



Ingénierie dirigée par les modèles pour la construction de dispositifs pédagogiques sur des plateformes de formation

Pierre-André Caron

► **To cite this version:**

Pierre-André Caron. Ingénierie dirigée par les modèles pour la construction de dispositifs pédagogiques sur des plateformes de formation. Autre [cs.OH]. Université des Sciences et Technologie de Lille - Lille I, 2007. Français. <tel-00156376>

HAL Id: tel-00156376

<https://tel.archives-ouvertes.fr/tel-00156376>

Submitted on 20 Jun 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.

THESE

Soutenue le 18 juin 2007
en vue de l'obtention du grade de

DOCTEUR DE L'UNIVERSITE DES SCIENCES ET TECHNOLOGIES DE LILLE

Discipline Informatique

par
Pierre-André Caron

**Ingénierie dirigée par les modèles pour la construction de dispositifs
pédagogiques sur des plateformes de formation**

Composition du jury

<i>Président :</i>	Lionel Seinturier,	Université de Lille1
<i>Rapporteurs :</i>	Pr Daniel Peraya, Pr Michel Riveill, Pr Pierre Tchounikine	TECFA, Université de Genève Université de Nice-Sophia Antipolis LIUM, Université du Maine
<i>Co-rapporteur :</i>	Dr Mireille Blay-Fornarino	Université de Nice-Sophia Antipolis
<i>Examineurs :</i>	Pr Gilles Leclercq	Trigone, Université de Lille1
<i>Directeurs de thèse :</i>	Pr Alain Derycke,	Trigone/LIFL, Université de Lille1
<i>Co-directeur de thèse :</i>	Dr Xavier Le Pallec,	Trigone/LIFL, Université de Lille1

Remerciements

Je tiens à remercier Monsieur Lionel Seinturier, professeur des Universités au Laboratoire d'Informatique Fondamentale de Lille (LIFL), pour m'avoir fait l'honneur de présider le jury de ma soutenance de thèse.

Je tiens aussi à remercier Monsieur Pierre Tchounikine, Professeur des Universités au Laboratoire Informatique de l'Université du Maine (LIUM), Monsieur Michel Riveil et Me Mireille Blay respectivement Professeur et Maître de Conférence à l'université de Nice Sophia Antipoli, Monsieur Daniel Peraya, Professeur ordinaire à l'Université de Genève (TECFA), tous ont acceptés le défi de rapporter une thèse dont l'ancrage entre deux domaines, les sciences de l'éducation et l'informatique, contribuait à rendre leur tâche très ardue.

Un grand merci à Mr Gilles Leclercq, Professeur des Universités au laboratoire Trigone de l'université de Lille, qui a accepté d'être membre de mon jury et dont la rencontre pendant ma thèse, a grandement influé sur mes travaux de recherche.

Ma recherche ne serait pas ce qu'elle est sans la coopération quasi fusionnelle avec mon encadrant Xavier Le Pallec, Maître de Conférence au laboratoire Trigone. Nos travaux, sans cesse complémentaires, se sont nourris l'un l'autre, depuis mon stage en Master Recherche jusqu'à ma soutenance. De lui, je dois beaucoup, cette page est enfin pour moi l'occasion de le dire publiquement.

Que dire d'Alain Derycke que nous ne sachions tous déjà ? Professeur des universités, co-directeur du laboratoire Trigone, directeur de l'équipe Noce, son humanité, sa direction à la fois exigeante et profondément libertaire en font un directeur de recherche idéal. Ces conseils sont toujours suggestions, accompagnements et occasions d'ouvrir de nouvelles perspectives. Son travail incessant, et sa disponibilité toujours acquise aux doctorants, en font pour moi un exemple à suivre. Je lui suis profondément redevable de m'avoir dirigé pendant cette thèse.

Une thèse c'est aussi une petite parenthèse dans une vie, cet espace temps consacré à ma recherche a été ensolleillée par de nombreuses rencontres faites au sein de mon laboratoire Frédéric, Sébastien, Mona, Jean-Claude, José, Yvan, Thomas et Chantal. Tous à un moment ou à un autre de ma thèse ont contribué à faire en sorte que je puisse aujourd'hui présenter mon travail. Pendant cet espace de recherche, d'autres rencontres ont également influencé mon travail, certains par quelques petites idées bien dérangeantes, d'autre par quelques remarques, tous par leur amitié : merci à Thierry, Pierre, Christophe, Jean-Marc, Elisabeth, Jean Philippe, Jean Pierre, Jean Michel, Vanda, Cyril, Amel, Anne, Christian, Christine, Laurence, Maryline, Bruno, Christophe, Richard et Jacqueline. Merci à tous.

Une thèse c'est un long voyage sur une mer intérieure, on y rencontre des capitaines mais aussi de simples marins qui tentent de naviguer à vue. Je souhaite aussi évoquer quelques compagnons rencontrés dans les ports, certains sont depuis devenus capitaines, d'autre naviguent encore, quelques uns ont disparu en mer...Hello les amis, la terre est proche, je pense à vous, Fatou, Sylvain, Carole, Elise, Vincent, Hassina, César, Alain, Sandra et Ludovic. Enfin, faut il le dire, derrière tout grand chercheur, il y a une femme exténuée, sans qui rien ne se ferait... Merci Delphine pour ta patience et ton amour, merci les enfants d'être tous aussi extraordinaires.

Merci à ma mère, merci à mon père: tu aurais été si fier...

à mes parents.

Résumé

Notre problématique en EIAH concerne la place de l'informatique quant à l'aide qu'elle peut apporter à des enseignants pour accommoder leur enseignement sur des plateformes de formation. Cette problématique, très générale, nous a amené à nous interroger sur l'organisation sociale des enseignants que nous souhaitons adresser, sur les pratiques de création enseignante, sur les objets pédagogiques que ces pratiques nécessitent, ainsi que sur les médias accueillant ces pratiques. De notre analyse nous avons mis en évidence plusieurs points, permettant définir clairement notre cadre de recherche. Notre approche est centrée sur des enseignants appartenant à de petites organisations et ne disposant pas d'ingénieur pédagogique. Ces petites organisations enseignantes pratiquent du bricolage pédagogique sur des plateformes de formations souvent modulaires et open source ou détournées à partir de portails légers.

Une des conséquences de notre positionnement est d'établir une écologie distincte entre les pratiques de créations enseignantes et l'élaboration des plateformes de formation. Notre travail vise à montrer que dans ce cadre restrictif mais réel il est néanmoins possible de définir une ingénierie des EIAHs permettant de coordonner ces deux écosystèmes.

À partir d'une taxinomie des fonctionnalités proposées par les plateformes de formation, nous singularisons, pour notre cadre de recherche, les activités de construction que mène un enseignant. Nous montrons qu'adopter une logique dispositif permet d'aborder un enseignement d'activités situées tout en respectant chez l'enseignant la possibilité de recourir à l'improvisation réglée de son enseignement.

Tisser un lien entre les intentions didactiques d'une équipe d'enseignant pratiquant le bricolage pédagogique dans une perspective artisanale et la réalisation concrète d'un dispositif pédagogique dans le contexte d'une plateforme constitue l'objectif de notre recherche.

Le travail que nous avons mené se distingue des travaux habituellement menés en EIAH et reposant sur l'exploitation et la définition de standards (LOM, SCORM, AICC, IMS-QTI, IMS-LD). Nous montrons la difficulté pour ces approches de prendre en compte des activités situées et l'improvisation enseignante et nous pensons qu'il est possible de leur substituer une approche privilégiant la notion de processus. L'infrastructure que nous définissons favorise un processus de concrétisation permettant de passer de l'expression d'un modèle de dispositif en langage naturel à son expression dans un langage spécifique à la plateforme puis à sa construction dans le contexte de la plateforme.

Pour implémenter cette infrastructure nous suggérons de profiter des possibilités d'automatisation offertes par l'informatique et en particulier celles offertes par une démarche d'ingénierie dirigée par les modèles (IDM). L'IDM est une approche spécifique du génie logiciel qui a pour objectif de définir un cadre théorique pour générer du code en utilisant des transformations successives de modèle. La démarche que nous présentons consiste, au sein d'une plateforme de modélisation, à définir des métamodèles à transformer ou à fusionner. Ces métamodèles permettent d'abstraire les notions nécessaires à l'expression des modèles de dispositifs. La définition de règles de transformations ou de fusion entre métamodèles, puis la construction d'un générateur de constructeur spécifique complète notre infrastructure.

Notre proposition repose en IDM sur l'exploitation des langages spécifiques de domaine, cette exploitation permet d'utiliser des modèles simples facilement appropriables et manipulables au sein d'un éditeur spécifique. Ces modèles peuvent constituer lors de la mise en place d'un processus participatif de création, des objets frontières, supports des tâches de création, de négociation et de résolution de conflits. Nous avons mené au cours de notre recherche une série d'étude de cas permettant de prouver la faisabilité de notre processus instrumenté. Ces études menées sur les plateformes Ganesha, WikiniMST et Moodle ont permis de juger l'adaptabilité du processus aux différents cycles de vie mis en jeu (cycle de création enseignante, cycle de vie des plateformes). Elles ont permis d'affiner notre proposition d'infrastructure en redéfinissant les rôles tenus par les différents acteurs.

Des études de cas ont permis d'élaborer chaque aspect de l'infrastructure que nous proposons, elles constituent une ébauche du travail d'adaptation instrumentale que nous poursuivons dans le cadre de différentes initiatives (PCDAI, Fernahiti, Piste, p-Learnnet, MetaWep).

En conclusion notre recherche a permis de définir un processus instrumenté permettant à une petite équipe d'enseignant de construire facilement des dispositifs de formation sur des plateformes de formations au sens large.

Les apports de notre thèse sont les suivants :

L'analyse des différentes approches ingénieuriste existant actuellement en EIAH, nous a permis de positionner scientifiquement notre recherche, en privilégiant une approche dispositive des plateformes de formation.

Nos travaux en IDM ont permis d'aborder dans des cas concrets les verrous constitués par la définition du modèle d'une plateforme d'exécution et par la construction sur cette plateforme d'un système modélisé.

Les solutions techniques que nous avons élaborées proposent un adressage très simple des plateformes de formation sous la forme d'un greffon de Web Service. Nous montrons dans cette thèse qu'il est techniquement possible via l'ingénierie dirigée par les modèles d'inter opérer avec de tels greffons. Le logiciel que nous avons implémenté GenDep est ainsi en grande partie généré par l'interprétation d'une part du modèle du greffon de Service Web et d'autre part du modèle de la plateforme de formation.

Enfin nous montrons que l'infrastructure que nous proposons est adaptée aux évolutions actuelles des EIAHs. L'usage, en enseignement, d'applications de type Web2.0 banalise des activités de type socioconstructiviste. De telles activités nécessite de pouvoir contextualiser finement les dispositifs à construire, la phase dialogue que notre constructeur implémente permet cette contextualisation.

En EIAHs un nouveau paradigme s'établit, il décrit des approches pédagogiques telle que l'"Evolutionary Learning", le "Pervasive Learning" et le "Micro Learning". Il s'appuie sur des applications hybrides adoptant des modèles de type "Mash-up" et prenant la forme d'assemblage précaire de fonctionnalité Web. Les caractéristiques de notre proposition : approche dispositive, phase de contextualisation et plasticité des protocoles de construction basé sur une architecture service sous-tendue par une approche IDM permettent de relever ces nouveaux défis.

Résumé court

Notre thèse dessine les contours d'une ingénierie des EIAHs adaptée à des équipes d'enseignants pratiquant le bricolage pédagogique dans une perspective artisanale. Elle repose sur infrastructure exploitant les ressources de l'ingénierie dirigée par les modèles. Cette infrastructure est composée d'une plateforme de modélisation et d'un générateur de constructeur spécifique. Elle permet, de modéliser, de contextualiser et de construire des dispositifs pédagogiques sur des plateformes de formation ou de simple application Web. Les concepts et solutions présentés dans cette thèse permettent de relever les enjeux posés par les récentes évolutions en EIAH, tant au niveau des approches pédagogiques, que des médias utilisés.

Table des matières

INGENIERIE DIRIGEE PAR LES MODELES POUR LA CONSTRUCTION DE DISPOSITIFS PEDAGOGIQUES SUR DES PLATEFORMES DE FORMATION	1
REMERCIEMENTS	3
RESUME	7
RESUME COURT	9
TABLE DES MATIERES	11
TABLE DES FIGURES.....	17
INTRODUCTION.....	21
INTRODUCTION	21
ORGANISATION DE LA THESE	22
DIFFERENTES LECTURES PEUVENT ETRE FAITES DE NOTRE THESE.....	25
CHAPITRE 1 LE CONTEXTE	27
1.1 CONTEXTE GENERAL	27
1.1.1 <i>Cadre scientifique de notre recherche :</i>	27
1.1.2 <i>Notre recherche en ingénierie des EIAHs.</i>	28
1.1.3 <i>Les limites de notre recherche.....</i>	29
1.1.4 <i>Des dispositifs pédagogiques</i>	29
1.1.5 <i>Bricolage et conception participative</i>	30
1.1.6 <i>Modélisation et Méta modélisation, MDA et DSML</i>	30
1.2 PROBLEMATIQUE, OBJET DE RECHERCHE	31
1.2.1 <i>Objet de recherche</i>	31
1.2.2 <i>Perspective adoptée, paradoxe, artisanat, ingénierie</i>	31
1.2.3 <i>Problématique.....</i>	32
CHAPITRE 2 ANALYSE DES PRATIQUES SOCIALES DANS LA CONCEPTION D'INSTRUMENTS POUR LES EIAHS	33
2.1 INTRODUCTION	33
2.2 DEFINIR UNE PLATEFORME DE FORMATION PAR CES FONCTIONNALITES	34
2.3 DIFFERENTS SIGLES POUR UNE PLATEFORME DE FORMATION	35
2.3.1 <i>Différents sigles.....</i>	35
2.3.2 <i>Sigles et travaux de recherche.....</i>	36
2.3.3 <i>Applications Web 2.0.....</i>	38
2.3.4 <i>Conclusion</i>	38
2.4 DIFFERENTS ROLES, VISION CENTREE SUR L'APPRENANT, SUR L'INGENIEUR PEDAGOGIQUE ET SUR LE PROCESSUS D'INGENIERIE.	39
2.4.1 <i>Rôles attachés à l'usage d'une plateforme de formation</i>	39
2.4.2 <i>Différents rôles en EIAH vus par Gilbert Paquette.....</i>	39
2.4.3 <i>Perception des rôles par Thomas Vantroys.....</i>	40
2.4.4 <i>Limites de ces perspectives</i>	41
2.5 CONCLUSION ET POSITIONNEMENT DE NOTRE RECHERCHE EN INGENIERIE DES EIAHS.....	42
CHAPITRE 3 PLATEFORMES DE FORMATION ET USAGES DES ENSEIGNANTS AU REGARD DES MODELES PEDAGOGIQUES	45
3.1 INTRODUCTION	45
3.2 DEFINITION.....	45
3.3 COMMENT UNE EQUIPE PEDAGOGIQUE CHOISIT ELLE UNE PLATEFORME DE FORMATION?	45
3.4 LES MODELES PEDAGOGIQUES.	46
3.4.1 <i>Béhaviourisme, associationnisme.</i>	46

3.4.2	<i>Cognitivism</i>	47
3.4.3	<i>Constructivism</i>	47
3.4.4	<i>Médiation et Socioconstructivisme</i>	48
3.5	PLATEFORMES DE FORMATION ET MODELES PEDAGOGIQUES.....	49
3.5.1	<i>LMS et modèles pédagogiques</i>	49
3.5.2	<i>CMS et modèle pédagogique</i>	50
3.5.3	<i>C3MS, outils modulaires et modèles pédagogiques</i>	50
3.5.4	<i>Application Web 2.0 et modèle pédagogique</i>	50
3.6	CONCLUSION.....	51
3.6.1	<i>Limite d'une typologie basée sur les modèles pédagogiques</i>	51
3.6.2	<i>Pas de standard pour construire des dispositifs complexes sur des plateformes de formation</i>	52
CHAPITRE 4 QUELS TYPES DE CONSTRUCTION SUR UNE PLATEFORME DE FORMATION? DIFFERENTES APPROCHES POUR L'ENSEIGNANT.....		53
4.1	L'APPROCHE DOCUMENTAIRE.....	53
4.1.1	<i>LOM Dublin Core education</i>	53
4.1.2	<i>SCORM</i>	54
4.1.3	<i>Limites du standard LOM</i>	55
4.1.4	<i>Limites du standard SCORM</i>	56
4.1.5	<i>Conclusion à l'approche documentaire</i>	57
4.2	L'APPROCHE BASEE SUR UNE METHODE D'INGENIERIE COGNITIVE.....	58
4.2.1	<i>Introduction</i>	58
4.2.2	<i>Une présentation d'IMS-LD</i>	59
4.2.3	<i>Modéliser avec IMS-LD?</i>	60
4.3	L'APPROCHE PAR ACTIVITE, LA NOTION DE SCENARIO.....	61
4.3.1	<i>Introduction</i>	61
4.3.2	<i>Un scénario pédagogique pour Daniel Schneider</i>	61
4.3.3	<i>Un scénario pédagogique pour Rob Koper</i>	62
4.3.4	<i>Un scénario pédagogique pour Gilbert Paquette</i>	62
4.3.5	<i>Analyse de la notion de scénario (Daniel Schneider)</i>	63
4.3.6	<i>Analyse de la notion de scénario (Rob Koper)</i>	64
4.3.7	<i>Analyse de la notion de scénario (Gilbert Paquette)</i>	64
4.3.8	<i>Conclusion de cette approche</i>	65
4.4	OPERATIONNALISATION ET CONTEXTUALISATION.....	65
4.4.1	<i>L'exemple du projet FORMID approche par scénario dans un cadre constructiviste</i>	66
4.4.2	<i>Conclusion</i>	67
4.5	ECRIRE UN SCENARIO PEDAGOGIQUE, UNE ACTIVITE ENSEIGNANTE?.....	67
4.5.1	<i>Introduction</i>	67
4.5.2	<i>Réutiliser, partager?</i>	68
4.5.3	<i>Les limites de la planification</i>	68
4.6	CONCLUSION.....	69
CHAPITRE 5 NOTRE PROPOSITION : DE L'INTERET D'UNE DEMARCHE DISPOSITIVE ..		71
5.1	INTRODUCTION, UNE PRATIQUE PARTICULIERE, L'EXPERIENCE ACQUISE AU SEIN DU LABORATOIRE...	71
5.2	ILLUSTRATION DE LA PROBLEMATIQUE SUR UN EXEMPLE, UN COURS DE REMISE A NIVEAU JAVA	72
5.2.1	<i>Présentation de l'exemple</i>	72
5.2.2	<i>Construction et re-construction du dispositif sur une plateforme de formation</i>	73
5.2.3	<i>La réutilisation du dispositif sur d'autres plateformes</i>	73
5.2.4	<i>Le partage de connaissance</i>	74
5.2.5	<i>Deux études pour valider nos hypothèses</i>	75
5.3	LA NOTION DE DISPOSITIF.....	76
5.3.1	<i>Introduction et définition</i>	76
5.3.2	<i>Dispositif et scénario</i>	76
5.3.3	<i>Ambivalence de la notion de dispositif</i>	77
5.3.4	<i>Conclusion</i>	78
5.4	ADEQUATION DE LA NOTION DE DISPOSITIF A NOTRE RECHERCHE.....	79
5.4.1	<i>Dispositif et bricolage pédagogique</i>	79
5.4.2	<i>Conclusion : un premier cahier des charges de notre proposition</i>	80
CHAPITRE 6 PRESENTATION DE L'INGENIERIE DIRIGEE PAR LES MODELES, LIENS AVEC NOS TRAVAUX EN EIAH.....		83

6.1	QU'EST CE QUE L'IDM	83
6.2	LES CONCEPTS FONDAMENTAUX DE L'IDM	85
6.2.1	<i>Quatre niveaux</i>	85
6.2.2	<i>Le niveau M3</i>	85
6.2.3	<i>Le niveau M2</i>	86
6.2.4	<i>Le niveau M1</i>	87
6.2.5	<i>Le Niveau M0</i>	87
6.2.6	<i>Conclusion</i>	87
6.3	DIFFERENTES RELATIONS.....	88
6.3.1	<i>La relation "représentation de"</i>	88
6.3.2	<i>La relation "être conforme à"</i>	88
6.4	LE MDA.....	89
6.4.1	<i>Présentation générale</i>	89
6.4.2	<i>Généralités sur la transformation de modèles</i>	90
6.4.3	<i>Avantages du processus MDA</i>	91
6.4.4	<i>Critiques à l'encontre du processus MDA</i>	91
6.4.5	<i>Conclusion</i>	92
6.5	D'UNE APPROCHE MDA A LA COMPOSITION DE DOMAINE	93
6.6	CONSTRUIRE UN SYSTEME POUR OPERATIONNALISER UN MODELE	95
6.6.1	<i>Opérationnaliser un modèle en génie logiciel</i>	95
6.6.2	<i>Opérationnaliser un modèle en IDM</i>	97
6.6.3	<i>Dans notre contexte</i>	97
6.7	CONCLUSION DE L'ETAT DE L'ART MODELISATION.	98
CHAPITRE 7 EIAH ET IDM		101
7.1	DIFFERENTES INITIATIVES EN EIAH, REFORMULEES SELON UN POINT DE VUE IDM	101
7.1.1	<i>Une méthode pour aider à la modélisation : Cepiah CEM</i>	101
7.1.2	<i>Le LDL</i>	102
7.1.3	<i>UML et Profil CPM</i>	104
7.1.4	<i>Outils de modélisation reposant sur un métamodèle IMS-LD</i>	106
7.1.5	<i>Ré engineering et Open Distributed Processing Reference Model</i>	106
7.1.6	<i>Conclusion</i>	108
7.2	UNE PREMIERE CONCLUSION, NOTRE SOLUTION AU REGARD DE LA MODELISATION, INTRODUCTION A NOS TRAVAUX DE DEVELOPEMENT.....	108
7.2.1	<i>Positionnement de notre recherche en EIAH, une première synthèse</i>	108
7.2.2	<i>L'ingénierie dirigée par les modèles</i>	109
7.2.3	<i>Les différentes problématiques abordées par nos travaux</i>	109
CHAPITRE 8 PROCESSUS DE BRICOLAGE ET CREATION PARTICIPATIVE POUR UNE APPROCHE PAR CONSTRUCTION DE DISPOSITIF.....		111
8.1	INTRODUCTION	111
8.2	REINTRODUIRE L'USAGER DANS UN PROCESSUS INDUSTRIEL DE CREATION, PLACE DE LA MODELISATION ET RECHERCHE D'OBJETS FRONTIERES.	112
8.2.1	<i>Usage et création participative</i>	112
8.2.2	<i>Objets frontières : modèles</i>	113
8.2.3	<i>Objets frontières : prototypes</i>	114
8.3	PROCESSUS DE CREATION ENSEIGNANTE	115
8.4	BRICOLAGE ET ENSEIGNEMENT	115
8.5	BRICOLAGE ET CONCEPTION PARTICIPATIVE.....	117
8.5.1	<i>Communication et pragmatisme</i>	117
8.5.2	<i>Conclusion</i>	118
8.6	CONCLUSION ET PROPOSITION.....	118
8.6.1	<i>Tensions</i>	118
8.6.2	<i>Un nouveau paradigme pour les EIAHs</i>	119
8.6.3	<i>Notre proposition = deux ateliers : Outillage et Construction</i>	119
8.6.4	<i>Le domaine d'application et d'expérimentation de notre ingénierie</i>	120
CHAPITRE 9 PROBLEMATIQUES IDM ABORDEES PAR NOTRE RECHERCHE		121
9.1	UTILISER L'INGENIERIE DIRIGEE PAR LES MODELES	121
9.1.1	<i>Comment utiliser l'IDM dans le cadre de notre proposition?</i>	121

9.1.2	<i>Place de l'IDM dans l'atelier Outillage</i>	123
9.1.3	<i>En guise de conclusion : "sur l'adéquation de l'IDM à notre proposition"</i>	124
9.1.4	<i>Problématiques IDM évoquées par notre proposition</i>	125
9.2	MODELE DE PLATEFORME ET METAMODELE DE LANGAGE.....	126
9.2.1	<i>Qu'est ce qu'une plateforme de formation au sens de l'IDM</i>	126
9.2.2	<i>Pourquoi réaliser des modèles de plateformes d'exécution en EIAH?</i>	127
9.2.3	<i>Comment définir un modèle de plateforme</i>	128
9.2.4	<i>De bonnes pratiques pour définir un modèle de plateforme en EIAH</i>	129
9.2.5	<i>De la transformation à la fusion de métamodèle</i>	133
9.2.6	<i>Conclusion à la problématique posée par la réalisation d'un modèle de plateforme</i>	138
9.3	CONSTRUIRE UN MODELE SUR UNE PLATEFORME, APPROCHE CONCEPTUELLE.....	139
9.3.1	<i>Introduction</i>	139
9.3.2	<i>Choix conceptuels pour réaliser un générateur de constructeurs spécifiques?</i>	139
9.3.3	<i>Construire un générateur de constructeur spécifique</i>	140
9.3.4	<i>Décrire et factoriser une collection d'éléments?</i>	142
9.3.5	<i>Conséquences IDM, deux interprétations</i>	143
9.4	EN GUISE DE CONCLUSION.....	144
CHAPITRE 10 REALISATION TECHNIQUE : MODELEURS ET CONSTRUCTEURS SPECIFIQUES DE DISPOSITIFS.....		147
10.1	IMPLEMENTER, S'IL N'EXISTE PAS, UN SERVEUR DE WEB SERVICES.....	150
10.1.1	<i>Adresser une plateforme par des services Web</i>	150
10.1.2	<i>Implémenter un serveur de Web Service</i>	151
10.2	A PARTIR DE LA DESCRIPTION TEXTUELLE, REALISER LE METAMODELE INDEPENDANT DE LA PLATEFORME, A PARTIR DES SERVICES ET DES TYPES COMPLEXES REALISER LE METAMODELE SPECIFIQUE ...	159
10.2.1	<i>Une plateforme de modélisation</i>	159
10.2.2	<i>Présentation de ModX</i>	159
10.2.3	<i>Exemple de réalisation d'un modeleur spécifique</i>	160
10.3	DEFINIR LES TRANSFORMATIONS.....	163
10.3.1	<i>Les transformations dans ModX</i>	163
10.3.2	<i>Génération d'un modeleur spécifique</i>	164
10.4	TESTER LA CHAINE D'EDITION EN CREAT ET TRANSFORMANT UN MODELE.....	165
10.5	CREATION DU CONSTRUCTEUR SPECIFIQUE.....	168
10.5.1	<i>Introduction</i>	168
10.5.2	<i>Deux tâches pour implémenter un constructeur spécifique</i>	169
10.5.3	<i>Etablir une première communication entre le constructeur et la plateforme</i>	170
10.5.4	<i>Construire, Recevoir des informations</i>	172
10.6	DEPENDANCE DE CONSTRUCTION.....	173
10.7	EN GUISE DE CONCLUSION A L'IMPLEMENTATION DE GENDEP.....	175
10.7.1	<i>Un identifiant</i>	175
10.7.2	<i>Les applications Web 2.0</i>	176
10.7.3	<i>Des fonctions prévues mais pas encore implémentées</i>	176
10.8	TESTER LE CONSTRUCTEUR SPECIFIQUE.....	176
10.8.1	<i>Modèle de dispositif</i>	176
10.8.2	<i>Paramétrage de GenDep et première communication</i>	177
10.8.3	<i>Créer des collections primaires</i>	178
10.8.4	<i>Créer des collections dépendantes</i>	180
10.8.5	<i>Comportement spécifique de GenDep, et construction</i>	180
10.9	CONCLUSION.....	182
10.9.1	<i>Résumé de notre ingénierie</i>	182
10.9.2	<i>Usage de notre ingénierie</i>	183
10.9.3	<i>La complexité du travail réalisé</i>	183
CHAPITRE 11 DETAIL DES ETUDES DE CAS MENEES.....		185
11.1	INTRODUCTION.....	185
11.1.1	<i>Préambule</i>	185
11.1.2	<i>Présentation des études par ordre chronologique</i>	185
11.2	LA PREMIERE ETUDE : LE TRAVAIL REALISE AVEC LA PLATEFORME GANESHA.....	186
11.2.1	<i>Résumé et liens avec la thèse</i>	186
11.2.2	<i>Problématique de l'étude</i>	186

11.2.3	<i>Présentation d'IMS-LD et intérêt pour notre problématique</i>	188
11.2.4	<i>Notre proposition pour cette étude de cas n°1</i>	188
11.2.5	<i>Réalisation autour du standard IMS-LD</i>	190
11.2.6	<i>Construire un modèle, sur Ganesh</i>	192
11.2.7	<i>Exemple de Mise en œuvre</i>	193
11.2.8	<i>Conclusion et constat de l'étude</i>	196
11.3	LA SECONDE ETUDE : LE TRAVAIL REALISE AVEC L'APPLICATION WEB, WIKINI MST	197
11.3.1	<i>Résumé et liens avec la thèse</i>	197
11.3.2	<i>Cadre scientifique de l'étude</i>	197
11.3.3	<i>Les prémisses de l'étude</i>	198
11.3.4	<i>Le cahier des charges imaginé pour la première phase de l'étude</i>	198
11.3.5	<i>Les motivations de l'approche que nous avons choisie pour la première phase du projet</i>	199
11.3.6	<i>Description de notre approche</i>	200
11.3.7	<i>Les réalisations</i>	201
11.3.8	<i>Usage de l'infrastructure, phase 1</i>	203
11.3.9	<i>Résultats à l'issue de la première phase</i>	203
11.3.10	<i>Une évolution de notre approche</i>	203
11.3.11	<i>La phase 2 de notre étude</i>	204
11.3.12	<i>Les dispositifs de la phase 2</i>	205
11.3.13	<i>Conclusion de l'étude de cas</i>	205
11.4	LA TROISIEME ETUDE : LE TRAVAIL REALISE AVEC LA PLATEFORME MODULAIRE MOODLE	206
11.4.1	<i>Résumé</i>	206
11.4.2	<i>Introduction</i>	207
11.4.3	<i>Mepulco-Université</i>	207
11.4.4	<i>Construction d'un dispositif sur Moodle respectant la méthode Mepulco-Université</i>	209
11.4.5	<i>Automatiser la construction de dispositifs sur Moodle</i>	211
11.4.6	<i>Conclusions et perspectives de cette étude de cas n°3</i>	214
11.5	CONCLUSION	216
	CHAPITRE 12 CONCLUSION, PERSPECTIVES	217
12.1	BILAN DES TRAVAUX REALISES	217
12.1.1	<i>Une approche dispositif des plateformes de formation</i>	217
12.1.2	<i>Une ingénierie basée sur l'IDM</i>	218
12.2	APPORTS DE NOS TRAVAUX DE THESE	219
12.2.1	<i>L'approche dispositif et sa cohérence avec le type d'enseignement que nous adressons</i>	219
12.2.2	<i>Modèle de plateforme et construction de dispositif</i>	220
12.2.3	<i>Réalisation Technique</i>	222
12.3	PERSPECTIVES	222
12.3.1	<i>Le cadre de l'ingénierie Bricoles</i>	222
12.3.2	<i>Un nouveau paradigme en EIAH</i>	223
12.4	CONCLUSION	226
	BIBLIOGRAPHIE	229
	GLOSSAIRE	247
	ANNEXE	253
	ANNEXE A : TABLEAU RECAPITULATIF DES LIENS PLATEFORME MODELES D'APPRENTISSAGES ET SITUATION INITIALE DE NOTRE RECHERCHE	253
	ANNEXE B : MODELES PEDAGOGIQUES	254
	ANNEXE C : TEXTE DE DESCRIPTION DE L'EAPC	256
	ANNEXE D : IMPLEMENTATION DES METHODES "DEPLOY" ET "FETCH"	258

Table des figures

Figure 1 Différents sous domaines permettant l'analyse d'une plateforme de formation.....	34
Figure 2 TELOS 4 niveaux d'organisation.....	37
Figure 3 Exemples d'utilisations de TELOS, différents acteurs.....	40
Figure 4 Rôles et domaines fonctionnels sur une plateforme de formation selon Thomas Vantroys.....	41
Figure 5 Cycle de création d'un objet pédagogique dans le modèle documentaire.....	58
Figure 6 Séquence : création d'un glossaire, activité élémentaire et contexte d'exécution	62
Figure 7 Package IMS comprenant le fichier XML et les ressources.....	62
Figure 8 Exemple de détail de scénario générique (un modèle de l'élaboration d'une banque de ressource).....	63
Figure 9 Caractérisation d'un scénario Daniel Schneider.....	63
Figure 10 Caractérisation d'un scénario Rob Koper.....	64
Figure 11 Caractérisation d'un scénario Gilbert Paquette.....	64
Figure 12 Différentes approche de l'enseignement relativement à la contextualisation, tiré de (Allert, 2004).....	66
Figure 13 Caractérisation d'un scénario Viviane Gueraud.....	67
Figure 14 Dispositif comprenant 7 ateliers accueillant le cours sur Accel.....	72
Figure 15 Caractérisation d'un scénario utilisable pour définir un dispositif.....	77
Figure 16 Deux processus menant à l'opérationnalisation d'un dispositif concret.....	79
Figure 17 Processus de concrétisation	82
Figure 18 Architecture quatre niveaux.....	85
Figure 19 Plateforme de modélisation, support à la vérification de conformité par spécification exécutable.	89
Figure 20 Exemple de cycle de vie MDA pour un modèle Bancaire.....	90
Figure 21 Exemple de transformations dans l'environnement ModX.....	91
Figure 22 Composition de 2 métamodèles et métamodèle des relations	94
Figure 23 Synchronisation de domaine métier et applicatif.....	94
Figure 24 Opérationnalisation par création d'une machine virtuelle.....	95
Figure 25 Processus basique pour développer un programme en utilisant un générateur d'application	96
Figure 26 Appliquer la démarche préconisée (Tolvanen, 2004).....	98
Figure 27 : point de vue IDM de la méthode Cepiah.....	102
Figure 28 Vue simplifiée du métamodèle LDL	103
Figure 29 Paquetages principaux du langage CPM, tiré de (Laforcade, 2004).....	104
Figure 30 Position de CPM dans une vision IDM des EIAHs.....	105
Figure 31 Différents points de vue d'une activité de modélisation tiré de (Corbiere & Choquet, 2005).....	107
Figure 32 Tensions résultant de la difficulté d'adopter des standards en EIAH	118
Figure 33 Présentation de la démarche.....	120
Figure 34 : Solution 1 Utiliser un métamodèle pivot.....	123
Figure 35 Solution 2 : Utiliser des métamodèles pédagogiques ad hoc.....	123
Figure 36 Solution 3 : Utiliser un métamodèle fusionné	124
Figure 37 Problématiques spécifiques à l'IDM	125
Figure 38 Relations entre modèles de plate-forme et métamodèle de système.....	127

Figure 39 Modèle de plateforme = Métamodèle du modeleur.....	128
Figure 40 fichier de description des services encapsulant certaines fonctionnalité de Ganesha	131
Figure 41 : détails du métamodèle Ganesha.....	131
Figure 42 Processus de concrétisation d'un modèle de dispositif, en utilisant des transformations de métamodèles	134
Figure 43 Problème posé, par l'approche par transformation.....	136
Figure 44 : Utilisation d'un métamodèle global	137
Figure 45 Différent niveaux de modélisation.....	138
Figure 46 Problématique et proposition pour relier modèle et système.....	140
Figure 47 Construire un générateur d'application	141
Figure 48 Proposition conceptuelle de générateur de constructeur spécifique.	141
Figure 49 Classe Member	143
Figure 50 génération des squelettes de classe	143
Figure 51 Représentation conceptuelle de l'environnement de modélisation et d'opérationnalisation.....	144
Figure 52 Réalisation technique pour adresser une plateforme de formation.....	148
Figure 53 Description du schéma de production d'une infrastructure adaptée à une plateforme spécifique.	149
Figure 54 Cas d'utilisation simple	152
Figure 55 Cas d'utilisation 2 acteur : user	152
Figure 56 Cas d'utilisation 2 acteur : Admin.....	152
Figure 57 Créer.....	153
Figure 58 Détruire	153
Figure 59 Lister	153
Figure 60 Associer	153
Figure 61 Quatre types d'activité d'administration.....	154
Figure 62 Exemple de type complexe	156
Figure 63 Exemple de classe définie par le Métamodèle.....	156
Figure 64 Métamodèle de Claroline.....	158
Figure 65 Représentation graphique des concepts de modélisation dans le cadre du projet PCDAI.....	160
Figure 66 Métamodèle du WikiniMST	161
Figure 67 détail de la vue "élément"	162
Figure 68 Détail du paramétrage des associations d'une vue.....	163
Figure 69 Exemple de modèle de dispositif PCDAI 1 conforme au métamodèle pédagogique	165
Figure 70 Détails des vues créés après transformation	165
Figure 71 Vue "Basic" du modèle de dispositif, PCDAI.....	166
Figure 72 La vue de type "Owner" représentant la notion "Négociateur" du modèle de dispositif, PCDAI 1	166
Figure 73 Profilage des attributs pour étudiant	167
Figure 74 Exemple d'indications à mettre en attribut.....	167
Figure 75 Schémas de situation du constructeur spécifique.....	168
Figure 76 : Serveur de services Web et WSDL pour l'application Ganesha.....	169
Figure 77 : Détail de l'implémentation des souches de services Web.....	169
Figure 78 Class User dans ModX	170
Figure 79 Implémentation de la classe User	170
Figure 80 Fichier de configuration de ce type de plateforme.....	171
Figure 81 Paramétrage de l'instance de la plateforme Ganesha.	172

Figure 82 Une première communication avec une instance de la plateforme.....	172
Figure 83 Utilisation des méthodes "fetch" et "deploy"	172
Figure 84 Métamodèle fusionné de l'application WikiniMST	173
Figure 85 Exemple de modèle illustrant le problème de dépendance.....	174
Figure 86 Ecriture d'une méthode deploy différé, réalisée par le lien.....	175
Figure 87 Dispositif espace personnel	177
Figure 88 Paramétrage d'une instance de la plateforme	177
Figure 89 Lecture du métamodèle, interprétation du modèle.	178
Figure 90 Génération itérative utilisant un masque de profilage	179
Figure 91 Communication avec la plateforme	179
Figure 92 Génération à partir de fichier CSV	180
Figure 93 Création d'une collection dépendante	180
Figure 94 Création d'un objet unique	181
Figure 95 Construction effective.....	181
Figure 96 Résultat de la construction.....	182
Figure 97 Ré-utilisation d'un cours sur diverses plateformes	187
Figure 98 IMS-LD un langage pivot pour l'interopérabilité des cours sur des plateformes de formation.....	189
Figure 99 Démarche de l'étude Ganesha	189
Figure 100 Compatibilité ModX Reload.....	190
Figure 101 Métamodèle IMS-LD avec grisés les ajouts motivés par la compatibilité avec Reload.....	191
Figure 102 Editeur de pattern facilitant la modélisation IMS-LD	192
Figure 103 Les deux approches explorées par cette étude	193
Figure 104 Diagramme d'activité simplifié de la phase/pièce 'Cours'	193
Figure 105 Diagramme de classe du cours.....	194
Figure 106 Premier métamodèle simplifié de l'application Ganesha.....	195
Figure 107 Modèle du dispositif conforme au métamodèle Ganesha.....	195
Figure 108 Dispositif construit sur la plateforme Ganesha.....	196
Figure 109 Cadre conceptuel, éléments à définir	199
Figure 110 Eléments à réaliser ou à identifier pour le projet PCDAI phase 1	202
Figure 111 Notre proposition pour la phase 2.....	204
Figure 112 Espaces personnels	205
Figure 113 Espace administratif.....	205
Figure 114 Espaces collectifs	205
Figure 115 La partie du cours dédiée au suivi pédagogique	210
Figure 116 Vue générale de l'infrastructure pour un utilisateur.....	211
Figure 117 Concepts de la méthode Mepulco-Université.....	212
Figure 118 Détail du métamodèle de Moodle	213
Figure 119 Paradoxes de notre thèse.....	214
Figure 120 Ecriture de la méthode "deploy" pour un objet de type "User"	258
Figure 121 Détail de l'implémentation de la méthode "fetch" pour la classe "userGenDep"	259

Introduction

"N'importe quel objet, même le plus ordinaire,
enferme de l'ingéniosité, des choix, une culture"
François Dagognet¹,

Introduction

Au cours de ces dernières années, nous avons pu assister à une banalisation de l'utilisation des plateformes de formation. De telles plateformes prennent à nos yeux des formes très diverses : plateforme classique, plateforme modulaire, gestionnaire de contenu, ou même simple application Web éventuellement de type Web 2.0. (O'Reilly, 2005). Cette banalisation est portée par l'accroissement de l'offre de formation à distance, par la multiplication des plateformes, par l'apparition de nouveaux outils, ainsi que par l'essor des applications open source et gratuite (Adkins, 2005). Une des conséquences de cette banalisation a été de déplacer les objets de recherche concernant les environnements informatiques pour l'apprentissage humain (EIAH), de la conception de ces environnements vers l'analyse de leurs usages. Dans un monde où les outils numériques de formation ne sont plus des objets à construire mais des objets à utiliser (donc à comprendre¹), nous nous proposons de dresser un court panorama de l'usage des plateformes de formation dans une vision centrée sur l'enseignant. En décrivant nos objets d'études, plateformes de formation et intervenants, il est possible de définir les activités qu'un enseignant peut mener sur ces plateformes. Ces activités semblent liées à l'organisation sociale au sein de laquelle se situe l'enseignant et à des conceptions diverses de l'acte d'apprendre. Or certaines activités de création menées par des enseignants peuvent être rattachées au cadre théorique du "bricolage enseignant".

