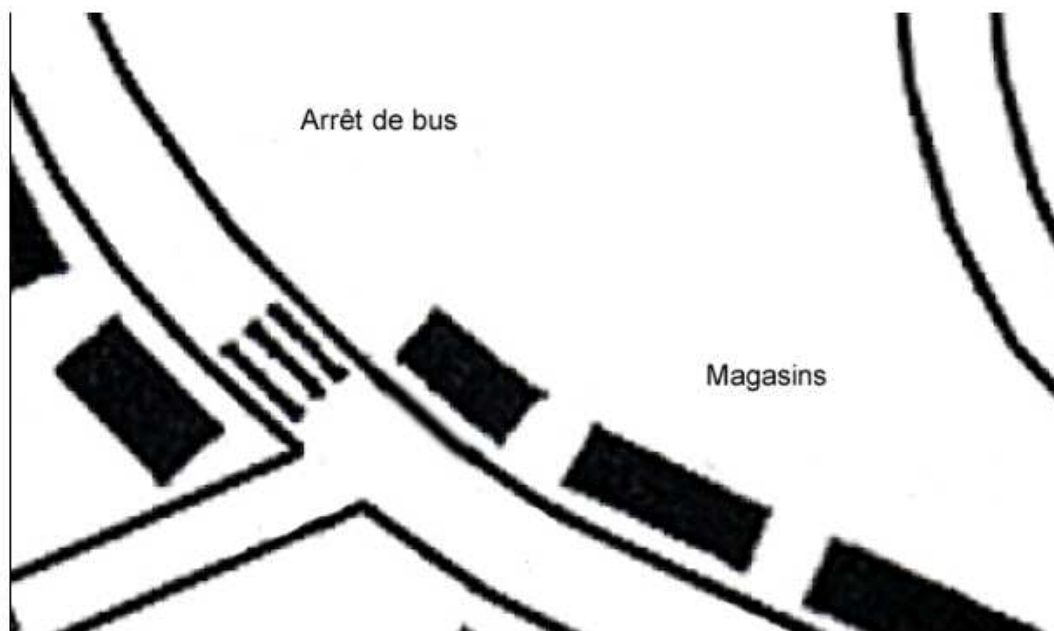


Figure Annexe B.4 : Gros plan de la zone de l'accident

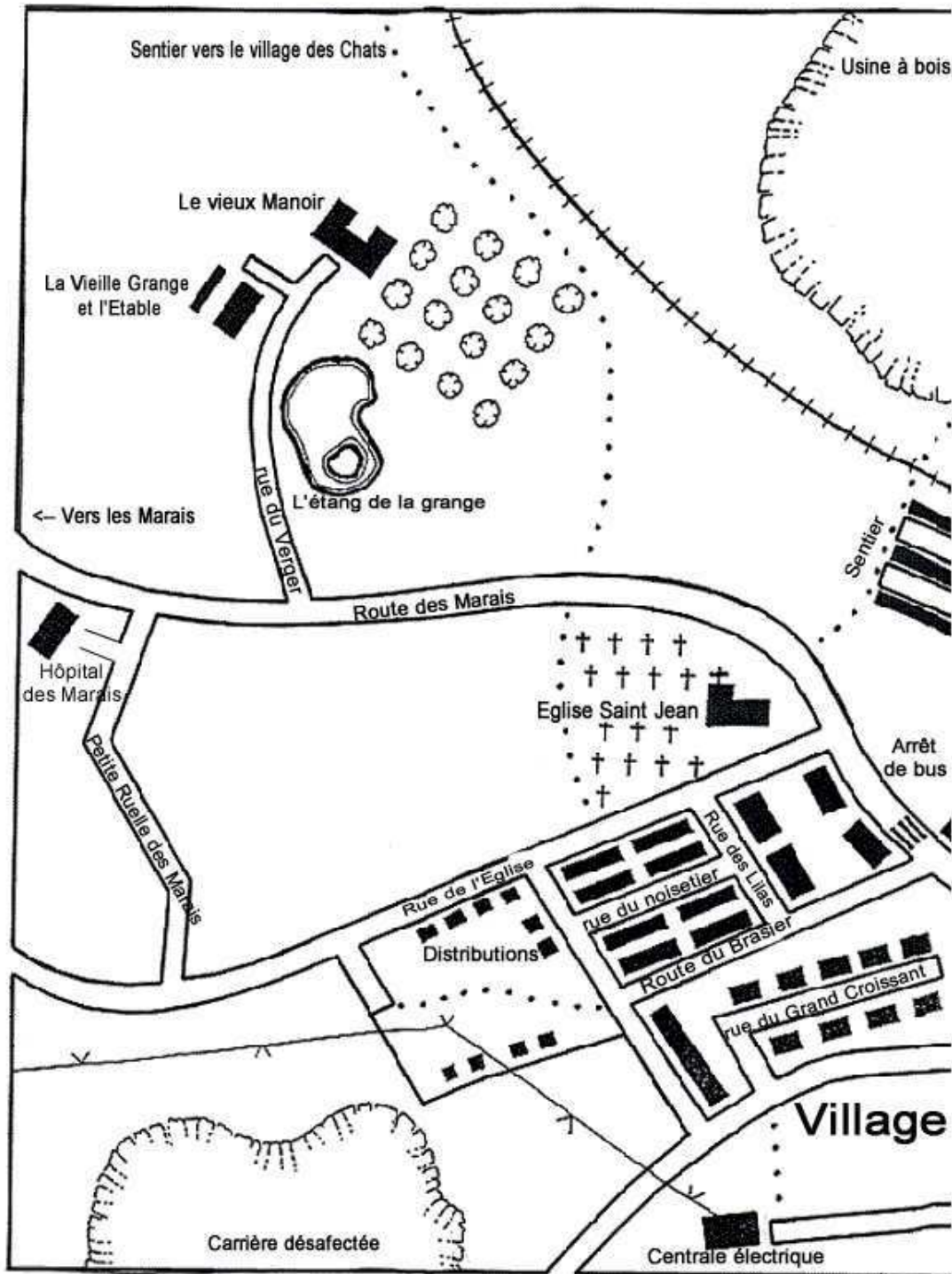


Figure Annexe B.5 : Carte du village des Mille-Flours où s'est déroulé l'accident (partie gauche)

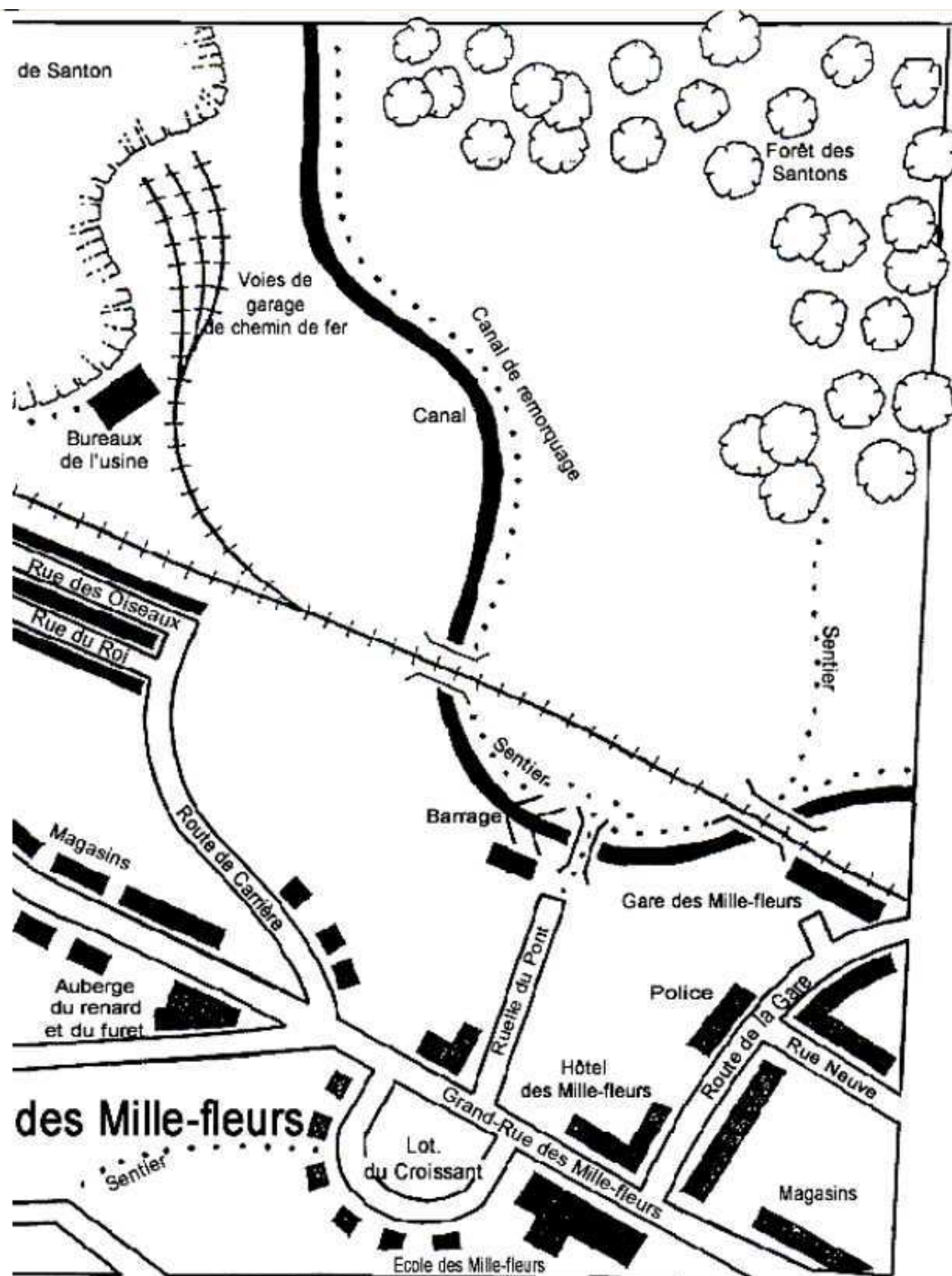


Figure Annexe B.6 : Carte du village des Mille-Fleurs où s'est déroulé l'accident (partie gauche)

Séquence Smash

Séance 3 : mise en commun des résultats

Acteurs/Rôles :

- Enseignant-tuteur,
- Groupes G1 à G3 (qui sont successivement auditeurs et speakers),
- Groupe G4 (qui est successivement auditeur et speaker).

Contexte :

L'analyse des témoignages est terminée. Les différents groupes d'apprenants ont produit un schéma présentant leur vision de cet accident. Ils ont donc produit :

- Un schéma complété dans lequel ils ont positionné les témoins et acteurs,
- Un ensemble de QCM complétés (un QCM par témoignage),
- Leur synthèse : un ensemble de questions (zones d'ombre) + leur vision de l'accident (culpabilités, raisons, ...).

Objectifs de la séance pour le prof :

- Vérifier que chaque groupe a bien prélevé les indices qui étaient dans leurs témoignages,
- Vérifier que, sur la carte, les éléments ont été placés comme il faut,
- Vérifier que chaque groupe a conscience de la fragilité de ses conclusions,
- Amener à une solution collective.

Motivation des groupes d'apprenants pour rentrer dans la séance :

- Exprimer sa vision, montrer qu'il a résolu l'énigme,
- Justifier ses choix, ses décisions (argumenter),
- Confronter sa vision à celle des autres groupes,
- Trouver des réponses aux zones d'ombre de la solution proposée par le groupe considéré.

Déroulement général de la séance :

- 1) Présentation de la séance et précisions sur l'ordonnancement pour le tour de parole + durée des exposés
- 2) Présentation par les groupes G1 à G3 de leurs productions (plan + conclusions) et correction de leur plan, compte tenu des témoignages incomplets dont ils disposaient.
- 3) Présentation par le groupe G4 du plan de l'accident tel qu'il l'a déduit de la totalité des témoignages dont il disposait. Critique collective (G1 à G4 + tuteur) de ce plan. Discussion orale sur les culpabilités, les responsabilités ...
- 4) Production écrite par G4 / (G5=G1+G2+G3) de leurs conclusions d'enquête suite aux discussions
- 5) Présentation par le tuteur du film de l'accident et commentaires sur les niveaux de responsabilités.

Figure Annexe B.7 : Les différentes phases de la séance 3

Séquence Smash

Séance 3 : mise en commun des résultats

Phase 1

Objet :

Présentation de la séance et précisions sur l'ordonnancement pour le tour de parole + durée des exposés

Rôle des acteurs :

- Le tuteur choisit l'ordre de tour de parole des groupes 1 à 3, en choisissant comme premier speaker un groupe dont les plans et synthèse sont corrects (ou proche de la solution),
- Le tuteur donnera la parole en dernier lieu au groupe 4 qui a eu tous les témoignages à disposition.

Objectifs d'apprentissage

Ressources :

- (En entrée) Pour ordonnancer les groupes, le tuteur a accès aux différentes productions (carte et QCMs des groupes 1 à 4).
- (En entrée) Le tuteur dispose des moyens pour évaluer la justesse des productions
- (En sortie) La feuille de route prévue par le tuteur (écrit / oral) est lancée

Evénements pédagogiques significatifs :

- Présentation de la séance démarrée (consignes, déroulement prévu de la séance),
- Feuille de route (intégrant consignes et déroulement prévu) transmise aux groupes G1 -> G3
- Feuille de route intégrant consignes et déroulement prévu) transmise au groupe G4

Déroulement de la phase 1 :

- a) Le tuteur fait une planification concernant l'ordonnancement des présentations en tenant compte de la qualité des plans produits par les groupes G1 à G3. Pour cela, il dispose des plans types corrects pour chaque groupe compte tenu des témoignages dont disposaient les groupes considérés.
- b) Le tuteur présente la séance aux 4 groupes d'apprenants. Il donne aussi l'ordre des interventions à respecter.
- c) Il remet à chaque groupe une feuille de route (la feuille de route du groupe 4 est différente de la feuille de route des groupes G1, G2 et G3).

Figure Annexe B.8 : Détails pour la phase 1 de la séance 3

Séquence Smash
Séance 3 : mise en commun des résultats
Phase 2

Objet :

Présentation par les groupes G1 à G3 de leurs productions (plan + conclusions) et correction de leur plan, compte tenu des témoignages incomplets dont ils disposaient.

Rôle des acteurs :

- Le tuteur est là pour gérer le déroulement de la séance et les événements pédagogiques qui se produisent,
- Les groupes 1 à 3 sont auditeurs et speakers selon qu'ils ont ou pas la parole,
- Les groupes speaker font leur présentation en montrant leur plan et en le commentant. Ils répondent à l'issue de leur exposé aux questions qui leur sont posées par le tuteur,
- Les groupes auditeur (dont fait partie le groupe 4) prennent des notes, lisent et commentent les documents qui leur sont remis.

Objectifs d'apprentissage (et autres rubriques utiles) :**Ressources :**

- (En entrée) Les témoignages utilisés par chaque groupe (remis au fur et à mesure),
- (En entrée) Les plans et feuilles de notes produits par les groupes 1 à 3 durant la séance,
- (En sortie) Les plans corrigés des groupes 1 à 3,
- (En sortie) Des notes et questions que se posent les différents groupes.