Instrumenter le cadre théorique du "bricolage enseignant", en l'appliquant à la création de dispositif pédagogique sur une plateforme de formation constitue l'objet central de notre thèse.

Une telle instrumentation nécessite d'étudier le processus de création enseignant et les pratiques des enseignants sur les plateformes de formation. Une des difficultés de notre travail consiste alors à définir la place exacte de l'informatique dans cette instrumentation. Nous montrons qu'adopter une logique dispositif et faire intervenir l'informatique dans le cadre de l'opérationnalisation d'une telle logique peut permettre aux enseignants de mettre en place une structure ouverte facilitant l'improvisation réglée de leur enseignement. Au sein d'une telle structure des enseignements de type constructiviste et socioconstructiviste peuvent être menés.

Respecter le cadre théorique du "bricolage pédagogique" c'est aussi accepter les exigences induites par ce cadre :

- La première en amont du processus de création enseignant conclue au respect de cette création enseignante à partir de laquelle tout cycle automatisé de construction de dispositif doit débiter. Ce n'est pas aux enseignants de s'adapter aux standards mais au cycle de création de s'adapter aux enseignants.

¹ Éloge de l'objet, Paris, Vrin, 1989, p. 12

- La seconde, en aval du cycle de création, consiste à étudier les plateformes de formation comme des objets naturels et non comme des prototypes dont on peut faire évoluer les propriétés en fonction des fonctionnalités que l'on veut utiliser.

Cette double exigence nécessite pour réaliser l'instrumentation envisagée, de mettre en place un processus de concrétisation. Le but de ce processus de concrétisation est de mener un enseignant de l'expression d'un dispositif abstrait à l'expression d'un dispositif concret dans le contexte spécifique d'une plateforme de formation. Ce processus impose à notre instrumentation une double adaptation, adaptation aux concepts de l'équipe enseignante, adaptation aux concepts des différentes plateformes de formation.

En parcourant ce manuscrit le lecteur pourra découvrir les différentes facettes de notre proposition, celle-ci repose sur une approche centrée sur la construction de dispositifs. Cette approche peut être instrumentée par un processus reposant sur l'ingénierie dirigée par les modèles (IDM).

L'objectif de l'ingénierie dirigée par les modèles est de définir une approche pouvant intégrer différents espaces technologiques. L'IDM permet la manipulation par des métamodèles de modèle productifs par opposition à des modèles descriptifs, contemplatifs. Le "Model Driven Architecture" (MDA) est un exemple d'ingénierie dirigée par les modèles, qui propose un ensemble de standards encadrant ce type d'ingénierie. Le MDA préconise une approche de modélisation basée sur un processus de concrétisation proche de notre recherche en EIAH. Nous proposons pour instrumenter notre approche centrée sur les dispositifs en EIAH, de définir un processus de type MDA dans le cadre élargie de l'IDM. Le processus MDA que notre ingénierie met en œuvre repose sur un générateur de modèleur spécifique ainsi que sur un générateur de constructeur spécifique. Cette ingénierie permet, par des mécanismes de transformation ou de fusion, de définir une infrastructure spécifique adaptée à la modélisation et à la construction de dispositif sur une plateforme de formation.

Les chapitres de cette thèse exposent en détail l'ensemble de notre démarche et de nos apports. Ils sont agencés comme suit :

Organisation de la thèse

La problématique abordée dans cette thèse est complexe : multidimensionnelle, multi-échelle et sociotechnique. Les premiers chapitres de notre thèse (du chapitre 1 au chapitre 4), correspondent à une analyse des EIAHs. Cette analyse va nous servir à construire progressivement notre problématique. En nous appuyant sur des connaissances scientifiques et pratiques, celle-ci sera définitivement constituée à la fin du chapitre 4

Le chapitre 1 présente le contexte de notre recherche, nous y rappelons les différents travaux menés par le laboratoire et situons notre thèse dans leur continuité. Nous expliquons ensuite comment notre recherche intervient au sein des problématiques actuelles abordées par les EIAHs. Nous précisons alors les termes, limites et domaines scientifiques rencontrés dans cette thèse. Nous décrivons l'objet de notre recherche : instrumenter des enseignants pour les aider à concevoir sur des plateformes de formation. Cet objet nécessite de préciser les problématiques intervenant dans la définition des objets, tâches et acteurs impliqués par notre recherche.

Partie 1 : Positionnement de notre recherche en EIAH

Le chapitre 2 s'attache à définir les différents sens que peut prendre une plateforme de formation et les différents rôles que des intervenants peuvent y jouer. Il est ainsi possible d'aborder la notion de plateforme de formation à partir des fonctionnalités qu'elle revendique, des sigles qui permettent de la nommer, des recherches scientifiques dont elle fait l'objet. Ce chapitre nous permet de préciser notre approche centrée sur l'enseignant par rapport à d'autres approches privilégiant l'apprenant ou l'ingénieur pédagogique. Nous définissons ainsi le cadre social dans lequel nous envisageons l'utilisation d'une plateforme de formation : il s'agit de petites organisations enseignantes pratiquant du bricolage pédagogique sur des plateformes de formations souvent modulaires et open source ou sur des applications Web "détournées" à des fins pédagogiques. Une des conséquences logicielles de ce positionnement est d'envisager l'adressage de telles plateformes en s'interdisant une quelconque modification de leur implémentation.

Le chapitre 3 propose d'explorer les conséquences de notre positionnement : Qu'est ce qu'une plateforme de formation pour un enseignant? Quel lien est-il possible d'établir entre un type de plateforme et les intentions didactiques d'un enseignant? Nous avons choisi d'aborder ces interrogations par le biais de l'étude des modèles pédagogiques. Nous présentons donc dans un premier temps les principaux modèles pédagogiques, en précisant le type d'activité qu'un enseignant peut mener sur une plateforme dans le cadre d'un modèle. Nous montrons alors que toute tentative de classification des plateformes de formation selon les modèles qu'elle favorise se heurte à des limites dues au caractère polymorphe de ces plateformes et à la difficulté pour un enseignant de positionner sa pratique selon un modèle pédagogique unique. Ce chapitre permet cependant d'établir un tableau de correspondance entre les fonctionnalités des plateformes et les différents modèles pédagogiques. Avec ce tableau nous pouvons situer plus précisément notre recherche autour de l'activité de construction pédagogique dans le cadre d'activité de type constructiviste et socioconstructiviste.

Le chapitre 4 permet d'explorer la notion de construction pédagogique sur une plateforme de formation. Une telle construction peut être documentaire, cognitiviste ou par activité. Ces trois approches offrent la possibilité à des enseignants de construire des entités très différentes. Ces approches peuvent être reliées aux standards actuels des EIAHs tels que LOM, SCORM et IMS-LD qui seront présentés. Il est également possible de les relier à différents modèles pédagogiques. Il apparaît alors que pour mener un enseignement constructiviste ou socioconstructiviste, ces différentes approches et les standards qu'elles supportent ne sont pas satisfaisants. A l'issue de ce chapitre nous présentons notre proposition : favoriser la construction de dispositifs pédagogiques sur des plateformes de formation.

Le chapitre 5 décrit en détail la notion de dispositif pédagogique. Nous montrons que non seulement il est possible de mener une réflexion sur la notion de dispositif en tant que cadre dans lequel se conçoit et s'exécute un scénario pédagogique mais qu'il est également possible de s'affranchir de la notion de scénario. Nous explicitons alors les trois différents sens que le mot dispositif peut prendre, c'est une offre de signification offerte comme cadre de pensée, c'est un objet frontière et enfin c'est un dispositif technique dans le contexte d'une plateforme de formation. Coordonner ces différents sens peut se faire dans le cadre d'un processus de concrétisation. Nous proposons d'instrumenter ce processus par l'ingénierie dirigée par les modèles et nous établissons un parallèle entre ce processus de concrétisation et le Model Driven Architecture.

Partie 2 : L'ingénierie dirigée par les modèles

Le chapitre 6 permet de décrire les différents concepts sur lesquels l'ingénierie dirigée par les modèles repose. Ce chapitre nous offre la possibilité de présenter quelques plateformes de modélisation et parmi celles-ci, notre plateforme ModX. Cette plateforme peut générer les modeleurs spécifiques permettant d'encadrer les activités de modélisation que nous menons en EIAH.

Nous présentons ensuite l'approche MDA. Le processus de concrétisation qu'elle définit, est instrumenté par des transformations de modèles dont les règles sont définies au niveau des métamodèles. Une telle approche se heurte pour notre recherche aux difficultés que les enseignants rencontrent lors de la manipulation du standard UML. Ces difficultés nous poussent à redéfinir l'approche MDA dans le cadre plus général de l'IDM. Dans le cadre de cette thèse, l'approche MDA est utilisée en relation avec des Langages Spécifiques de Domaine (DSL). Au sein de ces DSLs, opérationnaliser un modèle de dispositif dans le contexte d'une plateforme de formation constitue un enjeu véritable, il entraîne la définition d'une relation de conformité et une opérationnalisation propre au domaine envisagé.

Le chapitre 7 présente différentes initiatives en EIAH, pouvant être reconsidérées selon un point de vue IDM. Ce chapitre permet de situer notre recherche par rapport à ces travaux, abordant la notion de modélisation, sans toujours se situer dans un cadre aussi formel que le notre. A l'issue de ce chapitre nous formulons notre proposition.

Partie 3 : Mise en œuvre de notre proposition

Dans le chapitre 8, nous définissons la notion de processus de conception instrumentale participative et rattachons cette notion aux pratiques enseignantes. Pour montrer cet attachement nous montrons que l'activité créatrice enseignante est proche du bricolage et que celui-ci peut entraîner un processus de conception participative. Pour être mis en place un processus de création participative nécessite d'identifier les objets frontières supports des tâches de création, de négociation et de résolution de conflit. Dans un processus d'ingénierie classique cette place est occupée par les prototypes. Pour un processus de création enseignant caractérisé par un cycle et une finalité différente, les modèles peuvent jouer ce rôle. Cependant l'utilisation de modèles contemplatifs, trop détaillés ou trop techniques présente des limites que nous dénonçons. Notre proposition, inspirée de l'approche IDM, offre la possibilité d'utiliser des modèles simples plus facilement appropriables manipulables au sein d'éditeur spécifique.

Dans le chapitre 9, nous nous interrogeons sur la signification d'une démarche IDM dans un cadre qui ne conduit pas à la production de code sur une plateforme d'exécution parfaitement définie. Proposer une telle démarche dans le cadre des EIAH, c'est en particulier être capable de définir dans ce cadre, la notion de plateforme d'exécution. Une telle plateforme peut alors être reliée à un métamodèle. Celui-ci peut permettre la définition formelle de modèles décrivant des systèmes dans le contexte de la plateforme. Notre recherche propose également une définition de la relation de factorisation qui lie un modèle au système qu'il décrit. Cette définition repose sur des éléments de profilage définis au niveau du modèle. Ces éléments sont interprétés par le constructeur spécifique à la plateforme. Ils permettent la construction, sur la plateforme, des éléments du système modélisé.

Dans le chapitre 10, nous abordons, sur le plan technique et conceptuel, l'instrumentalisation des solutions de métamodélisation, de transformation et d'opérationnalisation que nous avons mis en place dans le cadre d'un atelier permettant l'outillage des enseignants. Cet atelier utilise la plateforme de modélisation ModX ; dans le cadre de notre recherche, ModX, nous permet

de définir des modélisateurs utilisables pour décrire et transformer des modèles de dispositifs pédagogique. Pour opérationnaliser de tels modèles sur une plateforme de formation, il faut alors définir une abstraction des fonctionnalités de construction qu'une telle plateforme propose. Cette abstraction conduit à la définition d'une interface de services réalisée sous forme d'un greffon de services Web. En utilisant les services ainsi définis, il est possible de déduire le métamodèle spécifique à la plateforme de formation. Services et métamodèle sont les paramètres nécessaires à l'outil réalisé pendant cette thèse : GenDep. Celui-ci procède alors à la génération d'un constructeur spécifique à la plateforme. Ce constructeur permet de construire dans le contexte de la plateforme les dispositifs modélisés.

Dans le chapitre 11 nous illustrons notre approche par trois études de cas que nous avons menées. La première a consisté à adapter la plateforme Ganesha. Cette adaptation menée en collaboration avec les concepteurs de la plateforme, a permis de finaliser les protocoles de communication entre le constructeur spécifique et la plateforme. La seconde étude consacrée à l'application Web, WikiniMST, a permis d'étudier les processus de transformation et de fusion à établir entre les différents métamodèles produits. Une infrastructure de modélisation et construction adaptée aux enseignants a été produite ; la prise en compte de la spécificité des application de type Web 2.0 a été abordée. La troisième étude, a concerné l'adaptation d'une méthode de suivi de projets étudiants à la plateforme modulaire Moodle. Cette étude a exploré les limites de la métamodélisation pour rendre compte d'une méthode et a proposé trois possibilités d'adaptations de notre ingénierie.

Dans le chapitre 12 qui constitue la conclusion, nous présentons les principaux apports de notre recherche dans le domaine des EIAHs et de l'IDM. Nous abordons ensuite les évolutions récentes des EIAHs, et nous montrons comment notre ingénierie peut être employée pour aborder ces évolutions. Nous présentons plusieurs travaux en cours adressant les problématiques posées par l'apprentissage pervasive, l'utilisation en enseignement d'agrégations hétérogènes de fonctionnalités Web et de micro-contenus.

Différentes lectures peuvent être faites de notre thèse...

Avertissement

Force est de reconnaître que si une thèse n'est pas toujours facile à écrire, il est également vrai que le lecteur éprouve parfois aussi des difficultés à la lire. Ce petit paragraphe est donc conçu pour en faciliter la lecture.

Nous avons adopté dans l'écriture de cette thèse un parti pris qui peut sembler contestable mais que nous désirons justifier par ces quelques lignes. Le parti pris adopté par la thèse consiste à séparer l'identification des problématiques abordées, les solutions conceptuelles que nous proposons, les solutions techniques que nous élaborons et les études de cas. Cet ordre qui a été repensé après l'écriture initiale de la thèse, a deux justifications.

La première justification réside dans notre volonté de mettre en évidence les rapports qu'entretiennent les notions de dispositifs, les notions de processus de "bricolage" et les notions de processus IDM. L'organisation actuelle de la thèse permet de mettre en évidence ces corrélations tant au niveau des problématiques que des solutions conceptuelles et techniques que nous proposons.

La deuxième justification réside dans notre volonté d'offrir au lecteur un guide pour mettre en œuvre notre ingénierie sur des plateformes de formation. Ainsi à partir du chapitre 8, chacun

des derniers chapitres de notre thèse représente un aspect différent de notre travail. Cette construction motivée par les différentes disciplines abordées par notre thèse, permet de faciliter la consultation ultérieure de celle-ci.

Comme d'autres auteurs l'on tenté avant nous², nous proposons ci-dessous un ordre de lecture différent de notre travail, celui-ci permettant ainsi de mettre en œuvre plusieurs niveaux de lecture.

Pour tous

Les premiers chapitres de notre thèse (du chapitre 1, au chapitre 4), permettent de construire progressivement notre problématique. Le chapitre 11 présente 3 études de cas où nous montrons l'usage de notre proposition. Ce chapitre clef, contient des illustrations simples de notre travail, il peut permettre lors d'une première lecture à comprendre à la fois l'apport de notre thèse en EIAH, et les problématiques abordées en IDM.

Pour le lecteur désirant consulter les différents apports de notre thèse en EIAH

Les dispositifs pédagogiques constituent, dans notre offre d'ingénierie, les objets dont nous souhaitons favoriser la construction. Une définition de ces objets est donnée chapitre 5. Nous détaillons, chapitre 8, le processus de création enseignant. Le chapitre 12 présente les perspectives de notre travail en EIAH.

Pour le lecteur désirant comprendre l'ingénierie qui sous tend notre recherche

Le chapitre 6 présente notre positionnement en Ingénierie Dirigé par les Modèles. Le chapitre 7 présente d'autres initiatives abordant la modélisation en EIAH. Les problématiques abordées par notre recherche sont présentées dans le chapitre 9. Enfin le détail de notre instrumentation est abordé dans le chapitre 10.

² Dans son œuvre "Use of Weapons" (Ian M. Banks, "l'usage des armes", livre de poche n°7189) Ian M. Banks propose une première lecture de son œuvre puis à la fin de l'ouvrage repropose un nouvel ordre de lecture, cette nouvelle lecture changeant le sens de l'histoire qu'il nous conte.

chapitre 1 Le contexte

1.1 Contexte général

1.1.1 Cadre scientifique de notre recherche :

Cette thèse s'est déroulée au sein de l'Equipe Noce (Nouveaux Outils pour la Coopération et L'Education) du Laboratoire Trigone à l'université de Lille I. L'équipe bénéficie d'une expérience d'une vingtaine d'années de recherche dans le domaine du travail coopératif assisté par ordinateur. Deux spécificités de l'équipe NOCE sont à l'origine de la thèse : l'interopérabilité de systèmes malléables et le e-Learning.

Un des constats fait par l'équipe est qu'au fur et à mesure qu'un groupe travaille ensemble, ses besoins évoluent, il faudrait donc que le système qu'il utilise évolue de concert. Cette réflexion l'a amenée à la conception, au développement, au déploiement et à l'expérimentation de plusieurs environnements répartis flexibles d'aide au travail coopératif. Le développement d'une architecture logicielle ODESCA par Frédéric Hoogstoël a permis de créer un environnement d'apprentissage collaboratif assisté par ordinateur nommé Co-learn (Hoogstoël, 1995).

Le travail de Grégory Bourguin (Bourguin, 2000) a permis l'élaboration d'une nouvelle plateforme de Travail Collaboratif Assistée par Ordinateur (TCAO), DARE (Bourguin & Hoogstoël, 1999) "*Distributed Activities in a Reflexive Environment*". Celle-ci, basée sur la théorie de l'activité (Bourguin & Derycke, 2005), autorise une plus grande malléabilité pour l'utilisateur par rapport à ODESCA grâce à un langage de modélisation plus accessible. La théorie de l'activité considère l'activité comme unité de base de l'analyse. DARE est un système réflexif, il permet à l'utilisateur de modifier l'activité à laquelle il participe.

Par ailleurs, toujours au sein de l'équipe NOCE, Thomas Vantroys (Vantroys, 2003) élaborait COW (Vantroys & Peter, 2005), un système de Workflow coopératif et réflexif, dont le métamodèle est très proche de celui du WfMC (Scott, 2003). C'est alors posé très naturellement le problème de synchroniser deux systèmes évolutifs et éventuellement d'établir des passerelles entre les divers environnements. Avec CAST "Creation for Adaptation Services Tool" de Xavier Le Pallec, une solution à ce problème fondamental a été envisagée. CAST (Le Pallec, 2002) est un cadre conceptuel pour la création de services d'adaptation, dont le but est de créer une couche logicielle entre deux systèmes de TCAO. L'objectif est de permettre à l'utilisateur d'un système de TCAO de voir les modèles d'un deuxième système traduit dans le formalisme de modélisation qu'il utilise actuellement. Une réalisation (Le Pallec, 2001) a été implémentée sur le support Meta-Object Facility (MOF) (OMG, 1997). Ceci a nécessité le développement d'un outil de prototypage dédié au MOF. Cet outil a évolué au cours de notre thèse pour finalement être totalement reprogrammé et prendre le nom de ModX. ModX (Le Pallec, Moura, Marvie, Nebut, & Tarby, 2006) est un outil de métamodélisation MOF qui peut éditer tout type de métamodèle et de modèle. Sa principale qualité outre son interface graphique est de permettre le prototypage rapide : ModX est en effet doté de mécanisme de réflexivité qui facilitent grandement le prototypage.

Des travaux ont été également menés au sein du laboratoire dans le domaine de la plasticité (Rouillard, Chevrin, Tarby, & Derycke, 2006), de la mobilité (Chevrin, Derycke, & Rouillard, 2006) et des interactions multicanaux et multimodales (Chevrin, 2006).

Les travaux de l'équipe NOCE se sont alors focalisés sur les Environnements Informatiques d'Apprentissage Humain (EIAH), et nous avons ainsi participé, pendant la durée de cette thèse, à différents projets dans lesquels le laboratoire a été impliqués :

- Le programme de recherche de la nouvelle équipe de recherche technologique éducation (ERTé) Apprentissages en Réseaux.
- Le projet PCDAI (Pratiques Collectives Distribuées d'Apprentissage sur Internet), projet pluri-partenaires financé par la Direction de la Technologie (du MENRT) dans le cadre de l'appel à projet "usages de l'Internet" dont le contrat va de juillet 2004 à juillet 2006. (D'Halluin & Delache, 2006), (D'Halluin & Delache, 2005b), (Leclercq, 2005).
- Le projet de plate-forme nationale pour les (EIAHs) (George & Derycke, 2005), ce projet fait partie du Réseau Thématique Pluridisciplinaire RTP39 "apprentissage, éducation formation" du département STIC du CNRS
- Le Shared Virtual Laboratory (SVL) du réseau d'excellence européen Kaléidoscope. La construction de ce laboratoire virtuel a été débutée par le laboratoire TRIGONE.

L'expérience du laboratoire en modélisation, en construction de plateforme pour le travail collaboratif assisté par l'ordinateur et son implication régionale, nationale et européenne dans des projets liés au domaine des EIAH sont à l'origine des travaux actuels étudiant les apports de la modélisation et de la méta modélisation aux EIAHs dans un contexte éventuellement ubiquitaire. Cette thèse s'inscrit dans la poursuite de ces travaux ; elle explore l'apport de l'ingénierie dirigée par les modèles pour la définition et l'opérationnalisation de dispositif pédagogique sur des plateformes de formation.

1.1.2 Notre recherche en ingénierie des EIAHs.

Le but des EIAHs est de favoriser l'apprentissage humain (Tchounikine, 2002). La caractéristique des EIAHs est de tenter d'atteindre ce but en mêlant deux types d'agents différents, les humains et les machines. Parmi les nombreux travaux qui explorent les différentes facettes de cette relation entre humain et machine dans le cadre des EIAHs, nous avons choisi d'explorer les relations impliquées par la conception de dispositifs pédagogiques par des enseignants et l'opérationnalisation de ces dispositifs sur un EIAH. Nos travaux se situent donc plutôt dans le domaine de l'ingénierie des EIAHs que nous abordons selon une perspective originale. C'est cette perspective que nous allons détailler dans cette section. Notre perspective est centrée sur les activités des enseignants construisant un enseignement sur une plateforme de formation dans une perspective artisanale. Nous pensons et défendons la thèse qu'il est possible dans une telle perspective d'établir une ingénierie des EIAHs.

"L'objet des travaux sur l'ingénierie des EIAH (environnements informatiques pour l'apprentissage humain) est d'étudier les principes de construction des EIAH et de produire un ensemble de méthodes, de techniques et d'outils visant à encadrer et systématiser leur processus de conception" (Tchounikine, 2006)

Nous pensons que produire un ensemble de méthodes, de techniques et d'outils visant à encadrer et systématiser le processus de conception des activités d'un enseignant sur une plateforme de formation, peut être réalisé sans changer la nature profonde de la création enseignante. Dans les sections suivantes, nous présentons quelques points qui permettront au

lecteur de comprendre en quoi nous nous démarquons des précédentes recherches menées en ingénierie des EIAHs. Le premier point que nous abordons concerne la notion de plateforme de formation. Nous abordons cette notion selon une approche très large dans le sens où pour nous tout environnement susceptible d'encadrer des tâches d'enseignement, selon les fonctionnalités qui seront rappelées dans le chapitre suivant, est susceptible d'être considéré comme une plateforme de formation. Les objets dont nous désirons favoriser la construction sont des dispositifs pédagogiques et dans la suite de ce chapitre nous précisons la nature de ces objets. Nous faisons référence dans cette thèse à l'originalité de l'ingénierie enseignante. Cette ingénierie peut être qualifiée d'artisanale et nous rapprochons cette notion de celle de bricolage pédagogique. Enfin nous donnons rapidement quelques clefs concernant l'ingénierie dirigée par les modèles qui permettent de comprendre les concepts que nous désirons mettre en œuvre pour définir notre instrumentation.

1.1.3 Les limites de notre recherche.

Dans le domaine des EIAHs, notre recherche concerne principalement les plateformes pédagogiques de formation accessible par le Web. Or ce terme peut recouvrir différents artefacts. Aussi proposons nous une définition d'une plateforme de formation pédagogique au regard des fonctionnalités qu'elle propose. Une telle définition recouvre différents sigles parmi lesquels : LMS, LCMS, CMS, LKMS, C3MS etc... Nous avons également mené des études de cas, sur des outils très spécifiques, centrés sur l'utilisation de Wikis ou de forums. Pour désigner tous ces artefacts nous utiliserons indifféremment le terme plateforme de formation.

Notre recherche n'est pas centrée sur l'apprenant, bien que nous savons toute l'importance d'une telle démarche, mais elle est centrée sur l'enseignant et les rapports que celui-ci entretient avec l'artefact plateforme de formation. Trois types d'acteurs ont participé à notre recherche : enseignants-chercheurs impliqués dans le projet PCDAI, ingénieurs pédagogiques et informaticiens. Nos outils ont été utilisés pour des formations distantes, au présentiel inexistant ou réduit.

Enfin les objets dont nous avons souhaité favoriser la construction sur une plateforme sont essentiellement des dispositifs pédagogiques, Nous ne focalisons pas notre recherche sur les objets de connaissances que ces dispositifs peuvent mettre en oeuvre. La notion de dispositif est dans notre recherche une notion fondamentale dont nous rappelons dans la section suivante les principales caractéristiques.

1.1.4 Des dispositifs pédagogiques

Un dispositif pédagogique tel qu'il est décrit dans notre thèse est un objet offrant au moins trois facettes différentes. Lorsqu'il est conçu indépendamment de l'environnement qui le supportera, c'est une "*offre de signification proposée à des usagers*" (Leclercq, 2005). Cette définition permet de mettre l'accent sur le caractère subjectif et évolutif d'un dispositif. En particulier, un dispositif pédagogique n'est pas neutre, sa description tend à souligner les intentions pédagogiques qui sous-tendent sa conception.

Cette offre peut être *spécifiée pour une plateforme donnée*. Le dispositif devient alors un objet frontière au sens de (Star, 1989). Pour l'enseignant, le dispositif constitue l'écorce de son scénario pédagogique, qu'il soit défini scrupuleusement ou simplement présumé. Pour l'ingénieur pédagogique, le dispositif est un ensemble de fonctionnalités qu'il doit parvenir à réifier sur une plateforme. Pour l'informaticien, le dispositif constitue un système cohérent avec les fonctionnalités ou les services qu'il a implémentés.

Les éléments de cette offre renvoient alors à des éléments d'un *dispositif concret dans le contexte d'une plateforme*. Un dispositif sur une plateforme de formation est constitué d'un ensemble d'éléments, construit par la plateforme et pouvant mettre en œuvre le scénario pédagogique de l'enseignant. Par exemple la création de différents espaces d'expression sur une plateforme (blog, forum, wiki) peut permettre de mener à bien une pédagogie impliquant des travaux de groupe.

Nous verrons par la suite que ces trois facettes d'un dispositif pédagogique, trouvent dans notre thèse un écho en ingénierie dirigée par les modèles et qu'il est possible d'établir un lien entre les trois temps qu'elles suggèrent et les différentes phases d'un processus de modélisation MDA.

1.1.5 Bricolage et conception participative

Contrairement aux perspectives prônant une approche industrielle de l'ingénierie pédagogique (De la Teja, Lundgren-Cayrol, & Paquette, 2006), (Bachimont & Crozat, 2004), (Koper & Tattersall, 2005), nous désirons mettre en place une instrumentation permettant l'accompagnement du bricolage enseignant dans une perspective artisanale (McCullough, 1998). Nous verrons dans la section suivante que ce choix implique de placer comme a priori à toute ingénierie pédagogique le respect des choix des enseignants aussi originaux soient-ils, et de considérer les plateformes de formation comme des objets finaux éventuellement adaptables et modifiables mais dans les limites d'une activité artisanale. Nous montrerons dans cette thèse qu'il est possible de mettre en place une telle instrumentation, celle-ci ne repose pas sur la définition de standards normatifs, mais sur les avancées récentes de l'ingénierie dirigée par les modèles.

1.1.6 Modélisation et Méta modélisation, MDA et DSML

L'Ingénierie Dirigée par les Modèles (IDM), propose une approche basée sur les modèles permettant d'intégrer de manière homogène différents espaces technologiques (Bézivin, Blay-Fornarino, Bouzeghoub, Estublier, & Favre, 2005) Au sein du laboratoire Trigone notre expérience en ce domaine concerne essentiellement les plateformes de modélisation et la réalisation de langage spécifique de domaine (DSL). Des recherches ont également été menées en collaboration avec le laboratoire d'Informatique fondamentale de Lille, que notre équipe vient d'intégrer. Ces recherches portent sur la formalisation des processus de modélisations incrémentaux (Marvie & Nebut, 2006).

Notre thèse vise quant à elle la phase de construction d'un système à partir d'un ou plusieurs modèles. Dans le domaine des EIAH, le nombre des plateformes d'exécution est particulièrement grand, près de 240 (Thot), ce qui nous laisse entrevoir la possibilité de concevoir un dispositif pouvant opérer sur plusieurs d'entre elles. C'est ce qui a motivé l'adoption d'une telle ingénierie. La ligne de production logicielle que nous avons définie, et outillée, est assez typique du Model Driven Architecture (MDA) vision de l'IDM selon l'OMG (Clark, Evans, & Kent, 2002) : définition d'un modèle abstrait, transformation vers un modèle dit "technologique", raffinement puis interprétation de ce modèle par un générateur (Tolvanen, 2004) (Tolvanen, 2005) et enfin construction (automatique) de tout ou partie du système modélisé.

La "version" EIAH du cycle MDA est la suivante : description en langage naturel d'un dispositif pédagogique, modélisation du dispositif dans un langage pédagogique, transformation en un modèle fonctionnel spécifique à la plateforme visée, raffinement, contextualisation et construction dans le contexte d'une instance de la plateforme cible

Nos travaux en IDM se sont inspirés des travaux menés sur les langages spécifiques de domaine. En effet l'approche par DSL repose sur une vision pragmatique de la modélisation, Ils reposent sur la définition de métamodèles décrivant une plateforme pour un domaine spécifique. Ceux-ci permettent l'édition de modèles décrivant des systèmes sur une plateforme. En adoptant comme méta méta modèle de modélisation le langage MOF, nous proposons dans notre thèse de produire des modeleurs et des constructeurs spécifiques à chaque plateforme de formation.

1.2 Problématique, Objet de recherche

1.2.1 Objet de recherche

L'objet de notre recherche consiste à :

Tisser un lien entre les intentions didactiques d'une équipe d'enseignants pratiquant le bricolage pédagogique dans une perspective artisanale et la réalisation concrète d'un dispositif pédagogique dans le contexte d'une plateforme.

Cet objet nécessite d'analyser informatiquement les schèmes d'utilisation d'une plateforme par un enseignant et d'imaginer une infrastructure permettant d'accompagner l'enseignant dans son travail. La perspective choisie "*définir une ingénierie permettant de rationaliser l'activité artisanale d'un groupe d'enseignant*" implique le respect des paradoxes présentés dans le paragraphe suivant.

1.2.2 Perspective adoptée, paradoxe, artisanat, ingénierie

Notre cadre de recherche impose deux paradoxes que nous illustrons ici et pour lesquels nous tentons d'explicitier les conséquences qu'ils induisent

Paradoxe 1: Un EIAH n'est pas a priori conçu pour mettre en œuvre les intentions pédagogiques d'un groupe spécifique d'enseignants.

Ce paradoxe est la conséquence immédiate du cadre artisanal dans lequel nous souhaitons étudier la création enseignante. Dans ce cadre, les enseignants n'ont pas les moyens de concevoir des plateformes pour médiatiser leur enseignement. Nous n'avons donc pas prévu de tester notre ingénierie sur des prototypes conçus spécifiquement pour notre étude. Notre recherche s'adresse à des EIAHs choisis et adaptés par une équipe d'enseignants impliqués dans une démarche artisanale dans la limite de leurs compétences. De tel EIAHs ont une vie propre en dehors de notre recherche et sont utilisés en dehors de celle-ci. Cette utilisation contribue d'ailleurs à placer notre recherche dans un contexte réel (Kommers, Grabinger, & Dunlap, 1996), (McCalla, 2004).

Le deuxième paradoxe de notre recherche est la réciproque du premier.

Paradoxe 2: Un dispositif pédagogique n'est pas prévu pour un EIAH spécifique.

Ce paradoxe s'appuie sur trois idées.

L'étrangeté : Bien que construit, un EIAH peut être considéré comme un objet semi naturel, dont la finalité, la définition et l'utilisation échappent a priori à l'enseignant chargé de l'utiliser. Un tel objet peut être étudié, interprété et son fonctionnement peut être modélisé, mais chacun doit garder à l'esprit le caractère subjectif et incomplet des représentations

mentales ainsi créées. Dans la mesure où l'équipe d'enseignants chargée d'utiliser l'EIAH n'a pas participé à sa réalisation on ne peut pas parler de véritable adéquation entre celui-ci et les enseignants.

La dualité : Intention pédagogique et EIAH forment un couple dont chaque partie modifie l'autre. Autant l'utilisation d'un EIAH permet de faire évoluer les intentions pédagogiques d'une équipe d'enseignants, autant l'EIAH se finalise dans l'usage que l'équipe en fait. Ces adaptations ont été à maintes reprises constatées en particulier dans le cadre de l'étude de cas lors du projet PCDAI (PCDAI, 2006)

La différence de cycle : Création pédagogique et création d'EIAH appartiennent à deux cycles de création différents dont la synchronisation nécessite des outils de négociation.

1.2.3 Problématique

Nos problématiques concernent la définition et la mise en œuvre d'une ingénierie pour accompagner la création enseignante. Le but que nous nous fixons nécessite de clarifier la notion de plateforme de formation, les rôles qu'un enseignant peut y jouer, la nature des activités de création qui s'y déroulent et l'objet de ces activités. Les choix conceptuels et technologiques de notre ingénierie, utilisés pour réaliser une infrastructure, doivent être adaptés à la création enseignante et favoriser par exemple le bricolage pédagogique et la création participative. Ces choix conceptuels et technologiques mettent en œuvre, dans notre proposition, une ingénierie dirigée par les modèles. Une telle ingénierie n'a que très rarement été appliquée au domaine des EIAHs. Cette adaptation nous permet d'aborder, dans cette thèse, les verrous IDM constitués par la définition d'un modèle de plateforme et la définition d'un mécanisme de factorisation.

chapitre 2 Analyse des pratiques sociales dans la conception d'instruments pour les EIAHs

2.1 Introduction

Un des objectifs de nos travaux est de proposer des outils informatiques permettant aux enseignants d'adresser, de paramétrer et de nourrir d'activités des plateformes de formation. Pour définir la forme exacte que peuvent prendre les interactions d'un enseignant sur une plateforme de formation, il nous faut parvenir à établir une définition claire de ces plateformes. Selon le domaine dans lequel nous nous situons, EIAH ou Ingénierie dirigée par les modèles, le terme plateforme recouvre des significations différentes. Dans ce chapitre nous essaierons de décrire les différents usages de ce mot en EIAH. En particulier, nous abordons l'ingénierie des EIAHs dans une perspective large. Dans *"Introduction à l'ingénierie des EIAHs"*, Pierre Tchounikine exclut du champ de l'ingénierie des EIAHs la conception de situations pédagogiques fondées sur des systèmes informatiques existants. *"Ceci exclut les travaux où l'on se focalise sur les conceptions de situations pédagogiques fondées sur des systèmes informatiques existant par ailleurs, par exemple l'utilisation (...) d'un collecticiel comme environnement d'apprentissage collaboratif"* (Tchounikine, 2006). Notre perspective est différente : notre définition d'une ingénierie pour les EIAHs renvoie à l'intentionnalité de l'enseignant, concepteur et usager de cet EIAH, que cette intentionnalité s'exprime par des actions de construction ou de détournement. L'ingénierie que nous proposons permet de prendre en compte cette intentionnalité et offre un cadre conceptuel pour son expression.

Nous verrons dans ce chapitre différentes définitions de la notion de plateforme de formation, il est difficile, voire inutile, d'aborder la question de l'ingénierie des EIAH par le biais d'une taxinomie des plateformes de formation. Cette question doit plutôt être abordée en terme de dimension sous jacentes à l'EIAH considérée (Tchounikine, 2006). Cette suggestion se rapproche de notre recherche puisque nous allons montrer qu'il est difficile d'établir une taxinomie des plateformes de formation. Tout au plus pouvons-nous proposer un tableau récapitulatif des différents sous-domaines qu'il est possible de prendre en compte lors de l'évaluation d'une plateforme de formation.

Ces sous-domaines, illustrés par la Figure 1, seront détaillés dans la suite de ce chapitre. Ils concernent les fonctionnalités permettant de définir une plateforme de formation et les différentes dénominations utilisées dans leur pratique et dans les travaux de recherche universitaires.

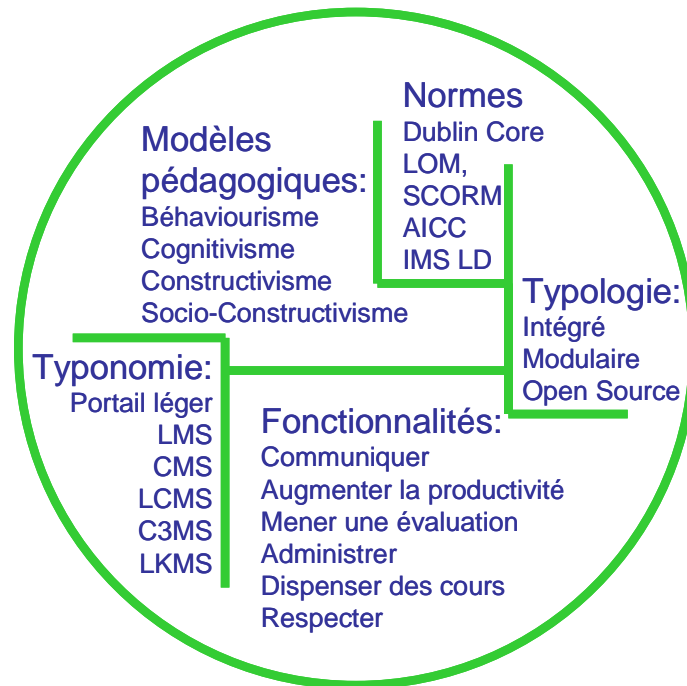


Figure 1 Différents sous domaines permettant l'analyse d'une plateforme de formation

Nous complétons notre étude par une typologie des plateformes de formation, et les caractérisons selon qu'elles soient intégrées, modulaires ou open source. Nous étudions la question de l'ingénierie pédagogique sur une plateforme de formation, en situant cette question par rapport aux normes et standard des EIAH ainsi que par rapport aux modèles pédagogiques. Notre étude nous permettra de situer les prémisses de la problématique qui est à la genèse de notre recherche, en définissant un tableau de correspondances entre fonctionnalité et modèle pédagogique.

Comment instrumenter l'enseignant pour lui permettre de construire des dispositifs riches, permettant de supporter des activités constructivistes et socioconstructivistes sur des plateformes de formations modulaires?

C'est à cette problématique, que notre proposition basée sur l'ingénierie dirigée par les modèles adaptée au EIAH entend répondre. Pour l'instant nous nous attachons, dans la suite de ce chapitre, à définir la notion de plateforme de formation.

2.2 Définir une plateforme de formation par ces fonctionnalités

En Septembre 2005, 240 plateformes de formation étaient recensées par (Thot). Le grand nombre de plateformes et leurs diversités motivent le désir que nous avons, pour cette recherche, de définir finement ce que nous entendons par plateforme de formation en EIAH. En effet ce nombre est à relativiser :

- En le minimisant : toutes les plateformes n'ont pas la même audience, par exemple parmi les nombreuses plateformes open sources utilisables, quatre plateformes occupent une place prépondérante (Adkins, 2005).
- En l'augmentant : de nombreux outils sont utilisés en tant que plateforme de formation sans être référencés en tant que tel, par exemple près de 600 "Content Management System" sont référencés (CMS)

Pour débiter notre étude, proposons la définition du terme "plateforme de formation" établie par l'Office de la Langue Française :

"Une plate-forme de formation est un système informatique destiné à automatiser les diverses fonctions relatives à l'organisation des cours, à la gestion de leur contenu, au suivi des progrès des participants et à la supervision des personnes responsables des différentes formations" (Dictionnaire_OLF)

Dans l'éditorial du numéro spécial *"Conception et usage des plateformes de formation"* de la revue STICEF, Sébastien George et Alain Derycke complètent cette définition pour prendre en compte l'émergence de nouveaux paradigmes liés à la mise en avant des activités pédagogiques *"une plate-forme peut être vue comme un système qui permet de gérer et de donner accès à un ensemble d'activités et de ressources pédagogiques"* (George & Derycke, 2005).

Les plateformes ont donc pour objectif de proposer un ensemble de fonctionnalités permettant de mener à bien un dispositif de formation (distant ou non). Il est possible de lister ces fonctionnalités. Nous nous baserons pour cela sur la liste établie par Bruce Landon à partir de laquelle le site de comparaison Edutools a été conçu (Edutools) (Landon, 2002).

Cet organisme propose de comparer les plateformes de formation selon six fonctionnalités principales :

1. **Communiquer** : forum, mail, blog...
2. **Augmenter la productivité** des apprenants : signet, recherche, calendrier...
3. **Mener une évaluation** dans le groupe ou individuel : groupe de travail, portfolio, autotest...
4. **Administrer** les personnes : authentification, inscription au cours...
5. **Dispenser des cours** : Suivi des élèves, Gestion des parcours, gestion des cours, test
6. **Respecter de la charte de la plateforme** : accessibilité, design des cours, respect des normes.

Ces fonctionnalités permettent de comparer, différentes plateformes de formation sur le plan technique. De nombreuses études partielles s'inspirent, ou ont permis la réalisation, de cet outil de comparaison. (Edutech, 1999) (Préau) (Ecoutin & Guidon, 2000) (Bouillaux et al., 2002) La *British Educational Communications and Technologies Agency* (Becta, 2006a) publie une spécification des plateformes de formation en insistant sur les fonctionnalités obligatoires et optionnelles qu'elles doivent fournir. Cette agence gouvernementale recense ainsi 45 fonctionnalités qu'une plateforme de formation doit respecter et pour 7 de ces fonctionnalités, elle spécifie le standard qui doit être utilisé. Ce travail est destiné aux établissements d'enseignement pour les aider à faire un choix parmi les différentes plateformes de formation disponibles, il est également destiné aux concepteurs de plateforme.

Un des premiers enseignements de ces différentes études est de montrer que la définition de l'artefact *"plateforme de formation"* est assez floue, on trouve en effet dans ces études des artefacts très différents. Ces artefacts peuvent prendre des formes diverses, allant de plateforme proposant des outils intégrés (WebCT (WebCT), Ganesha (Ganesha)) au portail modulaire de type PostNuke (Nuke) qui n'est pas destiné spécifiquement à l'enseignement. Un des objectifs que nous nous fixons dans les prochains chapitres est donc de décrire ces différents artefacts à partir de leur dénomination, et de montrer que la notion de plateforme de formation réside moins dans la spécificité des artefacts qui l'incarnent, que dans l'usage que des enseignants en font.

2.3 Différents sigles pour une Plateforme de formation

2.3.1 Différents sigles

Définir les plateformes de formation au travers des fonctionnalités qu'elles exhibent, ne permet pas de prendre en compte la grande diversité des artefacts utilisés pour mener à bien

un dispositif de formation. Aussi proposons nous dans cette section d'aborder dans un premier temps la dénomination des plateformes de formation. Dans la pratique, trois définitions anglo-saxonnes associées correspondent à des artefacts différents. Toutefois, elles sont semblables, en tant qu'outil, par leur usage. Ces trois termes recouvrent en partie la notion élargie que nous nommons "*plateforme de formation*" : Il s'agit des sigles LMS, CMS, et LCMS (Ertoran, 2006).

- Un LMS (learning management system) est un système Internet de gestion de cours. Il permet à des apprenants d'accéder de façon simple à des cours mis à disposition par les formateurs au moyen d'un navigateur. WebCT (WebCT), Ganesha (Ganesha) Claroline (Claroline), Moodle (Moodle) sont des LMS. La première plateforme est de type propriétaire, les autres sont de type Open Source.
- Un CMS (content management system) est un système de gestion de contenus qui permet une publication de contenus multimédia sur un réseau local ou distant, dans lequel une base de données sert à stocker les contenus. SPIP (SPIP) (système de publication pour Internet), Joomla (Joomla), Typo 3 (Typo3) sont des systèmes de gestion de contenus assez répandus.
- Un LCMS (learning content management system) (Greenberg, 2002) est un système intégré de formation à distance utilisant toutes les fonctionnalités d'un LMS et d'un CMS. Il inclut également une interface permettant la production de cours à partir de catalogue d'objet pédagogique. TopClass (TopClass), SimplyLearn (Simplylearn) sont des exemples de telles suites intégrées.

De tels artefacts, pourtant différents par leurs définitions et leurs sigles, peuvent être utilisés parfois indifféremment, comme nous le verrons dans la suite de ce chapitre. Dans la section suivante, nous explorons d'autres sigles permettant de désigner des plateformes de formation, ces sigles sont introduits par des travaux de recherche.

2.3.2 Sigles et travaux de recherche

Dans le domaine de la recherche EIAH, il est apparu d'autres sigles pour désigner des usages spécifiques de certaines plateformes de formation. Ces nouveaux sigles entendent, dans un contexte de recherche particulier, préciser les sigles précédents. Nous présentons dans les paragraphes suivants deux sigles différents illustrant deux visions très différentes de la notion de CMS.

Le sigle C3MS

Le terme C3MS (content, community, and collaborative management system) est utilisé pour définir un CMS auquel des fonctionnalités permettant la prise en compte du contenu, de la communauté et de la collaboration ont été adjointes (Schneider, 2003). Le CMS PostNuke (PostNuke) est ainsi utilisé comme plateforme de formation dans le cadre des enseignements dispensés par l'unité de recherche "Technologies de Formation et Apprentissage" (Tecfa) appartenant à la Faculté de Psychologie et des Sciences de l'Éducation de l'Université de Genève. À ce CMS, des modules permettant la mise en œuvre d'activités pédagogiques ont été ajoutés. Le Tecfa propose ainsi, sur son site, 95 modules pédagogiques pouvant être utilisés avec PostNuke, et il participe activement au développement de ces modules (Module_PostNuke), (Lattion, 2005), (Chakroun, 2003). Un CMS auquel on a adjoint des modules pédagogiques propose les fonctionnalités d'une plateforme de formation. L'exemple ci-dessous cité par Schneider, illustre cette situation pour le CMS PostNuke.

- Gestion de contenu
- Echange de connaissances
- Echange d'arguments
- Gestion et support de projets
- Gestion de connaissances
- Gestion de la communauté

La frontière entre un C3MS tel que nous venons de le décrire et les autres objets précédemment décrit (LMS, LCMS et CMS) est donc très floue. Elle est difficile à établir dans le cas de CMS modulaire. Le caractère modulaire de telles applications permet en effet l'ajout rapide de fonctionnalités souhaitées : par exemple le détournement d'un module de recette de cuisine permet dans PostNuke d'éditer des scénarios pédagogiques ; sur la plateforme Moodle, l'outil "Journal" a été utilisé pour générer l'outil "Devoir". La spécificité Open source de certains CMS permet d'utiliser une communauté de programmeurs pour le développement de fonctionnalités additionnelles. Cette spécificité contribue, en renforçant l'offre de modules, à rendre polymorphe ce type de plateforme (Brusilovsky & Nijhawan, 2002).

Le sigle LKMS

Dans le cadre du réseau de recherche canadien LORNET (Lornet), le projet DIVA (Paquette, 2005) propose la réalisation du système TELOS (TeleLearning Operating System), ce système est un générateur de portail Web soutenant l'apprentissage et la gestion des connaissances. Une vue du système TELOS et de ces quatre niveaux d'applications, tel qu'il est présenté par ces concepteurs (Paquette, 2005) est illustrée sur la Figure 2.

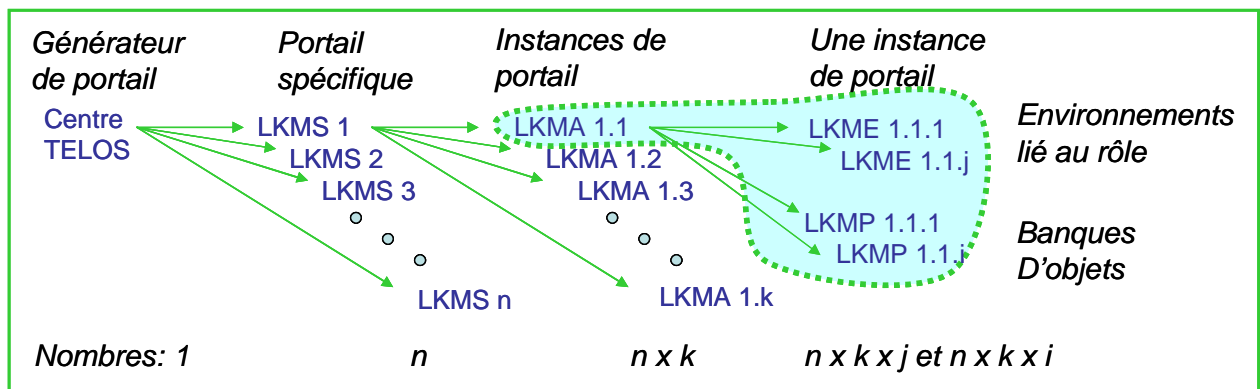


Figure 2 TELOS 4 niveaux d'organisation

Une plateforme produite par le générateur TELOS est appelé LKMS (Learning Knowledge Management System), chaque plateforme concrétise les besoins particuliers d'une organisation. La concrétisation d'un LKMS dans un contexte particulier s'appelle un LKMA (Learning Knowledge Management Application). Chaque LKMA possède des environnements différents selon le rôle tenu par l'utilisateur qui s'authentifie. Un tel environnement se nomme LKME (Learning Knowledge Management Environment). Les ressources produites se nomment (LKMP Learning Knowledge Management Products) et sont stockées dans des banques de données.

Derrière ces sigles, deux approches différentes

Il existe un parallèle évident entre ce que Daniel Schneider appelle C3MS et ce que Gilbert Paquette appelle LKMS, ne serait-ce que par le recours à des portails de type "Nuke" (Nuke), PostNuke dans le cas des C3MS et DotNetNuke (DotNetNuke) dans le cas des LKMS.

Les artefacts considérés, PostNuke et DotNetNuke, sont comparables dans leur forme originelle. Cependant, l'utilisation des LKMS s'inscrit dans une vision très ingénieriste de la formation. C'est ce qui distingue la vision de Gilbert Paquette de celle de Daniel Schneider. En effet les portails LKMS sont conçus par des spécialistes en technologie d'apprentissage à partir du générateur TELOS. L'ensemble du processus qui mène à la fabrication d'une instance de cours, sollicite différents rôles parmi lesquels : un informaticien, un ingénieur pédagogue, un concepteur, un facilitateur, et un apprenant. Nous reviendrons dans le chapitre 3 sur cette vision des rôles dans le cadre de TELOS. L'ensemble des entités composant les quatre niveaux d'organisation de TELOS sont observables depuis le centre de contrôle de TELOS. Les ressources produites, les LKMS, sont par conséquent réutilisables dans d'autres contextes.

A l'opposé, les C3MS composés par le Tecfa revendiquent une vision artisanale de la formation. La production d'un portail de type C3MS spécifique ne résulte pas d'un processus d'ingénierie pédagogique complexe. Il s'agit d'une enveloppe vide contenant des possibilités d'extension via le système de module du Tecfa ou par des modules compatibles avec PostNuke. Un tel portail est alors mis à la disposition d'une équipe d'enseignants par le Tecfa. Les activités que l'équipe pédagogique mène sur son portail n'ont pas de répercussion sur les autres portails, sauf à établir des listes de syndication spécifiques. Les données produites par les apprenants ne font pas l'objet de référencements disponibles pour tous les C3MS mis en place. L'ensemble du processus qui mène à la fabrication d'une instance de cours, relève ici clairement de bricolage pédagogique. Cette notion sera explicitée dans le chapitre 5 et le chapitre 8.

2.3.3 Applications Web 2.0

Une autre sorte d'artefact est également utilisable en tant que plateforme, il s'agit d'application Web, basée sur un nombre de fonctionnalités réduites : Forum, Wiki, dépôt de document, Blog. Le paramétrage et le détournement de telles applications permettent leurs usages en enseignement. Ces applications sont caractérisées par le pouvoir de création qu'elles offrent à leurs utilisateurs. Cette création concerne des productions de documents ainsi que le commentaire et l'évaluation de ces documents. Cette création concerne également des mécanismes de structuration individuelle (création de catégorie) ou collectif (carte conceptuelle élaborée à partir de simples tags). Le sigle utilisé pour définir ces applications est usuellement le Web 2.0 (O'Reilly, 2005). L'usage de ces applications en enseignement est désigné par le sigle eLearn 2.0 (Downes, 2005). Nous reviendrons sur ce type d'application dans la suite de cette thèse (sections 3.5.4, 8.6.2) et dans le cadre de l'étude de cas conduite sur l'application WikiniMST (chapitre 11.3).

2.3.4 Conclusion

Nous venons de voir que de nombreux outils procurent les fonctionnalités d'une plateforme de formation. Ces outils peuvent être des LMS classiques mais aussi des applications de type CMS, C3MS ou LKMS. L'exemple que nous venons de traiter nous montre que des artefacts pourtant semblables par leurs origines semblent induire des usages différents en terme d'ingénierie pédagogique. Pour définir une plateforme de formation, il semble donc important de s'intéresser aux usages qu'elle suscite et aux acteurs qu'elle mobilise plutôt qu'aux fonctionnalités qu'elle revendique. Nous proposons dans la section suivante de spécifier les différents rôles tenus par les acteurs d'une plateforme de formation.

2.4 Différents rôles, vision centrée sur l'apprenant, sur l'ingénieur pédagogique et sur le processus d'ingénierie.

2.4.1 Rôles attachés à l'usage d'une plateforme de formation

De nombreuses plateformes définissent trois types principaux de rôles : l'apprenant, le formateur, l'administrateur (Rasseneur, 2004) (Ecoutin & Guidon, 2000).

Différents rôles

L'apprenant consulte en ligne ou télécharge les contenus pédagogiques qui lui sont recommandés, organise et a une vue de l'évolution de son travail, effectue des exercices, s'auto évalue et transmet des travaux à corriger.

Le formateur crée des parcours pédagogiques types et individualisés, incorpore des ressources pédagogiques multimédias et effectue un suivi des activités des apprenants.

L'administrateur installe et assure la maintenance du système, gère les accès et les droits des uns et des autres, crée des liens avec les systèmes d'information externes (scolarité, catalogue, ressources pédagogiques, etc.).

Ces trois rôles sont respectés par de nombreuses plateformes, comme Ganesha et Claroline. Ils peuvent être complétés par plusieurs sous catégories, ainsi la plateforme Ganesha permet-elle d'ajouter au rôle de formateur des droits d'administrations ou d'agrégation.

2.4.2 Différents rôles en EIAH vus par Gilbert Paquette

Dans une perspective de recherche, cette distinction est également présente dans (Paquette, Ricciardi-Rigault, de la Teja, & Pasquin, 1997), celui-ci ajoute également le rôle d'informateur, et précise les rôles précédents

- L'informateur rend disponible les informations.
- Le formateur facilite l'apprentissage.
- Le concepteur construit, adapte, maintient le système d'apprentissage.
- Le gestionnaire gère les acteurs et les événements.

Dans (Paquette et al., 1997) Gilbert Paquette considère les plateformes de formation dans une perspective centrée sur l'apprenant : pour lui une telle plateforme est caractérisée par les interactions entre les apprenants et les autres rôles impliqués par le dispositif. La définition des rôles qu'il propose permet ainsi de spécifier un dispositif de formation en un ensemble de tâches élémentaires impliquant l'apprenant avec les autres rôles. Une plateforme de formation est un artefact autorisant ces différentes tâches.

Dans un article plus récent (Paquette, 2005), l'auteur complète la définition des rôles intervenant autour d'une plateforme de formation : il y décrit le système TELOS (cf. chapitre précédent). Ce système qui permet la génération de portail LKMS, définit six rôles principaux. La Figure 3 illustre un exemple d'interaction entre ces différents rôles lors de l'utilisation de TELOS.

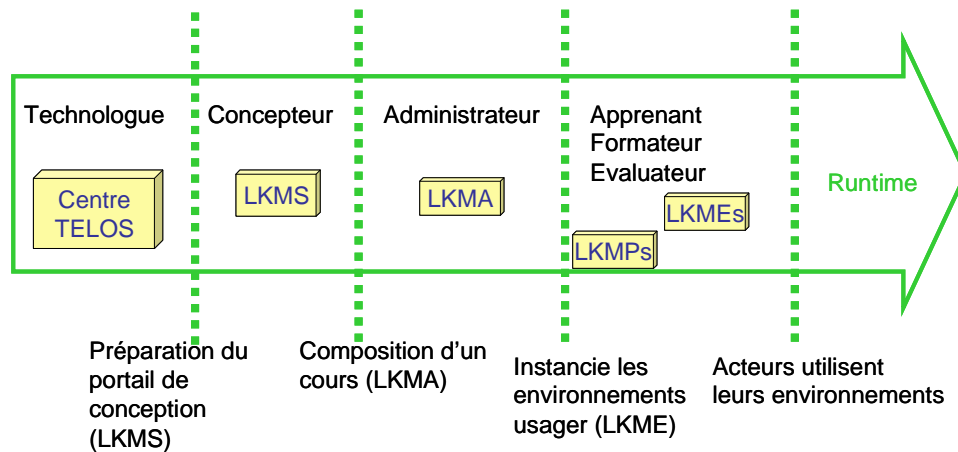


Figure 3 Exemples d'utilisations de TELOS, différents acteurs

Le **technologue** prépare le portail de conception destiné au **concepteur**. Celui-ci compose des cours que l'**administrateur** instancie. **Apprenant, formateur et évaluateur** peuvent alors mener leurs activités dans leurs environnements spécifiques.

Gilbert Paquette précise que lors de la conception d'un scénario pédagogique, l'enseignant doit anticiper les interactions entre les différents composants d'un scénario d'apprentissage, (activités, ressources, outils et rôles) afin qu'elles génèrent chez l'apprenant les opérations mentales voulues (Paquette, 2005). Cependant l'ensemble des rôles qu'il définit nous semble motivés par le désir de rationaliser l'élaborations des plateformes de formation au travers un processus d'ingénierie s'apparentant à un workflow.

"Par ailleurs, une nouvelle ingénierie pédagogique est nécessaire pour contrer la tendance au développement artisanal que l'on peut observer dans trop de formations sur l'Internet, le génie logiciel pouvant servir d'inspiration à cet égard" (Paquette, 2005)

Gilbert Paquette développe donc dans cet article une vision des plateformes de formation selon une perspective ingénieuriste. L'objet principal des rôles définis ici nous semble être le processus d'ingénierie lui-même.

2.4.3 Perception des rôles par Thomas Vantrois

Thomas Vantrois dans sa thèse *"Du langage métier au langage technique, une plateforme flexible d'exécution de scénarios pédagogiques"* aborde les plateformes de formation selon une perspective assez semblable (Vantrois, 2003). Pour catégoriser les différents rôles liés à une plateforme de formation, il commence par distinguer les différents domaines fonctionnels qu'ils séparent en quatre catégories: administration, modélisation, exécution et usage. Ces quatre domaines lui permettent de répartir les différents rôles mobilisés autour d'une plateforme de formation, cette répartition est illustrée par la Figure 4.

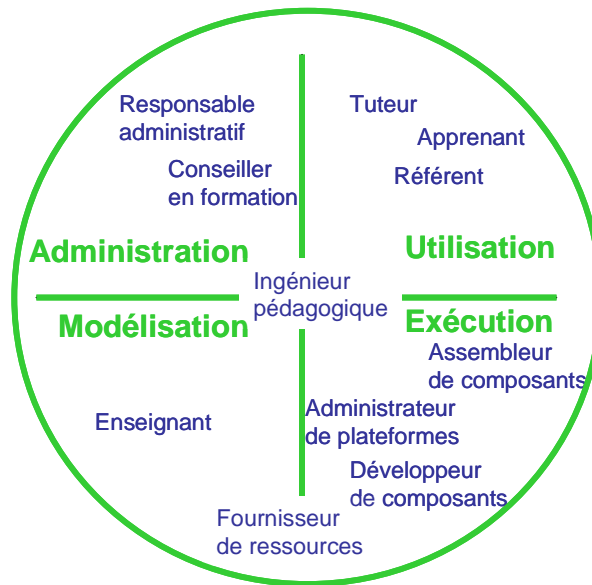


Figure 4 Rôles et domaines fonctionnels sur une plateforme de formation selon Thomas Vantroys

Ces onze rôles différents permettent de décrire les processus de conception et d'utilisation affectés à une plateforme de formation. Dans sa vision d'une plateforme de formation, Thomas Vantroys place le rôle tenu par l'ingénieur pédagogique au centre de tous les processus. Il le définit comme l'homme orchestre de la plateforme. Celui-ci fait le lien entre la définition, la réalisation et l'utilisation des modules d'enseignement. Il mène le même travail lorsqu'il s'agit d'ajouter de nouveaux composants à la plateforme. Pour cela il dialogue avec l'enseignant, l'assembleur et le développeur, ainsi qu'avec le tuteur.

Cette vision s'appuie en fait sur une perspective restreinte de la plateforme. Celle-ci est considérée comme un objet final autour duquel se structurent différents domaines fonctionnels cités précédemment. Dans cette perspective l'ingénieur pédagogique joue effectivement un rôle fondamental, puisqu'il intervient dans les quatre domaines. L'ensemble des activités qu'il mène consiste à rationaliser les activités générées autour d'une plateforme de formation, (analyse conception, modélisation, prescription). Dans cette vision, les activités générées ont pour objet la plateforme de formation.

Nous allons dans la section suivante aborder les limites de ces deux perspectives.

2.4.4 Limites de ces perspectives

Les deux perspectives, que nous venons de décrire (Gilbert Paquette, Thomas Vantroys), sous-entendent toutes les deux une approche industrielle.

"La cascade de génération des systèmes à partir du Centre TELOS est analogue à une usine qui construit des machines servant à construire d'autres usines (LKMSs), ces dernières construisant des voitures, des avions, etc. (LKMA); utilisés par des usagers finaux" (Paquette, 2005).

"Le processus de conception d'un module d'enseignement, comprend 6 phases principales. Nous pouvons le rapprocher du processus unifié de développement (Rational Unified Process : RUP)" (Vantroys & Peter, 2005).

Ces deux perspectives industrielles - que ce soit Gilbert Paquette dans le cadre d'Explora (Paquette, 2001) puis dans celui du générateur de LKMS, TELOS (Paquette, 2005) ou Thomas Vantrois pour l'adaptation du moteur de Workflow COW à la plateforme Campus Virtuel (Vantrois, 2003) - considèrent comme acquis la possibilité de concevoir une plateforme pour une communauté enseignante, ou de faire développer des fonctionnalités nouvelles pour cette plateforme de formation, dans le but de la mettre en adéquation avec les demandes des enseignants.

Ces perspectives industrielles ne nous semblent pas pertinentes pour décrire les offres de formations légères.

Dans le cadre de l'institut CUEEP (Centre Université- Economie d' Education Permanente, CUEEP) dans lequel notre laboratoire Trigone se situe, nous assistons à la coexistence de plusieurs dispositifs de formations pour une même promotion. Si nous adoptons une perspective centrée sur l'enseignant, il nous faut oublier le monde harmonieux dans lequel un enseignant n'interviendrait que sur un seul type de plateforme. Non seulement au CUEEP, les plateformes de formations utilisées sont multiples, mais un même enseignant peut intervenir dans l'organisation de plusieurs formations. Cette situation a une conséquence sur le choix des plateformes utilisées. Les plateformes de formations Open Source et modulaires sont privilégiées dans la mesure où elles peuvent permettre une adaptation au dispositif spécifique de formation. Cette adaptation peut être réalisée par développement propre ou par ajout de module. Cet ajout est favorisé par le caractère open source de la plateforme.

Cependant, les enseignants sont alors placés devant deux situations différentes mais ayant les mêmes conséquences. Si la plateforme est imposée par l'institution, elle concerne alors un grand nombre de formations et l'adaptation de la plateforme par les ressources de l'institution au désir de chaque enseignant est minimale. Si la plateforme est choisie par une petite équipe d'enseignants elle est a priori plutôt adaptée au projet des enseignants, mais ceux-ci ne peuvent pas alors bénéficier d'adaptations spécifiques. Or pour des formations légères, (nous avons des exemples constitués par 6 enseignants autour d'une promotion de 25 étudiants), cette adaptation a pu (par expérience) être réalisée de deux manières :

- Par l'ajout de modules à des portails de type C3MS.
- Par des paramétrages simples d'application de type Web 2.0. Celles-ci pouvant ainsi être utilisés comme des plateformes de formation (Caron, Le Pallec, & Sockeel, 2006b).

Dans le cadre artisanal des formations légères, il existe donc des alternatives à des ingénieries des EIAHs reposant sur des approches industrielles.

2.5 Conclusion et positionnement de notre recherche en ingénierie des EIAHs

Pour résumer ce chapitre, retenons que nous ne situons pas notre recherche dans une perspective centrée sur l'ingénieur pédagogique ou sur le processus de production d'une plateforme mais que nous considérons les plateformes de formation dans une perspective centrée sur une équipe d'enseignants construisant "artisanalement" leurs dispositifs pédagogiques.

Plutôt que de considérer la création en EIAH dans une perspective industrielle, nous avons décidé d'étudier l'aspect bricolage et artisanal que peut prendre un tel processus de création. Cet aspect nous semble mieux adapté à l'expérience que nous avons des EIAHs. Nous

montrons dans cette thèse qu'il est possible dans cette perspective de proposer des outils et des méthodes composant une ingénierie et permettant de rationaliser le travail des enseignants.

Nous considérons la création sur une plateforme de formation dans une perspective centrée sur les usages d'enseignants encadrant artisanalement des formations légères.

C'est dans cette perspective réduite que nous avons mené nos études de cas (cf. le chapitre 11). Cette perspective artisanale implique des contraintes quant à l'adaptation d'un EIAH aux intentions pédagogiques des enseignants. Cette adaptation ne peut être réalisée que par des structurations légères, celles-ci seront détaillées dans le chapitre 5.

Notre domaine de recherche correspond à des plateformes de formation au sens large choisies (installées, administrées) par des enseignants dans le cadre d'une utilisation artisanale. Notre travail tend donc à minimiser l'activité de l'ingénieur pédagogique (souvent absent de ce type de formation) et rationaliser celle de l'enseignant. Nous pensons qu'il est possible de définir une ingénierie pour l'artisanat pédagogique et que cette vision artisanale du métier d'enseignant est justifiée par le caractère spécifique de la création enseignante.

Nous poursuivons, dans le chapitre 3 et le chapitre 4, l'analyse de la notion de plateforme de formation selon une approche centrée autour des activités de création menées par les enseignants. Cette approche ne signifiant pas que nous privilégions une pédagogie centrée sur l'enseignant : nous le verrons avec la présentation à venir des modèles pédagogiques que nous avons mis en œuvre. Cette analyse va permettre de cerner les usages que les enseignants ont d'une plateforme de formation et mettre en évidence les problématiques EIAHs que notre proposition aborde.

chapitre 3 Plateformes de formation et usages des enseignants au regard des modèles pédagogiques

3.1 Introduction

Nous allons dans ce chapitre explorer l'usage que les enseignants font d'une plateforme de formation. Une des difficultés fondamentales de l'élaboration d'une ingénierie des EIAH est l'absence de substrat théorique consistant (Tchounikine, 2006) ; en analysant l'usage enseignant des plateformes de formation au regard des modèles pédagogiques nous espérons clarifier le principe d'ingénierie que nous désirons mettre en œuvre. Nous présentons, dans ce chapitre, différents modèles pédagogiques, puis nous tentons d'établir des liens entre ces modèles, l'objet plateforme et l'usage enseignant de cet objet.

3.2 Définition

Le point commun de tout EIAH est *"la relation d'une intention didactique et d'un environnement informatique"* (Tchounikine, 2002). C'est l'importance donnée à réalisation de cette intention didactique qui motive l'utilisation d'une plateforme de formation. En effet, le but d'une plateforme de formation est de procurer à un enseignant ou à un groupe d'enseignants une série d'outils lui permettant de mettre en œuvre de façon rationnelle (pour lui ou son institution) des activités d'apprentissage collectives ou individuelles. L'usage d'une plateforme de formation n'a de sens que par la volonté **d'une équipe d'enseignants** d'y héberger son enseignement.

En ce sens une plateforme de formation est un EIAH un peu particulier, dans la mesure où sa finalité première n'est pas l'apprentissage mais la mise en œuvre et le suivi de l'apprentissage.

Une plateforme de formation est certes destinée à une communauté d'*apprenants*, mais elle est avant tout choisie par une communauté d'*enseignants*, elle permet aux premiers l'accès aux EIAHs qu'elle héberge. Aux seconds, elle offre des outils permettant la mise en œuvre de leurs intentions didactiques (créer des activités, les dispenser, les contrôler et les évaluer). Cette remarque nous permet de proposer une définition d'une plateforme de formation en nous basant sur l'usage qu'un enseignant en a :

Une plateforme de formation est un environnement informatique collectif permettant à une communauté d'enseignants de produire les conditions qui favorisent l'apprentissage et le suivi des apprenants.

3.3 Comment une équipe pédagogique choisit elle une plateforme de formation?

Est-il possible d'explicitier la relation entre les intentions pédagogiques de l'équipe et la plateforme de formation qui servira de support à leurs expressions? Pour explorer les relations entre enseignants et plateforme de formation, nous avons décidé dans un premier temps, d'établir une typologie des plateformes de formation en fonction des modèles pédagogiques implicites ou explicites choisis par la communauté d'enseignants. Un tel travail a été mené

dans le cadre plus général des logiciels d'apprentissage (E. De Vries, 2001) ou dans le cadre des outils de collaboration (Wenger, 1998). Cette approche peut-être un peu caricaturale est cependant motivée par le désir que nous avons de nous situer, en EIAH, dans un cadre communément accepté :

- Différents travaux de recherche en EIAH font référence à ces modèles pédagogiques (Koper & Olivier, 2004), (Berggren et al., 2005), (Schneider, 2003), (Nodenot, 2006a).
- Des plateformes de formation comme Moodle, Postnuke, Ganesha, Acolad, Accel revendiquent une adéquation avec certains de ces modèles.
- Les standards Scorm (SCORM), IMS-LD (IMS-LD), se positionnent par rapport à ces modèles.

Dans ce chapitre nous situons notre approche par rapport aux modèles pédagogiques. Nous produisons, en annexe, un tableau permettant de mettre en relation les fonctionnalités offertes par une plateforme et les modèles pédagogiques.

Nous verrons cependant qu'une telle recherche si elle permet d'exprimer des généralités utiles, souffre de défauts importants dus essentiellement à deux facteurs :

- Le caractère polymorphe des plateformes de formation modulaire.
- La difficulté pour une équipe d'enseignant de se positionner dans un modèle pédagogique particulier.