Événements pédagogiques significatifs :

- Parole est donnée à un des groupes G1 à G3 (phase 1 des exposés),
- Début exposé(i),
- Fin exposé(i) (décidé par groupe, par tuteur ou horloge),
- Fin du temps imparti au groupe(i)

Déroulement de la phase 2 :

- a) La parole est donnée successivement aux groupes 1 à 3 dans l'ordre défini précédemment par le tuteur.
- b) Les groupes 1 à 3 font leur présentation les uns après les autres en mettant à disposition des autres groupes les témoignages sur lesquels ils ont travaillé.
- c) Les groupes auditeurs écoutent et prennent des notes mais n'interviennent pas.
- d) A la fin de chaque présentation, le tuteur intervient pour faire corriger toutes les erreurs de compréhension (liées à l'analyse des QCM) et de positionnement sur la carte.
- e) A la fin des 3 exposés, le tuteur met à disposition des groupes 1 à 3 les trois plans corrigés

Figure Annexe B.9 : Détails pour la phase 2 de la séance 3

Séquence Smash
Séance 3 : mise en commun des résultats
Phase 3

Objet :

Présentation par le groupe G4 du plan de l'accident tel qu'il l'a déduit de la totalité des témoignages dont il disposait. Critique collective (G1 à G4 + tuteur) de ce plan. Discussion orale sur les culpabilités, les responsabilités ...

Rôle des acteurs :

- Le tuteur est là pour gérer le déroulement de la séance et les événements pédagogiques qui se produisent,
- Les groupes 1 à 3 sont auditeurs durant la présentation du groupe 4,
- Durant la présentation du groupe 4, les groupes 1 à 3 notent leurs éventuelles questions,
- A l'issue de la présentation du groupe 4, le tuteur choisit parmi les questions écrites des groupes 1 à 3 celle qui va être posée (plusieurs seront posées en fonction du temps restant). Il passe la parole à un groupe qui va reformuler la question pour la dire oralement.
- Le groupe 4 répond à chacune des questions.
- Le tuteur fait des compléments s'il le souhaite entre chaque question.

Objectifs d'apprentissage (et autres rubriques utiles) :

Ressources :

- (En entrée) Les plans et feuilles de notes produits par le groupe 4 durant la séance,
- (En entrée) Les notes de chaque groupe 1 à 3 prises durant les présentations,
- (En entrée) Le plan correct,
- (En sortie) Le plan correct mis à dispo de chaque groupe pour les étapes suivantes,
- (En sortie) Des termes comme culpabilité, niveau de responsabilité, imprudence, ...

Evénements pédagogiques significatifs :

- Parole est donnée au groupe G4 (phase 2 des exposés),
- Début exposé(4),
- Fin exposé(4) (décidé par groupe, par tuteur ou horloge),
- Question à poser (question q, groupe i),
- Commentaire,
- Fin du temps imparti au groupe(i)

Déroulement de la phase 3 :

- a) La parole est donnée au groupe 4.
- b) Le groupe 4 fait la présentation de sa carte et de ses conclusions
- c) Le tuteur choisit parmi les questions des groupes 1 à 3 celles qui vont permettre de corriger la carte. Et les fait poser oralement. Il complète ou fait un complément si nécessaire
- d) Le tuteur choisit ensuite les questions qui portent sur les pbs de responsabilité, d'imprudence. Et les fait poser en commentant par introduction du vocabulaire adéquat
- e) Le jeu des questions/réponses se termine quand les questions pendantes des groupes 1 à 3 sont évacuées.