3.4 Les modèles pédagogiques.

Nous citons en annexe un tableau récapitulatif des différents modèles pédagogiques que nous considérons. Les quatre paragraphes suivants correspondent à une simplification de ce tableau, et réfère à une typologie assez simpliste selon quatre catégories : béhaviourisme, cognitivisme, constructivisme, socioconstructivisme. Cités par (Schneider, 2003), (Giordan & Girault, 1996), (Rosca, 1999)

(E. De Vries, 2001) retient pour sa typologie les termes Béhaviourisme, Cognitivisme, Constructiviste et Cognitivisme Situé, faisant de ce fait plus référence à Bruner (Bruner, 1967) qu'à Vygotski (Vygotski, 1985)

3.4.1 Béhaviourisme, associationnisme.

Le Béhaviourisme est basé sur l'observation du comportement des apprenants et de leurs résultats (Legros & Crinon, 2002). Il repose sur l'idée que l'on apprend par les conséquences de ses actes, en enseignement, il y aura donc apprentissage, si lorsqu'une tâche est effectuée par l'apprenant, celle-ci est évaluée très rapidement et que l'apprenant est informé de la réussite ou de l'échec de cette tâche, un programme de remédiation individuelle peut alors être mis en oeuvre. Remédiation individuelle, délai de correction contribue alors au renforcement. Il est possible de proposer un programme de renforcement individualisé assisté par ordinateur, donc proposant des corrections en temps réel, fondé sur ce concept. (Skinner, 1968) Une des conséquences de ce type de pratique est cependant la disparition "d'écrits longs", par exemple, réalisés en classe ou à la maison (Amigues, Faïta, & Saujat, 2004). Une plateforme de formation pour mettre œuvre correctement ce modèle doit donc proposer divers outils parmi lesquels on trouvera :

- Des outils permettant la délivrance de contenus, (apprentissage signifiant contenu à apprendre)
- Des outils permettant le design des contenus.
- Des tests d'apprentissage avec correction automatiques, permettant l'évaluation formative et normative. (Evaluation synchrone, individuelle)

- Le respect des normes permettant la prise en compte de test et de scénario reposant sur le séquençement simple ou individuel d'activité. (Evaluation, individualisation, remédiation)
- Un tableau permettant de positionner l'apprenant au regard des objectifs à apprendre. (Evaluation)
- Des outils administratifs permettant de placer l'apprenant au sein de son cursus scolaire, (positionnement).
- Des outils de communication synchrone et asynchrone permettant le préceptorat de l'apprenant. (Importance du préceptorat (Bloom, 1979))

Certains standards actuels en EIAH relève surtout de ce type de modèle : (LOM), (Dublin-Core), (SCORM), (IMS-LD). Se référer par exemple à (Jonassen, Peck, & Wilson, 1999) (Pernin, 2004) (Dessus & Schneider, 2006)

3.4.2 Cognitivisme

Le Cognitivisme désigne l'ensemble des activités et des processus internes inhérents à l'acquisition des connaissances (Legendre, 1993). L'apprentissage y est perçu comme un processus de construction des connaissances et non uniquement comme une activité de transmission. En enseignement une des conséquences de cette théorie est par exemple d'attacher de l'importance à la notion de pré-requis, pré-test, post-test, et à l'ordonnancement des documents à soumettre aux élèves, s'inscrivant dans une suite que l'enseignant juge cohérente, logique, et progressive. En particulier une plateforme favorisant un modèle cognitiviste d'apprentissage, doit permettre :

- Le référencement des documents fournis par des métadonnées permettant de définir des pré-requis.
- Des outils de design permettant l'ordonnancement des activités.
- Le positionnement des élèves par rapport aux objectifs à atteindre par des pré et post tests
- Des activités de remédiations.
- Des problèmes à résoudre mettant en œuvre des activité transversales.
- Des outils administratifs permettant de placer l'apprenant au sein de son cursus scolaire, (positionnement).

Un exemple d'exploitation du modèle cognitiviste dans le domaine des EIAHs est l'élaboration de la méthode d'ingénierie des systèmes d'apprentissage MISA (Paquette, De la Teja, & Léonard, 2005). Cette méthode permet l'analyse, la conception, la réalisation et la planification de la diffusion des systèmes d'apprentissage. La modélisation cognitive permet de classer les connaissances humaines en trois grandes catégories abstraites : les procédures, les concepts, les principes. Aussi la méthode MISA permet l'élaboration de scénarios pédagogiques en intégrant des concepts, des processus et des principes de trois domaines : le design pédagogique, le génie logiciel et l'ingénierie cognitive. Pour mettre en œuvre, sur la plateforme Explora, des scénarios pédagogiques, la méthode MISA peut être instrumentée par le logiciel MOT+. Celui-ci permet le design graphique de scénarios respectant la méthode. (Paquette, Marino, De la Teja, Léonard, & Lundgren Cayrol, 2005)

3.4.3 Constructivisme

C'est un élément fondamental du Constructivisme que d'expliquer que la construction des connaissances nouvelles ne peut se produire que dans les contextes de la vie réelle (Basque & Doré, 1998) Le constructivisme repose sur le principe que l'on apprend principalement par l'action. En enseignement cela se traduira donc par une place privilégiée donnée aux activités

plutôt qu'aux documents. Cela nécessitera sur une plateforme de mettre en œuvre des outils permettant de mener des activités et de rendre compte des activités menées.

- Possibilité d'incorporer des documents comprenant des simulateurs, micro-monde, objets à manipuler.
- Possibilité de déposer des comptes rendus de projets.
- Outils permettant de rendre compte de l'activité des apprenants.

Ces fonctionnalités sont à rapprocher des trois éléments fondamentaux établis pour la mise en place d'un enseignement constructiviste utilisant le Web (Wilson & Lowry, 2001).

- Permettre l'accès à des documents riches (Kommers et al., 1996)
- Encourager les interactions entre apprenants et documents
- Favoriser le travail de groupe, en particulier l'entraide, l'émulation et le questionnement.

De nombreux exemples de plateformes et de travaux sont basés sur le modèle constructiviste. Le projet FORMID (Gueraud, Adam, Pernin, Calvary, & David, 2004) vise à mettre en place des Situations d'Apprentissage Actif ou Situations Actives d'Apprentissage (SAA) dans lesquelles l'apprenant interagit étroitement avec un dispositif. Ce dispositif induit par les découvertes qu'il favorise la construction de connaissances. Ce projet repose sur l'utilisation d'Objet Pédagogique Interactif (OPI) pour désigner le dispositif logiciel support de ce type d'apprentissage. Orienté formateur, le projet propose d'assister ceux-ci dans la mise en place des SAA, cette assistance repose sur l'utilisation de scénarios pédagogiques et sur la réalisation d'un prototype d'environnement appelé plateforme FORMID.

3.4.4 Médiation et Socioconstructivisme

Ce modèle repose sur l'idée que l'on apprend en se confrontant aux pairs et que l'apprentissage tire sa source de l'interaction sociale. En enseignement, ce modèle suppose pour être mis en œuvre de proposer des activités permettant la confrontation (mise en scène du conflit socio cognitif). La théorie médiationnelle (Vygotski, 1985) propose, quant à elle, le concept de zone proximale de développement, étayage et médiation. Médiation et socioconstructiviste se différencient du constructivisme dans la mesure où selon Vytgovsky, les savoirs enseignés se distinguent des savoirs quotidiens construits à travers l'expérience pratique. Pour appréhender les savoirs enseignés l'apprenant n'agit pas directement sur le monde, mais par l'intermédiaire de différents types de représentation. L'école permet l'appréhension de ces représentations, langage, mathématiques etc...L'apprentissage de ces différents outils (offerts par l'école pour appréhender le monde) est essentiellement social. Mettre en œuvre un tel modèle sur une plateforme de formation nécessite de porter une attention accrue aux canaux de communication mis à disposition des apprenants.

- Outils d'administration permettant de créer entités et liens composant le dispositif pédagogique (cette partie sera détaillée plus loin dans le chapitre).
- Outils de rétroaction de groupe synchrone et asynchrone,
- Outils de production de groupe
- Outils de résolution de conflit
- Outils de workflow permettant de favoriser le conflit de le mettre en scène.

De nombreux travaux mettent en relation une plateforme de formation avec ce modèle. Ainsi (Derycke, Hoogstoel, & Vieville, 1997) décrit les fonctionnalités proposées par le Campus Virtuel pour la mise en œuvre de travail coopératif. Une des fonctionnalités du campus virtuel a été étendu, il s'agit d'Accel, cette plateforme de formation est utilisée actuellement dans notre institut, elle est basée sur la métaphore du Forum (D'Halluin & Delache, 2006). Dans le cadre de la formation des maîtres à L'UFR de Lille, nous avons également décrit l'utilisation

de Groupware en tant que plateforme de formation dans le cadre d'un apprentissage collaboratif (Caron & Midenet, 2002).

La plateforme Acolad (Apprentissage Collaboratif A Distance) permet l'intégration à la fois technique et pédagogique des moyens de communication, ainsi que l'implantation d'outils de coordination et de partage, pour des petits groupes d'apprentissage (Faerber, 2001). Le CMS PostNuke a été utilisé en tant que plateforme socio constructiviste par (Schneider, 2003). La plateforme Moodle a été directement conçue pour mettre en œuvre le modèle socio constructiviste. (Dougiamas & Taylor, 2002)

Etablir une corrélation étroite entre les fonctionnalités d'une plateforme et le modèle pédagogique qu'elle privilégie n'est pas chose aisée. Comme nous allons le voir dans la section suivante, cette difficulté résulte essentiellement de l'aspect modulaire des plateformes de formation, des évolutions et détournements qu'elles subissent au cours de leurs usages.

3.5 Plateformes de formation et modèles pédagogiques

3.5.1 LMS et modèles pédagogiques

Les LMS telle Ganesha, WebCT, Claroline, Open USS, proposent des fonctionnalités de dépôts offertes aux enseignants et disponibles aux élèves. Des outils de communication sommaire permettent dans le cas d'un enseignement semi présentiel (Competice) de mettre en œuvre des activités de groupe. La prise en charge du standard (SCORM), que nous décrirons dans le chapitre 4, permet la composition de cours structurés au scénario simple. L'importation de test de type QCM, respectant par exemple la norme (IMS-QTI) permet de mener des évaluations formatives et l'utilisation du standard (IMS-LD) permet de mettre en œuvre des scénarios de groupe.

L'ensemble de ces fonctionnalités semble correspondre à plusieurs des théories pédagogiques que nous venons de décrire, attachons nous à étudier quelques LMS en particulier.

Net Université (Giacomini-Pacurar, Trigano, & Alupoiaie, 2005) est une plateforme implémentant les spécifications IMS Learning Design (niveaux A, B et C) pour la représentation des contenus pédagogiques. Le Standard QTI de l'IMS (IMS-QTI) (Questions et Test Interoperability) a été utilisé sur cette plateforme pour l'intégration des exercices interactifs. Bien qu'une telle plateforme, par les fonctionnalités qu'elle propose, semble orientée vers l'apprentissage behaviouriste et cognitiviste, elle propose également la prise en charge d'activité constructiviste et socio constructiviste. La plateforme Net Université offre la possibilité d'ajouter des outils de communication à des cours en ligne, dans le but de motiver davantage les étudiants dans la réalisation de leurs tâches d'apprentissage.

L'environnement d'apprentissage à distance Univ-R^{ct} (anciennement Acolad (Acolad)) (tel qu'il est présenté par ses auteurs) est adapté au travail collaboratif en petits groupes, au tutorat à distance et à la mise en oeuvre d'un mode d'apprentissage par projet ("Problem Université Learning") au sein d'une communauté d'apprentissage.

La plateforme Moodle (Dougiamas & Taylor, 2002) (Dougiamas & Taylor, 2003), référencée comme LMS, a été orientée dans sa conception pour privilégier les modèles pédagogiques Constructivisme, Socioconstructivisme. A ces modèles pédagogiques, ils faut aussi ajouter les modèles Constructionisme, et "Lié\détaché", que les concepteurs de la plateforme définissent comme suit :

- Le constructionnisme avance que l'apprentissage est particulièrement efficace lorsque l'on construit quelque chose pour que d'autres l'expérimentent.
- Le lié/détaché s'intéresse aux motivations des individus dans le cadre d'une discussion : trois comportements y sont étudiés, le comportement lié est qualifié de subjectif, le comportement détaché est objectif, le comportement construit indique la capacité de choisir entre lié et détaché.

Malgré la conception de Moodle prévue pour les usages que nous venons de décrire, des exemples existent de détournement de ce LMS pour une utilisation Behaviouriste, ainsi trouve-t-on des adaptations permettant de d'importer et de prendre en compte des QCM Hot Potatoes, des tutoriaux expliquent comment utiliser cette plateforme comme répertoire d'objets pédagogiques. (Touvet, 2005)

3.5.2 CMS et modèle pédagogique

L'utilisation d'un CMS répond à des objectifs de gestion de document. Il facilite l'indexation et le dépôt de documents avec des métadonnées (LOM, Dublin Core Education). Dans le cadre d'une pédagogie behaviouriste et cognitiviste, l'utilisation d'un CMS permet de séparer la forme, du fond de la leçon. Ce type de plateforme favorise donc la réutilisation de documents pour la constitution d'une nouvelle leçon. (Université, Cailleau, Crozat, Majada, & Spinelli, 2002). Dans le cadre d'une pédagogie constructiviste et socio constructiviste (étude de cas, projet d'élèves, travaux de groupe), un CMS permet l'indexation des documents recueillis avec les métadonnées qui ont servi à les trouver, il offre un espace de partage et de dépôt pour les productions individuelles ou collectives. Les CMS sont ainsi utilisés pour mettre en place dans le cadre du CSCL des pratiques collaboratives en enseignement, ils peuvent par exemple être utilisés pour élaborer des environnements pour l'apprentissage constructiviste (CLEs) (Hung & Chen, 2003).

3.5.3 C3MS, outils modulaires et modèles pédagogiques

Un C3MS (PostNuke, Joomla) ou un LMS modulaire (Moodle, Claroline), propose une base minimaliste d'outils. Chaque fonctionnalité peut être ajoutée ou retirée. La modularité permet de rendre totalement polymorphe ces plateformes. Dans le cadre d'une mise en œuvre d'un modèle behaviouriste, il est alors possible de proposer des outils permettant l'importation et la mise en œuvre de test. Dans le cadre d'un enseignement plus collaboratif, l'ajout ou le retrait de fonction est préconisé dans la mesure où il est un indicateur de la vitalité de la communauté d'apprentissage (Schneider, 2003) (D'Halluin & Delache, 2006). Lorsque la demande est faite, de nouveaux outils peuvent être ainsi être ajoutés, comme les besoins d'éditer en groupe des documents, d'être avertis de la présence des autres en ligne en même temps que soi, d'être tenu au courant des contributions des autres et des commentaires sur ses propres contributions. Ces besoins sont désignés par l'expression "social awareness". Les modules qui favorisent la satisfaction de ces besoins en supportant l'émergence et la vie d'une communauté participe d'un même processus émergent (Preece, 2004). Les fonctionnalités à remplir par des modules participant à l'émergence d'une communauté de pratique, peuvent être spécifiées. Un prototype générique de plateforme supportant un apprentissage socio constructiviste a d'ailleurs été défini (Chewar, McCrickard, & Carroll, 2003).

3.5.4 Application Web 2.0 et modèle pédagogique

Une application de type Web 2.0 est une application souvent conçue initialement autour d'un nombre limité de fonctionnalités. Ainsi Blog, Wiki, Forum, e-portfolio, e-group, sont des outils aux fonctionnalités simples que l'on peut d'ailleurs rencontrer en tant qu'outils intégrés à de nombreuses plateformes de formation. Or il existe des exemples d'utilisation de ces outils, éventuellement modifiés, en tant que plateforme de formation. Ces artefacts subissent alors

des détournements (Clot, 2003) (Béguin & Rabardel, 1999) dont nous essaierons, dans le chapitre 4 et le chapitre 8, de comprendre l'origine et la nécessité. Nous pouvons citer comme exemple significatif, l'utilisation et le détournement de Forum (Bounie, Catté, D'Halluin, & Réthoré, 2005) (George, 2005), du Wiki (Caron, 2006) (Schneider, 2003), d'un e-portfolio (Sjunnesson, 2001), d'un Blog (Mortensen & Walker, 2002).

Les détournements dont font l'objet ces outils ne permettent pas de prévoir leur utilisation finale et donc d'estimer leur adéquation à un modèle pédagogique unique. Par exemple un forum favorise des activités relevant a priori du modèle socio constructiviste (E. De Vries, 2001), cependant il est possible de mettre en évidence le caractère particulier des échanges réellement induits par ce type d'outil. (George, 2001), (Ciussi, Simonian, & Augier, 2004) ont montré que les relations verticales apprenant-formateur sont nettement prédominantes et que les relations horizontales entre apprenants demeurent rares. Toujours pour l'outil Forum, (Touvet, 2005) explique l'usage en tant que dépôt de document SCORM qu'il peut en être fait. Dans le même esprit (D'Halluin & Delache, 2005b) cite des exemples de détournement de wiki pour stocker des photos.

Avec l'avènement de technologie de type Ajax (Garrett, 2005), et de mécanisme de MashUp (Floyd, 2006), (cf. chapitre 12) l'agrégation hétérogène de fonctionnalité Web 2.0 permet à l'apprenant de concevoir lui-même son environnement d'apprentissage. Cette nouvelle évolution contribue en EIAH, à rendre totalement caduque toute classification simple d'une application Web en fonction des modèles pédagogiques.

3.6 Conclusion

3.6.1 Limite d'une typologie basée sur les modèles pédagogiques

Outre les critiques intrinsèques portant sur la pertinence des quatre modèles pédagogiques précédents. (Giordan, 1994), établir une typologie des plateformes de formation se heurte essentiellement à deux difficultés.

La première est qu'il est difficile d'établir une relation stricte entre les catégories CMS, LMS, LKMS, C3MS, Application Web 2.0 et les modèles pédagogiques. Nous avons vu que même si l'artefact a été prévu pour correspondre à un modèle pédagogique particulier, les détournements qu'il subit le rendent apte à être utilisé dans le cadre d'un autre modèle. Ce point sera abordé de nouveau au chapitre 8.

La seconde difficulté est pour les enseignants à définir leur pratique comme relevant uniquement d'un modèle pédagogique. Comme le souligne Josiane Basque, chaque modèle pédagogique présente des aspects intéressants qui peuvent être retenus dans telle ou telle situation de design pédagogique (Basque, 1999). Les modèles pédagogiques doivent en effet être considérés comme des outils d'analyse plutôt que des outils permettant la construction d'un cours "*each learning scenario can be viewed and analyzed from any of these positions*" (Allert, 2004) De même Thierry Nodenot constate : "*Dans leur pratique quotidienne les enseignants utilisent avec discernement le potentiel de chaque théorie*" (Nodenot, 2006a). Pour une institution, s'enfermer dans un modèle pédagogique, comporte le danger de se couper des pratiques réelles des enseignants. "*A trendy mixture of constructivist, social-interactionist, and situationist approaches...is often translated into a total banishment of teaching by telling, an imperative to make cooperative learning mandatory to all, and a complete delegitimization of instruction that is not problem-based or not situated in a real-life context.*" (Sfard, 1998)

Les modèles pédagogiques, tout comme les distinctions entre différents types de plateformes de formation, contribuent à la description des usages menés par les enseignants sur ces plateformes. Néanmoins il n'est pas possible d'établir des correspondances strictes entre usages, plateformes et modèles. Il est par contre possible d'établir un tableau décrivant les usages des fonctionnalités d'une plateforme de formation en fonction des modèles pédagogiques. Un tel tableau a été réalisé, il est consultable en annexe de cette thèse (cf annexe). En particulier ce tableau illustre le manque apparent de standard pour aider à l'expression d'activité socioconstructiviste sur une plateforme de formation. Cette constatation, constitue la genèse de notre recherche en EIAHs. Dans le chapitre 4 suivant, nous étudierons donc les rapports entre l'activité de création qu'un enseignant peut mener sur une plateforme de formation et les standards usuels des EIAHs. Dans la dernière sous-section de cette conclusion, nous posons les premiers jalons de notre proposition : substituer à la notion de standard la mise en œuvre par une infrastructure adaptée, un processus instrumenté permettant de construire sur des plateformes de formation des dispositifs pédagogiques.

3.6.2 Pas de standard pour construire des dispositifs complexes sur des plateformes de formation.

Une des caractéristiques d'une activité socioconstructiviste est de mobiliser des dispositifs complexes. Cette complexité peut s'exprimer de deux manières :

La première fait appel à des fonctionnalités proposées par la plateforme de formation. Schneider, (Schneider, 2003) Dougiamas (Dougiamas & Taylor, 2003), D'Halluin (D'Halluin & Delache, 2006) expliquent comment le recours à des portails Web, riche et modulaires permet de mettre en œuvre à des activités de ce type. Dans le catalogue édité par le TECFA (Synteta, Schneider, & Frété, 2002) il n'est par rare qu'une activité mobilise une dizaine d'outils différents.

La deuxième manière d'exprimer un dispositif complexe est de choisir des outils a priori moins riches, Wiki (Caron et al., 2006b), Blog (D'Halluin & Delache, 2006), Forum (George, 2005) et de les structurer pour une activité précise. Ainsi pour l'environnement CoWeb, basé sur un Wiki, Mark Guzdial a montré la possibilité de créer des activités particulières en effectuant des restructuration du Wiki ou en débutant les activités sur des Wikis pré-remplis (Guzdial, Rick, & Kehoe, 2001); il a d'ailleurs édité un catalogue d'activités possibles dans le cadre de cet environnement (Guzdial, 2000). Nous verrons que l'étude de cas (cf. le chapitre 11) basés sur l'exploitation de l'application Web WikiniMST s'appuie sur ce type de structuration. Construire un dispositif sur une plateforme consiste ainsi à mener une pré structuration de la plateforme, cette pré-structuration permet de constituer des **entités** : groupe, étudiants, formateur, espace instrumenté, documents, politiques d'utilisation puis de définir des **liens** entre ces entités. **Entités** et **Liens** sont des éléments fondamentaux de la construction des dispositifs, ces éléments seront détaillés dans le chapitre 5 chapitre 8 et le chapitre 10.

Favoriser des activités socioconstructivistes sur une plateforme de formation peut donc passer par l'instrumentation de la construction de dispositif complexe.

Nous verrons dans ces chapitres qu'instrumenter la construction de dispositifs complexes dépasse largement la problématique posé par la mise en place ou non de d'activités socioconstructivistes, néanmoins nous avons jugé opportun de rappeler ici la genèse de notre recherche.

chapitre 4 Quels types de construction sur une plateforme de formation? Différentes approches pour l'enseignant.

Nous venons de le voir, les plateformes de formations sont nombreuses, polymorphes et échappent à une simple taxinomie basée sur leur fonctionnalités ou même sur les modèles pédagogiques qu'elles privilégient. Parmi les usages que des enseignants font des plateformes, il nous a semblé intéressant d'étudier les constructions qu'ils peuvent y réaliser. Dans le but de proposer des outils pour assister cette construction, il nous faut nous pencher sur les différentes approches permettant d'adresser les plateformes de formation malgré leur hétérogénéité. Jean Philippe Pernin distingue trois types d'approches pour rationaliser le déploiement des TIC dans l'apprentissage, (Pernin & Lejeune, 2004a), (Pernin & Lejeune, 2004b). Bien que nous ne partagions pas complètement cette classification, nous allons l'utiliser car elle est connue de la communauté EIAH française. Les trois approches qu'il décrit adressent l'apprentissage humain par le biais des plateformes de formation, il s'agit de :

- l'approche Documentaire,
- l'approche Centrée Processus,
- l'approche par Activités.

Nous présentons ci-dessous ces trois approches, en les situant par rapport aux standards technologiques EIAH existants. Cette étude nous permettra d'introduire notre proposition, "**l'approche par dispositif**". Cette approche par dispositif s'appuie sur des pratiques enseignantes : le bricolage pédagogique, elle peut être instrumentée par l'ingénierie des modèles, il est possible de définir pour cette approche un standard basée sur la définition des services proposés par une plateforme de formation. Nous détaillerons cette approche dans le chapitre 8 et le chapitre 10.

4.1 L'approche documentaire

L'approche documentaire propose d'encadrer par des standards : la description, le référencement, la mise à disposition et le séquençement de documents sur une plateforme de formation. Ce modèle permet à un enseignant d'adresser une plateforme au travers de la création de document (html, doc, pdf ou autre), du référencement et de l'agrégation de contenu. Inspiré par le modèle pédagogique behaviouriste et cognitiviste, l'approche documentaire nécessite de définir l'interfaçage des documents avec la plateforme de formation choisie. Ces deux activités principales (création/référencement et agrégation/interfaçage) font maintenant l'objet de normes internationales précises.

4.1.1 LOM Dublin Core education

La norme Learning Object Metadata (LOM), qui reprend l'initiative Dublin Core (Dublin-Core) elle-même complétée par les métadonnées "education", a pour objet l'ajout de métadonnées à des documents pédagogiques dans le but d'en favoriser le stockage, le référencement et la réutilisation. De telles métadonnées peuvent être complétées par des métadonnées particulières comme c'est par exemple le cas en France pour les documents du CNDP (Sceren). La norme LOM a ainsi été souvent mise en œuvre en France selon des profils particuliers, par exemple à Université de Technologie de Compiègne (Bertrand & Hennequin, 2004), (Bachimont, Cailleau, Crozat, Majada, & Spinelli, 2002). Actuellement un profil français unique de la norme LOM (LOM France) est en cours d'élaboration (AFNOR, 2005).

Ce type de référencement, adapté à des documents faiblement structurés, laisse libre l'enseignant d'ordonner lui-même ses contenus par rapport aux outils disponibles. L'usage de cette norme renvoie donc à un profil d'enseignant que Jean-Philippe Pernin classifie comme : enseignant prospecteur, référenceur et organisateur de ressources.

De nombreux dépôts d'objets d'apprentissage respectant la norme LOM sont actuellement disponibles, le site Thot référençait en septembre 2005, 52 dépôts proposant des millions d'objets (Thot). La fondation européenne Ariadne (Ariadne), a construit une infrastructure permettant le partage d'objets pédagogiques au travers de viviers de connaissances. Ce projet regroupe au sein d'un consortium près de 30 partenaires européens (dont 20 universités) et collabore au sein du projet Globe (Global Learning Objects modèles Exchange) (Globe) avec les projets Merlot aux Etats Unis) (Merlot), EdNA Online en Australie (Edna), LORNET au Canada (Lornet) et NIME (NIME) au Japon.

Pour compléter ce modèle de métadonnées, des normes ont également été élaborées, elles permettent de préciser l'agrégation de documents, leur séquençement et leur interfaçage avec une plateforme de formation.

4.1.2 SCORM

Scorm (Shareable Content Object Reference Model) (SCORM) est une spécification regroupant un ensemble de normes déjà existantes (AICC), (LOM), (IMS-Packaging). Elle est proposée par "Advanced Distributed Learning", sous l'initiative de l'armée Américaine, et définit l'agrégation d'objets pédagogiques (Shareable Content Object : SCO), la navigation de l'un à l'autre, ainsi que la communication entre ceux-ci et la plateforme de formation.

Par exemple, un SCO "composite" peut lier deux SCO ensembles, comme un cours et un quiz. Il est alors possible de définir des métadonnées décrivant l'ensemble, puis de définir un séquençement simple ou conditionnel entre les différents SCO. L'encapsulation Scorm permet ainsi à la plateforme de naviguer de l'un à l'autre, et au client de modifier des variables de la plateforme associées à ce cours, comme par exemple le score ou l'état d'avancement du cours. Cette modification se fait par la mise en œuvre sur le client de scripts exprimés en ECMAScript (ECMAScript).³

Le standard Scorm a été créé afin de permettre le partage et la réutilisation des objets d'apprentissage. Puisque les objets Scorm contiennent un protocole universel de communication avec la plateforme, ils n'ont pas besoin d'être re-crés lorsque l'on choisit de changer de plateforme. De plus, comme Scorm intègre la spécification LOM, il est possible de rechercher un objet pédagogique dans des bibliothèques, par rapport aux métadonnées qui caractérisent son contenu. Des logiciels auteurs permettent très facilement à l'enseignant d'intégrer ses documents dans des fichiers compatibles Scorm : Dreamweaver et son extension CourseBuilder, Flash suite (Adobe) sont des exemples d'éditeurs très utilisés, pour produire des contenus sur une plateforme de formation. La production de petits composants permettant l'évaluation des élèves (qcm, questionnaire à trou etc....) en étant adjoint au contenu permet de compléter un package Scorm, l'outil Reload (Reload) permet la construction manuelle de tels packages. Des plateformes de formation importent et interfacent alors ce type d'objet pédagogique. (Par exemple Ganesha, Claroline, Moodle)

³ ECMAScript est une tentative de normalisation du noyau du langage javascript.

Le modèle documentaire, propose à l'enseignant une place de choix dans la maîtrise des objets pédagogiques; il montre néanmoins des limites que nous allons préciser. Nous résumons dans les deux sous sections suivantes, les limites du modèle documentaire.

4.1.3 Limites du standard LOM

Dès son adoption par IEEE en 2002 le standard LOM a fait l'objet de réserves portant sur :

- La définition d'un objet pédagogique (granularité, structuration).
- La pédagogie qu'elle implique.
- L'absence de prise en compte des activités induites.
- Le paradigme "share, reuse".
- Les différents profils du standard.

La définition du terme objet pédagogique est sujet à débat. Ainsi dans *"Three Objections to Learning Objects and E-learning Standards"*, Norm Friesen explique que le concept d'objet pédagogique est ambigu, il montre que l'utilisation du LOM dans le cadre du standard SCORM renvoie à une conception de l'enseignement commerciale, correspondant plutôt à de l'apprentissage individuel. La neutralité affichée du standard lui semble incompatible avec le mot enseignement, qu'il présente comme devant être activement engagé. Enfin les efforts de standardisation du LOM et de SCORM portent selon lui l'empreinte de l'idéologie du complexe militaro-industriel américain (Friesen, 2004). Jean Philippe Pernin insiste sur manque de séparation formelle entre les entités de structuration et les contenus (ressources, media), il montre que le LOM n'est pas compatible avec toutes les approches, et relève le problème posé par l'indexation d'un matériel brut (Pernin, 2003). Yolaine Bourda présente les problèmes de structuration et granularité que posent la définition des objets pédagogiques. (Bourda, 2001). David Wiley aborde quant à lui le problème de décontextualisation que ces objets pédagogiques requièrent pour être utilisés. *"Focusing on removing or separating as much context as possible from learning objects in order to maximize the reuse of the learning objects can be problematic."* (Wiley et al., 2004). La non prise en compte de l'activité induite par l'objet pédagogique a été également stigmatisé par Claude Vieville et Brigitte de La Passadiere (Vieville & Passardière, 2003). Les auteurs ont montré le problème posé par la mise en œuvre du référencement de ressources utilisées dans des formations à l'université de Lille1. Ils ont ainsi remis en cause des éléments de cette norme et ont été amenés à ajouter quelques métadonnées. La norme LOM ne prenant en compte que les objets d'apprentissage; une tentative a été menée pour mettre en place une grammaire capable de représenter l'activité d'apprentissage.

L'objectif du LOM est d'indexer des objets d'apprentissage pour les réutiliser, or ce paradigme selon lequel la réutilisation d'objet pédagogique pourrait correspondre à une pratique enseignante ne correspond pas à la réalité (Chartron, 2004) (Pernin, 2004). La réalité correspondant plutôt à une situation d'autoproduction des ressources pédagogiques, et non de partage.

"The dream of improving educational outcomes via re-usable e-learning content is proving to be a myth at best, a nightmare at worst, as millions of dollars are spent on content development across the world, and yet few teachers ever re-use (let alone adapt) the products of this development." (Dalziel, 2005)

Enfin, le constat a été fait que le LOM est rarement utilisé tel que définit par le standard IEEE de 2002 (IEEE, 2002); de nombreux profils d'application viennent raffiner les définitions, ajouter des éléments, spécifier des valeurs, rendre obligatoires des éléments facultatifs - ce qui

rend extrêmement difficile la préservation de l'interopérabilité avec le schéma de base. Au Canada, l'initiative CanCore (CanCore) propose un profil d'application des métadonnées du LOM; en France le groupe de travail G4 de l'Afnor est chargé de la définition d'un profil d'application. La base choisie pour ce profil a été ManUeL, un profil d'application du LOM réalisé pour C@mpusScience (Passadière & Jarraud, 2004). Bien que la publication d'une norme expérimentale ait eut lieu en août 2005 (AFNOR, 2005), le groupe G4 poursuit ses recherches pour proposer une solution séparant aspect conceptuel et aspect représentation. En effet les travaux du groupe ont mis en évidence les incohérences associées à la confusion de ces deux aspects lors de l'établissement des profils particuliers du LOM.

4.1.4 Limites du standard SCORM

Le standard SCORM comporte certaines limites déjà décrites lors de l'étude du standard LOM, en particulier au sujet du type d'activité que ce standard décrit, (Friesen, 2004) (Dessus & Schneider, 2006). On retrouve de même, les réserves concernant le paradigme *"Share and Reuse"* porté par la métaphore de la brique Lego (Bourda, 2001). Cette métaphore se heurte à l'hétérogénéité des contenus tant sur le plan du nombre des institutions impliquées dans leur production que sur le plan des pédagogies utilisées ou des techniques mise oeuvre. Nous ne sommes pas dans un monde où les briques lego ne sont produites que par une seule entreprise. (Crozat, Majada, & Spinelli, 2003) Enfin, le standard Scorm faisant référence au LOM, on retrouve des critiques portant sur la difficulté de faire apparaître au sein d'une séquence le contexte pédagogique. *"The role of context is simply too great in learning, and the expectation that any educational resource could be reused without some contextual tweaking was either naive or stupid"* (Wiley, 2002).

Des limites spécifiques au standard Scorm peuvent cependant être énoncées, ces critiques portent essentiellement sur :

- Le principe d'agrégation et la nécessaire contextualisation
- Les choix techniques relayé par le standard

La norme SCORM prévoit trois grands niveaux de granularité, les "assets", les "SCO" et les "tagged content aggregation". Un asset représente le niveau le plus bas de granularité, c'est par exemple une image, une animation flash. Un SCO représente un regroupement d'asset, encapsulé par du code ECMAScript permettant le dialogue avec la plateforme. Un tagged content aggregation représente un package comportant des SCO, et un séquençement exprimé sous forme d'un fichier XML permettant de les présenter et de gérer leur accès. Le problème vient que bien souvent un SCO est égal à un fichier HTML (Delestre, Frénot, Mottelet, & Vayssade, 2002) Or un fichier html contient les informations à présenter mais aussi la façon de les présenter, ce qui en terme d'agrégation pose des problèmes d'IHM (Bachimont & Crozat, 2004). Un autre problème posé par l'agrégation de contenu concerne la nécessité de faire apparaître des éléments de contextualisation pédagogique (absent dans les "assets" pour en préserver l'interopérabilité) au niveau des packages. (Gebers, 2005), cette possibilité est absente des premières versions du standard. La version 2004 du standard prévoit la possibilité de placer des "assets" aux trois niveaux de hiérarchie, ce qui permet donc d'intégrer, lors de l'agrégation, des "assets" particuliers comportant les renseignements de contextualisation pédagogiques souhaités, mais qu'en est il alors de l'indépendance, dans une optique de réutilisation, du package obtenu?

Techniquement le choix de lier le standard SCORM à un standard de script pour l'interfaçage avec des plateformes, a permis, de proposer une mise en oeuvre très simple du standard, mais

est un frein à son utilisation dans un monde évoluant vers des applications distribuées et des aspects multiples d'accès à l'information (téléphone, assistant personnel, ordinateur). Lors de l'utilisation d'un package SCORM dans un environnement composé de diverses fonctionnalités Web agrégées par des technologies de type MashUp (Floyd, 2006), les contraintes du protocole de communication sur lequel repose le Standard SCORM ne permettent pas d'établir une communication correcte avec la plateforme (Engelbrecht, 2003). SCORM a été pensé pour être exécuté seulement via un navigateur Web, l'intégration de techniques, comme les frames HTML et l'ECMAScript l'empêche d'être utilisé par d'autre mode de navigation. Ce protocole de communication sur lequel repose SCORM pose également des problèmes lors de la mise en œuvre d'interactions Multimodales, il ne permet pas l'usage d'objets plastiques, comme le montre des travaux menés dans notre laboratoire (Bouyer, 2006).

4.1.5 Conclusion à l'approche documentaire

L'approche documentaire est essentiellement portée par deux standards, le LOM et le SCORM, pour que cette approche soit efficace il faut qu'elle relève des défis portant sur l'indexation, l'interopérabilité, l'intégration des services, la mutualisation. (Pernin & Lejeune, 2004b). Or nous venons de voir que chacun de ces points rencontre des limites. Nous avons abordés les choix techniques impliqués par le standard et montré les limites engendrées en terme de profil spécifique, architecture distribuée et capacité à relever les nouveaux défis portés par les EIAHs plastiques et ubiquitaires.

Si l'on aborde ces standards dans une perspective enseignante, ils renvoient une image double du métier de l'enseignant :

- L'enseignant concepteur et prescripteur de contenu
- L'enseignant prospecteur, "référenceur" et organisateur de ressource (Pernin & Lejeune, 2004b).

Dans le cas de l'enseignant concepteur, les expériences menées depuis près de six ans ont montré que la création d'objet pédagogique avait lieu essentiellement pour des usages internes. (Bourda, 2001) Une nouvelle limite liée à la rationalisation apparaît : l'enseignant se trouve alors contraint de respecter la logique précise d'apprentissage relayée par son institution ce qui conduit à un appauvrissement de son rôle créatif, relégué à la stricte création des contenus, il ne participe pas, ni à la conception des interfaces, ni à la conception des objets pédagogiquement riches (objet interactif, micro monde, simulateur). (Pernin & Lejeune, 2004b) (Stacey, 2003) Si l'on se limite à regarder le rôle qu'il tient lorsqu'il est organisateur de ressources, il faut constater qu'en laissant à l'enseignant le soin d'agréger et d'encapsuler des objets pédagogiques, on ne fait que déplacer le problème que pose leurs créations.

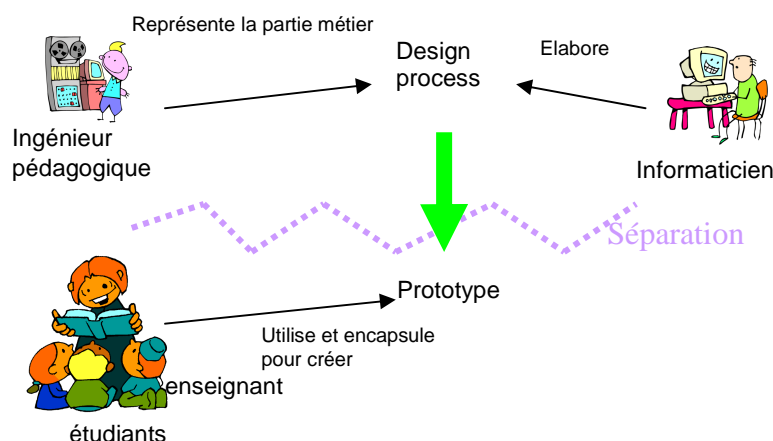


Figure 5 Cycle de création d'un objet pédagogique dans le modèle documentaire

Un objet pédagogique est donc créé par trois personnes différentes, comme l'illustre la Figure 5. Tout d'abord, un ingénieur pédagogique conçoit le contenu et la forme de chaque objet pédagogique. Celui-ci est ensuite créé par un informaticien avec un outil approprié (par exemple CourseBuilder, Flash suite, ...). Une fois cette étape réalisée, l'enseignant crée son cours en encapsulant les différents objets nécessaires qu'il a collectés. Ce processus ne permet pas à l'enseignant une réelle créativité dans la mesure où il ne maîtrise pas la définition des objets initiaux. L'enseignant ne fait que prospecter et assembler des ressources mises à sa disposition.