Figure Annexe B.10 : Détails pour la phase 3 de la séance 3

Annexe C

Les paquetages du méta-modèle CPM

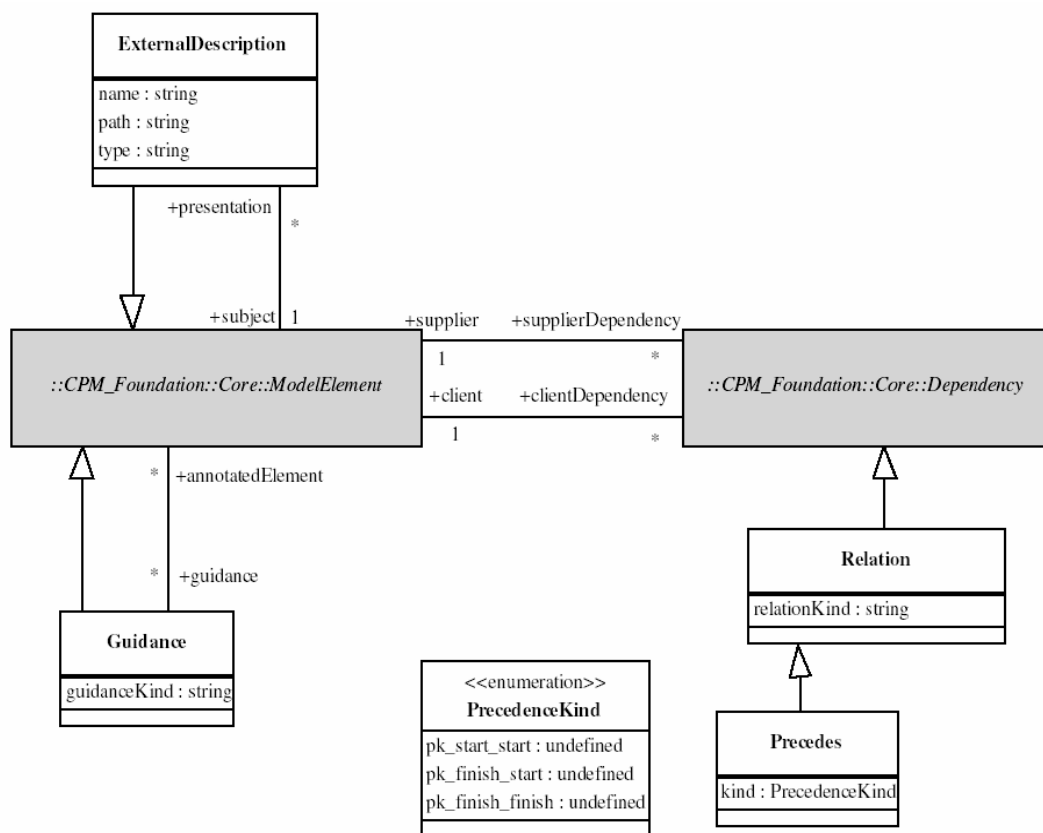


Figure Annexe C.1 : Le paquetage CPM_BasicElements

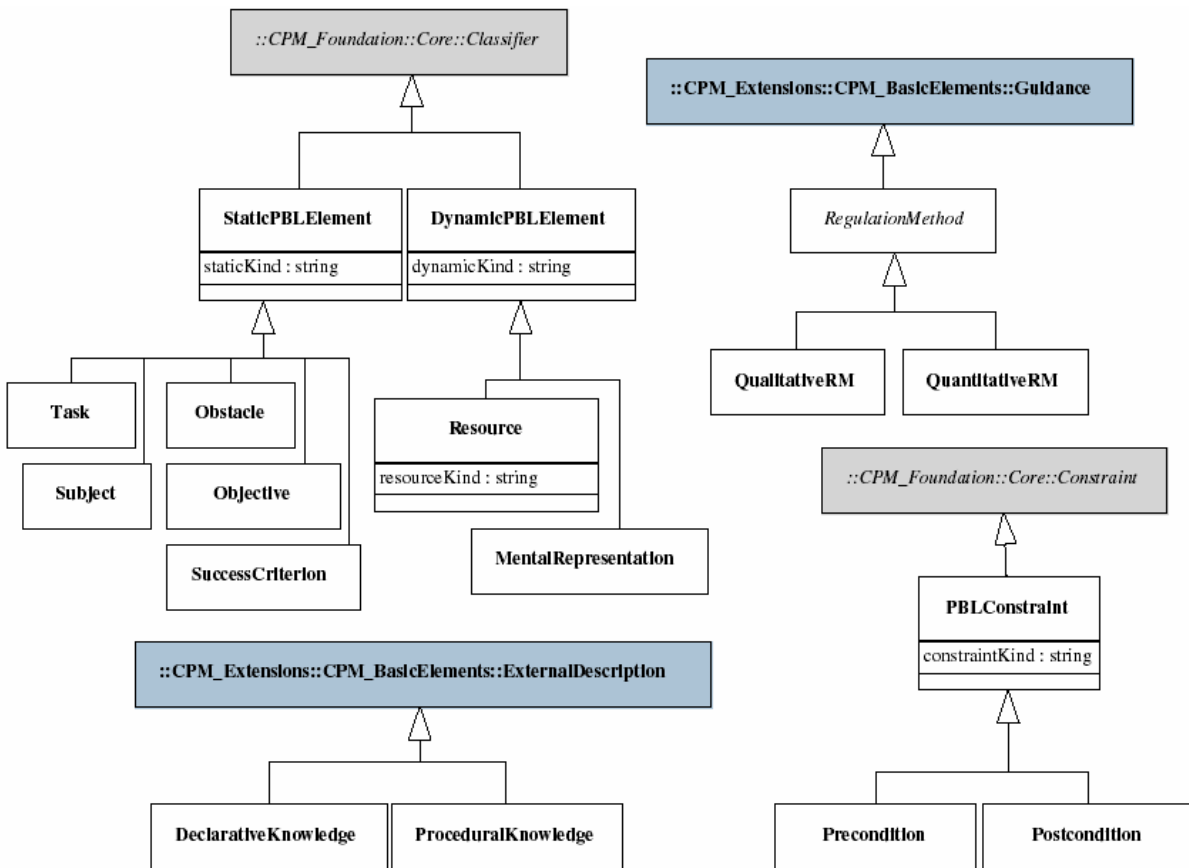


Figure Annexe C.2 : Le paquetage CPM_PedagogicalPackage

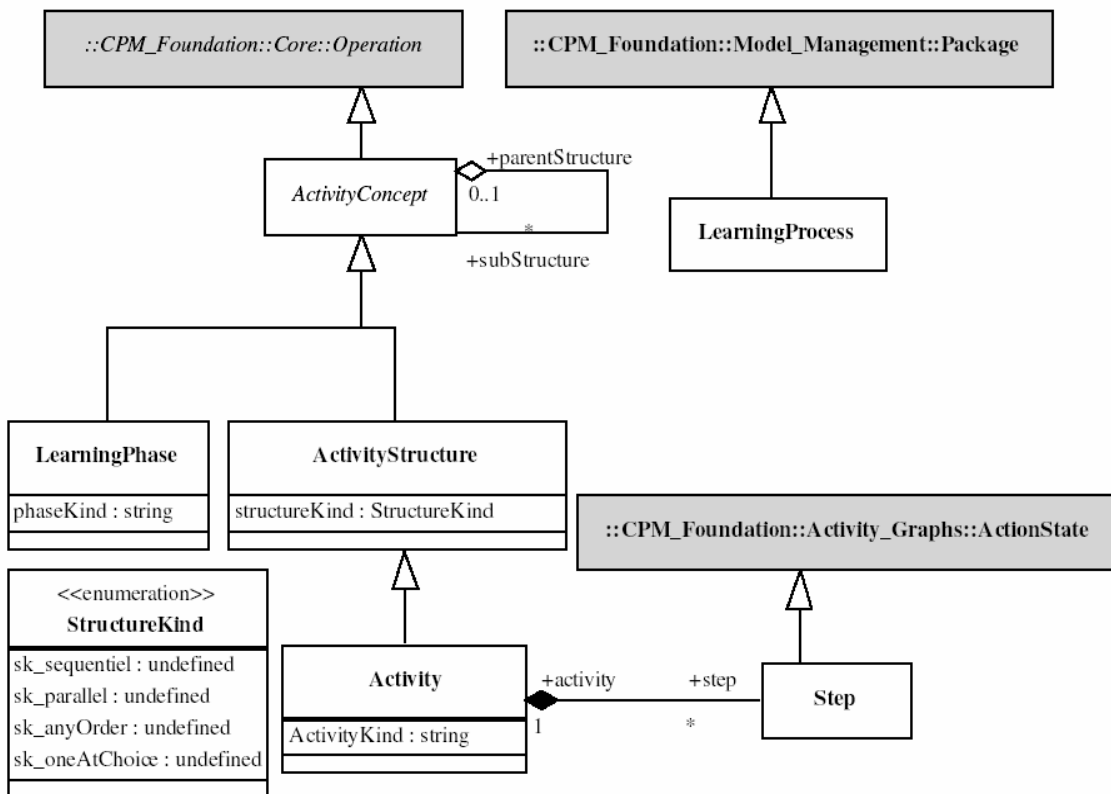


Figure Annexe C.3 : Le paquetage CPM_StructuralPackage

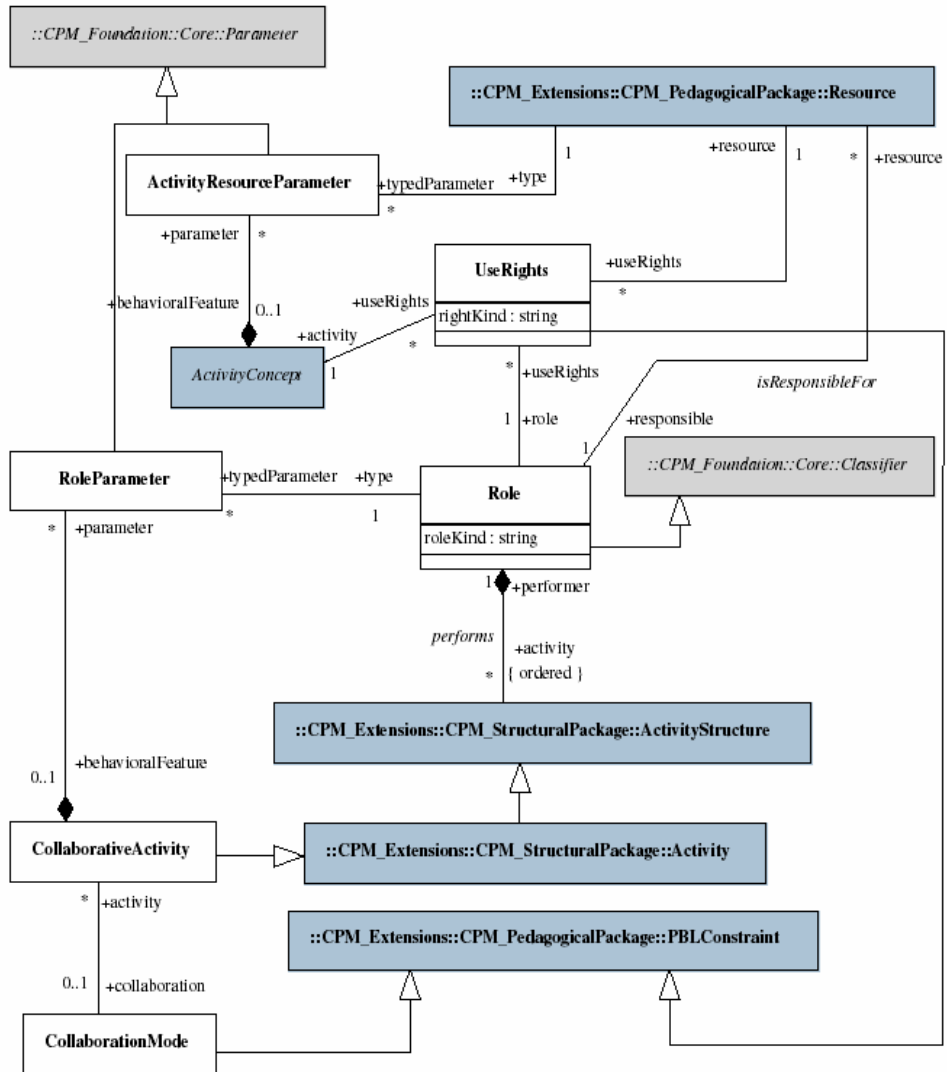


Figure Annexe C.4 : Le paquetage CPM_SocialPackage

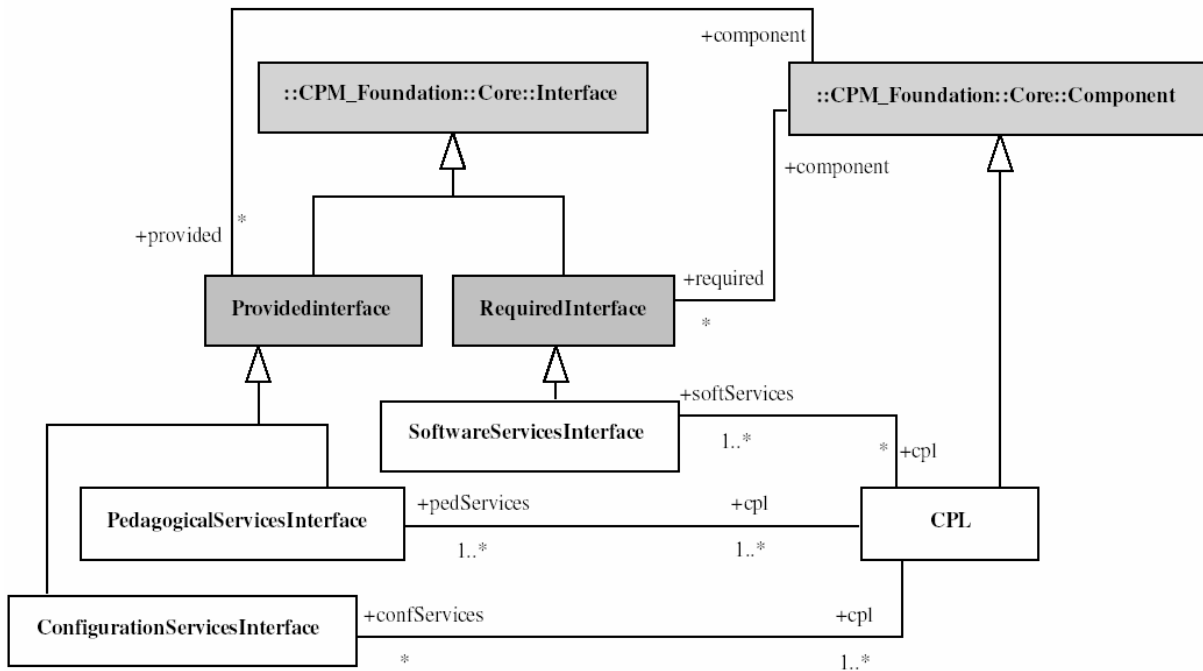


Figure Annexe C.5: Le paquetage des composants CPL

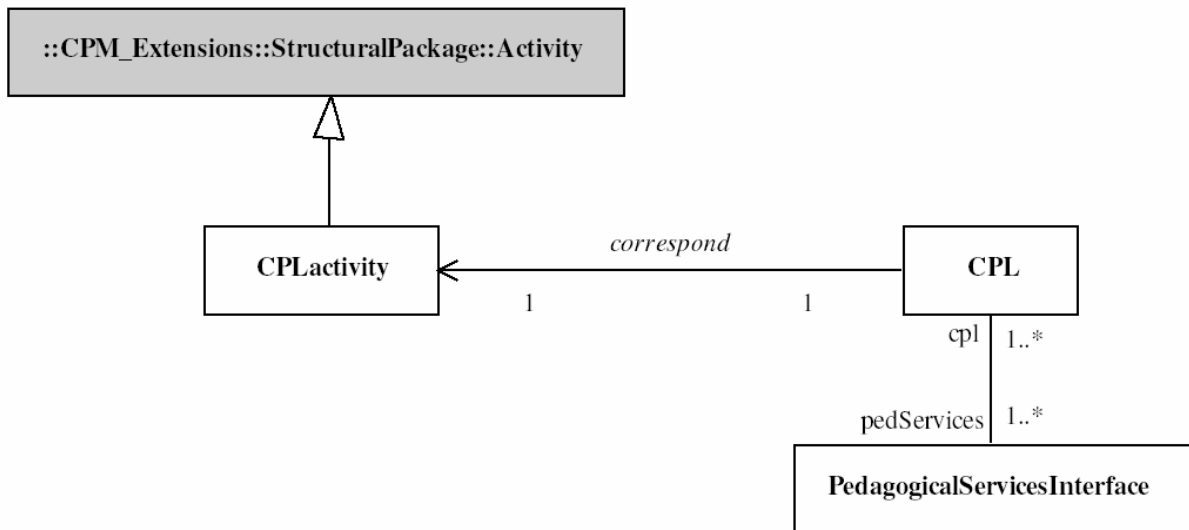


Figure Annexe C.6: Lien entre le concept d'activité et le composant CPL

Annexe D

Extraits du méta-modèle d'UML utiles pour la compréhension du profil CPM

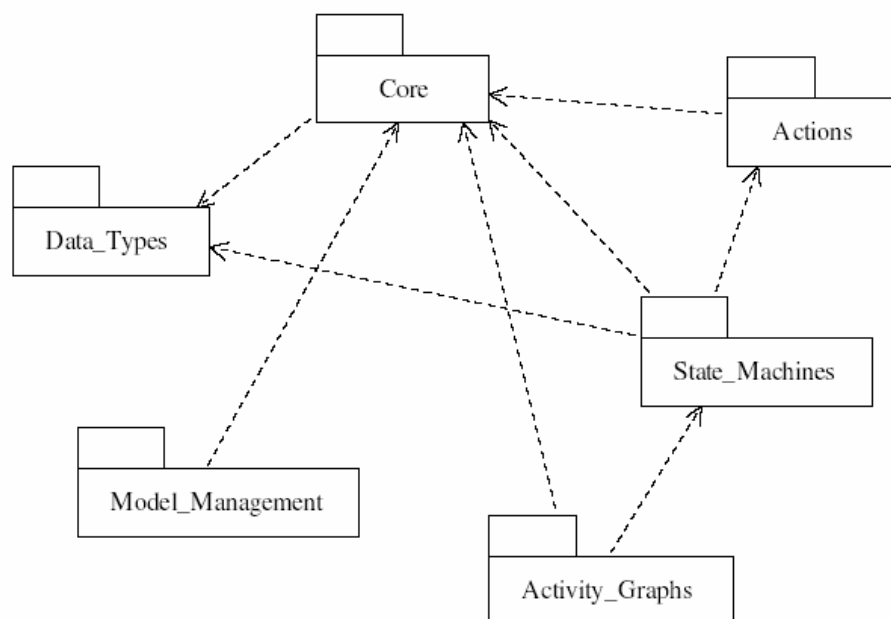


Figure Annexe D.1 : Les dépendances entre sous-paquetages composant le paquetage CPM_Foundation

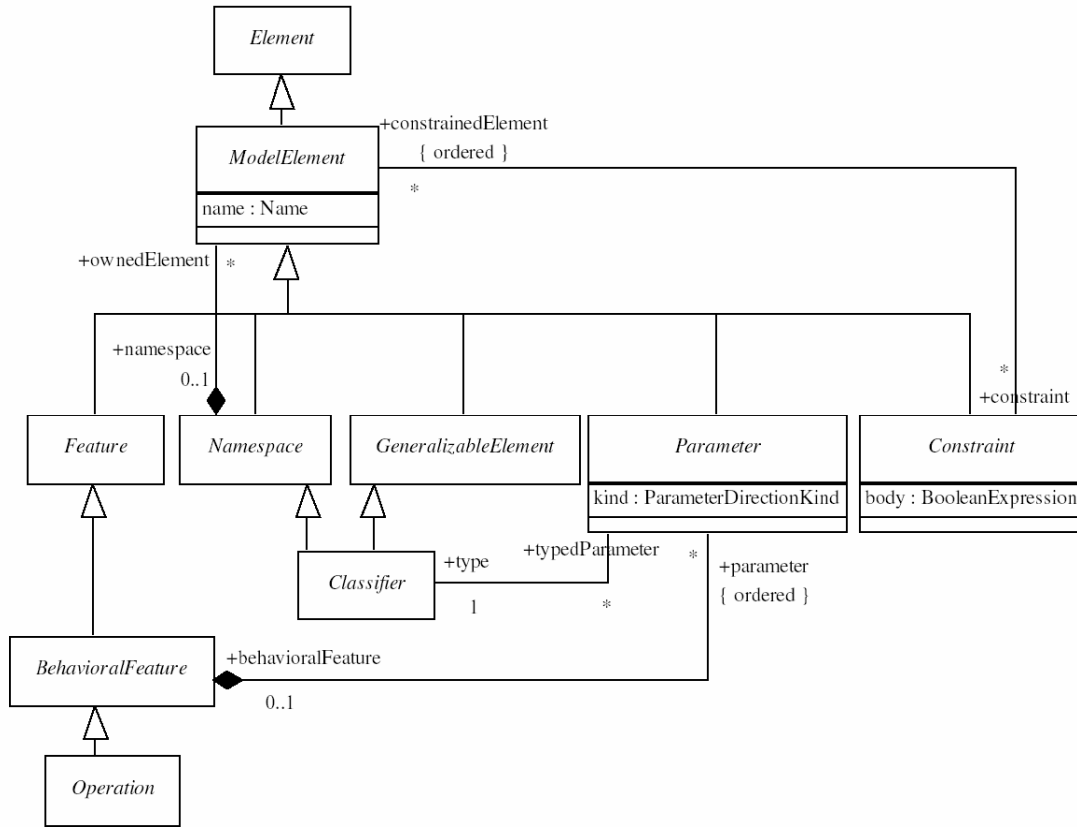


Figure Annexe D.2 : Le paquetage Backbone (constitutif du Core d'UML)

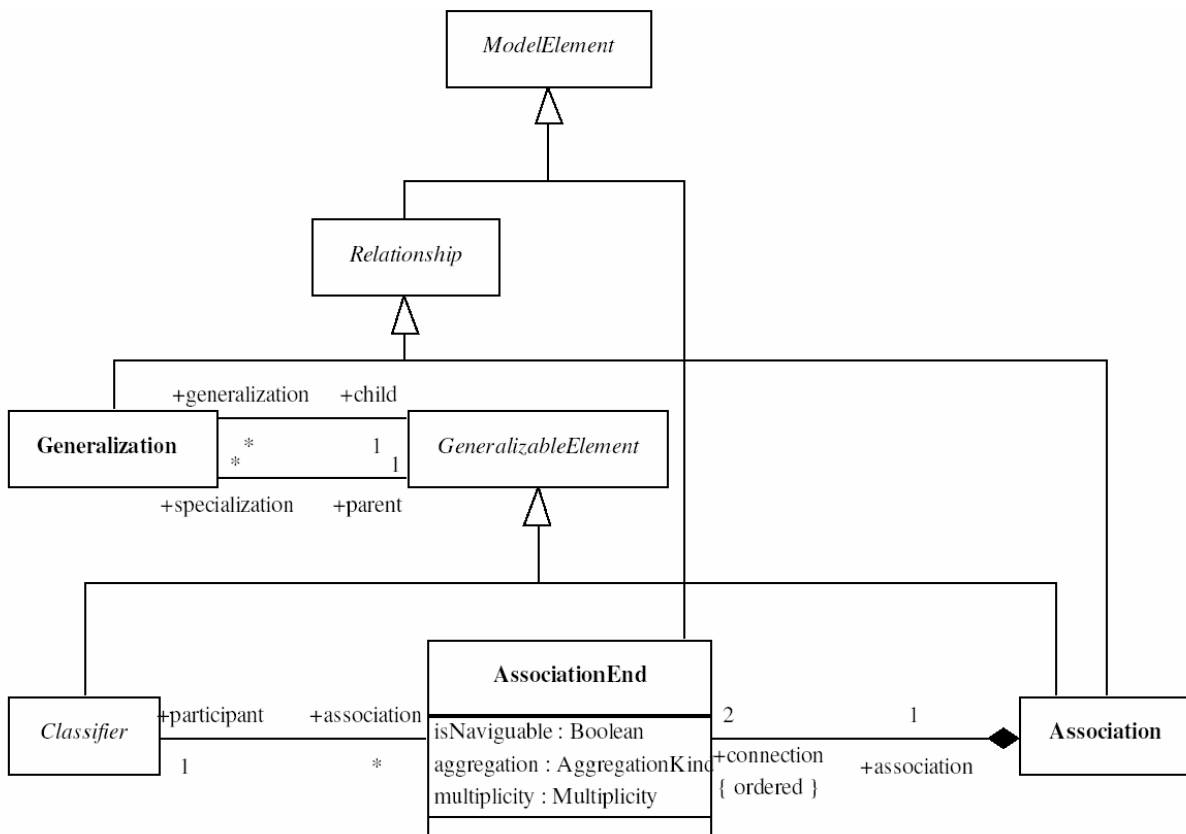


Figure Annexe D.3 : Le paquetage Relationships (constitutif du Core d'UML)

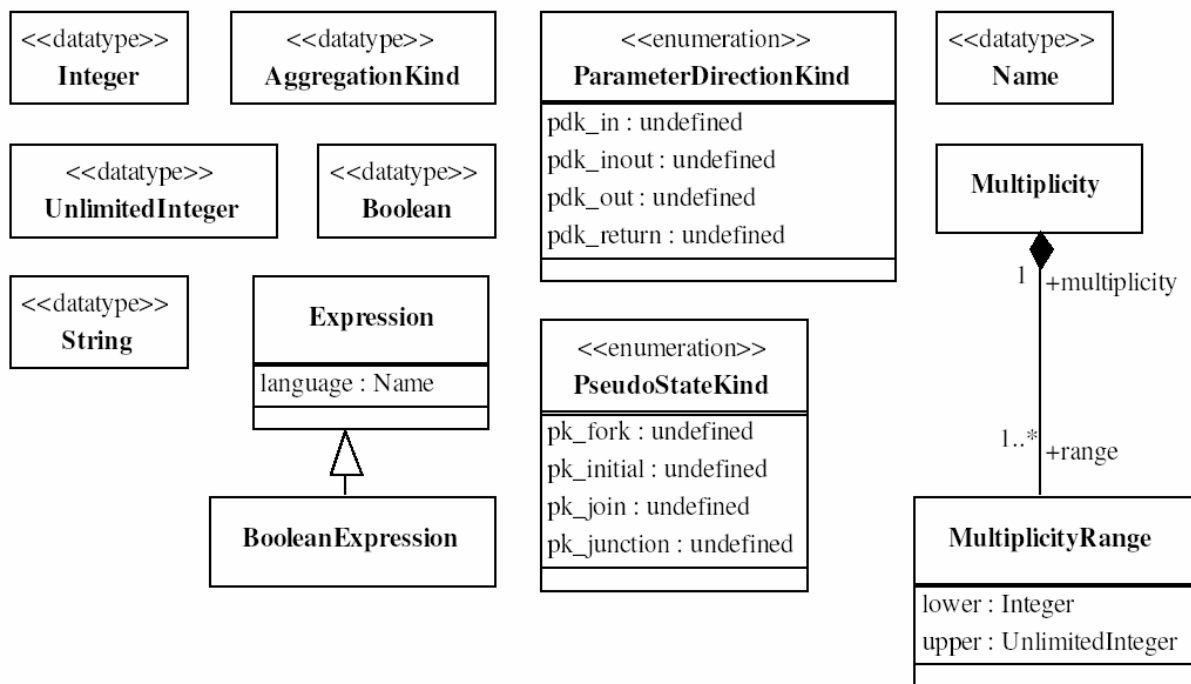


Figure Annexe D.4 : Le paquetage Data_types

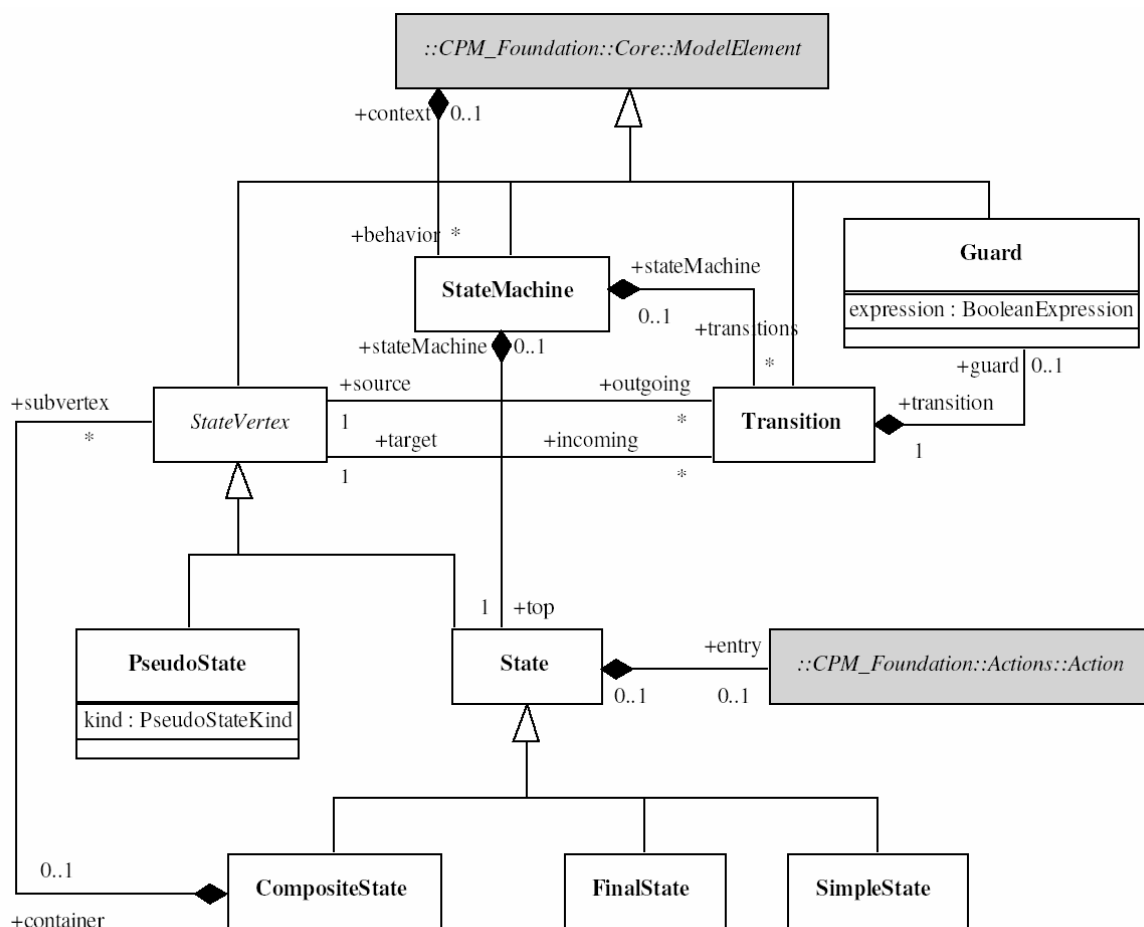


Figure Annexe D.5 : Le paquetage State_Machines

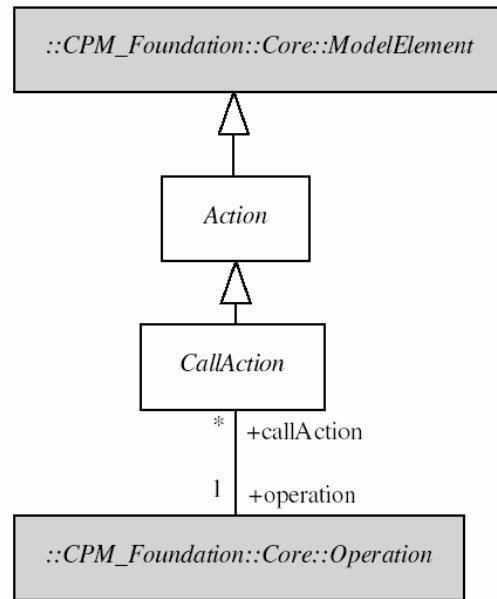


Figure Annexe D.6 : Le paquetage Actions

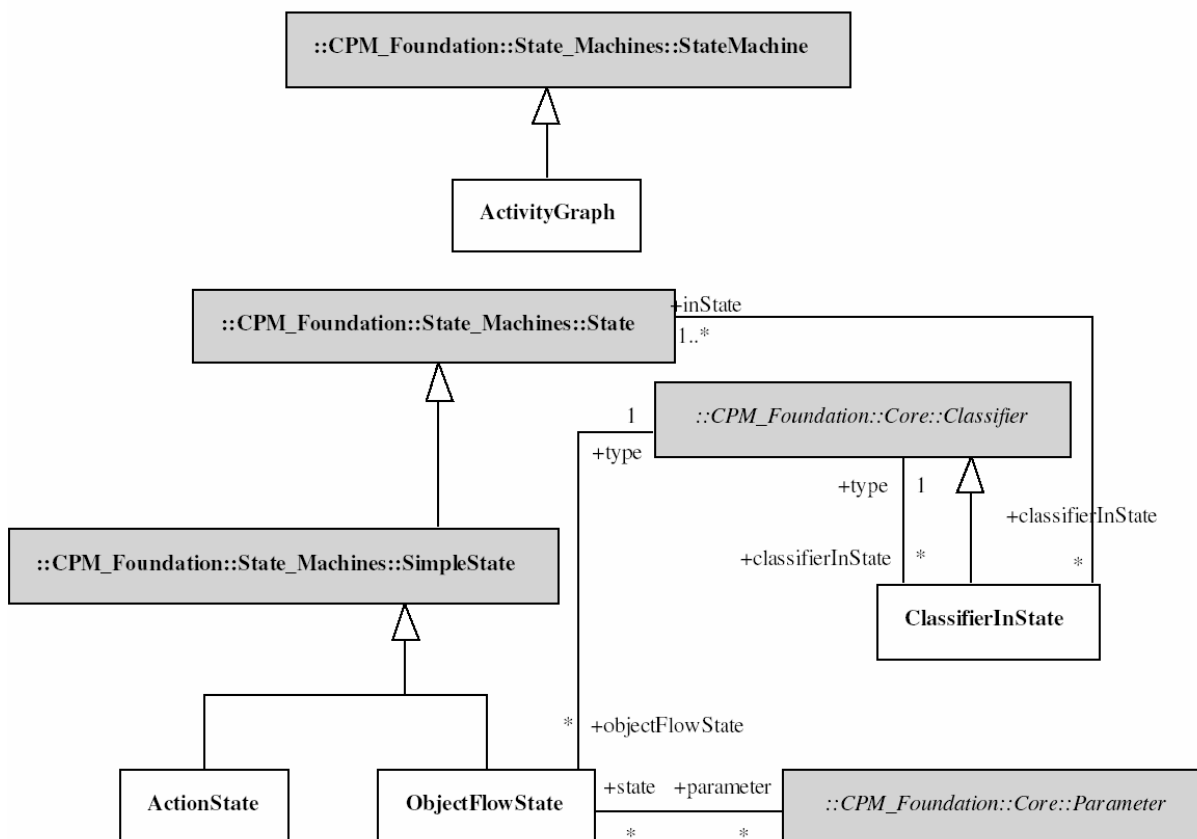


Figure Annexe D.7 : Le paquetage Activity_Graphs

Annexe E

Extraits des modèles réalisés avec le profil CPM pour la situation Smash

Les exemples qui suivent portent sur l'acte 2 de la situation Smash, la thèse de Pierre Laforcade ayant détaillé principalement l'acte 3 de cette même situation (voir chapitre 9 de (Laforcade 2004)). Les premiers diagrammes fournissent un simple rappel du contexte de l'acte 2 qui est détaillé ensuite.

E.1 Définition de la tâche globale et éléments du déroulement de Smash

Enoncé. La Figure Annexe E.1 décrit la tâche globale commune à tous les apprenants en décomposant cette tâche en 4 phases successives (voir cahier des charges de l'annexe B) à l'aide d'un diagramme d'activités. Ce diagramme met en relation les phases avec les critères de succès identifiés ; on peut noter les changements d'état des instances d'*ObjectFlowState* entre l'acte 2 et l'acte 3 qui sont conformes au diagramme d'états (non présenté) des classes *Rapport d'enquête* et *Schéma Accident*.

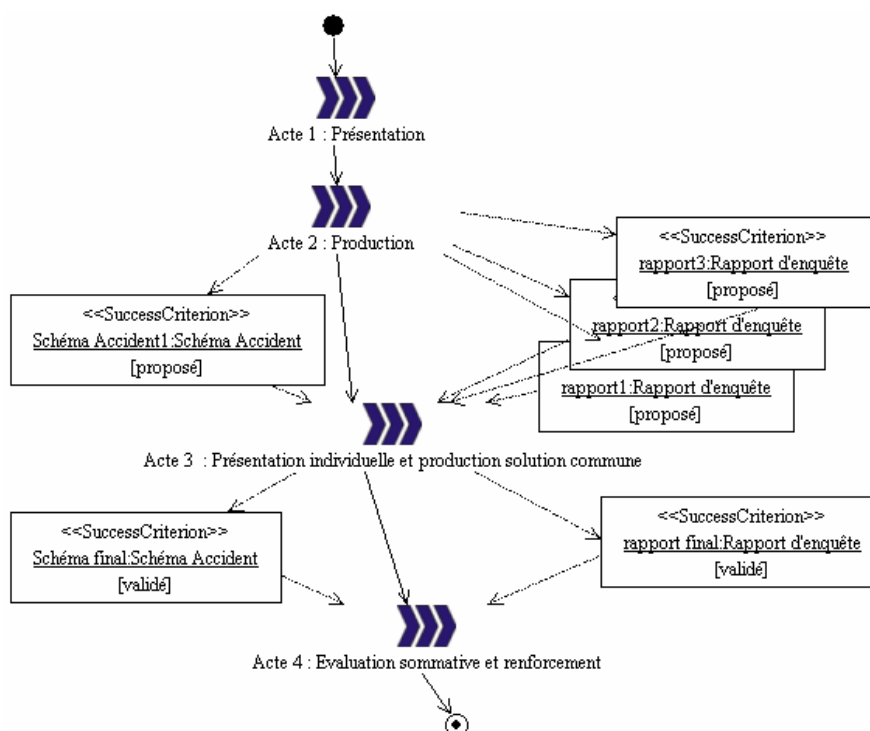


Figure Annexe E.1 : Déroulement général de la situation Smash

Utilisation du langage CPM. Dans la figure précédente, le stéréotype <<LearningPhase>> est utilisé pour les quatre instances d'ActionState du diagramme d'activités ; ces instances sont également marquées par la valeur marquée *phaseKind* avec la valeur *Act* (les valeurs marquées ne sont pas visibles dans la figure). Les transitions entre *ActionState* indiquent la séquence des actes. Les critères de succès sont définis via le stéréotype <<SuccessCriterion>> (attaché à l'élément UML *ObjectFlowState*) et associés aux actes par des transitions.

Sur le même principe, la Figure Annexe E.2 montre le déroulement général de l'acte 2 qui sera détaillé par la suite.

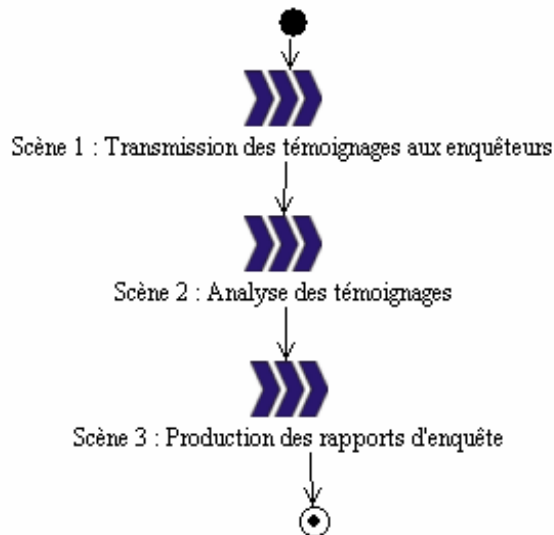


Figure Annexe E.2 : Déroulement général de l'acte 2

E.2 Vers une division du travail

Enoncé. Dans cet acte, les rôles d'enquêteurs joués par les apprenants se décomposent en quatre groupes, le rôle *RôleChefPolice* étant joué par le tuteur. Trois groupes d'enquêteurs ont un rôle similaire (cf les trois instances identiques de critères de succès dans la Figure Annexe E.1), le rôle *RôleEnquêteur 4* étant spécifique (c'est lui qui doit produire le schéma de l'accident qu'on établit couramment lors des constats d'accident). La Figure Annexe E.3 fait apparaître ces activités alors que la Figure Annexe E.4 montre de quelle façon les rôles généraux *Tuteur* et *Apprenant* sont contextualisés lors de l'acte 2.