La Figure 5 illustre la séparation entre l'activité créatrice menée par l'enseignant et celle qui est menée en amont pour élaborer les objets pédagogiques interactifs. Pour que l'enseignant puisse être associé au processus de création d'objets riches, il faut qu'il soit associé à l'ingénieur pédagogique et à l'informaticien dès le début du processus de création pédagogique. Une telle démarche est proposée par les méthodes d'ingénierie cognitive, cette démarche fait l'objet de la section 4.2 suivante.

4.2 L'approche basée sur une méthode d'ingénierie cognitive

4.2.1 Introduction

Cette approche consiste à appliquer une méthode d'ingénierie dans le but de rationaliser l'analyse, la conception, la réalisation et la planification de la diffusion des systèmes d'apprentissage. Cette méthode que propose notamment Gilbert Paquette et l'équipe du LICEF/CIRTA, intègre des concepts, des processus et des principes de trois domaines : le design pédagogique, le génie logiciel et l'ingénierie cognitive (Paquette, De la Teja, Lundgren-Cayrol, Léonard, & Ruelland, 2002). Son but est de décrire à la manière du génie logiciel, une approche pour concevoir et construire des systèmes d'information, en les adaptant à la conception des systèmes d'apprentissage.

Nous verrons, dans le chapitre 6 chapitre 8 et le chapitre 9, qu'il est effectivement possible de considérer un dispositif pédagogique sur une plateforme de formation comme un système d'information particulier. L'ingénierie que nous proposons d'appliquer, l'ingénierie dirigée par les modèles, se distingue de l'approche que nous décrivons. En effet, l'ingénierie dirigée par les modèles permet d'établir une distinction entre considération métier et considération technique. En particulier le recours à ce type d'ingénierie nous permet de faire intervenir les enseignants dès le début du cycle de conception, sans qu'ils aient à s'exprimer dans un cadre

aussi formel que la méthode MISA (méthode d'ingénierie d'un système d'apprentissage) proposé par Gilbert Paquette. Dans le cadre de la méthode MISA, la modélisation pédagogique repose sur une classification de la connaissance humaine en trois catégories : les procédures (actions, étapes, processus etc.); les concepts (outils ou informations nécessaires aux actions ou produites par ces actions) ; les principes (règles, conseils, consignes, conditions, contraintes qui régissent l'une ou l'autre des connaissances).

Un logiciel MOT (et MOT+) (Paquette et al., 2002) permet de manipuler graphiquement facilement ces concepts et de les lier. La méthode MISA permet ainsi définir le "devis" que doit respecter le système d'apprentissage à construire. Ce devis sert de guide pour le choix ou la réalisation des briques logiciel constituant le portail qui in fine sera mis à la disposition des apprenants et formateurs. Une application du modèle de la méthode MISA se trouve dans le développement de l'atelier ADISA (Paquette, Rosca, Teja, Léonard, & Lundgren-Cayrol, 2001). Le modèle a alors servi à définir les interfaces des divers outils de l'atelier.

Ileana de la Teja a montré que la méthode MISA était compatible avec le standard IMS-LD dans la mesure elle partage avec ce standard de nombreux concepts (De la Teja et al., 2006). L'étude qu'Ileana de la Teja a menée, a montré la flexibilité du standard, et son expressivité. Les travaux actuels porte donc sur une généralisation du modeleur graphique MOT+LD pour prendre en compte les niveaux B et C du standard IMS-LD. Dans la sous-section suivante nous présentons ce standard.

4.2.2 Une présentation d'IMS-LD

Créé en 1997, le consortium IMS a pour objectif de définir et promouvoir des spécifications destinées à faciliter l'interopérabilité des technologies de l'éducation. Au sein des différentes spécifications d'IMS se trouve Learning Design. Cette spécification, issue du langage Educational Modeling Language de l'Open University of Netherlands définit un langage qui permet la description de processus pédagogiques essentiellement électroniques (online pedagogies).

Les points forts d'IMS-LD (IMS-LD) seraient les suivants :

- Support de tout type de pédagogies (Koper, 2001).
- Langage de description pédagogique suffisamment informatique (formel) pour être interprété par un ordinateur dans le but de créer de mettre en scène des activités sur des plateformes.
- N'est pas rattaché à une classe particulière de plateforme

De nombreux outils outillent la norme, ces outils sont de trois sortes :

- Interpréteur : CopperCore, SLED
- Editeurs simples : Reload, CopperAuthor, Cosmos, Alfaneteditor ou graphique : MOT+, ASK LDT, LAMS
- Environnements intégrés : Edubox (EML) et Alfanet (McAndrew, Nadolski, & Little, 2005)

Le langage IMS-LD utilise la métaphore théâtrale pour expliciter un scénario pédagogique spécifique. Le séquençement d'un scénario pédagogique se fait aux travers des méthodes, pièces et actes. Le scénario démarre par la définition d'un ou plusieurs scénarios (Method). Un scénario est constitué de plusieurs pièces (Play) qui sont indépendantes des unes des autres. Chaque pièce est composée d'actes (Act). C'est dans les actes que sont définis les partitions (role_part). Une partition est l'affectation d'une activité à un rôle. Pour un rôle, une

seule affectation d'activité est possible par acte. Au sein d'un acte, toutes les activités se déroulent en parallèle. La fin d'un acte indique la fin des activités incluses. Une pièce est donc un ensemble d'affectation d'activités aux rôles cadencés par les actes de la pièce (les actes constituant un ensemble ordonné). Il y a deux types d'activités "utilisables" : l'activité d'apprentissage (`learning_activity`) qui définit une activité qui sera fait par tous les acteurs (i.e. jouant le rôle) en parallèle ; l'activité d'encadrement (`support_activity`) qui définit une activité qui sera réalisé autant de fois qu'il y a d'élèves (ex : une correction de devoirs). Il y a deux types de rôles : ceux utilisés pour définir des rôles tenus par les élèves (`learner`) et ceux par le personnel enseignant (`staff`). Enfin chaque partition définit l'environnement (`environment`) dans lequel sera effectuée l'activité : documents pédagogiques, services électroniques... Ceux-ci constituent la majorité des concepts de description de niveau A. IMS-LD définit deux autres niveaux. Le niveau B permet un séquençement plus fin en associant des propriétés à n'importe quel élément et en définissant des conditions sur les méthodes, les scènes, les actes ou les partitions. Le niveau C permet une gestion événementielle des actions par l'ajout du concept de notification.

Nous allons maintenant préciser notre cadre de recherche en étudiant plus précisément le standard IMS-LD.

4.2.3 Modéliser avec IMS-LD?

IMS-LD est un standard où la place de la modélisation UML est prédominante. Le standard est en premier lieu défini par rapport à un métamodèle exprimé en UML (IMS-LD). Modéliser un scénario pédagogique consiste donc à produire un fichier XML respectant le schéma IMS-LD, celui-ci traduisant sous une forme XML le métamodèle UML du standard. On imagine aisément, les difficultés, pour des enseignants, à mener cette tâche. De très nombreuses propositions ont donc été formulées pour accompagner l'enseignant. Parmi celles-ci citons quelques propositions qui portent sur la mise au point de gabarit ou de pattern (Hernández-Leo et al., 2005), sur la classification de scénarios de référence (Paquette, 2006), sur la modélisation graphique au service de l'ingénierie cognitive (MOT+ (De la Teja et al., 2006), ExploraGraph (Villiot-Leclercq, 2006)), sur des méthodes de modélisations reposant sur un moteur d'inférence (net Université projet Cepiah (Giacomini-Pacurar et al., 2005)), sur la définition d'un profil UML adapté au Problem Based Learning (Laforcade, Nodenot, & Sallaberry, 2005) et sur la modélisation directement intégrée au moteur d'exécution (CopperCore, Lams, futures versions de Moodle (Burgos, Arnaud, Neuhauser, & Koper, 2005)).

Deux types de critiques sont néanmoins formulés à l'encontre du standard,

- Les premières stigmatisent la difficulté de modéliser dans un langage a priori très complexe et éloigné des pratiques enseignantes. Les initiatives présentées ci-dessus tendent à réduire le fossé entre pratique et modélisation.
- Les secondes portent sur la difficulté de traduire des situations d'apprentissage collaboratif à l'aide du langage IMS-LD; d'où la proposition de nombreuses extensions qui nuisent à la standardisation voulue.

Ainsi Christian Martel propose-t-il une solution s'appuyant sur le métamodèle LDL (Learning Design Language) (Martel, Vignollet, Ferraris, David, & Lejeune, 2006) conçu pour permettre l'articulation des aspects organisationnels et sociaux tels qu'ils sont supportés par la plateforme de formation. Le langage basé sur la métaphore des enceintes (Martel, 1998) est

opérationnalisable à grande échelle sur le portail de formation "le cartable électronique" (Martel et al., 2004)

Thierry Nodenot, montre la difficulté qu'il y a à modéliser une activité simple de type "Problem Base Learning" (PBL) en IMS-LD. Cette difficulté résulte d'un modèle d'information insuffisant lorsqu'on l'associe à des activités collaboratives. Il propose la modélisation d'un scénario pédagogique à l'aide du langage "Cooperative Pbl Metamodel" (CPM), ce langage est doté d'un éditeur mais ne dispose pas encore de moteur d'exécution (Nodenot, 2006b), (Laforcade, 2004).

C'est cet apparent paradoxe entre le standard IMS-LD centré sur la notion d'activité et la difficulté de rendre réellement compte des activités collaboratives que nous allons explorer dans les chapitres suivants. Pour cela nous allons étudier les différentes définitions et usage de la notion de scénario pédagogique. Les différences constatées nous semblant assez révélatrices des positions à adopter pour proposer une approche permettant d'adresser des activités collaboratives.

4.3 L'approche par activité, la notion de scénario

4.3.1 Introduction

Dès sa création IMS-LD a été perçu comme établissant une rupture avec l'approche documentaire pour privilégier une approche par activité. Le standard sépare en effet très clairement les contenus du séquençement des activités. *"(...) the current learning object centric view of the e-learning world should evolve to a learning activity centric view. Here, centre stage is given to the performance of individual and group learning activities designed to attain learning objectives and, in the process, making use of learning objects."* (Tattersall & Koper, 2004) Activités et rôle sont centraux dans ce nouveau standard. Ils mettent le focus sur ce que l'apprenant doit réaliser plutôt que sur les objets qu'il doit manipuler (McAndrew, 2003). Cette nouvelle approche a eu comme conséquence de focaliser la communauté EIAH sur la notion de scénario. Un débat semblable au débat autour de la définition du Learning Object a permis de proposer différentes définitions pour un scénario pédagogique. Nous proposons de nous attacher à décrire trois exemples de définitions concrètement opérationnalisés dans le cadre des EIAHs.

4.3.2 Un scénario pédagogique pour Daniel Schneider

Lors de la réalisation du module pScénario de PostNuke, Daniel Schneider a défini un scénario pédagogique comme étant une séquence orchestrée de phases dans lesquelles les apprenants ont des tâches à effectuer et des rôles spécifiques à jouer. (Schneider, 2003). Cette définition est précisée dans le Catalogue édité par le TECFA (Frete, Schneider, & Synteta, 2002) en établissant pour chaque scénario présenté une mise en correspondance stricte entre des activités qualifiées d'élémentaires et des modules spécifiques de PostNuke (Synteta et al., 2002). La Figure 6 illustre cette mise en correspondance pour une activité de type glossaire.

GLOSSARY activity (scenario)		
Phases	Generic activities	
1	participants identify interesting "words"	IntroWork, BrainStorm
2	agree on a provisional list	EditGlossary or EditPage
3	search for information and share links	SearchWeb, EditLink
4	synthesis and editing	CoEdit
5	teacher feedback	CheckWork
6	editing of final definitions	CoEdit

Generic activities	description	Available C3MS bricks
CoEdit	creation of collaborative documents	Wiki, ContentExpress GlossaryTool
BrainStorm	generate ideas	Wiki, News Engine, forums, Bulletin Boards
...

Figure 6 Séquence : création d'un glossaire, activité élémentaire et contexte d'exécution

4.3.3 Un scénario pédagogique pour Rob Koper

Rob Koper définit un scénario pédagogique comme concrétisation du savoir didactique des enseignants lors de la réalisation d'un acte pédagogique, un cours, une séquence, l'ensemble d'un parcours. (Koper & Tattersall, 2005). Concrètement un scénario pédagogique est un package (cf. Figure 7) contenant :

- un fichier XML (manifest) qui décrit la méthode, la pièce, les actes, les rôles, les activités, l'environnement, les propriétés, les conditions et ou les notifications de la spécification, qui indique en outre les ressources qui lui sont liées
- le groupe de documents ou les ressources mentionnées dans le fichier XML (Burgos et al., 2005).

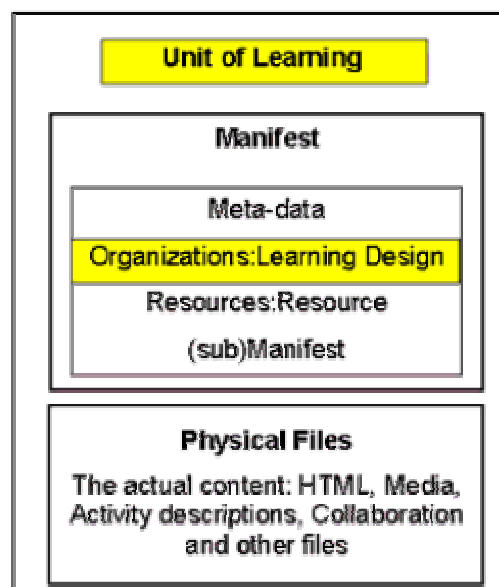


Figure 7 Package IMS comprenant le fichier XML et les ressources

4.3.4 Un scénario pédagogique pour Gilbert Paquette

Pour Gilbert Paquette la notion de scénario pédagogique est directement issue de la méthode d'ingénierie pédagogique MISA, "par le design de scénarios pédagogiques, le concepteur

établi les liens entre les sources d'informations et les différents acteurs" (Paquette et al., 2002). Techniquement chaque rôle possible d'un acteur est décrit par un graphe générique de tâches. Un tel graphe (cf. Figure 8) est appelé "scénario générique". Le scénario spécifique d'un acteur pour un système particulier est alors obtenu en particulierisant le scénario générique. Ileana de la Teja a montré qu'il était possible de rendre compatible méthode MISA avec les concepts du standard IMS-LD (De la Teja et al., 2006).

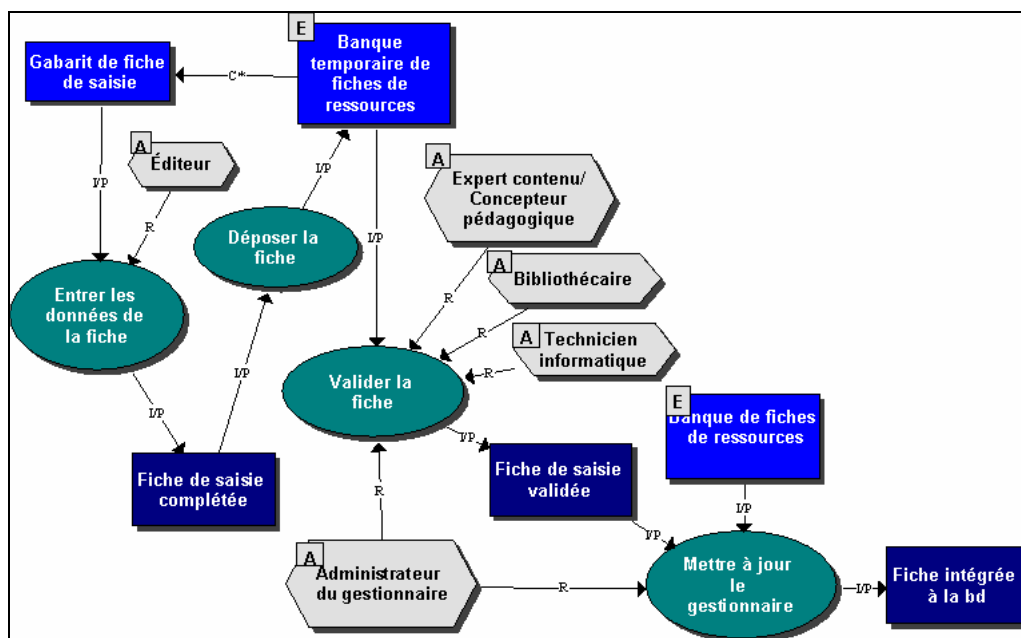


Figure 8 Exemple de détail de scénario générique (un modèle de l'élaboration d'une banque de ressource)

Ses trois notions adressent des aspects très différents de ce que peut être un scénario :

4.3.5 Analyse de la notion de scénario (Daniel Schneider)

Pour Daniel Schneider, un scénario est un guide pour l'enseignant et l'apprenant, ce guide présente une liste de tâches élémentaires à mener avec une description du dispositif permettant d'opérationnaliser ces tâches. Si l'on reprend la taxonomie proposée par Anne Lejeune et Jean-Philippe Pernin : (Pernin & Lejeune, 2004b) il s'agit d'un scénario ayant les caractéristiques suivantes (cf. Figure 9) :

Formalisation : <i>-informel, formel, automatisable</i>
Abstraction : <i>-abstrait, concret</i>
Finalité : <i>prédictif, descriptif</i>
Granularité : <i>activité, séquence, parcours</i>
Personnalisation : <i>générique, adaptatif</i>
Contrainte : <i>contraint, ouvert, adaptable</i>

Figure 9 Caractérisation d'un scénario Daniel Schneider

Le degré de personnalisation d'un tel scénario ne peut être jugé, puisque celle-ci n'est pas décrite, elle fait l'objet d'ajustement directement en cours d'exécution. Le caractère non automatisable limite son utilité dans un contexte informatisé, son opérationnalisation devant être réalisé à la main. Les références à des services concrets existant sur l'application PostNuke limitent son interopérabilité.

4.3.6 Analyse de la notion de scénario (Rob Koper)

Pour Rob Koper, la notion de scénario a évolué entre les intentions d'origines et leurs réalisations effectives sous la forme du standard IMS - LD. Si les premières intentions de Rob Koper étaient de définir un langage formel pour le design pédagogique privilégiant la notion d'activité, le standard IMS-LD s'est clairement focalisé sur le désir de rendre opérationnalisable le standard au travers du séquençage simple ou individuel de ressources pédagogiques. Le standard est centré activité dans la mesure où il conçoit l'apprentissage comme résultant d'une planification d'un acte d'enseignement (Nodenot, 2006b), mais il néglige de traiter correctement la contextualisation nécessaire à la réalisation d'activité située (Allert, 2004). Ainsi les dispositifs impliqués par les activités ne disposent d'aucun mécanisme pour s'opérationnaliser et même s'exprimer. Une simple référence dans "IMS learning design and information model" précise par exemple que :

"IMS Enterprise can be used for mapping learners and support staff to roles when instantiating a learning design". Ce qui semble un peu court pour mener effectivement l'opérationnalisation d'une cohorte particulière sur une plateforme donnée. Un scénario IMS-LD peut donc être décrit ainsi (cf. Figure 10) :

Formalisation : <i>informel, formel, automatisable</i>
Abstraction : <i>abstrait, concret</i>
Finalité : <i>prédictif, descriptif</i>
Granularité : <i>activité, séquence, parcours</i>
Personnalisation : <i>générique, adaptatif</i>
Contrainte : <i>contraint, ouvert, adaptable</i>

Figure 10 Caractérisation d'un scénario Rob Koper

Un scénario IMS-LD est automatisable mais nous verrons que si les données peuvent être effectivement manipulable par un système d'information, elles sont insuffisantes pour mener concrètement l'opérationnalisation de scénarios collaboratifs.

4.3.7 Analyse de la notion de scénario (Gilbert Paquette)

Gilbert Paquette considère deux niveaux de scénarios, les scénarios génériques obtenus par l'application de la méthode MISA, et les scénarios spécifiques obtenus en combinant les scénarios génériques et en les particularisant au domaine d'étude. "Par la suite, la description de tels processus permet d'identifier les ressources (par exemple, les logiciels, les moyens de communication, les guides d'étude, documents pédagogiques, etc.) dont ils ont besoin pour remplir les rôles correspondant" (Paquette et al., 2002) La chaîne de scénarios considérée par Gilbert Paquette peut être décrite ainsi (cf. Figure 11) :

Formalisation : <i>informel, formel, automatisable</i>
Abstraction : <i>abstrait, concret</i>
Finalité : <i>prédictif, descriptif</i>
Granularité : <i>activité, séquence, parcours</i>
Personnalisation : <i>générique, adaptatif</i>
Contrainte : <i>contraint, ouvert, adaptable</i>

Figure 11 Caractérisation d'un scénario Gilbert Paquette

Si les scénarios produits sont effectivement automatisables, leur opérationnalisation passe par une expertise humaine, il est nécessaire d'identifier manuellement les ressources évoquées. Le

niveau d'abstraction peut être qualifié d'abstrait et de concret selon que l'on se situe face à un scénario générique ou spécifique. L'originalité de la démarche par rapport aux deux précédentes est de fournir un cadre très précis, l'ingénierie cognitive, pour guider l'élaboration des scénarios.

4.3.8 Conclusion de cette approche

Par rapport à ces trois visions différentes de la notion de scénario comment pouvons nous situer notre travail?

L'objectif de notre démarche en ingénierie des dispositifs est de permettre l'opérationnalisation concrète des dispositifs impliqués par un scénario automatisable. Nous verrons que notre proposition, présentée dans le chapitre 8 et le chapitre 10, permet d'aborder tout type de plateforme (LMS stricte, portail modulaire, portail léger). Non seulement notre proposition complète, les solutions présentées ci-dessus, mais elle propose également un cadre d'analyse "ouvert" des intentions didactiques d'une équipe d'enseignants. A partir de ce cadre d'analyse, un outil de modélisation spécifique est produit, il permet d'élaborer des dispositifs pédagogiques d'abord génériques, puis spécifiques à une plateforme et enfin concrets dans le contexte de la plateforme. (Caron et al., 2006b)

Attachons-nous maintenant à préciser le concept de contextualisation et les rapports qu'il entretient avec le standard IMS-LD.

4.4 Opérationnalisation et Contextualisation.

L'opérationnalisation d'un fichier IMS-LD sur une plateforme de formation pose des problèmes importants de contextualisation. S'il est en effet possible de décrire dans le standard l'appel à des ressources présentes dans le package ou disponibles sur Internet, il est impossible de faire référence à un service particulier proposé par une plateforme spécifique. Ce contexte peut par exemple aussi être une propriété particulière que les ressources entretiennent entre elles. Ainsi dans le cas de la mise en œuvre d'un scénario de type Jigsaw⁴ (Aronson & Patnoe, 1997) il faut construire au niveau des personnes une interdépendance cognitive. Cette propriété, pourtant fondamentale pour l'opérationnalisation concrète du scénario, ne peut pas être exprimée en IMS-LD.

On ne dispose pas actuellement d'exemple d'opérationnalisation de scénario pédagogique IMS-LD faisant appel à des services concrets dans un cadre indépendant de son modeleur. Les exemples actuels Lams, CopperCore, Edubox, net Université, Explora, n'utilisent pas de services contextualisés, ou bien construisent directement l'environnement d'exécution à partir des services existants sur la plateforme. Le scénario IMS-LD exporté n'étant alors, qu'une version appauvrie, décontextualisée, du scénario spécifique à la plateforme.

Le problème de contextualisation posé par le langage IMS-LD est un problème récurrent des standards de l'interopérabilité. Pour préserver la réutilisation, ces standards passent sous silence les références au contexte précis permettant la réalisation de l'activité. Cette critique a déjà été formulée pour le LOM, et pour le standard SCORM, elle est également vraie pour IMS-LD. Malgré la neutralité du langage, l'objectif d'interopérabilité revendiqué le place clairement du côté des modèles d'apprentissage de type enseignement/évaluation (instructional design) (Nodenot, 2006b). Heidrun Allert, explique d'ailleurs que le fait même

⁴ Le Jigsaw est un pattern pédagogique de type CSCL reposant sur la constitution de groupes rassemblant des experts de domaines différents.

de définir une structure de métadonnées et de spécifier un schéma conceptuel reflète inévitablement un modèle spécifique d'apprentissage. Le fait de ne pas situer les activités par rapport à leur contexte, empêche le scénario d'être le reflet réel d'une activité constructiviste ou socioconstructiviste (Allert, 2004). L'auteur propose un tableau permettant d'établir une typologie des approches instructionnelles, situées et humanistes au regard de la contextualisation. Nous le reproduisons ci-dessous (cf. Figure 12).

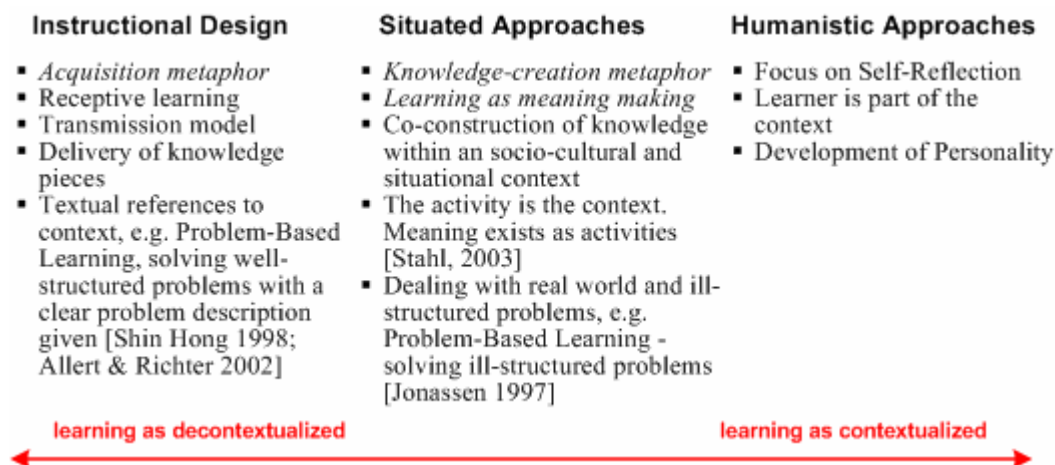


Figure 12 Différentes approche de l'enseignement relativement à la contextualisation, tiré de (Allert, 2004)

Ce tableau est utile à plus d'un titre, il permet de relier théorie pédagogique, type de plateforme et design pédagogique. Outre l'accent mis sur le rapport entre socioconstructivisme et contextualisation, le tableau illustre la faiblesse des standards pour rendre compte et opérationnaliser des situations collaboratives. Il met en évidence le paradoxe qui peut exister entre un enseignement co-construit, à base de problèmes ouverts, et un langage ayant l'ambition de décrire intégralement une situation pédagogique.

"The specification must be able to fully describe the teaching-learning process in a unit of learning, including references to the digital and non-digital learning objects and services needed during the process." (IMS LD 2003)

A cet égard, il est intéressant d'analyser le projet FORMID (Gueraud et al., 2004) et de voir en quoi ce projet illustre particulièrement bien la perspective élaborée par Heidrun Allert.

4.4.1 L'exemple du projet FORMID approche par scénario dans un cadre constructiviste

Ce projet, mis en œuvre par le laboratoire Clips IMAG de Grenoble, définit des scénarios pédagogiques permettant l'opérationnalisation d'activités situées sur la plateforme FORMID. Les scénarios réalisés à l'aide de l'éditeur OASIS (Cortés & Guéraud, 1998) (propre au projet FORMID) concernent l'accomplissement par les apprenants d'une tâche d'interaction avec un Objet Pédagogique Interactif (OPI). Un OPI est un objet permettant de générer des interactions riches avec un apprenant : par exemple un OPI peut être un simulateur, un micro monde, un mini environnement de calcul formel, etc....

La définition des scénarios, tels qu'ils sont considérés par le projet FORMID, est différente des conceptions présentées dans la section 4.3 précédente. En effet plutôt que de spécifier un certain nombre d'étapes à atteindre par un apprenant ou un groupe d'apprenant, il spécifie un objectif à atteindre et un certain nombre d'étapes à observer. Les mécanismes d'observation sont décrits précisément dans le scénario, ils permettent une opérationnalisation réelle des observations. L'architecture proposée, architecture client serveur, permet l'encapsulation des l'OPIs par des services et se rapproche de la solution que nous avons élaborée pour

l'opérationnalisation de dispositifs sur des plateformes de formation (cf. partie 2). Viviane Guéraud a elle-même décrit la vision des scénarios pédagogiques proposés par le projet FORMID à l'aide de la taxinomie utilisée dans le chapitre précédent (Lejeune & Pernin, 2004). Un scénario tel que considéré par le projet Formid (cf. Figure 13) suit le tableau suivant : (Guéraud, 2006).

Formalisation : informel , formel, automatisable
Abstraction : abstrait , concret
Finalité : prédictif , descriptif
Granularité : activité, séquence , parcours
Personnalisation : générique, adaptatif
Contrainte : contraint, ouvert, adaptable

Figure 13 Caractérisation d'un scénario Viviane Gueraud

L'auteur explique que le critère de contrainte n'est pas pertinent pour décrire un tel scénario, en effet même si a priori les scénarios produits laissent totalement libre le cheminement suivi par l'apprenant, le dispositif offre la possibilité à l'enseignant de spécifier des objectifs intermédiaires à atteindre par l'apprenant. Si l'on se réfère au tableau d'Heidrun Allert, le dispositif permet la création de scénario pouvant décrire des activités situées mais aussi des activités de type enseignement/évaluation, ("*Problem Based Learning solving ill structured problem vs. Problem Based Learning solving well structured problem*") (Allert, 2004).

4.4.2 Conclusion

Notre étude semble conclure sur la difficulté d'établir des scénarios prédictifs contraints pour prendre en compte des activités constructiviste et socioconstructiviste. Cette difficulté de modélisation est-elle spécifique aux activités envisagées par l'enseignant ou constitue-t-elle une difficulté inhérente au métier d'enseignant? En d'autres termes, établir un scénario prédictif contraint, constitue-t-il un acte *facile* pour un enseignant?

Nous abordons ci-dessous cette question, notre proposition basée sur l'improvisation pédagogique et le bricolage sera présentée dans le chapitre 8.

4.5 Ecrire un scénario pédagogique, une activité enseignante?

4.5.1 Introduction

Essentiellement deux arguments motivent l'intérêt de concevoir des scénarios pédagogiques pour favoriser un enseignement sur des plateformes de formation.

Le premier met en avant l'aspect opérationnel, et omni-descriptif du processus de création permettant de produire un scénario pédagogique. Cet argument repose sur l'idée qu'il est possible d'établir un scénario utilisable pour tout type de situation pédagogique. Nous avons vu, dans le chapitre précédent, qu'il était difficile d'opérationnaliser et même de décrire certaines situations, cette difficulté étant due à la non prise en compte par le scénario des éléments du contexte.

Le second argument, concerne la pratique enseignante et suppose que la phase de planification d'un enseignement est d'autant plus féconde qu'elle est détaillée. Une telle planification (le scénario pédagogique) est non seulement utile mais réutilisable et partageable. C'est ce dernier argument que nous explorons dans cette section.

4.5.2 Réutiliser, partager?

La plupart des recherches s'accordent sur l'intérêt de planifier un enseignement mais sont partagées sur la validité de la corrélation entre meilleure planification et meilleur enseignement (Dessus, 2002). La planification d'un enseignement n'est pas forcément réutilisable et partageable, en effet contrairement à des processus d'ingénierie classique la création en enseignement est un cycle illimité de re-création. (Caron, Derycke, & Le Pallec, 2005a). La réinvention d'activités et du matériel correspondant est en effet un élément important dans l'enrichissement et l'appropriation personnelle du rôle professionnel (Huberman, 1980) Huberman explique que non seulement établir un scénario strict de son enseignement dans un but de réutilisation n'est pas forcément utile, mais que ce scénario remet en cause le rôle de l'enseignant au sein de sa classe. Cet auteur explique ainsi qu'une fois en classe, la préoccupation majeure de l'enseignant doit être de réagir aux élèves et non de suivre un scénario (Huberman, 1980). *"L'activité effective de l'enseignant et des élèves n'est jamais l'interprétation d'une pièce écrite par le maître"* nous confirme Philippe Perrenoud (Perrenoud, 1983). Nous laissons aux lecteurs le soin de juger l'ironie de la phrase écrite vingt ans avant la métaphore théâtrale choisie par le standard IMS-LD. Philippe Perrenoud décrit les activités de l'enseignant comme se situant entre improvisation réglée et bricolage. Ces activités ne sont pas, pour lui, de simples mises en pratique de recettes, de modèles ou de schémas d'action conscients, mais elles sont plutôt gouvernées par un système de schèmes de pensée et d'action qui sous-tend les multiples microdécisions prises en classes, sur le vif. Ces schèmes permettant d'engendrer une infinité de pratiques, ces pratiques peuvent ainsi s'adapter à des situations toujours renouvelées, sans jamais se constituer en principes explicites. (Bourdieu, 1972). Cette improvisation s'appuie également sur des routines qui permettent de prédire le comportement des élèves dans la situation d'indétermination de la classe et de résoudre des problèmes d'interrelation durant la phase interactive. (Lacourse, 2004).

Dans le contexte des EIAHs, utiliser un langage de modélisation pédagogique comme IMS-LD, semble incohérent avec une telle pratique. Le peu de place laissé à l'improvisation enseignante dans les scénarios modélisés avec IMS-LD, rend en effet ce langage difficile à utiliser par des enseignants. Ceux-ci agissent moins selon un "learning design" que selon des "grammaires de l'expérience" (Dessus & Schneider, 2006) Les constituants d'une telle grammaire ont la particularité de pouvoir être changés à chaque étape d'analyse et d'actions (Tochon, 1996). Une thèse menée au sein de notre laboratoire de recherche explore d'ailleurs cette voie (Moura & Derycke, 2007).

4.5.3 Les limites de la planification

Nous pouvons ici conclure que la scénarisation stricte et contrainte d'activité pédagogique ne correspond pas à une pratique enseignante. Cependant posons-nous la question de l'évolution d'une telle pratique. En effet avec Philippe Perrenoud nous savons que *"la transformation des contraintes et conditions objectives de l'enseignement font plus pour changer les pratiques que la diffusion d'idées ou de recettes pédagogiques nouvelles"*. (Perrenoud, 1983) Si l'usage des technologies de l'information est susceptible de changer les pratiques enseignantes, il est possible aussi de considérer que pour un enseignant, le changement et l'évolution sont déjà des pratiques habituelles que les TIC ne sont qu'un révélateur de sa capacité à évoluer (Charlier, 2000). Pour autant, peut-on affirmer que la scénarisation pédagogique constituera à l'avenir un élément important des pratiques enseignantes? Dans son habilitation à diriger les recherches, Thierry Nodenot, propose trois arguments permettant de conclure que la scénarisation pédagogique peut biaiser l'activité de conception. (Nodenot, 2006a) L'approche structurée par scénario pédagogique peut tendre à contraindre la créativité des enseignants, Thierry Nodenot fait ainsi référence au problème très simple que lui pose en enseignement

l'utilisation de PowerPoint par ses étudiants (à titre d'illustration de ce problème citons l'essai de Rafi Haladjian : "Devenez beau, riche et intelligent, avec PowerPoint, Excel et Word" (Haladjian, 2003).) Thierry Nodenot conclut son argumentaire par "*Approcher une situation d'apprentissage par un scénario de type structuration en actes, scènes et activités a toutes les chances d'amener à une organisation hiérarchique de cette situation.*"

L'auteur précise son propos dans son deuxième argument : L'approche par scénario pédagogique risque de privilégier une approche analytique qui va à l'encontre de la découverte d'idées, de situations porteuses de sens pour les apprenants. Il conclut son argumentaire en stigmatisant le fait que le recours au scénario ne conduise pas les enseignants à explorer le contexte d'utilisation des instruments prescrits par l'activité.

4.6 Conclusion

Pour conclure ce chapitre nous pouvons affirmer que la scénarisation pédagogique, ne constitue pas une pratique enseignante tant au niveau de l'opérationnalisation précise du scénario, dans le contexte d'un enseignement, que de la réutilisation ou du partage de ce scénario. L'activité enseignante faite d'improvisations réglées, s'appuyant sur des schèmes entrecoupés de routine, ne correspond pas à celle véhiculée par le learning design. Cette dernière considère l'enseignant comme partie intégrante d'un processus d'ingénierie, processus menant à la réalisation d'un scénario pédagogique, opérationnalisable strictement. Même si avec l'utilisation de plus en plus fréquente des EIAHs, il est possible qu'on observe une mutation des pratiques enseignantes. Dans ce nouveau contexte, la scénarisation d'activité risque de contraindre la créativité des enseignants et de fausser leurs perspectives lors de la préparation d'une séquence pédagogique.

Nous proposons, dans le chapitre 5 suivant, de rendre compte de différentes pratiques menées sur des plateformes de formation et de montrer la place que peut tenir standard, méthodes et outils dans la nécessaire rationalisation des pratiques enseignantes en EIAH. A cette occasion nous présenterons notre proposition basée sur la modélisation de dispositifs pédagogiques.

chapitre 5 Notre proposition : de l'intérêt d'une démarche dispositive

5.1 Introduction, une pratique particulière, l'expérience acquise au sein du laboratoire.

Nous avons vu dans les chapitres précédents que pour un enseignant désirant mener une rationalisation de son travail sur une plateforme de formation, différentes approches existaient. Cependant ces différentes approches (documentaire, ingénierie cognitive, activité) ne permettent pas dans certains cas de lui apporter une véritable aide. Pour l'enseignant désirant mettre en œuvre un enseignement constructiviste ou socio-constructiviste impliquant des activités situées, et donc contextualisées, les standards actuels sont également peu utilisables. Nous proposons dans ce chapitre de présenter les travaux actuels se rapportant à l'objet de notre recherche, "l'approche par dispositif". Dans un premier temps nous présentons un exemple permettant d'illustrer la problématique posée par la réutilisation, la migration et le partage de dispositif.