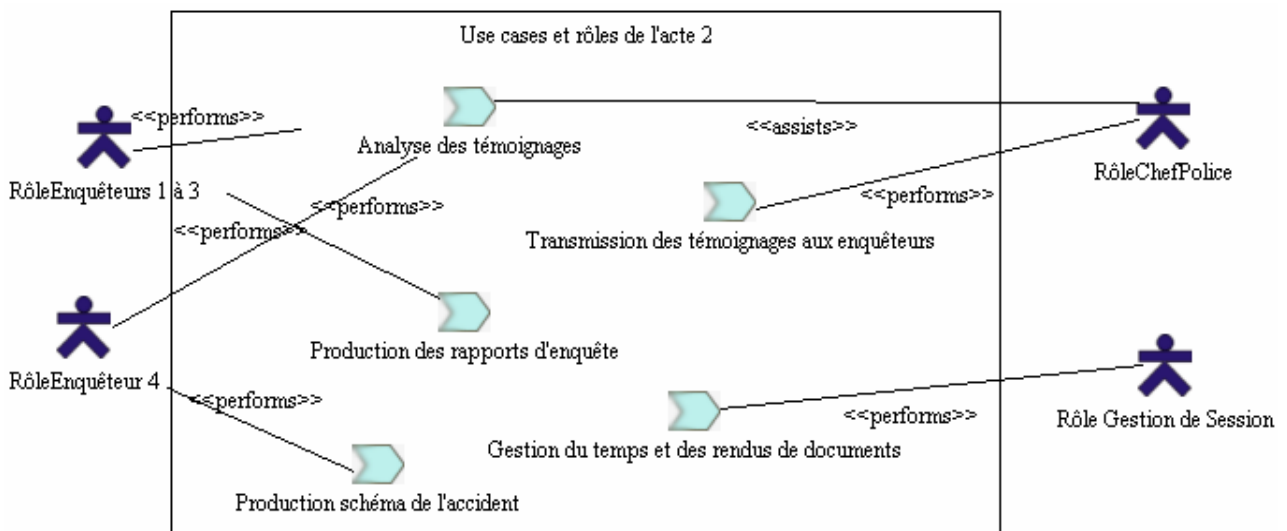


Figure Annexe E.3 : Découpage en activités de l'acte 2

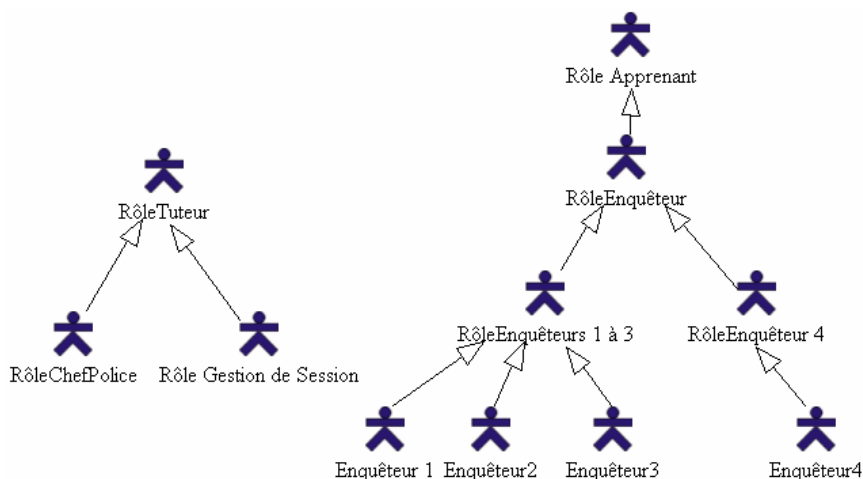


Figure Annexe E.4 : Les types de rôles dans l'acte 2

Utilisation du langage CPM. La Figure Annexe E.3 est un diagramme de cas d'utilisation (la Figure Annexe E.4 est un diagramme de classes utilisant le stéréotype CPM <<Role>>). Les activités sont définies comme des *Use-Case* stéréotypés <<Activity>>. Les rôles sont reliés aux activités par les relations stéréotypées <<assists>> et <<performs>>. Telles qu'elles apparaissent dans le diagramme de la Figure Annexe E.3, les activités ne sont donc pas individuelles et doivent être précisées. Dans cet objectif, il convient de mieux décrire le domaine enseigné.

E.3 Eléments du modèle du domaine

Enoncé. La situation d'apprentissage est basée sur l'existence d'une série de témoignages complémentaires (6 au total) dont l'un est présenté en annexe B. Chacun de ces témoignages apporte de l'information importante sur des acteurs (et leurs véhicules), des lieux, ... pour comprendre le déroulement de l'accident qui est au centre de l'enquête menée. Quelques diagrammes du modèle de ce domaine sont donnés ci-dessous, les diagrammes d'instances de la Figure Annexe E.6, de la Figure Annexe E.7 et de la Figure Annexe E.8 s'appuyant sur le diagramme de classes de la Figure Annexe E.5 :

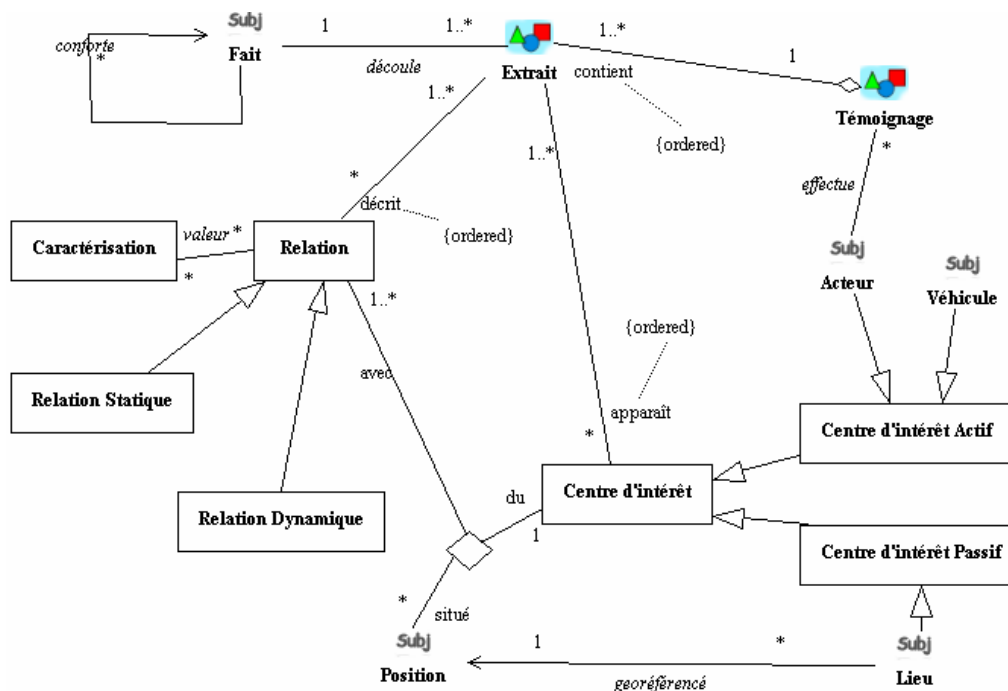


Figure Annexe E.5 : Un diagramme de classes pour rendre compte des extraits des témoignages (Nodenot, Gaio et al. 2005a)

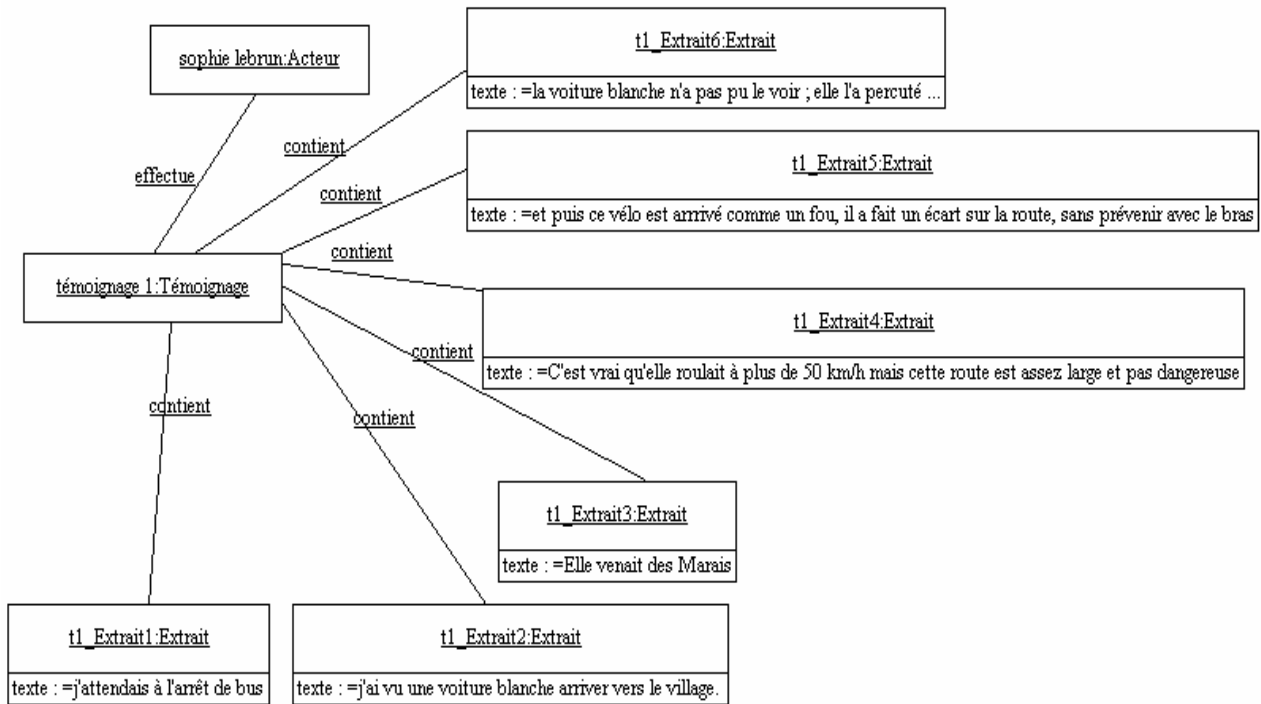


Figure Annexe E.6 : Un diagramme d'instances faisant apparaître les extraits du témoignage 1

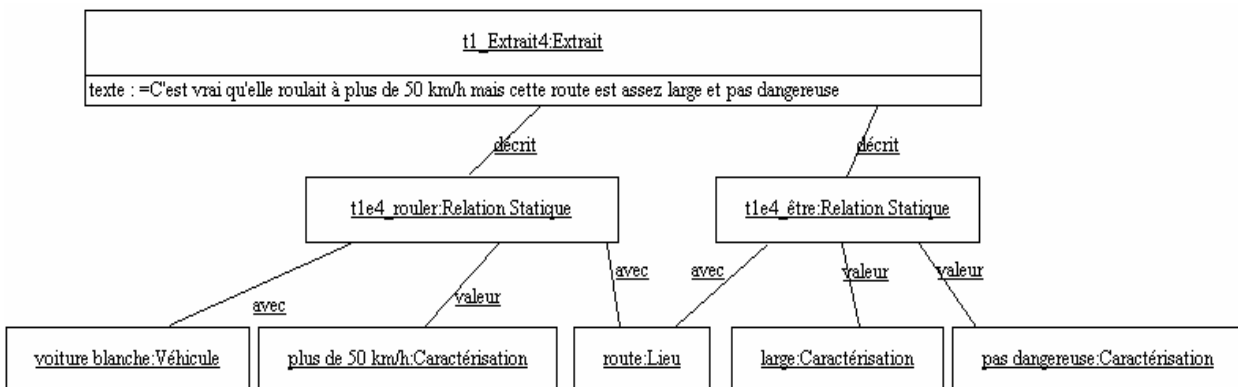


Figure Annexe E.7 : Un diagramme d'instances mettant en relation les éléments de l'extrait 4

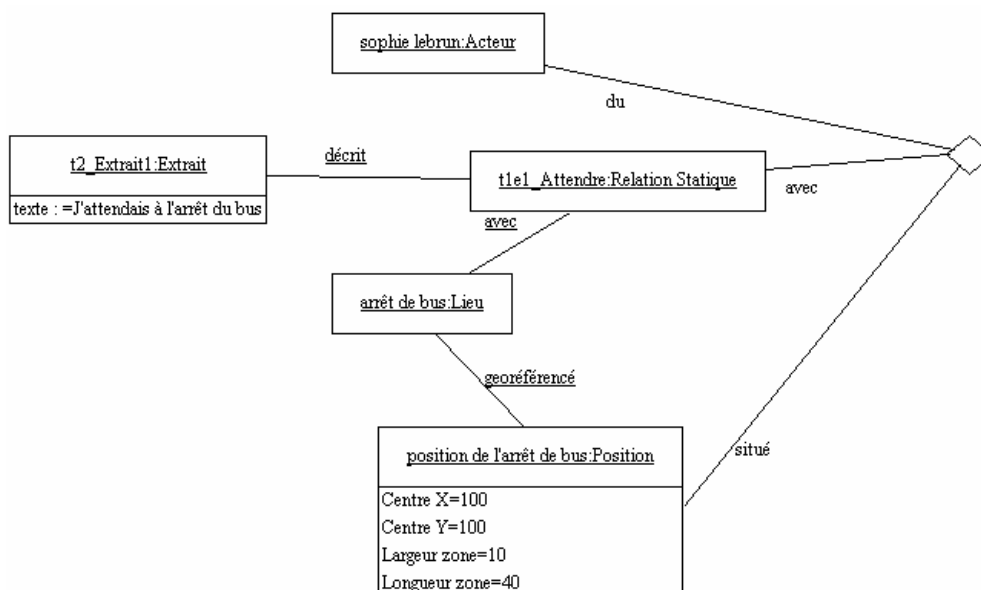


Figure Annexe E.8 : Un autre diagramme d'instances décrivant des éléments géocentriques

Utilisation du langage CPM. Les éléments de la Figure Annexe E.5 apparaissant sous forme iconique représentent des classes qui seront réutilisées dans les diagrammes décrivant la façon dont le scénario d'apprentissage est conçu. Les classes *Extrait* et *Témoignage* sont stéréotypées <<Resource>> car elles désignent des éléments qui seront consultés (exploités) durant l'acte 2. Les classes *Acteur*, *Véhicule* et *Lieu* sont stéréotypées <<Subject>> car elles désignent des éléments significatifs sur lesquels porteront des discussions entre apprenants/des apprentissages durant l'acte 2. Les différents diagrammes d'instances décrivent le contenu des témoignages, ce qui permettra d'exploiter le contenu de chaque témoignage pour évaluer les productions des apprenants (voir notamment les modèles décrivant les diagrammes de croyances des apprenants dans la Figure Annexe E.11) et adapter ainsi le scénario au comportement de l'apprenant.

Enoncé. Pour chacun des sujets *Acteur*, *Véhicule* et *Lieu*, l'analyse effectuée a permis de préciser quelles notions seront au cœur des apprentissages envisagés. La Figure Annexe E.9 circonscrit les comportements à faire analyser aux apprenants : la fatigue au volant, les comportements dangereux et illicites, les problèmes de savoir vivre. Chacun de ces comportements peut ainsi être pointé par des faits s'appuyant sur des extraits des témoignages analysés (l'association entre le sujet *Fait* et la ressource *Extrait* est définie dans la Figure Annexe E.5).

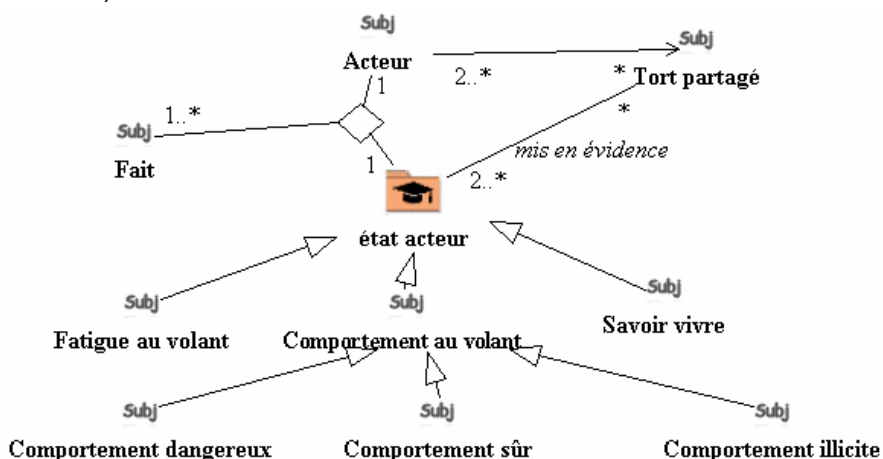


Figure Annexe E.9 : Un diagramme de classes pour décrire les comportements d'un Acteur intervenant dans l'accident

Le diagramme d'instances de la Figure Annexe E.10 montre comment le diagramme de classes précédent est utilisé pour associer certains extraits du témoignage 1 aux comportements que les apprenants vont devoir identifier.

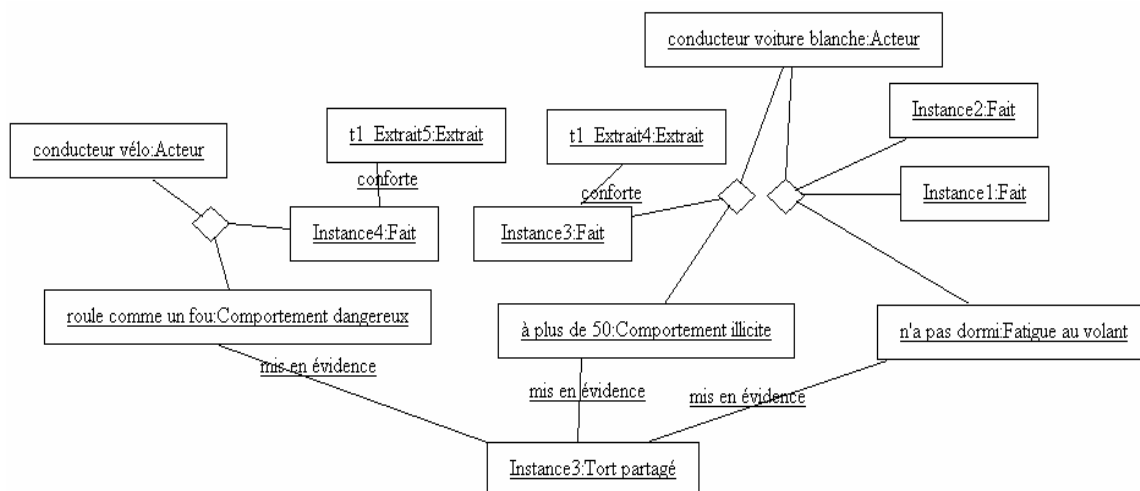


Figure Annexe E.10 : Un diagramme d'instances décrivant les torts partagés du conducteur de vélo et de la voiture blanche

E.4 Eléments du modèle des tâches assignées aux apprenants

Enoncé. Dans la Figure Annexe E.11, les apprenants qui jouent les rôles *RôleEnquêteurs 1 à 3* et *RôleEnquêteur 4* sont amenés à échanger (cf les ressources *Question* et *Réponse*) et à effectuer des déclarations :

1. sur la méthodologie de résolution de la situation-problème (sujets *Problème*, *Hypothèse*, *Solution*).
2. sur leur compréhension du déroulement de l'accident : ils produisent des instances de *Fait* en s'appuyant sur des *Extraits* issus des témoignages.

L'ensemble de ces déclarations constitue un *Graphe de croyances* que le système de tutorat (automatique et/ou humain) va pouvoir confronter aux faits avérés (ceux découlant de l'analyse du domaine présentée précédemment – voir l'exemple de la Figure Annexe E.10).

Le tutorat va s'appuyer sur ce principe de recouvrement entre le modèle de l'apprenant (représenté par le graphe de croyances pour les rôles *Enquêteurs 1 à 3* ou le schéma de l'accident pour le rôle *Enquêteur E4*) et le modèle du domaine.

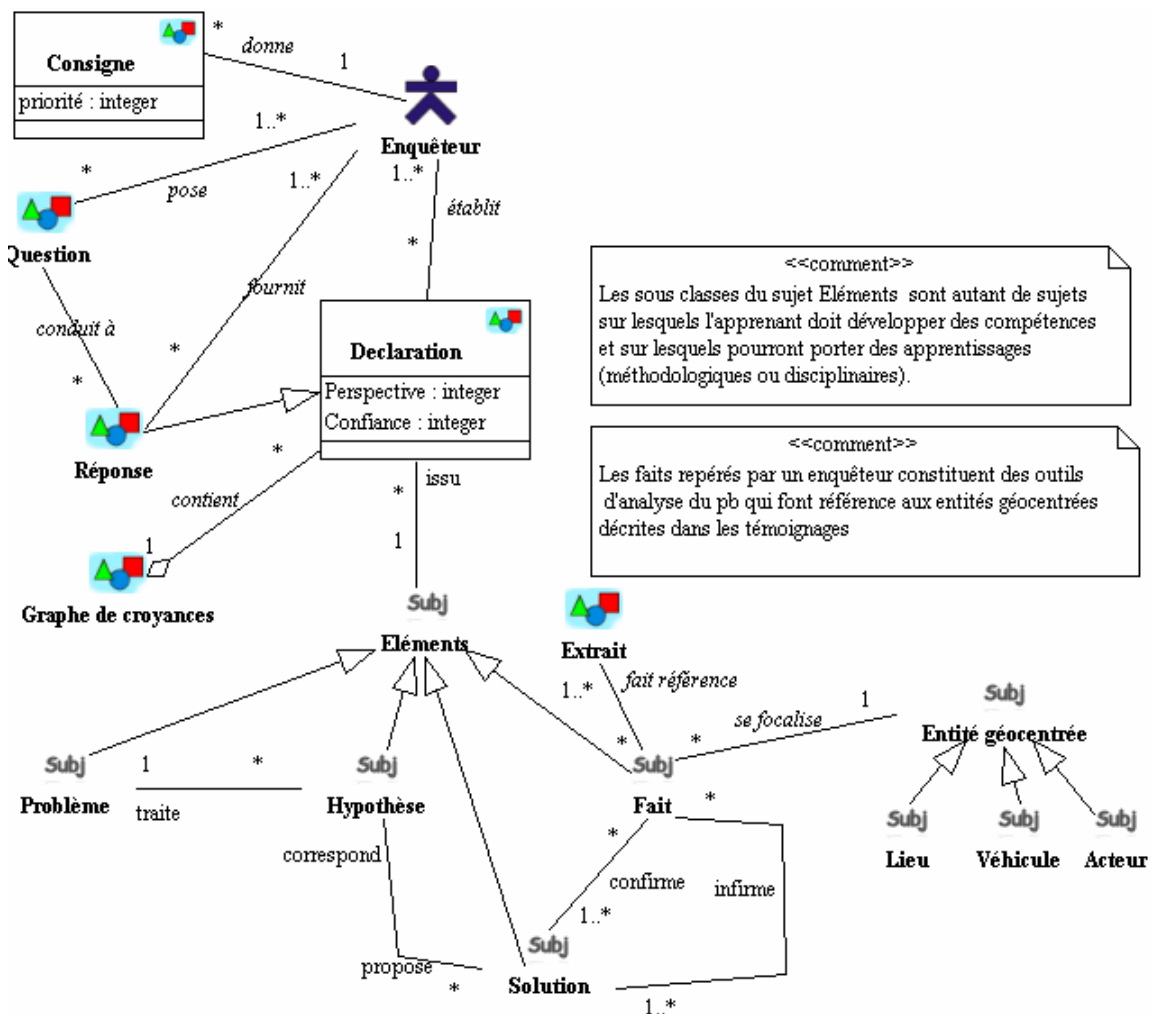


Figure Annexe E.11 : Diagramme de classes pour prendre en compte les croyances des apprenants et les autres ressources qu'ils créent / utilisent durant la scène 2

La Figure Annexe E.12 vient compléter l'analyse en précisant les contraintes que devront respecter les instances de *Question*, *Réponse* et *Consigne* qui seront créées par les enquêteurs.

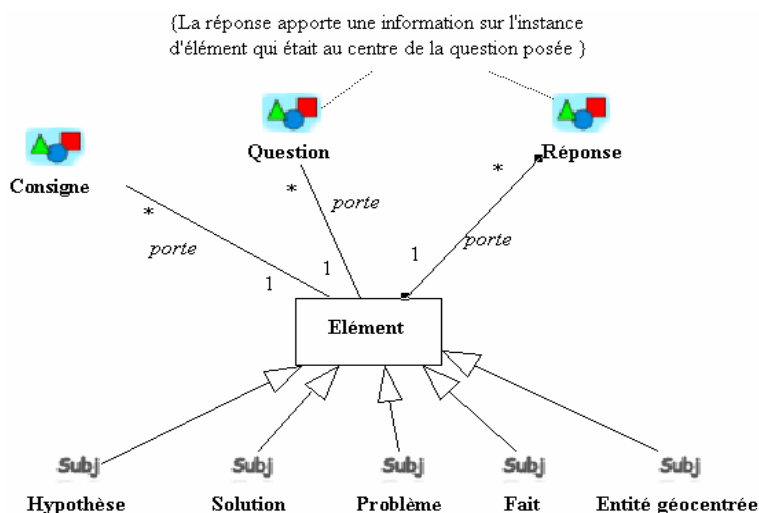


Figure Annexe E.12 : Les contraintes que doivent respecter les instances des ressources

La Figure Annexe E.13 présente les ressources utilisées spécifiquement par le rôle Enquêteur 1 pour mener à bien l'activité *Analyse des témoignages* identifiée initialement dans la Figure Annexe E.13.

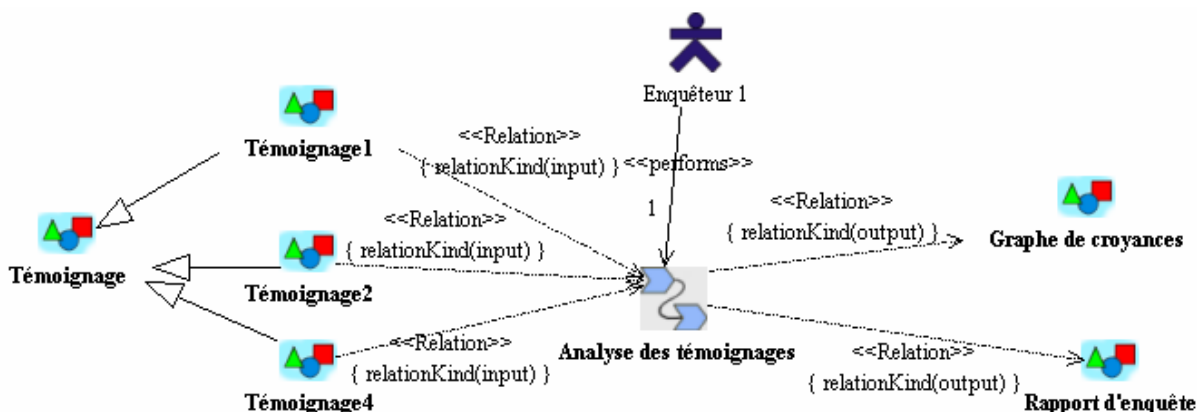


Figure Annexe E.13 : Les ressources affectées au rôle Enquêteur 1

La Figure Annexe E.14 précise la ressource *Rapport d'enquête* qui apparaît dans la figure précédente.

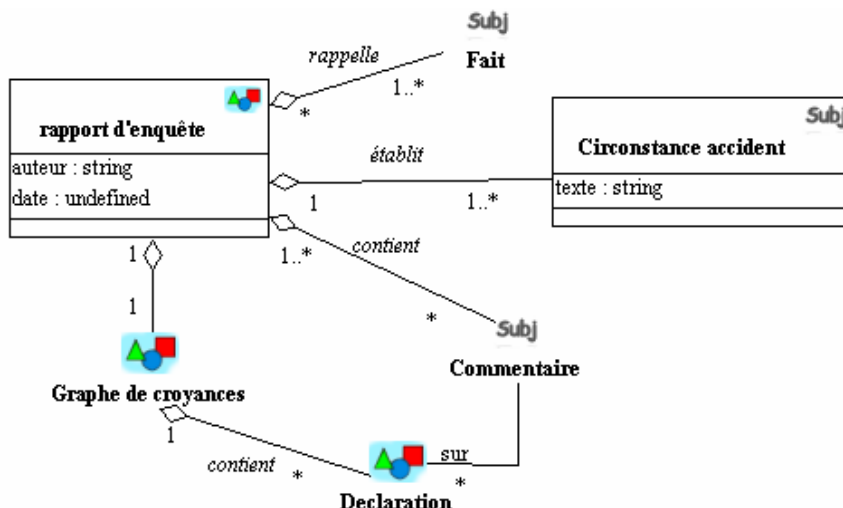


Figure Annexe E.14 : Précisions sur le contenu du rapport d'enquête

E.5 Les fiches de tâches individuelles et collectives

Enoncé. La fiche de tâche individuelle décrit l'ensemble des activités conduites par chaque rôle d'une scène donnée de l'acte 2. Cette fiche de tâche est le résultat d'un processus d'analyse de chacune des activités globales apparaissant dans le diagramme de Use-Case de la Figure Annexe E.3. Chaque activité globale est ainsi représentée sous forme d'une scène étudiée via un ensemble de modèles, cet ensemble conduisant au final à obtenir un diagramme de classes recensant l'ensemble des opérations (au sens UML) concourant à la scène étudiée. Les quelques exemples présentés ci-dessous donnent les fiches de tâches individuelles obtenues pour la scène 2 de l'acte 2 (chacune des opérations apparaissant dans ces diagrammes de classes encapsulant d'autres opérations identifiées lors de l'analyse des activités). La Figure Annexe E.15 présente les fiches de tâches obtenues au final, les autres figures de ce paragraphe correspondant à certains modèles produits pour aboutir à ces fiches de tâches.