Le positionnement du laboratoire Trigone sur des problématiques liées aux dispositifs adaptés au "Computer Supported Collaborative Learning" et basés sur la théorie de l'activité (Bourguin & Derycke, 2005) ont participé, au sein du laboratoire, à la mise en place d'une approche, l'approche par dispositif. Cette approche a été présentée dans le cadre des EIAHs lors de la participation de notre laboratoire à l'AS CNRS "Fondements théoriques et méthodologiques de la conception des EIAH" (Tchounikine, 2004). Les pratiques éducatives et les dispositifs mis en place par les enseignants chercheurs de notre laboratoire sont donc fortement influencés par cette approche (D'Halluin & Delache, 2005a), (D'Halluin & Delache, 2005a), (Leclercq, 2005). A partir de leurs expériences dans le cadre du Centre Université-Economie d'Education Permanente (CUEEP) auquel notre Laboratoire Trigone est rattaché, nous avons très vite postulé que la difficulté d'enseigner en e-Learning ne trouvait pas sa solution uniquement par la maîtrise des contenus et par le suivi fin des apprenants. Ce paradoxe (dont nous avons fourni un argumentaire dans les chapitres précédents) a reposé dans un premier temps sur la constatation pragmatique qu'enseigner en présentiel, dans la salle de classe, posait des problèmes totalement différents, en terme d'infrastructure, qu'enseigner par des moyens distants.

Si, dans la salle de classe, la publication du savoir est souvent le seul fait du professeur, pour des raisons pratiques : il maîtrise le code photocopieuse et le choix des manuels. En enseignement à distance, médiatisé par une plateforme de formation, ce n'est plus vrai. Chaque apprenant à accès à Internet et peut délivrer bien plus facilement un savoir aux autres. Cependant, pour un tel enseignement, les canaux de communication sont peu nombreux : ils ne permettent pas de mettre en scène l'ensemble des interactions qui ont lieu en classe lors d'une séance d'enseignement. C'est ce type d'interaction qu'il nous faut particulièrement privilégier en enseignement à distance. Or de telles interactions nécessitent la mise en œuvre de dispositifs complexes, création de groupe, de droit, contextualisation de document par des outils, etc... Cette constatation est à l'origine de notre problématique en ingénierie des dispositifs.

Dans ce chapitre nous allons, dans un premier temps, éclairer notre problématique initiale par un exemple concret, puis nous explorerons la notion de dispositif et les relations que cette notion entretient avec la notion de scénario. Nous situerons ensuite notre problématique comme complémentaire d'une démarche menée sur les standards, en reprenant les différents travaux se rapportant à notre recherche.

5.2 Illustration de la problématique sur un exemple, un cours de remise à niveau Java

5.2.1 Présentation de l'exemple

Dans le cadre d'une version à distance du Master professionnel, spécialité Informatique, parcours e-Services, la plateforme Accel est utilisée, (Bounie et al., 2005).

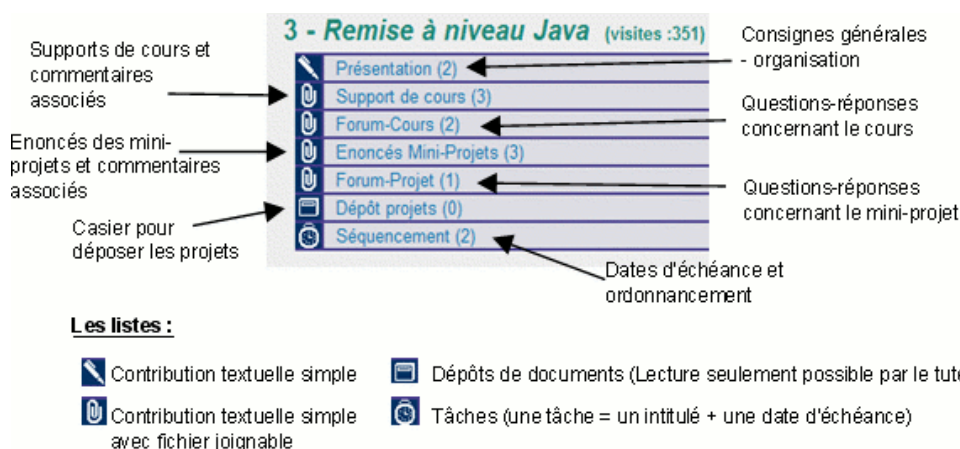


Figure 14 Dispositif comprenant 7 ateliers accueillant le cours sur Accel

Un des cours mis à la disposition des étudiants de ce Master est un cours de "remise à niveau programmation en Java". Ce cours a pour but d'harmoniser le niveau en programmation orientée objet des différents étudiants. Le parcours de ce cours peut se résumer comme suit : **une première partie** où les élèves étudient des documents de cours et peuvent interagir entre eux, le tuteur peut poser des questions ou y répondre et émettre des remarques ; **une seconde partie** où chaque élève doit effectuer un certain nombre d'exercices. Ceux-ci seront corrigés par un autre élève dans un premier temps puis par le tuteur dans un second temps ; **une troisième et dernière partie** concerne l'accomplissement d'un mini projet. Les élèves disposeront encore une fois d'un espace pour discuter sur leurs problèmes. Ils déposeront leur devoir dans un espace privé.

Le dispositif requis (cf. Figure 14) comporte 7 ateliers, ceux-ci sont nécessaires pour mener à bien ce cours sur la plateforme Accel. Cette complexité engendre des difficultés dans trois situations.

- Lorsqu'il faut reconstruire le dispositif de formation
- Lorsqu'il faut faire migrer ce dispositif sur une autre plateforme de formation
- Lorsqu'il faut partager le dispositif avec d'autres enseignants

Ces trois situations sont décrites et illustrées par notre exemple dans les sous-sections suivantes; de chaque situation, nous tirons un enseignement qui nourrit notre recherche. Ces enseignements seront développés dans la section 5.3.

5.2.2 Construction et re-construction du dispositif sur une plateforme de formation

A chaque nouvelle promotion, il faut reconstruire le dispositif de formation :

Il faut inscrire les étudiants et recréer les sept ateliers qu'il faut pré-structurer : indications et présentation dans l'atelier **présentation**, placer les documents de cours dans l'atelier **support de cours**, créer les différentes tâches dans l'atelier **séquencement**... L'espace entre deux formations (un an) ne permet pas une capitalisation de l'expérience acquise lors de la construction et de fait ralentit généralement le processus de création des ateliers, ce qui en fait une tâche longue et fastidieuse. Une fonction de sauvegarde pourrait être utile : en sauvegardant les ateliers au moment de leur définition, nous aurions une sauvegarde uniquement de la structure non parasitée par des éléments propres à une promotion (ex : questions ou réponses d'étudiants, matériel co-construit). Toute modification effectuée au dispositif pendant l'exécution du cours (par exemple un atelier ajouté à la demande des étudiants) ne pourrait malheureusement pas être pris en compte sous peine de sauvegarde de "parasites" : inconvénient majeur d'une fonctionnalité de sauvegarde.

Notre première hypothèse est qu'il est possible de définir l'expression d'un dispositif. Cette expression peut constituer pour lui un schéma de persistance, elle peut éventuellement être enrichie, elle peut servir à mener une opérationnalisation du dispositif. C'est cette expression que nous désignons par **le modèle du dispositif**.

Le modèle du dispositif est une version, dépendante de la plateforme, du scénario pédagogique conscient ou inconscient que l'enseignant veut réaliser avec ses étudiants. Or il n'existe aucune norme permettant de construire ou sauvegarder des dispositifs techniques sur des plateformes de formations. La plateforme Accel n'offre pas de possibilité de sauvegarde. La plateforme Ganesha propose un outil spécifique (Moolinex) permettant la création de cohortes. La plateforme Moodle propose un outil de sauvegarde propriétaire qui permet également les sauvegardes partielles. Par exemple, des activités individuelles peuvent être sélectionnées pour la sauvegarde, avec, en option, la possibilité d'inclure les données des étudiants pour chaque activité. La plateforme Claroline ne propose pas de sauvegarde, mais en sauvant le répertoire consacré à la formation, ainsi que la base de données MYSQL de cette formation, il est possible de réinstaller manuellement l'ensemble du dispositif de formation. Cette sauvegarde peut être complétée, en sauvant aussi la base de données principale, par l'entière des comptes utilisateurs créés.

Pour être utilisable dans les situations pratiques que nous avons pu observer, un modèle de dispositif doit pouvoir accompagner la migration sur d'autres plateformes de formation. C'est ce point que nous abordons dans la sous section 5.2.3 suivante.

5.2.3 La réutilisation du dispositif sur d'autres plateformes.

Quelques uns des cours du master e-services en EAD sont réutilisés dans d'autres structures de formations à distance, une partie des contenus est ainsi repris pour une formation se déroulant sur une la plateforme Claroline. Ce changement de plateforme est dû à des contraintes de partenariat qui ont prévalu lors de l'élaboration du cadre de la formation. Le développement au sein des universités des enseignements distanciels nous porte à croire que la réutilisation de formations sur d'autres plateformes est une situation qui est en passe de devenir fréquente.

Trois problèmes apparaissent lorsque l'on souhaite réutiliser un modèle de dispositif technique sur une autre plateforme :

- Il faut retrouver, à partir d'un dispositif existant, le scénario pédagogique sous-jacent .
- Il faut maîtriser les concepts ou fonctionnalité de cette nouvelle plateforme et pouvoir les exprimer pour adapter conceptuellement le dispositif
- Il faut pouvoir modifier automatiquement le modèle de dispositif technique.

Scénario sous-jacent : à partir d'un dispositif il est difficile de retrouver les choix pédagogiques qui ont permis de produire le produit, la transformation d'un scénario pédagogique même très grossier en un dispositif implique des pertes sémantiques. Retrouver le scénario original à partir d'un dispositif existant sur la plateforme Accel est impossible. Ceci nécessite donc soit une bonne mémoire de la part du concepteur, soit une description textuelle précise.

Maîtriser les nouveaux concepts : Mettre en place un dispositif pour accueillir le cours de remise à niveau Java sur une autre plateforme que Accel (par exemple Ganesha) demande une maîtrise des concepts de la plateforme Ganesha. Or concepts et outils sont différents sur ces deux plateformes : Ganesha n'offre pas par exemple la possibilité d'associer un forum pour une activité précise ; Les contenus, par contre, peuvent être définis indépendamment des formations et leurs compatibilités SCORM permettent une recombinaison facile et une traçabilité plus subtile des étudiants. Les multiples transferts qui ont dû être réalisés pour le master e-Services EAD, ont montré que la phase d'apprentissage d'une plateforme par les tuteurs ne peut être réduite. Ce dernier point contribue à rendre délicat l'aspect logistique inhérent au changement de plateforme.

Pouvoir automatiser la migration : la retranscription manuelle est non seulement plus longue qu'une retranscription automatique mais surtout elle est susceptible d'engendrer erreurs et oublis.

Notre deuxième hypothèse est que pour être réutilisé sur une autre plateforme, il faut pouvoir lier le modèle de dispositif à un scénario pédagogique, il faut disposer d'un formalisme pour exprimer les concepts de la nouvelle plateforme de formation, un modèle de dispositif doit pouvoir être manipulé automatiquement .
--

5.2.4 Le partage de connaissance

La conception d'un dispositif de formation bénéficie rarement de dispositifs existants développés par d'autres enseignants. . Nous avons vu dans les sections précédentes que le partage est un des grands enjeux des EIAH, ce partage nécessite l'utilisation de formalisme indépendant des plateformes de formation, ce qui pose des difficultés pour exprimer des activités pédagogiques fortement contextualisées. Pour qu'un dispositif de formation puisse être partagé il faut pouvoir l'exprimer dans un formalisme compris de tous, indépendant des plateformes de formation, mais pourtant constructible, automatiquement, ou du moins facilement, sur ces plateformes. Des standards comme SCORM remplissent en partie cet enjeu, cette réussite a un prix : restreindre le domaine de description, à la description de simples contenus décontextualisés, définir un couplage très faible avec la plateforme de formation.

Notre dernière hypothèse est qu'il est éventuellement possible de définir des règles permettant de passer d'un dispositif exprimé dans un langage indépendant des plateformes à un dispositif dans un langage dépendant de la plateforme de formation. Nous pensons qu'il doit être possible d'établir entre un modèle de dispositif et la plateforme de formation, un dialogue riche en partie automatisable.

5.2.5 Deux études pour valider nos hypothèses

Au cours de notre thèse, trois études de cas ont été menées (chapitre 11). Les deux études résumées ci-dessous concernent plus spécifiquement les points abordés dans les sous-sections précédentes (5.2.1, 5.2.2, 5.2.3, 5.2.4); Ces deux études montrent que le fait de travailler avec un langage indépendant peut constituer une étape permettant l'expression d'un dispositif. Cette étape peut être complétée pour parvenir à l'expression technologique d'un modèle de dispositif. Un tel modèle fonctionnel peut alors facilement être opérationnalisé sur une plateforme de formation.

La première (chapitre 11.2) a consisté à utiliser le formalisme IMS-LD pour une expression indépendante de la plateforme du scénario pédagogique et du dispositif. Dans notre contexte de déploiement et de construction de dispositif, IMS-LD peut permettre aux enseignants de définir un modèle de leur dispositif pédagogique une fois pour toute, ceci de manière précise et indépendamment de toutes technologies/plateformes et quelque soit la pédagogie sous-jacente du cours. IMS-LD par les concepts qu'il manipule (activité, séquence, service, message) nous a semblé dès sa création se situer dans une logique d'ingénierie des dispositifs. C'est cette logique que nous avons voulu exploiter par notre approche. Nous verrons, dans le chapitre 8 et le chapitre 10, que l'utilisation d'IMS LD que nous proposons ne correspond pas à ses utilisations actuelles. Le recours à ce standard ne constitue d'ailleurs qu'une étape du processus que nous avons instruit. (Caron, 2006)

La seconde étude (chapitre 11.3) a eu lieu dans le cadre du projet PCDAI. Ce travail a permis de définir conceptuellement deux processus.

- Le premier a pour objectif de créer un atelier spécifique à un type de préoccupations pédagogiques, les acteurs de ce processus étant : un enseignant, un ingénieur pédagogique et un informaticien. Ce processus, voit d'abord la description des types des préoccupations pédagogiques par l'enseignant et l'ingénieur pédagogique puis la définition, par tous les acteurs, des règles de transformation permettant de définir ces préoccupations dans le contexte d'une plateforme.
- Le second a pour objet la mise en œuvre d'intentions pédagogiques par un enseignant. Le principe est simple : définition des intentions dans un modèle de dispositif indépendant puis utilisation des règles de transcriptions. (Caron et al., 2006b)

Dans ces deux études, le modèle de dispositif indépendant obtenu a été transformé en un modèle de dispositif opérationnalisable dans le contexte d'une plateforme spécifique de formation. La notion de dispositif, s'est révélé être au cœur des pratiques que nous avons observées. Cependant ce terme générique est assez confus, il est question dans notre exemple de dispositif technique sur une plateforme de formation, de dispositif pédagogique, d'exprimer un modèle de dispositif dans un langage abstrait et d'observer les liens entre un tel modèle et un scénario pédagogique. La section suivante s'attache à définir cette notion de dispositif.

5.3 La notion de dispositif

5.3.1 Introduction et définition

Le dictionnaire en ligne "Trésor de la Langue Française" définit le terme dispositif par : " Un dispositif est *"un ensemble d'éléments agencés en vue d'un but précis"*. (TLF). Dans le domaine particulier des EIAHs le "Lexique de la Formation Ouverte et/ou à distance" (Lexique_Formation) définit un dispositif de formation comme étant *"un ensemble de moyens matériels et humains, correspondant à une forme de socialisation particulière destinée à faciliter un processus d'apprentissage"*. Ces deux définitions soulignent la dimension duale d'un dispositif : **agencement de ... et au service de ...**

Concrètement et de façon première, nous énonçons une définition relativement simple d'un dispositif sur une plateforme de formation.

Il s'agit d'un agencement particulier dans le contexte d'une plateforme de formation de composants, de fonctionnalités et de services, cet agencement est au service d'une intention pédagogique et d'une communauté d'apprentissage

5.3.2 Dispositif et scénario

La relation qui lie intention et dispositif dans le cadre des EIAHs a été en partie explorée par Daniel Peraya. *"Un dispositif consiste en une organisation de moyens au service d'une stratégie, d'une action finalisée, planifiée visant à l'obtention d'un résultat."* (Peraya, 1998). L'auteur explique qu'un dispositif comporte par essence les intentions qui ont motivé sa création, mais qu'il possède ses propres intentions, son fonctionnement et ses modes d'interactions propres. Ainsi sa structure, les moyens matériels qu'il met à la disposition des utilisateurs, modélisent, à partir de leurs caractéristiques propres, les comportements de ceux-ci. En ce sens un dispositif existe et continue de remplir son contrat indépendamment des intentions pédagogiques qu'ils l'ont fait naître. Cette nouvelle dimension est fondamentale car elle justifie l'idée qu'il puisse exister des modèles de dispositif et que l'on puisse séparer les auteurs de l'acte de modélisation qui a conduit à la définition d'un modèle de dispositif, et les personnes qui l'utiliserons. En ce sens un dispositif pédagogique construit sur une plateforme de formation est une entité aussi complexe qu'un scénario pédagogique. Nous allons explorer dans ce qui suit cette complexité.

Tout dispositif technique, tout artefact, mobilise durant sa conception une connaissance, une représentation, et au sens le plus large un modèle de l'utilisateur et de son activité. (Béguin & Cerf, 2004). Pour passer de ce modèle de l'utilisateur et de son activité à un modèle de dispositif, il est possible de passer par un modèle intermédiaire reliant activité et dispositif, un tel modèle nous semble pouvoir être en EIAH un scénario pédagogique. Introduire au côté du dispositif, un scénario pédagogique c'est rappeler qu'un dispositif est une construction pratique au service d'une fin, ici représentée par un scénario. Il est alors possible de considérer un dispositif comme la mise en œuvre pratique *"d'une logique de type dramatique qui combine la mise en scène des protagonistes, des rôles et des circonstances avec les règles du déroulement de l'action."* (Linard, 1998).

Les rapports entre dispositifs et scénario peuvent être définis ainsi :

- Le dispositif met à la disposition du scénario des moyens logistiques et des ressources (techniques, humaines, etc.) pour être mis en œuvre.
- Le scénario implique de se poser la question suivante : "*quels moyens et quelles ressources le dispositif dans lequel je travaille met-il à ma disposition pour mettre en œuvre mon scénario ?*" (Brassard & Daele, 2003)

Nous verrons, dans le chapitre 8, le chapitre 9 et le chapitre 10, qu'il est possible de définir une chaîne de modélisation incluant des scénarios pédagogiques et menant des intentions pédagogiques à la construction d'un dispositif. La finalité d'un tel scénario est de prévoir les ressources et moyens à utiliser sur la plateforme, il n'a pas besoin d'être très contraint. Il peut laisser à l'enseignant un espace d'improvisation dans le cadre d'un dispositif précis, il doit cependant pour pouvoir définir un dispositif constructible, faire référence à des services, ou fonctionnalités, très spécifiques à la plateforme. Pour reprendre la typologie des scénarios utilisés dans le chapitre 4 précédent, un tel scénario peut être (cf. Figure 15):

Formalisation : informel , formel, automatisable
Abstraction : abstrait , concret
Finalité : prédictif , descriptif
Granularité : activité, séquence, parcours
Personnalisation : générique , adaptatif
Contrainte : contraint , ouvert, adaptable

Figure 15 Caractérisation d'un scénario utilisable pour définir un dispositif

Pour être manipulé informatiquement, nous avons besoin qu'un tel scénario soit formel, automatisable et **opérationnalisable**, dans le sens où nous souhaitons qu'il participe à la **construction** d'un dispositif pédagogique. Un tel scénario doit être concret dans la mesure où il doit faire référence concrètement au dispositif à construire sur la plateforme. Les autres critères découlent de l'étude menée dans les sections précédentes de ce chapitre.

Nous venons de le voir, une approche centrée sur la notion de dispositif, peut être complémentaire des approches décrites dans le chapitre 4. Dans cette perspective un dispositif est au service d'une intention exprimée par un scénario pédagogique. Cependant cette vision de la notion de dispositif établit une dichotomie entre moyen et fin, pratique et conceptuel, elle relègue la notion de dispositif à la description de systèmes techniques. Or il est possible d'explorer plus avant la notion de dispositif, pour dépasser la définition très fonctionnelle que nous avons établie. Un dispositif dans le contexte d'une plateforme peut être en effet vu comme la concrétisation physique d'un dispositif abstrait sans qu'il soit nécessaire de rattacher l'un ou l'autre à l'expression concrète d'un scénario pédagogique.

5.3.3 Ambivalence de la notion de dispositif

Un dispositif pédagogique n'est pas une simple entité permettant de mettre en œuvre un scénario pédagogique. Il est également possible de considérer un dispositif comme la concrétisation d'un dispositif abstrait représentant les intentions pédagogiques d'une équipe d'enseignant. Cette nouvelle considération permet de mener une réflexion autour de la notion de dispositif sans préjuger de l'ordonnancement des activités qui vont s'y dérouler. De façon générale, la notion de dispositif réduit la dichotomie entre technique et symbolique (Berten, 1998) André Bertin souligne ainsi l'ambivalence d'un dispositif entre les ressources mis en œuvre et "*intentionnalité flottante et transversale*", il le décrit également comme "*un paysage*

cognitif, tolérant à l'erreur", indiquant non seulement la nature conceptuelle d'un dispositif, mais également le caractère ouvert que nous stigmatiserons dans les sous-sections suivantes. Le dispositif est une notion mixte intermédiaire entre usage et concept, il peut être considéré de façon objective, organisation de moyens techniques mis à la disposition d'un but, il peut aussi être considéré de façon subjective, expression de "*l'intentionnalité des concepteurs, des usagers*" (Linard, 2002).

Pour décrire cette dimension conceptuelle, Gilles Leclercq, qui mène ses recherches en sciences de l'éducation dans notre laboratoire, qualifie un dispositif "*d'offre de signification*" offerte aux utilisateurs, (enseignants apprenant), pour lui un dispositif est une proposition organisée pour penser, concevoir, agir (Leclercq, 2005). L'étude que nous avons menée dans le cadre du dispositif PCDAI (décrit dans le chapitre 11.3) permet d'illustrer significativement ce que nous entendons par "*offre de signification*".

Le projet PCDAI vise à permettre des formes d'apprentissage plus actives sur internet. L'expérience que nous décrivons s'est déroulée dans le cadre du sous chantier : Co-construction et évolution d'une infrastructure, déploiement et mise en exploitation. Elle concerne l'accompagnement collaboratif à distance de stage et de rédaction de mémoire professionnel destiné à des étudiants suivant une licence professionnelle. Pour réaliser cet accompagnement, les enseignants imaginent un dispositif qu'ils définissent par le terme : EAPC (Explorateur d'Actions Personnelles et Collectives). Dans l'espace potentiel de l'EAPC imaginé par les enseignants, chacun doit pouvoir créer un espace personnel. Cet espace personnel doit être singularisé (automatiquement) par l'identité de la personne (nom, prénom, année). Pour utiliser l'EAPC, cet espace personnel doit être obligatoirement être pré-structuré par 5 balises :

*Construire son tutorat,
Formaliser sa mission,
Mener des investigations de terrain,
Mener des investigations conceptuelles
Capitaliser les références bibliographiques.*

Les auteurs du projet ont choisi d'exprimer leur dispositif en terme de balise, d'acteur, d'appartenance. Ces concepts constituent des éléments du langage permettant l'expression de l'offre de signification. Dès son élaboration, le but du dispositif est énoncé et les activités prévues sont décrites :

Le dispositif ainsi créé permet de constituer une collection d'espace personnel qui forment ensemble un espace collectif. La première grande famille d'activité consiste donc à composer une mémoire personnelle qui soit en même temps une mémoire collective. La deuxième grande famille d'activité consiste à exploiter la mémoire collective. Elle se décompose en trois sous-activités :

- *Inviter quelqu'un à consulter l'EAPC*
- *Rechercher quelque chose dans l'EAPC*
- *Faire son marché*

L'intention et les activités ainsi définies ajoutent les notions de droits et de collectif aux concepts nécessaires à l'expression dispositif. Nous verrons (de façon générale dans le chapitre 8, le chapitre 10, et de façon spécifique à cette étude dans le chapitre 11.3) qu'à partir d'une description en langage naturel, il est possible de parvenir à un dispositif concret dans le contexte d'une plateforme de formation, en définissant de façon formelle le langage et l'environnement permettant d'exprimer le dispositif.

5.3.4 Conclusion

Nous avons montré dans les deux dernières sous sections qu'il est possible d'approcher un dispositif concret par l'expression du scénario pédagogique prévu ou par la définition d'un dispositif conceptuel qu'il s'agit alors d'opérationnaliser dans le contexte d'une plateforme.

Il est possible de résumer ces deux approches par le tableau suivant (cf. Figure 16) :

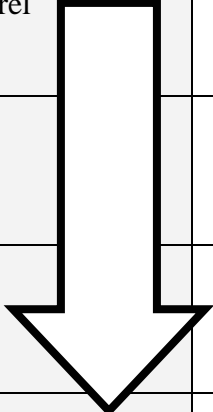
Expressions...	Solutions...	passant par l'expression du dispositif abstrait	passant par l'expression du scénario abstrait
en langage naturel		Offre de signification, objectif, acteurs, activité	Séquençement, scénario abstrait, activités, ressources, acteurs
automatisable		Dispositif abstrait, exprimé dans un langage formel défini par l'équipe pédagogique	Scénario abstrait, automatisable, exprimé dans un langage normalisé
Spécifique		Dispositif spécifique obtenu par transformation	Dispositif spécifique obtenu par analyse du scénario et confrontation avec les concepts de la plateforme
Concrète		Dispositif concret dans le contexte de la plateforme permettant une pédagogie ouverte	

Figure 16 Deux processus menant à l'opérationnalisation d'un dispositif concret

Si initialement notre recherche a porté sur l'étude des dispositifs pédagogiques, c'est que la difficulté de mettre en place ceux-ci semblait constituer un frein à la pratique d'enseignement s'inspirant des théories médiationnelles. Nous allons voir dans la section suivante en quoi une approche centrée sur la notion de dispositif peut favoriser un tel enseignement.

5.4 Adéquation de la notion de dispositif à notre recherche

5.4.1 Dispositif et bricolage pédagogique

Il est assez paradoxal d'imaginer que le fait de proposer un dispositif, donc un cadre conceptuel a priori normatif, va permettre à des enseignants d'exprimer plus facilement leur créativité, d'improviser et de mettre en œuvre une pédagogie constructiviste ou socio-constructiviste. La notion de dispositif a été popularisée internationalement par Michel Foucault dans l'ouvrage *"Surveiller et punir : naissance de la prison"* Dans cet ouvrage, un dispositif est constitué : *"d'un ensemble hétérogène de chose dites et non dites"* (Foucault, 1975). André Berten, analyse cette vision du dispositif et souligne son caractère coercitif, *"le côté pouvoir normalisateur du dispositif le rend malveillant et inamical"* (Berten, 1998). L'auteur souligne cependant les aspects positifs que peut également prendre également un dispositif. Il propose de voir un dispositif comme un espace de potentialité, tolérant à l'erreur, permettant aux usagers de construire, d'improviser leur apprentissage tout en étant guidé par l'intentionnalité flottante résidant dans le dispositif. Il oppose alors à la rigidité des modes d'apprentissage pensés comme transmission du savoir, une approche privilégiant la notion de dispositif pensé comme un instrument de médiation favorisant des compétences cognitives générales et non spécialisées.

Claude Lévi Strauss, en comparant esprit bricoleur et esprit ingénieur, insiste sur l'aspect émancipatoire du dispositif (Lévi-Strauss, 1962). Dans le domaine de l'enseignement, un dispositif constitue pour l'enseignant un support individuel de bricolage créateur, il lui fournit un espace d'effectivité concret mais non contraint. Un tel espace, encadré par un dispositif, est avant tout un espace de potentialité, les services, l'organisation des groupes, les outils sont disponibles mais aucun scénario ne contraint à leur utilisation. *"Le dispositif est une œuvre ouverte qui invite l'enseignant, l'apprenant à l'interpréter dans son usage"* (Paquelin, 2005). Le dispositif peut être ainsi vu comme un espace de potentialisation par opposition au

scénario qui est un espace de planification. *"Il fait simplement exister un espace particulier préalable dans lequel ce "quelque chose" peut se produire"* (Peeters & Charlier, 1999).

Nous étudierons, dans le chapitre 8, les différentes facettes du bricolage enseignant, retenons qu'il est favorisé par une approche centrée sur les dispositifs. Une telle approche peut être initiée par un scénario ouvert ou par un dispositif abstrait, elle permet de proposer un espace pré-structuré techniquement et conceptuellement, celui-ci un espace de potentialité. Dans cet espace, l'enseignant, l'apprenant peuvent mener des activités d'appropriation de ce qui est "déjà là", cette mise en usage permet la co-construction du dispositif et des savoirs. (Paquelin & Choplin, 2003)

Cette approche par dispositif, que nous proposons consiste donc à mener une réflexion sur la potentialité du dispositif que l'on conçoit et non sur un ordonnancement d'activité simple ou adaptatif. Elle invite à se poser la question de la place et la définition d'une infrastructure informatique pour accompagner le travail de l'enseignant dans la construction d'un dispositif. *"Une infrastructure peut être définie comme un substrat social et technique qui stabilise et permet rapidement des activités instrumentées intentionnelles dans un domaine donné"* (Derycke, Hoogstoel, Peter, & Vantroys, 2006)

Certes l'outil informatique peut séquencer dans un workflow strict ou adaptatif les activités des apprenants, cet aspect fait d'ailleurs l'objet d'autres recherches au sein de notre laboratoire (Vantroys & Peter, 2005), (Derycke & Chevrin, 2007), pour autant est-ce souhaitable? Nous proposons de définir une infrastructure pour accompagner la création et la construction de dispositif sur des plateformes de formation, nous verrons, dans chapitre 8, que cette entreprise est complexe et qu'elle se heurte en particulier au problème de l'interopérabilité des plateformes de formation. Nous proposons dans la sous section suivante d'établir un premier cahier des charges de l'infrastructure que nous souhaitons produire.

5.4.2 Conclusion : un premier cahier des charges de notre proposition.

Pour conclure ce chapitre, nous reprenons les divers résultats énoncés dans les sections précédentes et nous énonçons un premier cahier des charges pour mettre en œuvre notre proposition. Ce cahier des charges s'énonce en quatre points que nous allons détailler ci-dessous.

Un dispositif est une offre de signification offerte aux usagers. Pour pouvoir engendrer une infrastructure menant à la construction d'un dispositif technique concret dans le contexte d'une plateforme, cette offre doit être automatisable. Elle doit pouvoir être manipulée informatiquement et les concepts, constituant cette offre, doivent pouvoir être produits.

1. Il nous faut mettre à la disposition des enseignants un formalisme permettant l'expression d'un dispositif abstrait, ou un outil leur permettant l'édition de leur propre formalisme.

L'élaboration d'un dispositif mobilise différents acteurs, enseignant, ingénieur pédagogique, informaticien, nous verrons, dans le chapitre 8, qu'un modèle de dispositif constitue un objet frontière (Il est perçu par tous, mais appréhendé différemment par chacun). Pour l'enseignant le dispositif constitue l'ensemble des moyens conceptuels et techniques mis à la disposition de son scénario pédagogique, ou simplement l'environnement conceptuel et technique porteur des potentialités au sein desquelles il désire improviser. Pour l'ingénieur pédagogique, le

dispositif est un ensemble de fonctionnalités qu'il doit parvenir à réifier sur une plateforme. Pour l'informaticien, le dispositif est un usage qui participe à l'élaboration de sa plateforme.

2. Pouvoir définir les concepts, les fonctionnalités d'une plateforme au sein d'un langage et permettre l'expression du dispositif dans ce langage constitue un enjeu important de notre recherche.
3. Il faut également disposer d'un mécanisme permettant le transformer un dispositif exprimé par un groupe d'enseignants dans leur sémantique propre en un dispositif exprimé dans une sémantique plus proche des concepts de la plateforme.

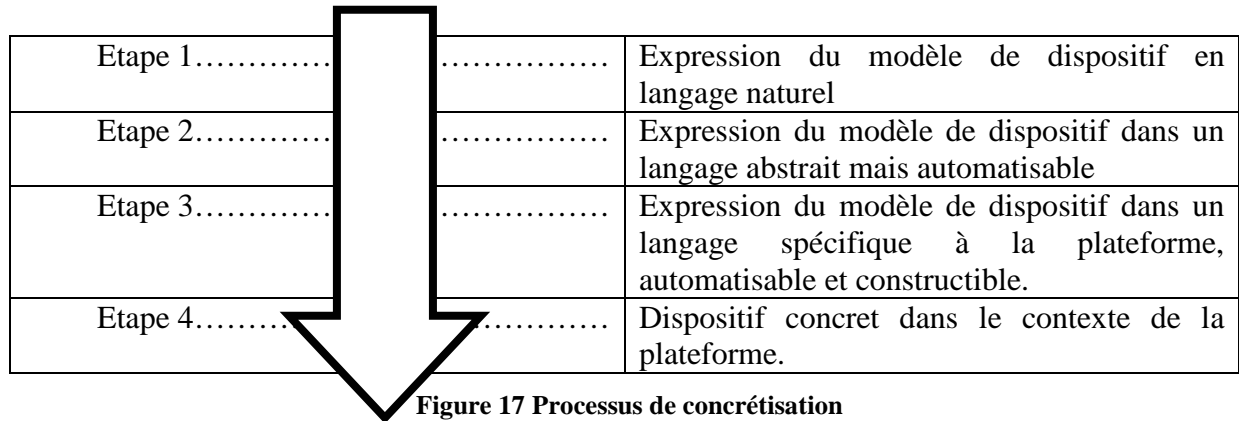
Enfin un dispositif sur une plateforme de formation est constitué d'un ensemble d'éléments, concrets construits par la plateforme et pouvant mettre en œuvre le scénario pédagogique de l'enseignant ou définir l'espace de potentialité qu'il désire proposer.

4. Il nous faut permettre l'élaboration d'un mécanisme concret de construction d'un dispositif sur une plateforme de formation (quelque soit le dispositif, quelque soit la plateforme) à partir de son modèle

Les quatre points de notre cahier des charges font référence à de nouvelles notions que nous n'avons fait qu'aborder jusqu'à présent il s'agit de **Définition de langages**, de **Modélisation**, de **Transformations** et **Construction**. Ces nouvelles notions peuvent permettre d'aborder la problématique de la réutilisation et du partage de dispositif pédagogique sous un angle qui ne passe pas par l'établissement d'une nouvelle norme, mais par l'établissement d'une infrastructure privilégiant un processus de création. Nous avons démontré dans les chapitres précédents l'incompatibilité entre la notion de norme et la notion de concrétisation, or l'établissement d'une norme ne constitue pas forcément une solution unique à cette problématique. L'ingénierie documentaire a montré depuis longtemps qu'un même contenu, exprimé dans un format de création exploitant une structuration logique, permet de mettre en œuvre une publication multi-supports. Un projet de transformation de contenu comme PISPAN (Gebers, 2005) a ainsi exploité les possibilités de transformations associées à un format de créations propriétaires et logiques pour produire des contenus dans différents formats d'exploitations. L'initiative a néanmoins conclu à la nécessité des standards dans un souci d'interopérabilité. Or nous verrons dans le chapitre suivant qu'un des buts de l'ingénierie dirigé par les modèles est justement de proposer une solution innovante aux problèmes d'interopérabilités.

<p>Nous pensons qu'il est possible de relier <u>langage</u>, <u>modélisation</u>, <u>transformation</u> et <u>construction</u> au sein d'une infrastructure pouvant être adaptée à chaque plateforme de formation. Cette proposition en privilégiant un processus de concrétisation permet de contourner les problèmes posés par les normes et standards.</p>

Dans ce chapitre, nous avons décrit un processus permettant la construction d'un dispositif à partir de sa définition. Ce **processus de concrétisation** permet de prendre en compte un enseignement constructiviste, et socio-constructiviste (cf. Figure 12 page 66). Pour accompagner ce processus nous suggérons de profiter des possibilités d'automatisation offertes par l'informatique.



Il est possible d'établir un parallèle entre ces quatre étapes du processus de concrétisation et l'ingénierie dirigée par les modèles (IDM) (cf. Figure 17). En effet la manipulation de langage, de modèles, la problématique des transformations et de l'opérationnalisation de modèle sont des sujets traités par l'IDM. Un processus semblable à celui que nous venons d'énoncer est d'ailleurs proposé à l'initiative de l'OMG sous le nom de MDA (Model Driven Architecture). Nous proposons de mener dans le chapitre 6 suivant un court état de l'art de l'ingénierie dirigée par les modèles. Cet état de l'art nous permettra de développer chapitre 8 notre proposition pour instrumenter l'approche par dispositif : **l'ingénierie pédagogique dirigée par les modèles.**

chapitre 6 Présentation de l'ingénierie dirigée par les modèles, liens avec nos travaux en EIAH

6.1 Qu'est ce que l'IDM

L'ingénierie dirigée par les modèles (IDM), en anglais Model Driven Engineering (MDE), est une approche spécifique du génie logiciel qui a pour objectif de définir un cadre théorique pour générer du code en utilisant des transformations successives de modèle. La réflexion sur l'IDM fait suite à la définition de l'approche Model Driven Architecture (MDA) (OMG, 2003) par l'OMG (OMG) (cf. section 06.4), elle entend ne pas se limiter à un jeu de standards spécifiques à un organisme particulier, et désire proposer autour de la notion de modèle et de métamodèle une vision unificatrice permettant de rassembler :

- **Différents courants de modélisation**, méthodes de modélisation, langages de modélisation spécifiques à des domaines, techniques de génération de code (via assistant, ou fichier de configuration...). (Greenfield & Short, 2003)
- **Différentes communautés informatiques** : Génie logiciel, Web Sémantique, Ingénierie des systèmes interactifs, Ingénierie des systèmes d'information etc. (Favre, 2006)
- **Différentes disciplines plus anciennes**, Sémiotique, Linguistique, etc... pouvant apporter à l'ingénierie dirigée par les modèles, des recherches abouties mettant en oeuvre des solutions différentes de celles que l'IDM propose.