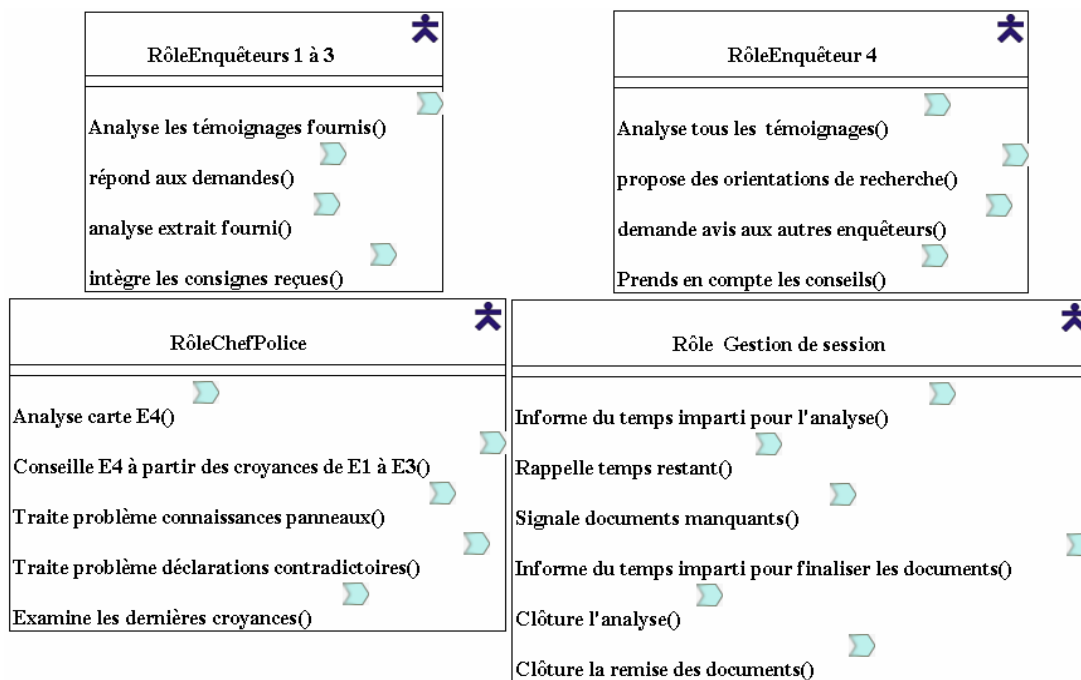


Figure Annexe E.15 : Fiche partielle des tâches obtenues à l'issue de l'analyse des activités

La partie gauche de la Figure Annexe E.16 décrit le scénario général d'analyse des témoignages qui est envisagé, la partie droite de la figure envisageant un scénario nécessitant des compléments d'informations (supports de cours sur la signification des panneaux, prise en compte d'éléments d'analyse contradictoires).

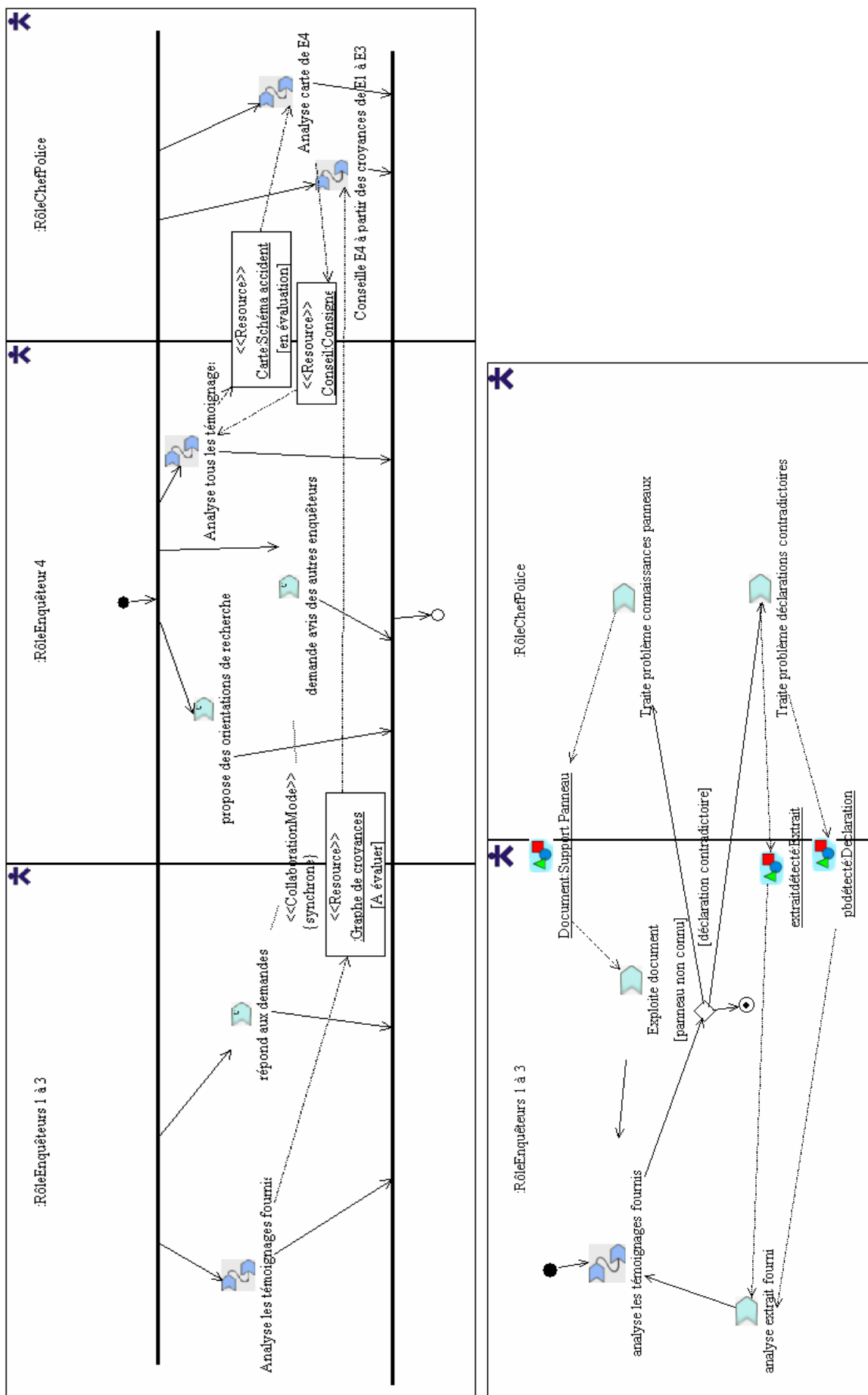


Figure Annexe E.16 : Etude de la scène « Analyse des témoignages »

Les différentes activités de ces diagrammes sont des *SubActivity State* ou des *ActionState* au sens UML. Selon l'activité considérée, ces activités sont stéréotypées différemment : le stéréotype `<<Activity-Structure>>` a été choisi pour pouvoir raffiner l'analyse de certaines activités (voir par exemple l'activité *Analyse les témoignages fournis* et l'activité *Analyse carte E4*) ; le stéréotype `<<Collaborative Activity>>` a été choisi lorsque manifestement l'activité considérée implique un autre type de rôle (voir par exemple l'activité *répond aux demandes* ou l'activité *propose des orientations de recherche*) ; le stéréotype `<<Activity>>` a été choisi dans les autres cas. Les différentes étapes (stéréotype `<<Step>>`), ressources (stéréotype `<<Resource>>`), critères de succès (stéréotype `<<Success Criterion>>`), ... de l'activité considérée peuvent faire l'objet de vues spécifiques afin de compléter l'analyse.

Les diagrammes ci-dessous raffinent l'activité *Analyse les témoignages fournis* de la Figure Annexe E.16 en utilisant des diagrammes d'activité exploitant les stéréotypes CPM.

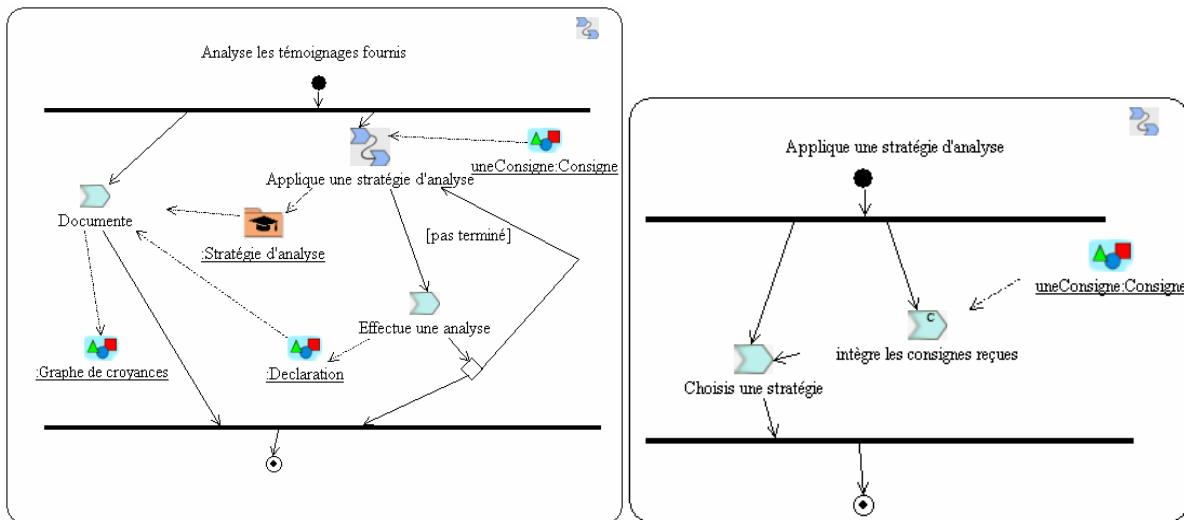


Figure Annexe E.17 : Deux niveaux de détail de l'activité *Analyse les témoignages fournis*

E.6 Prise en compte des activités collectives

Enoncé. Les éléments du langage permettent de préciser les formes de collaboration entre activités. Dans la Figure Annexe E.18, les deux activités qui avaient été mises en évidence lors de l'étude de la scène « Analyse des témoins » (cf partie gauche de la Figure Annexe E.16) sont reprises pour décrire les formes de collaboration souhaitées : droits d'accès sur la ressource *Question*, forme de collaboration envisagée (ici une collaboration synchrone), et rôles des partis impliqués dans la collaboration : l'initiateur des demandes aura accès à des services spécifiques mis à disposition par le composant logiciel (stéréotype `<<Composant CPL>>`) qui supportera la collaboration.

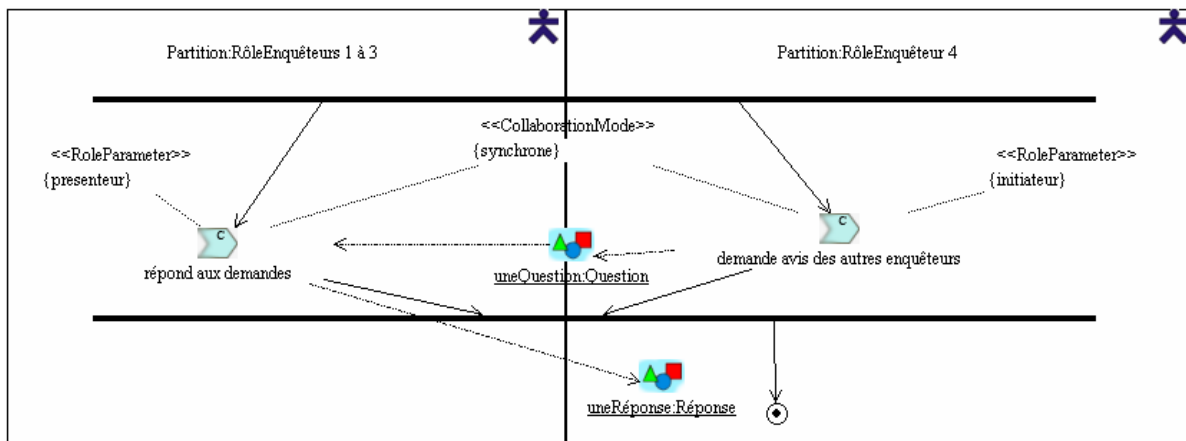


Figure Annexe E.18 : Une collaboration synchrone précisant les rôles des acteurs impliqués

La Figure Annexe E.19 montre un exemple plus complexe mettant en jeu deux collaborations asynchrones : le graphe de croyances mis à jour par le rôle *RôleEnquêteurs 1 à 3* est analysé par le rôle *RôleChefPolice* qui en déduit les extraits de témoignages que le rôle Enquêteur E4 doit approfondir. Ce dernier propose en retour des orientations de recherche aux rôles *RôleEnquêteurs 1 à 3*.

Ce type de modèle ne pouvait pas être produit tant que les classes stéréotypées <<Activity-Structure>> de la Figure Annexe E.16 (partie gauche) n'étaient pas détaillées : l'activité collaborative *intègre les consignes reçues* provient de l'analyse détaillée de l'activité *Applique une stratégie d'analyse* qui est elle-même imbriquée dans l'activité *Analyse les témoignages* (voir Figure Annexe E.17)

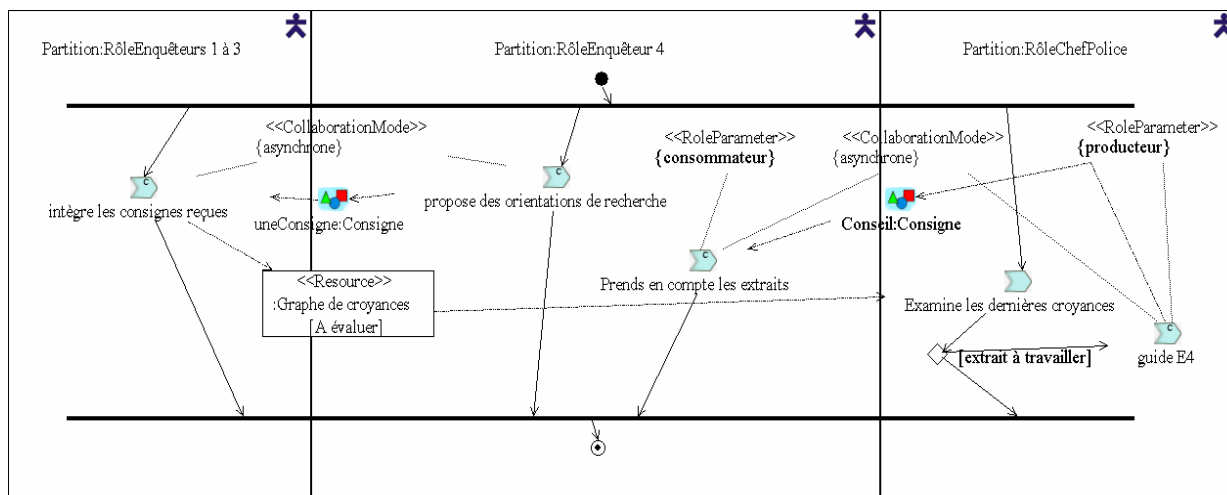


Figure Annexe E.19 : Des collaborations asynchrones précisant le rôle des acteurs impliqués

E.7 Description détaillée des différents états d'une ressource

Enoncé. Comme on peut le voir sur la figure précédente, les diagrammes d'activités proposés exploitent des ressources (cf *Graphe de Croyances*, *Consigne*) qui sont des classes UML stéréotypées <<Resource>> dont les états peuvent être précisés (cf l'état [A évaluer] de la ressource *Graphe de croyances*). Des préconditions et des postconditions peuvent ainsi être placées sur les activités décrites sur la base des états pris par ces ressources. La figure suivante présente un exemple de diagramme d'états pour la ressource *Graphe de croyances*. Les transitions correspondent à des événements (ajout d'une déclaration par un enquêteur) et des actions (opérations) à exécuter (évaluer la déclaration de l'apprenant).