Les objectifs de l'IDM sont donc de plusieurs ordres :

Pratique :

L'IDM a pour objectif la manipulation par des métamodèles de modèle **productifs** par opposition à des modèles descriptifs, contemplatifs. Cette productivité se décline de deux manières : un modèle doit être manipulable informatiquement, il doit aussi être opérationnalisable. Cette capacité à manipuler informatiquement les modèles permet l'appréhension de systèmes (applications, bases de données, dispositifs pédagogiques) selon différents **points de vue**, et la définition des mécanismes permettant de lier ces points de vue. Ces mécanismes peuvent être de plusieurs types, transformation, fusion, synchronisation. L'IDM a également pour objectif d'explicitier et de rendre opérationnelle la relation qui lie un modèle à l'élément qu'il représente sur différentes plateformes technologiques. Des mécanismes permettant l'opérationnalisation de modèle sur des plateformes spécifiques peuvent être définis dans le cadre de l'IDM. Bien que ces mécanismes reposent souvent sur des compilateurs, des interpréteurs, ou des générateurs de code, toute forme de construction peut être abordée par l'IDM, nous présenterons dans les trois études menées chapitre 11 notre solution basée sur la réalisation d'un générateur de constructeurs spécifiques.

Conceptuel et formel :

Défini initialement à partir du MDA et donc inspiré par une approche orienté objet, l'IDM a l'ambition de proposer une approche utilisable dans des espaces technologiques différents. Il lui faut ainsi définir des concepts et des relations permettant de manipuler des modèles dans des environnements ne proposant pas exclusivement des concepts objets. L'opérationnalisation d'un modèle dans de tels espaces technologiques ne passe pas forcément

par la production de code mais plutôt vers la construction des éléments qu'il représente. Ces concepts et ces relations seront expliqués dans la section suivante.

Intégratrice :

Un des objectifs de l'IDM est de construire, par son approche, un pont interdisciplinaire. On retrouve en effet des réflexions sur la formalisation, les transformations et l'opérationnalisation de modèles dans de nombreuses disciplines et sous des noms différents. En génie logiciel, le MDA propose d'aborder la modélisation à l'aide du langage UML, par des transformations successives, il est alors possible de produire des artefacts pouvant s'exécuter par exemple sur des plateformes telle que EJB (OMG, 2004), et .Net (Abd-Ali & El Guemhioui, 2005).

Les principes de l'IDM peuvent être appliqués aux langages spécifiques de domaine (DSL). Ces langages permettent la définition et l'opérationnalisation simple de modèles dans des domaines spécifiques. L'application des principes de l'IDM à ces langages, a permis la définition de plateformes de modélisation génériques, de telles plateformes basées sur l'utilisation d'un méta méta modèle permet ainsi la production formelle de langages spécifiques. (GMF), (GME), (Merlin), (GEMS), (MetaEdit), en sont des exemples. Notre laboratoire avec la thèse de Xavier le Pallec (Le Pallec, 2002) mène depuis 1999 des recherches autour des plateformes de modélisation, ces travaux ont permis la réalisation d'une plateforme (ModX), reposant sur la méta méta modèle MOF 1.4. Dans le domaine des bases de données, la gestion de métadonnées, en privilégiant la manipulation des structures aborde des problématiques traitées par l'IDM, (Alagic & Berstein, 2001). En EIAH, les travaux que nous avons présentés au chapitre 4 concernant la définition du langage IMS-LD (Koper, 2001), la définition de langage LDL (Ferraris, Lejeune, Vignollet, & David, 2005), la réingénierie de scénario (Corbière & Choquet, 2004) et l'utilisation de profil UML spécifique (Laforcade, 2005) font référence à des préoccupations abordées par l'IDM. Le chapitre 7 présentera plus en détail ces différentes recherches en EIAH, en les présentant selon une approche IDM. Nos travaux actuels, dont cette thèse se fait l'écho, propose dans le cadre des EIAHs, une approche IDM de la construction de dispositif sur une plateforme de formation. Les articles suivant, en constituent les jalons (Caron, 2003), (Caron et al., 2005a), (Caron, Le Pallec, & Sockeel, 2006a), (Caron, 2007).

Restructuratrice :

Il est possible de reconsidérer les travaux menés auparavant en génie logiciel, mais plus généralement en informatique, au regard des concepts fondamentaux de l'IDM. Ces concepts peuvent par exemple être appliqués à des travaux de réingénierie portant sur des applications patrimoniales (Favre & Musset, 2006). Les auteurs montrent que l'article fondateur de la réingénierie (Chikofsky & Cross II, 1990) pourrait être réécrit sans en changer le sens en remplaçant les mots représentation, vue et abstraction par le mot modèle. Il est possible de proposer en sciences, un point de vue centré sur les modèles, pour interpréter des systèmes formels. De tels modèles peuvent permettre l'anticipation de comportements ou la construction de règles empiriques. Ainsi l'IDM peut se positionner par rapport à quelques notions clés, rencontrées en sciences, comme simplification et abstraction, langage, métamodèle, qualité, vérité. (Terrasse, Savonnet, Leclercq, Grison, & Becker, 2005) Ce travail de positionnement a été effectué en ingénierie des systèmes, il a permis d'enrichir l'IDM, en particulier en suggérant d'accorder une place prédominante aux métamodèles.

De façon, plus générale, le rapport de synthèse de l'AS CNRS MDA conclut sur la pertinence de l'IDM dans d'autres champs que celui de l'informatique :

"L'IDM pourrait devenir le domaine dans lequel les résultats de recherche relatifs à la modélisation seraient collectés, mis en forme, homogénéisés, standardisés avant d'être mis à la disposition des diverses communautés scientifiques et industrielles"

"Il est probable également que les travaux sur l'IDM auront une influence profonde sur le contour des disciplines scientifiques telles qu'elles sont perçues aujourd'hui et, espérons le, favoriseront une unification de domaines actuellement cloisonnés." (Bézivin, Belaunde, & Marvie, 2004)

L'IDM est une approche qui a pour objectif d'offrir un cadre conceptuel aux pratiques de modélisation que l'on peut rencontrer en informatique et dans d'autres disciplines. Nous allons détailler dans la section suivante ce cadre conceptuel.

6.2 Les concepts fondamentaux de l'IDM

6.2.1 Quatre niveaux

L'IDM est fondé sur une architecture à quatre niveaux, initialement proposée dans le cadre du MDA. Cette architecture repose sur la définition d'un méta méta modèle (Crawley, Davis, Indulska, McBride, & Raymond, 1997), (OMG, 1997), (Bézivin & Gerbé, 2001). Elle fait maintenant l'objet d'un large consensus en IDM (Seidewitz, 2003), (Bézivin, Blay-Fornarino et al., 2005), on retrouve ainsi des références à cette architecture dans de nombreux travaux désirant se situer par rapport à l'IDM, par exemple, dans les Domaine Dirigés par les modèles (Ionita, Estublier, & Vega, 2005), les Base de Données à Base Ontologique (Dehainsala, Jean, Dung, & Pierra, 2005), en EIAH (Wang & Zhang, 2003), (Caron et al., 2005a), (Ferraris et al., 2005).

La Figure 18 présente cette architecture et propose pour chaque niveau quelques exemples.

Méta Méta modèle	MOF, eCore, profil UML, graphes conceptuels...	M3
Méta Modèle	UML, Java, EJB, .NET, IMS-LD, LDL, CPM...	M2
Modèle	Fichier XML, programme java, Fichier IMS-LD Modèle de dispositif pédagogique...	M1
Monde réel	Instance de programme, Dispositif pédagogique dans le contexte d'une plateforme...	M0

Figure 18 Architecture quatre niveaux

6.2.2 Le niveau M3

Le niveau supérieur, M3, définit les notions de base permettant l'expression des métamodèles (M2), et des modèles (M1), il permet d'exprimer les règles de conformités qui lient les entités du niveau M1 à celle du niveau M2. Le niveau M3 peut être réflexif c'est-à-dire que dans ce cas il se définit lui-même. Des méta métamodèle comme le MOF (Meta Object Facility) standardisé par l'OMG, eCore définit dans le cadre du "Eclipse Modeling Framework" (EMF) (Budinsky, Steinberg, & Ellersick, 2003), différents formalismes comme des profils UML (Unified Modeling Language), des graphes conceptuels, sNets, CDIF, OWL (Dinh, Gerbé, & Sahraoui, 2006) sont des exemples de méta métamodèles pouvant appartenir au niveau M3. Parmi ces métras métamodèles, le MOF possède un statut à part dans la mesure où bien que défini par l'OMG dans le cadre du MDA, il est maintenant largement utilisé indépendamment de ce standard. Des plateformes de modélisation générique implémentant le MOF permettent la production de modeleur spécifique à un domaine reposant sur un métamodèle appartenant au deuxième niveau (Bézivin, Brunette, Chevrel, Jouault, & Kurtev", 2005). Par exemple, la

plateforme de métamodélisation GME (Ledeczi, Bakay et al., 2001) implémente la version 1.4 du MOF (Emerson, 2005) (mais est compatible avec eCore d'EMF (Bézivin, Brunette et al., 2005)), ce méta métamodèle constitue le support des règles de conformités entre métamodèles et modèles au sein de l'environnement GME. Au sein de la plateforme, il est possible avec l'éditeur MétaGME de définir des métamodèles spécifiques à des domaines. De tels métamodèles permettent de produire des modeleurs graphiques reposant sur leur définition et sur les règles de conformité du MOF. L'outil que nous utilisons dans cette thèse est l'outil ModX (ModX), (Le Pallec, 2002), (Caron et al., 2005a), c'est une plateforme de métamodélisation rapide reposant sur l'implémentation du MOF 1.4. Cette plateforme développée dans le cadre de notre équipe de recherche définit un environnement au sein duquel il est possible d'éditer dynamiquement des métamodèles et modèles conformes au MOF. Tout changement intervenant sur la définition d'un métamodèle se répercute sur les modèles associés. L'éditeur de ModX offre également la possibilité de définir des transformations entre métamodèles.

6.2.3 Le niveau M2

Le niveau M2, est le lieu de définition des métamodèles, le MDA repose par exemple sur un métamodèle principal UML extensible par exemple par des profils. Les métamodèles que nous produisons avec ModX permettent quant à eux de définir des modeleurs spécifiques, basés sur un ou plusieurs métamodèles décrivant un domaine particulier. De tels modeleurs permettent alors de mener des activités de modélisation au sein de ces domaines. Nous décrivons ainsi dans cette thèse des exemples menés sur des plateformes de formations (Ganesha, Claroline, Moodle, WikiniMST), et permettant de construire des dispositifs pédagogiques spécifiques.

L'approche IDM consiste à manipuler des modèles pour en préciser le sens et les rendre opérationnalisables. Cette manipulation nécessite de définir au niveau M2, entre les métamodèles, des règles de transformation, de fusion, de composition, ou de tissage. Lorsque les métamodèles le permettent, il est possible de définir entre eux des règles de transformation. Ces règles permettent de mener des transformations entre modèles, c'est-à-dire la génération automatique d'un ensemble de modèles cibles à partir d'un ensemble de modèles sources (Czarnecki & Helsen, 2003). Un standard de moteurs de transformation est spécifié par l'OMG : QVT (Query/View/Transformation) (OMG, 2001), les initiatives MTL (MTL), ATL (Bézivin & Jouault, 2005) sont des exemples de l'implémentation de cette spécification. La plateforme de modélisation que nous utilisons, ModX, possède un environnement de transformation basée sur le MOF et utilisant des règles exprimées directement en Java. La fusion de métamodèles permet de manipuler dans le même environnement de modélisation des modèles issus par exemple de préoccupations métiers différentes. Pour mener à bien une fusion entre métamodèles il faut pouvoir résoudre les problèmes de cohérence qu'ils suscitent; la conformité des sous métamodèles n'entraînant pas la conformité du tout (Blay-Fornarino & Franchi, 2005). Il est possible de réaliser cette fusion en effectuant des compositions de domaines, les recouvrements peuvent alors servir comme points d'ancrage pour exprimer les nouvelles relations ajoutées par la composition. (Vega Baez, 2005). En EIAH, la fusion de métamodèles est abordée pour mettre en relation des préoccupations, pédagogiques, sociales et informatiques. (Vachet, 2006).

La démarche que nous présentons dans notre thèse consiste à définir des règles de transformations entre métamodèles, puis à considérer le métamodèle final comme un métamodèle fusionnant deux préoccupations "métier". Ce métamodèle est utilisé pour définir des modèles de dispositifs pédagogiques, il est également utilisé comme modèle permettant de définir une machine virtuelle spécifique à un domaine. Cette possibilité de considérer un métamodèle de niveau M2 comme un modèle de niveau M1 est en particulier évoqué par (Marvie, Duchien, & Blay-Fornarino, 2005). Nous reviendrons sur cet aspect de notre travail dans la section 6.6 de ce chapitre, ainsi que dans le chapitre 9 et le chapitre 10.

Un métamodèle peut être considéré comme un langage spécialisé pour un aspect d'un système, or, comme la construction d'un système informatique revient à effectuer un tissage d'aspects (Kiczales et al., 1997), il est possible par des tissages d'aspects de construire à partir de différents modèles un système. Le maintien en cohérence de tous ces modèles, ainsi que le passage automatisé entre certains d'entre eux, forme alors une nouvelle problématique pour le développement du logiciel centré autour de cette notion d'aspect (Marvie et al., 2005).

6.2.4 Le niveau M1

C'est au niveau M1 que les modèles sont édités. Un des objectifs de l'IDM est de produire des modèles qui puissent être validés formellement, cette validation favorisant leur manipulation par des outils informatiques et leur opérationnalisation (Blay-Fornarino & Franchi, 2005). C'est cette validation qui distingue l'IDM des méthodes de modélisations antérieures. Un modèle est valide s'il est conforme à son métamodèle, les règles de conformités étant définies au niveau du méta méta modèle. De telles règles sont par exemple : le respect de contraintes de cardinalité exprimé par le métamodèle. La conformité d'un modèle avec son métamodèle peut être de plusieurs types, empirique, littérale, théorique, outillée, mais pour que cette conformité soit productive il est souhaitable qu'elle soit basée sur une représentation de métamodèle qui soit formel et exécutable. Une plateforme de modélisation implémentant un méta méta modèle permet d'établir de telles règles de conformité bien fondées entre métamodèles et modèles. (Blay-Fornarino & Franchi, 2005).

6.2.5 Le Niveau M0

Le niveau M0, n'appartient pas au monde de la modélisation mais au monde du réel, il est rarement traité, mais l'exploitation de modèle pour la construction automatique de système est identifié comme un des verrous auxquels est confronté l'IDM (Bézivin, Blay-Fornarino et al., 2005). Nous verrons dans les chapitres suivants les différentes possibilités pour construire un système à partir d'un modèle. Etablir le métamodèle d'une plateforme réelle à partir de l'étude des systèmes en cours sur la plateforme constitue également un verrou (Marvie et al., 2005). Nous abordons dans le chapitre 9 ces deux verrous.

6.2.6 Conclusion

Les quatre niveaux que nous venons de décrire font maintenant l'objet d'un consensus au sein de la communauté IDM. Ils mettent en scène deux relations que nous allons décrire dans la section suivante, la première est la relation de "*conformité*" elle lie un modèle (M1) à son métamodèle (M2); la seconde est la relation "*représentation de*", elle lie un modèle à l'élément (M0) qu'il représente, rendre cette relation opérationnelle, c'est-à-dire construire à partir d'un modèle le système représenté constitue pour le domaine des EIAH un des résultats de cette thèse.

6.3 Différentes relations.

L'approche IDM s'appuie sur deux relations fondamentales : "*représentation de*" et "*être conforme à*", attachons nous dans un premier temps à décrire la première relation.

6.3.1 La relation "*représentation de*"

Cette relation, qui lie le modèle au système qu'il représente, nécessite de définir soigneusement la notion de modèle. Il est couramment admis qu'un modèle est la simplification subjective d'un système existant ou qui existera, cette simplification est alors sous tendue par une intention et le modèle obtenu doit alors être capable de répondre à des questions à la place du système qu'il représente. (Bézivin & Gerbé, 2001), (Bézivin et al., 2004).

On trouve dans cette définition finalement toutes les caractéristiques d'un modèle descriptif tel qu'il peut être manipulé par des méthodes classiques de modélisation. Or un des objectifs de l'IDM est de manipuler des modèles formalisés pour les rendre productifs. Etre productif pour un modèle signifie que l'on peut automatiser son traitement et diriger, par les modèles et les métamodèles, la construction du système qu'il représente. La relation "*représentation de*" implique donc dans le cadre de l'approche IDM deux caractéristiques fondamentales.

- La première concerne le cadre formel au sein duquel doit s'effectuer la définition du modèle, c'est l'objet de l'IDM de définir ce cadre, un modèle doit être "*conforme*" à son métamodèle, cette relation sera abordée dans le paragraphe suivant.
- La seconde concerne le caractère opérationnalisable de la relation "*représentation de*", si en génie logiciel la génération de code est la pratique la plus couramment utilisée, ce n'est pas le cas pour d'autres disciplines informatiques, en particulier lorsque le modèle ne représente pas du code.

Etudier comment un élément de modèle peut factoriser une collection d'éléments du système modélisé constitue un véritable challenge souvent abordé dans un cadre peu formel par les DSL. Nous proposons dans notre thèse d'illustrer cette dimension de construction de la relation "*représentation de*" dans le cadre de la construction de système sur des plateformes de formation.

6.3.2 La relation "*être conforme à*"

Manipuler informatiquement et opérationnaliser des modèles nécessitent qu'ils soient exprimés dans un langage clairement défini. L'IDM propose par son architecture, un cadre formel permettant cette définition, il est par exemple possible, en respectant les règles du MOF, de construire à partir d'un métamodèle défini en MOF, un modèle conforme. Cette relation de conformité peut être supportée directement par la plateforme de modélisation utilisée. Dans le cadre de nos travaux nous utilisons la plateforme de modélisation ModX celle-ci implémente la version 1.4 du MOF et permet une vérification de conformité par "spécifications exécutables" (Blay-Fornarino & Franchi, 2005).

La Figure 19 illustre, dans l'espace de technologique de modélisation qui est le notre, la modélisation de dispositifs pédagogiques en EIAH. Elle présente les deux relations que nous venons de décrire.

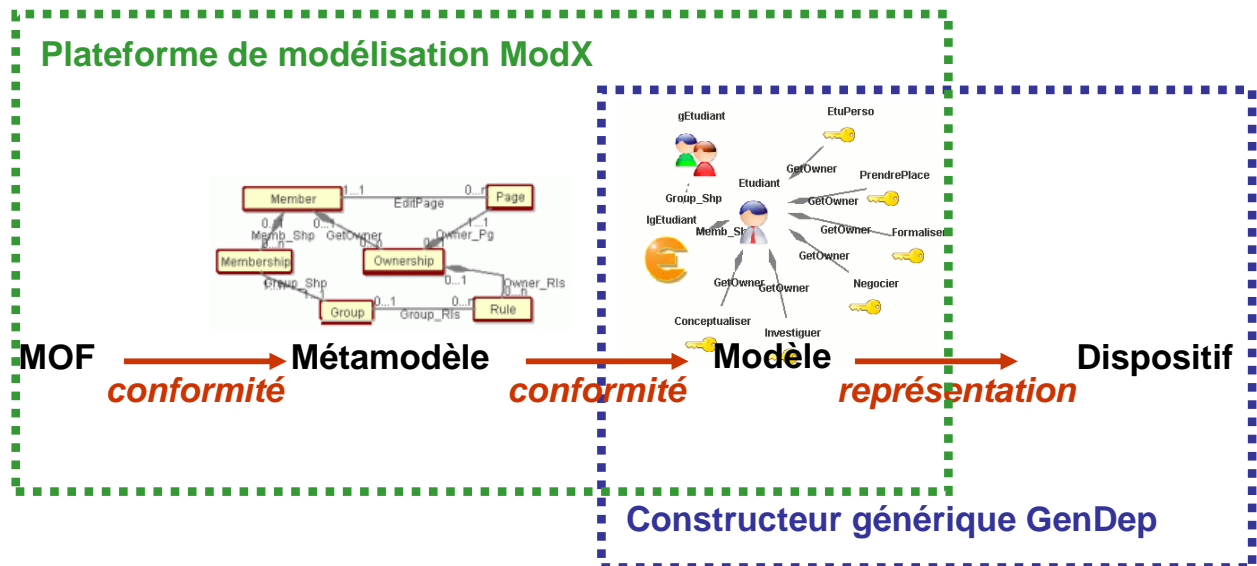


Figure 19 Plateforme de modélisation, support à la vérification de conformité par spécification exécutable.

Les modèles manipulés en IDM possèdent deux caractéristiques qui les distinguent des modèles classiquement utilisés par des méthodes de modélisation, ils sont définis formellement par la relation de conformité qui les lie à leur métamodèle, ils sont opérationnalisables informatiquement. Cette dernière caractéristique a fait l'objet de recherche dans le cadre de cette thèse, des résultats ont été obtenu en EIAH. Notre approche de la modélisation en EIAH a consisté à utiliser une plateforme de modélisation (ModX) pour produire des modeleurs spécifiques à notre domaine technologique, les résultats de cette recherche seront détaillés dans le chapitre 10. Nous avons appliqué à cette approche IDM, un processus de concrétisation simple s'inspirant du processus MDA. La section suivante présente ce processus.

6.4 Le MDA

6.4.1 Présentation générale

Le Model Driven Architecture (MDA) (OMG, 2003) est un exemple particulier d'ingénierie dirigée par les modèles. Défini par l'OMG (OMG), il se révèle être un jeu de standard pour la création de métamodèles spécifiques à un domaine ou à une plateforme (MOF et profils UML), pour la modélisation (formalisme UML et contraintes OCL), le support d'échanges de modèles (XMI) et la transformation de modèles (CWM, QVT).

Outre ce jeu de standard, le MDA est une sorte de guide structuré qui fait actuellement référence en IDM. Cet état de fait provient essentiellement du fait que l'OMG catalyse, centralise, et synthétise bon nombre de travaux sur l'IDM.

Le MDA préconise une approche de modélisation basée sur un processus de concrétisation proche de notre recherche en EIAH. Ce processus de "référence" est illustré sur la Figure 20. Celui-ci débute par une définition des besoins, une définition de la structure et du comportement de la future application dans un ou des modèles complètement dédiés au domaine concerné (Clark et al., 2002).

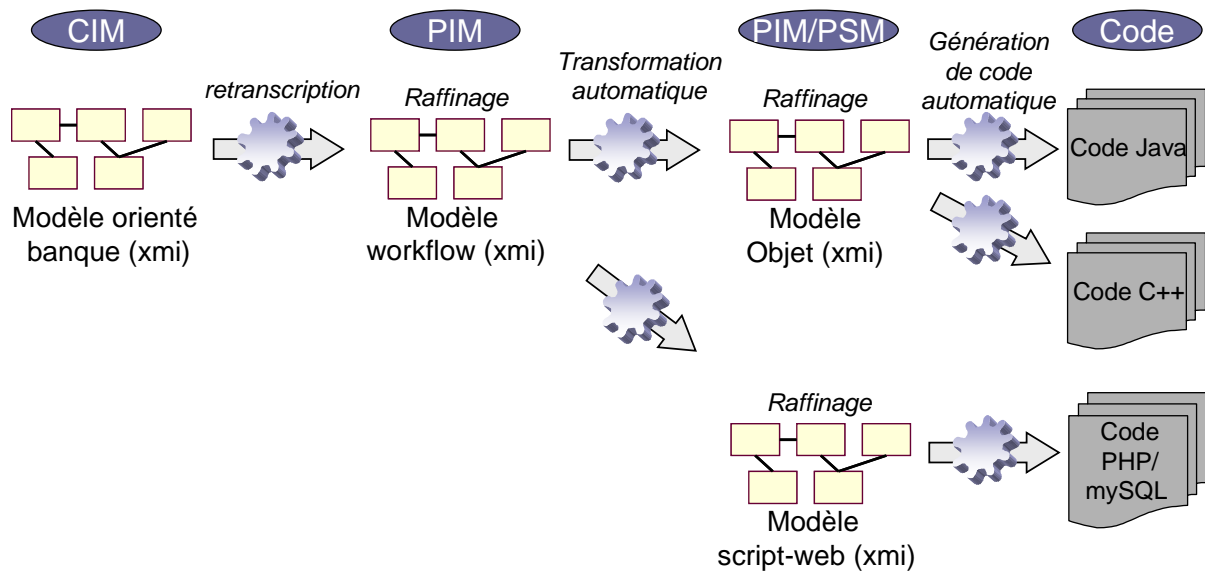


Figure 20 Exemple de cycle de vie MDA pour un modèle Bancaire

Ces modèles, appelés Computer Independent Model (CIM), ne présentent aucune trace d'informatique. Par exemple, dans le domaine Bancaire, il pourrait s'agir d'un texte où il serait indiqué que certaines compensations peuvent être traitées le soir. Un modèle CIM sert de base à la définition d'un modèle PIM (Platform Independent Model) où le métamodèle sous-jacent est cette fois-ci de type informatique mais non rattaché à une technologie particulière (ex : un modèle objet). Le passage d'un modèle CIM à un modèle PIM est normalement manuel et sujet à discussion entre experts du domaine et informaticiens. Mais rien n'empêche une projection automatique (Zhang, Mei, Zhao, & Yang, 2005). Le modèle PIM est, dans une troisième étape retravaillé afin d'indiquer des aspects propres à l'informatique (indications inadéquates dans le modèle CIM) au travers d'un paradigme particulier (ex : l'objet, le procédural...). La quatrième étape consiste à produire, au travers d'une transformation automatique (interactive ou non), un modèle spécifique à une plateforme, PSM (Platform Specific Model). Un modèle EJB (modèle UML utilisant le profil EJB) est un exemple de modèle de type PSM. Après raffinement de ce type de modèles, celui-ci sert à générer automatiquement une partie du code final d'implémentation (l'autre partie étant à écrire à la main). L'approche MDA, repose sur des mécanismes fiables de transformation de modèles, dans la sous-section suivante, nous présentons quelques généralités sur cette notion.

6.4.2 Généralités sur la transformation de modèles

La transformation de modèles est une opération qui sert à produire un modèle (appelé modèle cible) à partir d'un modèle (appelé modèle source). Elle est constituée, selon le MDA de deux étapes (cf. Figure 21). La première est une spécification des règles de transformations, décrivant la correspondance entre les concepts des métamodèles sources et les métamodèles sources. La deuxième étape est une application des règles pour générer la transformation du modèle cible en modèle source (Bézivin, 2001) (Czarnecki & Helsen, 2003).

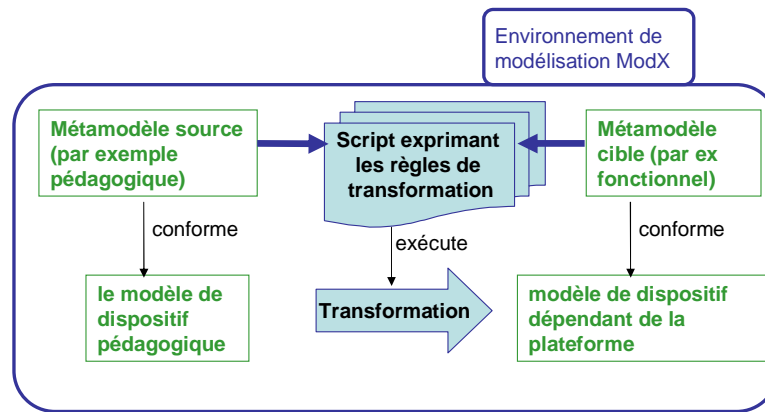


Figure 21 Exemple de transformations dans l'environnement ModX

Query/View/Transformation (QVT) (OMG, 2001) est un standard défini par l'OMG. Il s'agit d'un langage standardisé pour exprimer les transformations de modèles. Différents langages de transformation implémentent le langage QVT ou s'en rapprochent. Ainsi ATLAS Transformation Language (ATL) est un langage de transformation de modèles implémentant le standard QVT de l'OMG (Bézivin, Dupé, Jouault, Pitette, & Rougui, 2003), (Bézivin, Brunette et al., 2005). Il est disponible en tant que plug-in dans le projet Eclipse GMT Project et est en voie d'intégration à une solution standardisée par l'environnement Eclipse. Ce nouveau projet est nommé M2M (Model to Model), il propose de définir un outil open source "QVT-like" (m2m). D'autres implémentations comme TRL (Sriplakich, 2003) ou ModTransf (Bondé, Dumoulin, & Dekeyser, 2004) proposent également des moteurs de transformation reposant comme ATL sur la transformation de fichier XMI.

6.4.3 Avantages du processus MDA

Le processus MDA présente quatre avantages : (Selic, 2003)

- **Réutilisation.** Chaque modèle augmente la réutilisabilité en factorisant des informations de plus en plus proches de la réutilisation. Le modèle CIM servira à toutes réalisations informatiques ; le modèle PIM servira et pourra être le point de départ à toutes réalisations adoptant le même paradigme ; le modèle PSM servira à toutes implémentations sur des plateformes d'une même technologie pour un certain nombre de version de cette dernière.
- **Capitalisation.** Les mécanismes de transformation automatique (PIM-PSM) ou les consignes de retranscription (CIM-PSM) sont normalement "standard" dans le sens où elles s'enrichissent chaque année. Cet aspect présente un haut degré de capitalisation.
- **Intuitivité.** Le modèle PSM permet d'appréhender plus facilement la réalisation finale.
- **Progressivité.** Le modèle PSM va permettre une génération de code assez efficace.

6.4.4 Critiques à l'encontre du processus MDA

Les réserves exprimées à l'encontre du MDA portent essentiellement sur :

- **La difficulté de maîtriser le métamodèle UML jugé trop généraliste.** Microsoft par exemple a fait le choix avec "Software Factories" de privilégier des langages de domaines (Domain Specific Languages ou DSL) de petites tailles, facilement manipulables, transformables et estime que son utilisation d'UML sera surtout descriptive. (Greenfield & Short, 2003) Cependant MDA ne fait que préconiser l'utilisation d'UML et n'exclut pas que d'autres langages puissent être utilisés. De

même le MDA privilégie l'utilisation de profil UML pour définir des métamodèles PSM car il simplifie les règles de transformation à établir entre PIM et PSM, mais permet que les métamodèles (en particulier PSM) soient exprimés autrement. (Blanc, 2005)

- **Le caractère simpliste de l'approche CIM PIM PSM.** Ce reproche s'appuie pour une part sur l'argument précédent, il semble naïf d'imaginer, que des modèles exprimés en UML puissent être appréhendés par tous les experts métier intervenant dans la définition d'un PIM. (Bézivin, Blay-Fornarino et al., 2005) D'autre part les travaux actuels portant sur les compositions de modèles ont montré la difficulté de relier entre eux non seulement les différents modèles, représentant les différents aspects d'une application, mais aussi de synchroniser des processus de conceptions différents ne respectant pas forcément l'approche **CIM PIM PSM** (Estublier, Vega, & Ionita, 2005). Il faut cependant rappeler que le MDA constitue une approche, et non une méthode, fondée sur l'utilisation de modèles pour assurer la séparation des préoccupations. Appliquer réellement le MDA nécessite inévitablement de définir une méthode. Cette méthode bien que s'inscrivant dans le cadre naïf du MDA peut être complexe. (Blanc, 2005) Elle peut par exemple passer par la mise en place de guidelines, de best practices, il est possible de définir des macro processus (Kent, 2002) ou des processus de modélisation incrémentaux (Marvie & Nebut, 2006). Un tel cadre de travail permettant de définir et d'outiller des microprocessus incrémentaux, c'est-à-dire des processus permettant la construction incrémentale d'un modèle, a par exemple été mis en place en EIAH dans notre laboratoire sur la plateforme de modélisation ModX pour faciliter l'appréhension du métamodèle IMS-LD (Le Pallec et al., 2006).

6.4.5 Conclusion

L'approche MDA propose un cadre structuré permettant de mener des activités de modélisation. Elle a permis en particulier de focaliser l'IDM sur la nécessité de définir formellement les règles de transformation entre métamodèles. Au cours de nos travaux, nous avons respecté ce cadre structuré tout en substituant au métamodèle UML des métamodèles plus simples à utiliser pour les divers intervenants du processus de modélisation. Les transformations que nous définissons entre PIM et PSM sont exprimées au sein de l'éditeur de ModX sous forme de programme Java et expriment des correspondances MOF entre les métamodèles. Elles permettent au niveau M1 de transformer des modèles de dispositifs pédagogiques métiers en des modèles de dispositifs spécifiques à une plateforme de formation particulière.

En IDM de nombreuses approches s'inspirent de l'approche de concrétisation MDA. Il est possible d'utiliser en IDM, des transformations pour appliquer des patrons de conception (Le Guennec, Sunyé, & Jézéquel, 2000) (Jézéquel, Gérard, Mraidha, & Baudry, 2005). Le processus de conception d'un modèle peut alors être vu comme l'application progressive de patron de conception permettant l'affinage du modèle. Ce processus progressif d'affinage menant à un modèle d'implantation reste par sa démarche de concrétisation proche du MDA. Les transformations sont également utilisées pour séparer les préoccupations entre différentes vues ou au contraire réaliser des compositions de métamodèles (Vega Baez, 2005) Cependant si le MDA aborde les problématiques liées à la composition de domaines par la définition de règles de transformation, il ne préconise pas de méthode pour l'opérationnalisation de modèle PSM. Dans le cadre plus large de L'IDM, il est possible d'aborder la composition de modèle avec une approche privilégiant l'opérationnalisation des modèles produits. C'est ce que nous présentons dans les sections suivantes.

6.5 D'une approche MDA à la composition de domaine

La composition de modèle propose une approche un peu différente de l'approche MDA. Plutôt que de considérer un processus simple de concrétisation menant des métamodèles métiers aux métamodèles spécifiques à l'application, la composition de modèle propose de mener en parallèle un processus de concrétisation métier et un processus de concrétisation technique, le premier processus étant synchronisé au second. L'avantage de cette approche est de minimiser la production de code. La problématique abordée par la composition de domaine concerne essentiellement trois points :

Le premier point concerne **les différents sens de la relation "être conforme à"**. Une des conséquences de travailler simultanément sur des modélisations exprimées dans des domaines différents est de devoir gérer les différentes significations prises par cette relation de conformité, ce travail est possible s'il est mené au sein d'un environnement supportant les différents type de modélisation. Il faut donc disposer d'un environnement basé sur un méta métamodèle formel et permettant la définition de plusieurs métamodèle. Parmi les différentes approches IDM, un tel environnement se situe plutôt du côté des DSL (Wile, 2001) que du côté du MDA. Un tel environnement n'interdit cependant pas de mener une démarche proche de la démarche MDA. La plateforme de modélisation ModX que nous utilisons permet d'exprimer plusieurs métamodèles en MOF et de les sauvegarder en XMI. Sur cette plateforme nous appliquons un processus de concrétisation caractéristique d'une approche MDA pour définir des modèles de dispositif.

Le deuxième point abordé par la composition de domaine concerne **les problèmes de synchronisation de métamodèles**. Comme précédemment la synchronisation de métamodèle doit être menée au sein d'un environnement supportant les différents types de modélisation. Cette synchronisation peut être traitée simplement en recherchant un métamodèle unique fusionnant les différents métamodèles. Dans cette optique la composition consiste à réunir par transformation les différents modèles, cette réunion nécessite cependant une nouvelle génération du code. Une autre option consiste à synchroniser dynamiquement les différents métamodèles. La plateforme GME propose par exemple la possibilité de définir des opérateurs entre différents métamodèles pour servir de point de jonction lors de leur composition (Ledeczi, Maroti et al., 2001). Cependant cette plateforme ne produisant pas des modèles directement exécutables, la réutilisation obtenue par la composition de métamodèle est relative, il faut générer à chaque nouvelle composition le code de l'application, activité qu'il faut compléter manuellement. (Jézéquel et al., 2005). Il est possible, si les modèles sont exécutables, ou interprétables automatiquement de définir des mécanismes de synchronisation au niveau des métamodèles qui se répercuteront automatiquement au niveau des modèles et de l'application. Cette condition est nécessaire pour mener une composition conceptuelle de métamodèle. (Estublier et al., 2005).

Le troisième point concerne **les différents sens de la relation "représente"** et son opérationnalisation. Nous venons de voir que pour mener à bien une composition synchronisée de métamodèles, il faut disposer de modèles opérationnalisables. C'est-à-dire manipulables informatiquement et permettant de construire automatiquement les systèmes décrits. Pour relier deux métamodèles, il est possible ainsi de définir un métamodèle de relation, et d'exprimer dans ce métamodèle les relations à établir entre les modèles que l'on veut réunir (cf. Figure 22). Il faut alors définir un mécanisme de réalisation de ces relations, en interprétant les relations, ce mécanisme permet la construction du système. (Estublier et al., 2005)

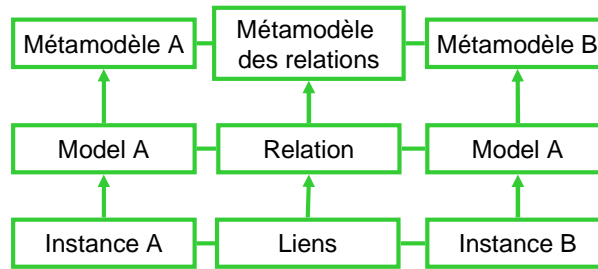


Figure 22 Composition de 2 métamodèles et métamodèle des relations

La composition renvoie alors deux définitions de la relation "représente", la première est classique, il s'agit d'une instanciation de l'application, la seconde concerne l'interprétation du modèle des relations pour déterminer les éléments à instancier. Cette double définition est également utilisée pour coordonner via des synchronisateurs, métamodèle du procédé et métamodèle UML d'une application (cf. Figure 23). Le modèle du procédé permet alors de piloter l'instanciation du modèle de l'application (Sanlaville & Estublier, 2005).

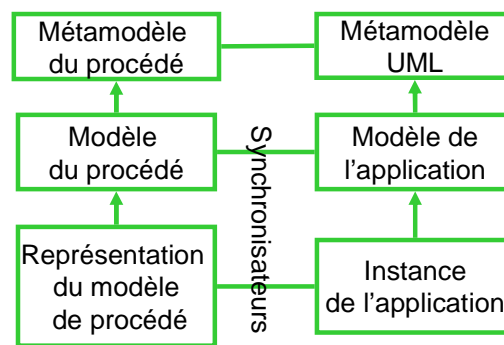


Figure 23 Synchronisation de domaine métier et applicatif

Enfin il est possible pour rendre opérationnalisable un modèle de construire la machine virtuelle qui l'interprétera à partir de métamodèle définissant le domaine, cette solution simple consiste à considérer le métamodèle du domaine comme un modèle d'application. Cette solution évite en particulier l'utilisation de synchronisateurs (cf. Figure 24), (Vega Baez, 2005).