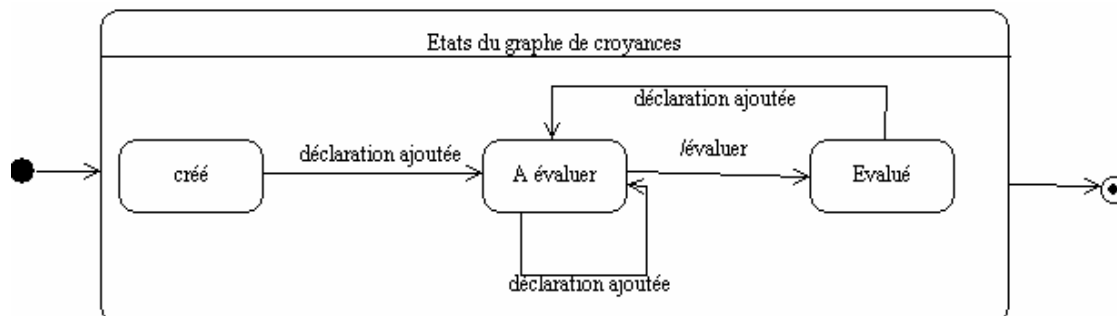


Figure Annexe E.20 : Diagramme d'états de la ressource *Graphe de croyances*

La Figure Annexe E.21 décrit un autre diagramme d'états applicable à la ressource *Question*. Ce diagramme d'état décrit le comportement que doit avoir/devoir (non précisé ici) le rôle *RôleEnquêteur 4* pour conduire le débat avec les autres enquêteurs.

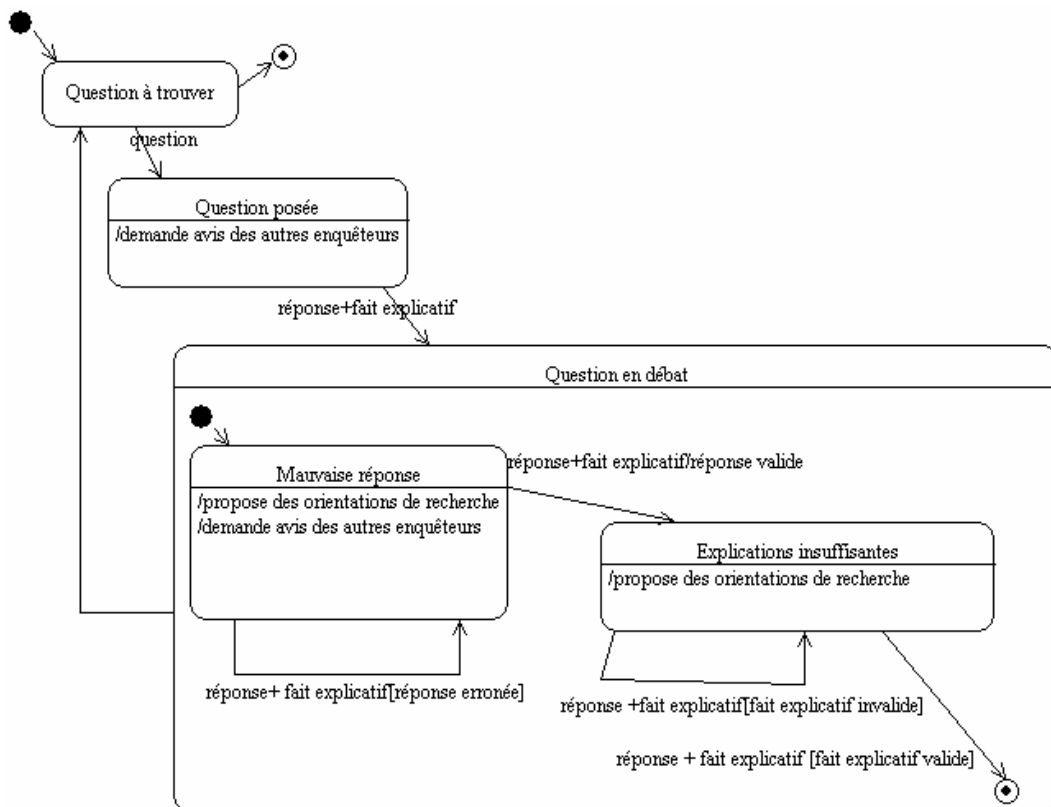


Figure Annexe E.21 : Diagramme d'états de la ressource Question

E.8 Des modèles de situation au modèle de composant éducatif

Enoncé. La Figure Annexe E 22 reprend l'étude des activités collaboratives : *répond aux demandes* et *demande avis des autres enquêteurs*. Chacune de ces activités va mettre à disposition des acteurs concernés un ensemble particulier de services (cf *Interface IDemande* et *Interface IRépond*) fournis par le composant CPL *Question/Réponse_S*. L'interface *IDemande* rassemble les services pour initier une discussion, alors que l'interface *IRépond* permet simplement de participer à une discussion existante (le détail des services offerts pour chacune des interfaces n'est pas détaillé dans cette vue).

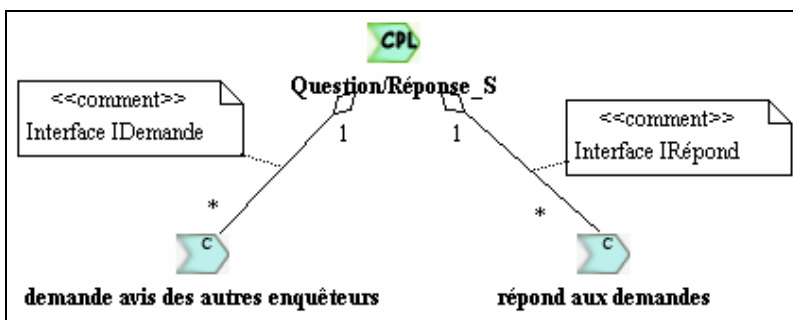


Figure Annexe E 22 : Modélisation des interfaces utilisées par deux activités collaboratives

Ce composant *Question/Réponse_S* exploite un composant de *Chat* tel qu'il peut être proposé par une plate-forme de FOAD donnée. Ce composant de *Chat* est donc spécialisé pour rendre possible les activités collaboratives proposées, comme le montre la Figure Annexe E 23 qui décrit les relations entre ces deux composants. L'analyse s'effectue ici au niveau PIM (Platform Independent Model) et non au niveau PSM (Platform Specific Model).

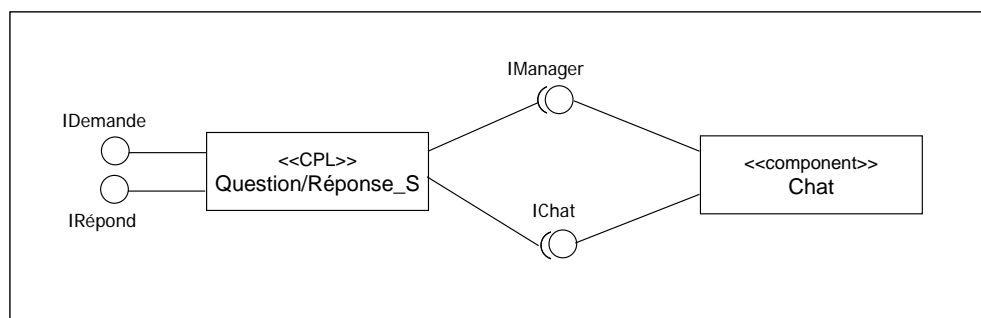


Figure Annexe E 23 : Un modèle décrivant le composant CPL Question/Réponse_S

L'interface *IManager* et l'interface *IChat*, qui permettent respectivement de créer des « salles » spécifiques pour les sessions de Chats et de gérer les échanges entre acteurs dans le cadre d'une session de Chat donnée, sont décrites en détail dans le paragraphe 10.3.1.1 de (Laforcade 2004).

E.9 Organisation du modèle de conception avancé sous Objecteering

Enoncé. La Figure Annexe E.24 illustre l'organisation du modèle de conception. Il s'agit d'une vue de l'organisation interne des éléments qui a montré sa pertinence aussi bien pour la description de l'acte 3 de la situation Smash (voir thèse de Pierre Laforcade) que pour la description de l'acte 2 (dont certains modèles viennent d'être présentés). Cette vue est liée à l'outil que nous avons choisi (l'AGL Objecteering), mais tous les AGL UML existants proposent un *explorateur* de ce type. La figure proposée est constituée de trois copies d'écran :

- Le cadre du haut montre les trois paquetages principaux du modèle de conception que nous avons construit pour la situation Smash :
 - le paquetage *Rôles globaux* décrit les rôles utilisés pour l'ensemble des modèles ;
 - le paquetage *Ressources globales* décrit les ressources utilisées pour l'ensemble du scénario ;
 - le paquetage *Modèles de la situation* correspond à la racine des modèles.
 Un diagramme d'activité modélisant le premier niveau des actes est rattaché au paquetage *Modèles de la situation*. Le détail de chaque acte est ensuite décrit dans un paquetage spécifique.
- Le cadre du milieu correspond à une vue détaillée du paquetage *Acte 2* du cadre précédent. Il est composé de :
 - un paquetage *Ressources locales* qui décrit des ressources utilisées uniquement pour l'acte 2 ;
 - un paquetage décrivant les rôles locaux joués dans cet acte ;
 - un paquetage *Détail des scènes de l'acte 2* qui décrit en détail le niveau inférieur des scènes ;
 - la Classe *Acte 2* à laquelle sont rattachées les scènes sous la forme d'*Operation* et un diagramme d'activité correspondant à la notation *proxy* du découpage en scènes de l'acte 2.
- Le cadre du bas correspond à une vue détaillée du paquetage *Détail Scène 2* du cadre précédent. Il se compose d'une classe *Scène 2* à laquelle sont rattachés :
 - les différents rôles intervenant dans la scène, sous la forme d'*Actor* ; à chaque rôle est rattachée la liste des activités qu'il réalise (sous la forme d'*Operation*).
 - un diagramme d'activité correspondant à la notation *proxy* du découpage en activités de la scène 2.

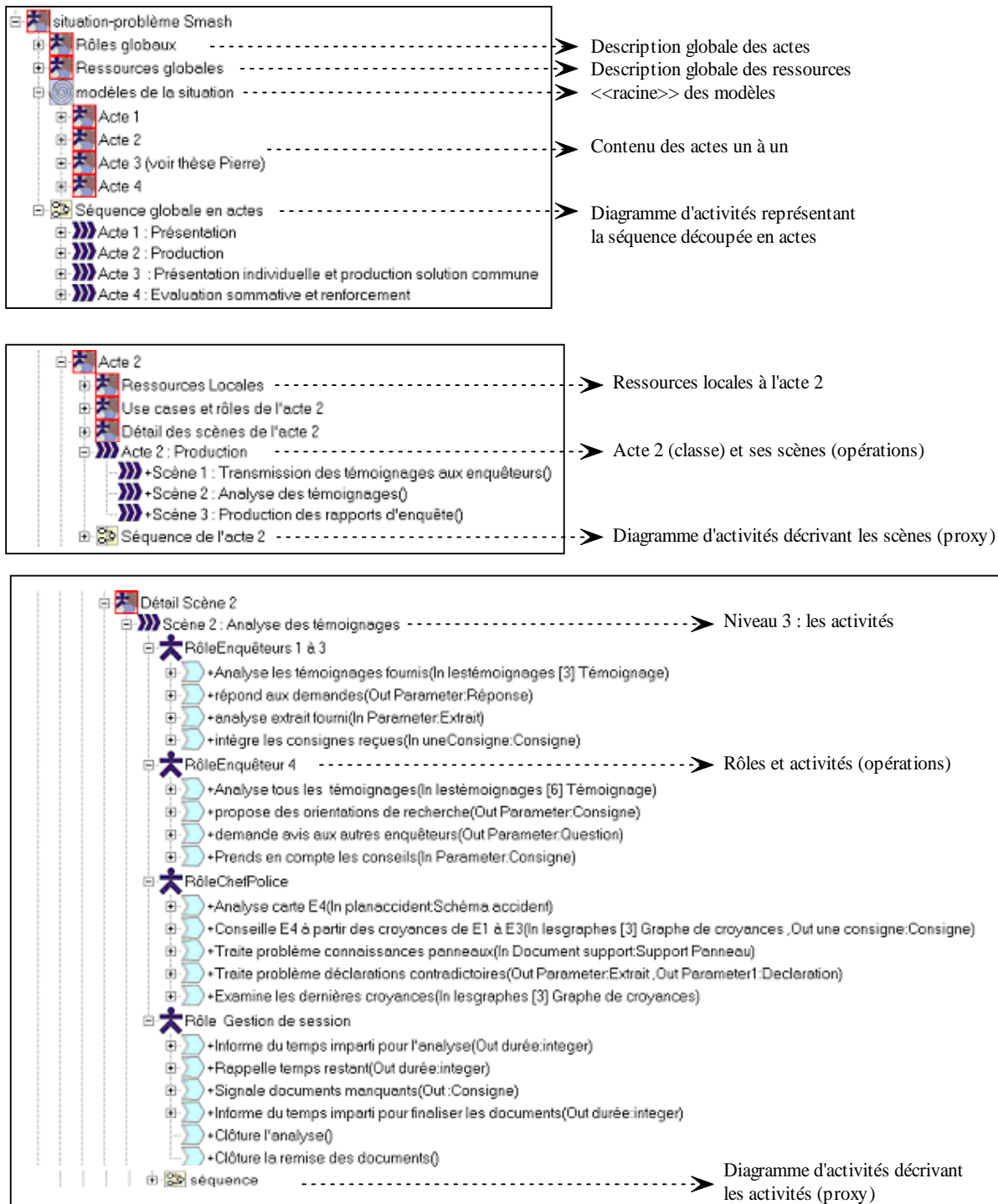


Figure Annexe E.24 : Organisation des éléments de modélisation avec l'AGL Objecteering