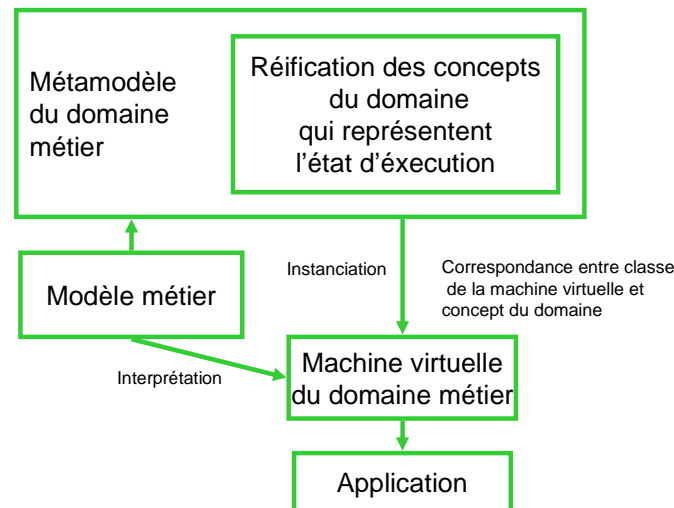


Figure 24 Opérationnalisation par création d'une machine virtuelle

Un des intérêts pour notre recherche de présenter les problématiques posées en IDM par la composition de domaine, est que cette approche aborde de façon pragmatique l'opérationnalisation des modèles produits. En effet comme nous venons de le rappeler :

Pour que la réutilisation de la démarche soit acquise il faut qu'elle permette une réelle automatisation de la phase d'opérationnalisation des modèles.

La dernière solution présentée, construire une machine virtuelle à partir de la réification, intégrée au métamodèle du domaine, de l'état d'exécution des concepts du domaine, est complexe à mettre en œuvre. Néanmoins notre solution qui consiste, dans notre environnement de modélisation, à construire un générateur de constructeurs spécifiques à partir du métamodèle du domaine s'en rapproche, cette solution sera présentée dans le chapitre 9 et le chapitre 10. Nous présentons dans la section 6.6 suivante, différentes propositions abordant l'opérationnalisation de modèles, ces propositions ne correspondent pas à notre domaine d'application; elles permettent cependant de présenter les différentes problématiques liées à l'opérationnalisation d'un modèle.

6.6 Construire un système pour opérationnaliser un modèle

6.6.1 Opérationnaliser un modèle en génie logiciel

De nombreuses techniques permettent, en génie logiciel l'opérationnalisation d'un modèle sur une plateforme d'exécution, mais principalement trois techniques sont employées : l'interprétation de modèles, la compilation de modèles et la génération de code (Jézéquel et al., 2005) :

L'interprétation de modèle nécessite la construction d'une machine virtuelle spécifique à chaque domaine applicatif. Cette option peut s'avérer intéressante lorsqu'il s'agit de produire des lignes logicielles et donc de réutiliser fréquemment les mêmes bibliothèque de composant. En couplant cette machine virtuelle à un éditeur de modèle, il est alors possible de créer des instances dynamiques d'une application. Nous avons vu dans la section précédente que la construction de la machine virtuelle peut être directement déduite des concepts inclus dans le métamodèle métier.

La compilation de modèles consiste à transformer directement le modèle en une application exécutable sur une plateforme d'exécution donnée. Cette compilation nécessite l'existence de compilateur pour chaque plateforme considérée. Actuellement la constante évolution des langages de modélisation et le grand nombre de plateforme d'exécution rend peut praticable cette option.

La génération de code est la technique d'opérationnalisation la plus fréquente. Les processus généraux impliquant un générateur sont résumés par la figure ci-dessous : (Cleaveland, 1988)

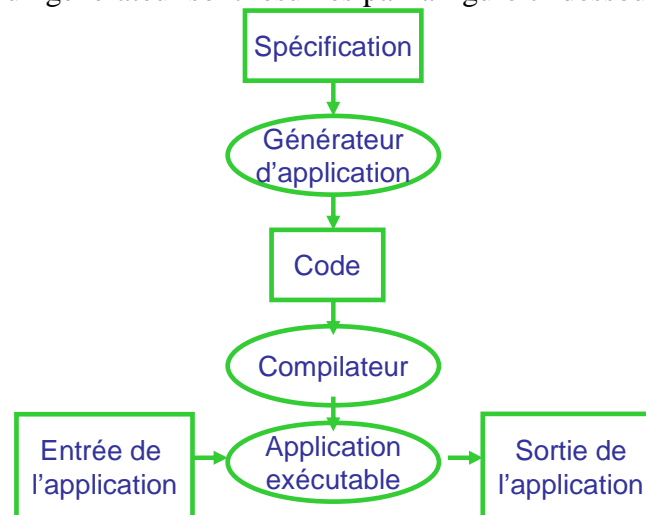


Figure 25 Processus basique pour développer un programme en utilisant un générateur d'application

Le processus débute avec une spécification qui décrit le comportement de l'application, cette spécification est interprétée par le générateur qui produit alors du code. L'application est alors obtenue par compilation du code produit par le générateur (cf. Figure 25).

Deux démarches concurrentes permettent de simplifier l'implémentation du générateur. La première consiste à restreindre la plateforme d'exécution à un ensemble de composants ou de services. On obtient alors un squelette d'application, c'est-à-dire un ensemble de classes et de mécanismes associés à une architecture logicielle qui facilite la génération de l'application. En appliquant une démarche IDM, il est possible de représenter les spécifications par un modèle conforme à un métamodèle spécifique au domaine étudié, l'intérêt de l'approche étant de minimiser l'implémentation du générateur en le reliant au métamodèle du domaine, ce métamodèle représente alors une abstraction du fonctionnement des composants ou services constituant le squelette d'application du domaine applicatif. Relier un squelette d'application à un métamodèle nécessite d'isoler les abstractions du domaine applicatif, ce travail a par exemple été réalisé par un processus MDA pour générer l'interface WSDL de Web Service (Vara, de Castro, & Marcos, 2005). En IDM les générateurs d'applications sont souvent employés au sein d'un DSL. En effet la création d'un générateur forme, avec la définition d'un langage et le design d'un éditeur, la "triade sacrée" des DSL. Les processus de création d'un DSL suivent alors les étapes suivantes : (Bhanot, Paniscotti, Roman, & Trask, 2005) (Tolvanen, 2006)

- Isoler les abstractions de la plateforme applicative, étudier son fonctionnement.
- Spécifier les concepts du langage et leurs règles (créer le métamodèle du domaine)
- Créer une représentation graphique associée ce métamodèle (notation)
- Définir les générateurs de codes

De tels générateurs nécessitent d'analyser les fichiers, composants, bouts de code, fonctionnalités, services, constituant le squelette d'application de la plateforme cible, C'est ce squelette d'application qui servira de référence lors de la génération du code. Après avoir défini le métamodèle, il faut lier ce métamodèle au squelette d'application. Cette liaison peut par exemple être réalisée sous forme de scripts. Cet ensemble de scripts, associés au squelette d'application de la plateforme, constitue alors le générateur. Une fois que le modèle de l'application est défini, le générateur construit l'application, le résultat doit alors être complété manuellement (Muszynski & AG, 2005). Juha-Pekka Tolvanen (MetaEdit) insiste sur la nécessité de minimiser le rôle tenu par le générateur, en intégrant au métamodèle, les décisions de haut niveau sujettes à variation et en réservant au squelette d'application les implémentations stables de bas niveau, les scripts composant le générateur ne devant servir qu'à faire la glue (Tolvanen, 2004) (Tolvanen, 2005).

6.6.2 Opérationnaliser un modèle en IDM

Les différentes techniques utilisées en Génie Logiciel, pour opérationnaliser un modèle, sont généralisables au domaine plus vaste de l'IDM. Cependant opérationnaliser un modèle dans le contexte d'une plateforme d'exécution peut signifier des choses très différentes selon le domaine applicatif dans lequel on se situe. De façon générale, une plateforme d'exécution définit un interpréteur pour l'exécution de systèmes. Elle est elle-même un système d'information qui offre des services à d'autres systèmes appelés applications (Marvie et al., 2005). Ces différents aspects que peut prendre une plateforme d'exécution entraînent différentes significations en IDM pour les mots compilation et génération de code. Il est possible par exemple d'appliquer une démarche IDM pour générer des interfaces WSDL à partir de la modélisation de Web Services, la plateforme d'exécution étant alors une plateforme logicielle. L'opérationnalisation d'un modèle consiste dans cet exemple à écrire un fichier XML décrivant les services Web modélisés et respectant la norme supportées par la plateforme d'exécution, c'est-à-dire la norme WSDL (Vara et al., 2005). Des plateformes d'exécution peuvent également être constituées d'éléments physiques et logiciels, les travaux portant sur la définition d'un métamodèle, pour l'automatisation du déploiement d'applications logicielles, ont ainsi montré la possibilité de définir un modèle de plateforme permettant d'abstraire le problème de cohérence rencontré en déployant des applications. Un modèle de déploiement peut alors être transformé en procédé de déploiement interprétable par un moteur de workflow (WfMC). L'opérationnalisation du modèle passe alors par l'exécution par le moteur de workflow du procédé de déploiement dans le contexte de la plateforme d'exécution, constituée ici d'ordinateurs et d'applications déjà installées (Merle, 2004).

6.6.3 Dans notre contexte

Appliquer une démarche IDM à la construction de dispositif pédagogique sur des plateformes de formation nécessite de définir clairement la plateforme d'exécution considérée. Nous verrons, dans le chapitre 9, que cette définition constitue un véritable verrou scientifique, mais qu'il est possible de proposer des bonnes pratiques ainsi qu'une solution conceptuelle et technique basée sur une architecture orientée service pour faciliter cette définition. Dans le cadre de notre solution, l'opérationnalisation d'un modèle de dispositif passe alors par la génération d'une file d'appel de services Web. Il est possible d'automatiser presque totalement la construction du générateur en utilisant l'interface WSDL fournie par la plateforme de formation, et le métamodèle spécifique à la plateforme d'exécution.

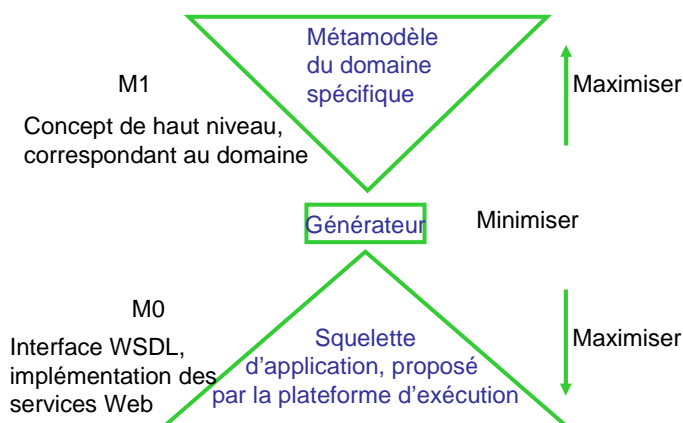


Figure 26 Appliquer la démarche préconisée (Tolvanen, 2004)

L'interface WSDL peut être considérée comme une description du squelette d'application, ici constituée par l'implémentation des services Web. Le métamodèle du domaine est constitué d'une fusion du métamodèle métier et du métamodèle déduit de l'interface WSDL précédente. Cette construction nous permet d'appliquer la démarche préconisée par Juha-Pekka Tolvanen, (cf. Figure 26) minimiser le plus possible l'implémentation du générateur, maximiser le squelette d'application en lui associant les implémentation de code, maximiser le pouvoir de modélisation du métamodèle en favorisant la définition d'expression de haut niveau. Notre thèse détaille dans le chapitre 9 cette solution conceptuelle, le chapitre 10 aborde l'implémentation effective de ce générateur.

Les deux solutions passant par la création de machine virtuelle ou de compilation n'ont pas été retenues pour une application dans le domaine des EIAHs. La compilation est exclue puisque nous ne produisons pas de code. Construire des machines virtuelles suppose de disposer d'un nombre limité de plateforme d'exécutions, ou de pouvoir exprimer toutes ces plateformes par un même squelette d'application que la machine pourrait adresser. Le grand nombre de plateforme de formation, leur diversité présentée dans les premiers chapitres interdisent ce type de solution. Qui ne pourra être envisagée que lorsque le problème de l'interopérabilité des plateformes de formation aura été résolu.

6.7 Conclusion de l'état de l'art modélisation.

L'approche IDM est une approche spécifique du génie logiciel qui a pour objectif de définir un cadre théorique pour générer du code en utilisant des transformations successives de modèle. Cette approche est caractérisée par une architecture à quatre niveaux, M3 occupée par un méta méta modèle, M2 les métamodèles, M1 les modèles, M0 le mode réel. La spécificité de l'approche IDM par rapport aux autres méthodes de modélisation est de privilégier des modèles automatisables et opérationnalisables. Pour cela il faut disposer d'un cadre formel de modélisation. Dans ce cadre formel que propose l'IDM, un modèle est conforme à son métamodèle si celui-ci s'exprime et exprime les relations qui le lie à ses modèles dans les termes définis par le méta méta modèle choisi. L'approche IDM est mise en œuvre par des plateformes de modélisation implémentant un méta méta modèle et permettant de définir des modeleurs spécifiques à un métamodèle. Parmi les différentes approches IDM, l'approche MDA, est à la fois un jeu de standards proposé par l'OMG et un guide structuré de modélisation. Le MDA propose une approche par concrétisations successives permettant de passer, par des transformations, d'une description indépendante des formats informatiques à une description informatique mais "métier", puis à une description informatique dépendante de la plateforme d'exécution visée, pour finir par la production du code de l'application. C'est

cette approche dont nous avons décidé de nous inspirer, tout en l'implémentant dans le cadre plus général de L'IDM. Appliquer une démarche IDM à la construction de dispositif pédagogique nécessite de donner un sens, pour ce domaine qui ne conduit pas à la production de code exécutable, à la relation de conformité et l'opérationnalisation de nos modèles. La relation de conformité peut être exprimée au sein de notre plateforme de modélisation reposant sur une implémentation du MOF. L'opérationnalisation de nos modèles révèlent deux verrous que nous abordons dans notre recherche :

Comment en EIAH abstraire une plateforme d'exécution (le système d'information représenté par la plateforme de formation) pour définir un métamodèle spécifique à la plateforme?
Quel sens donner à l'opérationnalisation d'un modèle?

Des solutions conceptuelles et techniques sont envisagées par notre recherche, elles passent par l'élaboration de bonnes pratiques et par la construction de générateurs spécifiques à chaque plateforme de formation. Le chapitre suivant présente différents travaux menés en EIAHs selon un point de vue IDM. Certains des auteurs cités ne situent pas leurs travaux par rapport à l'IDM, ce travail de présentation que nous faisons représente donc une interprétation de notre part.

chapitre 7 EIAH et IDM

Avec les travaux de Pierre Laforcade (Laforcade, 2004), décrit section 7.1.3, nos travaux sont les premiers en France à aborder explicitement certaines problématiques EIAHs par l'IDM. Cependant il est possible de reconsidérer une série de recherches, abordant les mêmes problématiques, selon un point de vue IDM. C'est ce que nous entreprenons dans ce chapitre, qui nous permet ainsi de situer notre approche "*Utiliser l'ingénierie dirigée par les modèles pour construire des dispositifs pédagogiques*" parmi d'autres approches que nous décrivons ci-dessous.

7.1 Différentes initiatives en EIAH, reformulées selon un point de vue IDM

7.1.1 Une méthode pour aider à la modélisation : Cepiah CEM.

Le projet Cepiah de l'Université de Compiègne est une méthode d'aide à la Conception et à l'Évaluation des Polycopiés Interactifs pour l'Apprentissage Humain (CEPIAH). Cette méthode a été implémentée sous la forme d'un guide interactif accessible sur le Web, composé de trois modules : Aide à la Conception, Aide à l'Évaluation, Modèles Prédéfinis de sites Web éducatif (Giacomini-Pacurar, Trigano, & Alupoiaie, 2006).

Il nous est possible de reformuler cette méthode dans un contexte IDM, celle-ci permet, dans le cadre d'un DSL, orienté vers la production de cours en ligne d'aider l'enseignant dans son processus de modélisation. Cette méthode est à comparer à une approche utilisant différents métamodèles liés par des règles de transformation et menant à une concrétisation progressive d'un dispositif pédagogique. L'originalité de la méthode Cepiah repose sur plusieurs points.

- Le métamodèle du DSL est déduit de l'analyse de 175 sites pédagogiques.
- Des modèles prédéfinis permettent d'aborder la modélisation d'un site par le biais de patron.
- Deux formulaires : l'un consacré aux concepts pédagogiques, l'autre à l'IHM permettent via leurs interprétations par un moteur d'inférence de produire un modèle de scénario respectant les deux métamodèle sous jacents.

L'ensemble de ces trois points suffit à rendre originale la démarche de modélisation. Le modèle de scénario est décrit selon deux métamodèles la première description repose sur une implémentation du métamodèle IMS - LD, dans le cadre du méta métamodèle XML. La seconde description concerne l'Interface Homme-machine du scénario. Un modèle de scénario est opérationnalisé sur la plateforme Net Université, qui est alors en charge de la composition des deux métamodèles et qui instancie un générateur de scénario à partir du modèle décrit par la méthode Cepiah (Giacomini-Pacurar et al., 2005).

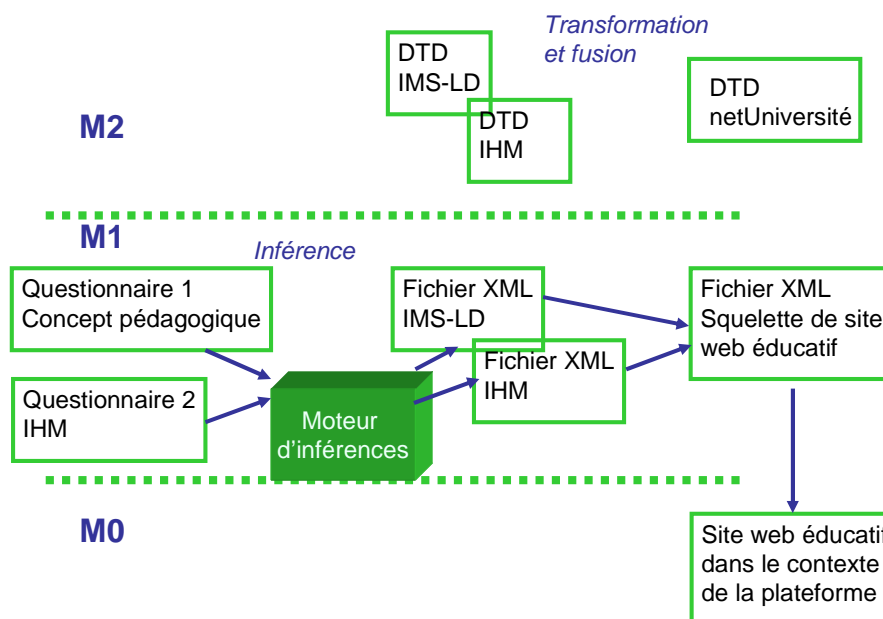


Figure 27 : point de vue IDM de la méthode Cepiah

Cette démarche pragmatique, a l'intérêt d'illustrer le processus d'opérationnalisation d'un modèle de scénario sur une plateforme de formation. En particulier elle identifie les éléments du modèle qui sont susceptibles d'être instanciés sous forme d'une collection d'éléments (étudiants, enseignants, activités associée), de ceux qui seront instanciés sous forme d'un élément unique mais partagé (outils ressources) (cf. Figure 27).

D'autres approches préconisent, pour la création de site éducatif, la séparation des aspects par la formalisation de métamodèles différents. Ainsi la méthode CEM (Courseware Engineering Methodology) permet de guider des auteurs novices dans la modélisation d'activités pédagogiques (Uden, 2003). Elle repose sur l'exploitation de quatre métamodèles, le métamodèle pédagogique renvoie au aspects pédagogique du cours, le modèle conceptuel gère les aspects fonctionnels impliqués par le cours, le métamodèle de séquençage traite de l'organisation des activités, le métamodèle IHM permet la présentation du cours au travers une interface Web. Un modeleur est proposé par la méthode et les modèles de cours sont directement opérationnalisés par le générateur CEM.

7.1.2 Le LDL

Le laboratoire SYSCOM de l'université de Savoie, a développé un langage de description spécifique pour décrire les activités collaboratives et pédagogiques le Learning Design Language (LDL) (Ferraris et al., 2005) (Martel et al., 2006).

Dans une perspective IDM, il nous est possible de reconsidérer ces travaux : le LDL peut ainsi être défini comme un langage de domaine centré enseignant. Ce langage s'appuie alors sur un métamodèle défini pour le "Computer Supported Cooperative Work", ce métamodèle repose sur un nombre minimal de concepts :

Interaction : Une interaction est définie par l'action réalisée par un acteur à destination d'un ou de plusieurs autres acteurs.

Enceinte : L'enceinte est le "lieu" dans lequel se déroule l'activité.

Position : Une position traduit le point de vue d'un acteur sur un autre acteur ou sur une de ses productions.

Rôle : Les rôles sont définis par les interactions que les acteurs ont entre eux.

Structure : Les structures décrivent l'enchaînement des interactions.

Observables : Ceux sont les valeurs sélectionnées par le scénario résultant de l'observation de ses différents composants.

Règles : elles permettent lors de leur application de vérifier la complétion des actions, l'état des positions au cours de l'activité, etc.

Notre collaboration avec le laboratoire SYSCOM a permis de définir avec notre plateforme ModX, le métamodèle LDL (cf. Figure 28).

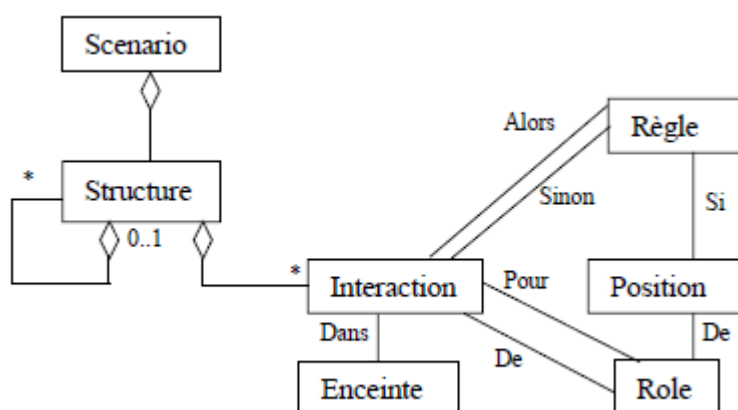


Figure 28 Vue simplifiée du métamodèle LDL

Construire le métamodèle LDL, à l'aide de la plateforme de modélisation (ModX), a permis d'associer au métamodèle, un modèle permettant l'édition de modèle de scénario LDL. Un modèle de scénario LDL est un fichier XML qui contient des informations relatives aux acteurs (ou rôles) du scénario (apprenant, enseignant, groupes etc.), aux ressources (artefacts, services, etc.) ainsi que la description des tâches à réaliser. L'opérationnalisation du modèle est effectuée par un paramétrage de l'infrastructure LDI (Learning Design Infrastructure) développée historiquement sur la plateforme Zope/Python. Cette infrastructure est capable d'exécuter le scénario; sur des plateformes compatibles LDI, elle transforme le scénario LDL en un scénario exécutable puis par l'intermédiaire du scénario, elle organise le contenu et l'appel aux différents services et fonctionnalités de la plateforme en fonction des interactions réalisées par les utilisateurs. L'infrastructure LDI est capable d'adresser différentes plateformes parmi lesquelles : le cartable électronique utilisé à l'université de Savoie et la plateforme Moodle.

L'environnement ainsi créé, permet de développer une chaîne éditoriale de scénarios pédagogiques et offre un cadre d'opérationnalisation à toute démarche IDM désirent produire des scénarios opérationnalisables sur les plateformes adressées par LDI. Ainsi est-il par exemple possible d'aborder la modélisation de situations interactionnelles (Vachet, 2006) et de relier cette modélisation au langage LDL. Le domaine considéré, (situations interactionnelles, mobilité, ressources d'interaction) a été décomposé en différents aspects, sous la forme de Langages Spécifique de Domaines (DSL) indépendants. Les métamodèles ainsi définis ont été composés par fusion avec le métamodèle LDL.

L'approche proposée par le laboratoire Syscom, repose sur le même constat de la nécessité de centrer la démarche de modélisation sur l'enseignant. Cependant elle se distingue de notre approche dans le sens où elle propose de modéliser des scénarios pédagogiques. Ce choix dont nous avons décrit les limites, impose de définir pour opérationnaliser les modèles de scénario, des interactions très fortes avec la plateforme d'exécution. Ces interactions sont rendues possibles par la réalisation de la LDI. Ce sont les interactions très fortes et d'assez bas niveau entre la LDI et l'API des plateformes d'exécutions, qui permettent alors l'opérationnalisation effective des scénarios.

7.1.3 UML et Profil CPM

Le LIUPPA (Laboratoire Informatique de l'Université de Pau et des Pays de l'Adour) a développé le langage de modélisation, Cooperative Problem-based learning Meta-model (CPM). Le langage permet la modélisation amont de situations problèmes coopératives (Laforcade, 2004), (Laforcade, 2005), (Laforcade et al., 2005), (Nodenot, 2006b) (cf. Figure 29). Le métamodèle utilisé pour exprimer de telles situations doit permettre l'expression :

- De la **structuration hiérarchique** des activités d'une PBL coopérative.
- Des éléments utilisés pour **décrire l'objet de la PBL**, c'est-à-dire les objectifs pédagogiques mais également tous les autres aspects pédagogiques des PBL (obstacles, ressources, contraintes, pré requis, connaissances, etc.).
- Des éléments permettant de décrire les **aspects sociaux des PBL** (rôles, activités collaboratives, restrictions, droits, différents modes de coopération, etc.).

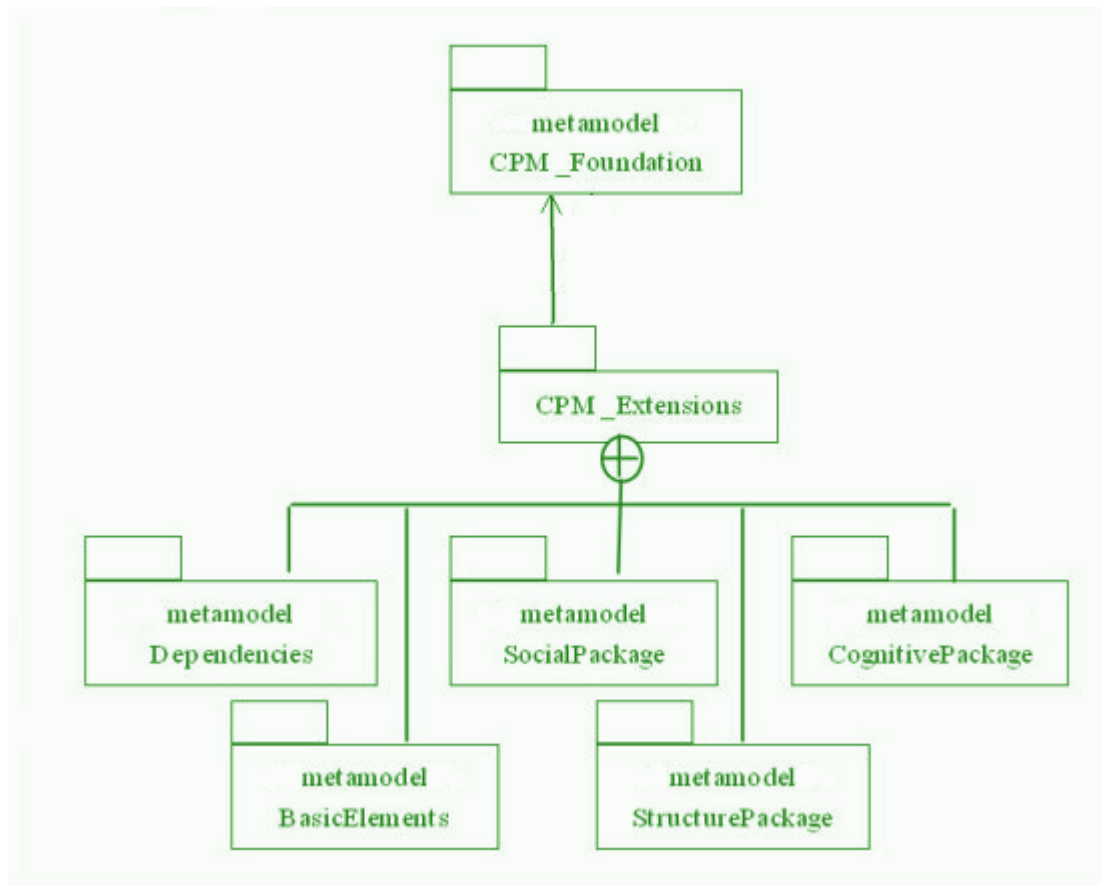


Figure 29 Paquetages principaux du langage CPM, tiré de (Laforcade, 2004)

Le langage CPM est une proposition de langage basé sur UML, Le métamodèle CPM est défini par un profil spécifique UML. Ce profil comprend :

Une syntaxe abstraite : représentée par le métamodèle CPM qui définit les concepts et leurs relations ;

Une syntaxe concrète : représentée par le profil CPM qui définit la notation des concepts et de leurs relations ainsi que leur utilisation dans les diagrammes UML ;

Une sémantique : celle-ci est définie au niveau de la terminologie dans le métamodèle (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).

Le profil CPM définit des stéréotypes, c'est-à-dire que pour une méta classe spécifique du métamodèle d'UML, le stéréotype crée une nouvelle méta classe basée sur la méta classe UML existante. Il définit également des valeurs marquées, ce qui permet l'attachement arbitraire d'informations à une instance. Enfin le profil permet de personnaliser via la définition d'icônes la notation UML. Un modelleur reposant sur ce profil a été créé via l'outil Objecteering Profile Builder.

Il nous est possible de reconsidérer les travaux de Pierre Laforcade selon une approche MDA. Avec cette perspective, (cf. Figure 30), le modèle CPM permet de passer d'une description CIM des situations problèmes à une description PIM de ces situations. En effet le métamodèle CPM n'est pas destiné à être opérationnalisé, mais le cadre formel qu'il propose lui permet de définir des modèles automatisables. Ceux-ci peuvent par transformation être opérationnalisés via d'autres métamodèles plus spécifiques aux plateformes d'exécution. La Figure 30 illustre selon notre point de vue, le placement de CPM dans une approche IDM des EIAHs.

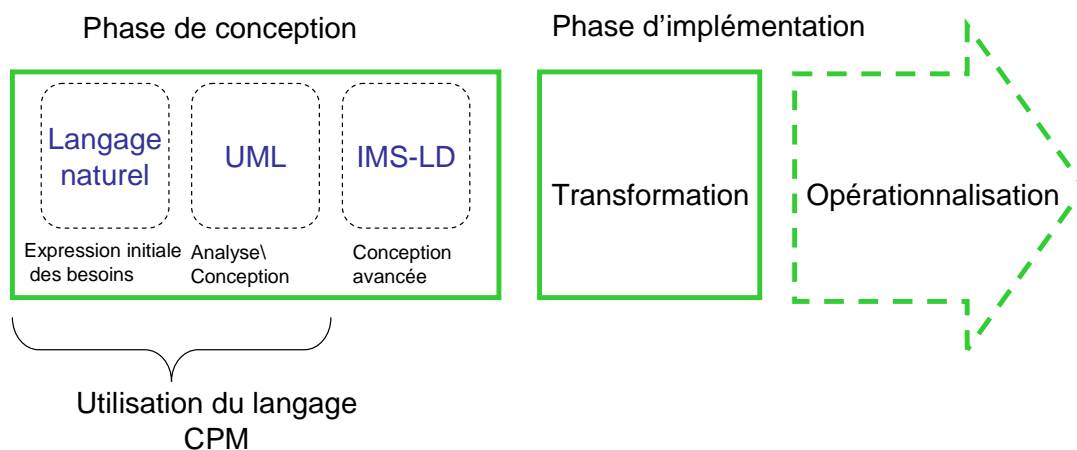


Figure 30 Position de CPM dans une vision IDM des EIAHs

Il se situe par exemple en amont d'une modélisation, de type IMS-LD, l'opérationnalisation du modèle transformé devant être réalisée dans le cadre de ce métamodèle.

L'approche est intéressante, car elle aborde la modélisation de situations dites "constructivistes". Les expérimentations menées ont montré l'expressivité du langage. Cependant l'approche envisagée ne permet pas l'opérationnalisation de scénarios produits et le choix d'utiliser un profil UML n'a pas permis de produire un DSL complet puisque le modelleur ne peut être utilisé sans l'outil Objecteering Profile Builder.

7.1.4 Outils de modélisation reposant sur un métamodèle IMS-LD

Différents modelleurs proposent de produire des modèles plus ou moins conformes au métamodèle IMS-LD. Nous avons présenté dans les chapitres précédents

- L'utilisation de la méthode Misa et du modelleur MOT+ pour générer du modèle de scénarios pédagogiques conformes au métamodèle IMS-LD, (Paquette et al., 2002), (De la Teja et al., 2006).
- L'utilisation d'ExploraGraph pour les définir au sein de l'environnement Explora (Villiot-Leclercq, 2006).
- Dans le cadre du Cepiah un modelleur permet également de produire des modèles de scénario conformes au métamodèle IMS-LD (Giacomini-Pacurar et al., 2006).
- LAMS (Learning Activity Management System), est un environnement de modélisation open source qui est inspiré par le métamodèle IMS-LD (Dalziel, 2005). Le système propose un modelleur graphique permettant l'édition de scénario pédagogique, un générateur permettant l'opérationnalisation de ces scénarios sur la plateforme LAMS. L'opérationnalisation n'est pas dépendante de la plateforme car l'environnement LAMS peut être utilisé avec de nombreuses autres plateformes.

Ces quelques exemples illustrent la diversité des approches adressant le métamodèle IMS-LD. Toutes font reposer le support de conformité des modèles produits sur la conformité XML à la DTD IMS-LD.

L'ensemble de ces outils de modélisations utilise une interprétation ad hoc du métamodèle IMS-LD, mais ne se place pas dans la perspective d'une approche IDM de ce métamodèle. Des méthodes (cognitive, empirique ou à base de pattern) permettent l'appropriation du métamodèle, un générateur opérationnalise les modèles sur une plateforme, propriétaire ou au sein d'un environnement que l'on peut attacher à une plateforme.

L'originalité de notre approche est de proposer une chaîne automatisable permettant de passer d'un métamodèle proche des intentions pédagogiques enseignantes, à une plateforme de formation, en passant éventuellement par un métamodèle pivot IMS-LD.

7.1.5 Ré engineering et Open Distributed Processing Reference Model

Le Laboratoire d'Informatique de l'Université du Maine (LIUM) a mené des recherches sur l'amélioration du modèle de conception d'une application d'apprentissage à partir de l'observation des parcours effectués par les étudiants (Corbière & Choquet, 2004), (Corbière & Choquet, 2005). La démarche choisie est "Open Distributed Processing Reference Model" (ODP-RM) (Farooqui, Logrippo, & Meer, 1996) cette démarche permet de supporter le processus de modélisation d'un système complexe et distribué en demandant aux concepteurs d'instancier sur leur domaine un ensemble de concepts génériques (composition / décomposition d'objets, état et comportement d'un objet, points de vue et correspondance entre points de vue...). Ces concepts sont déclinés par rapport à trois actes de modélisation :

- La **spécification** du système, où les concepteurs classifient les objets du système par des liens de composition.
- La **modélisation** du système qui définit, à différents niveaux d'abstraction, les modèles d'interaction entre objets.
- La **structuration** du système où les différentes structures à implémenter dans le système sont définies.

La démarche ODP-RM permet de séparer les activités de modélisation selon des points de vue aussi différents que métier, informationnel, computationnel, ingénierie et technologique. Ces points de vue permettent une appréhension plus facile d'un système complexe (cf. Figure 31).

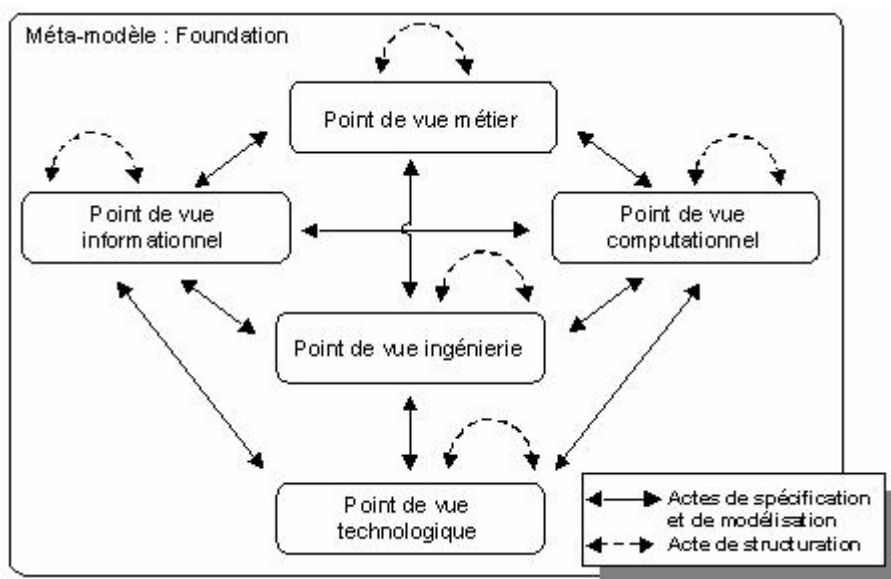


Figure 31 Différents points de vue d'une activité de modélisation tiré de (Corbiere & Choquet, 2005)

La démarche ODP-RM appliquée aux différents standards des EIAH, a pu montrer leurs complémentarités mais aussi leurs insuffisances, Ainsi IMS-LD se situe clairement selon un point de vue **informationnel**, le standard SCORM décrit le point de vue **ingénierie**, le point de vue **computationnel** est par exemple illustré par des documents comme le "Best Practice" de IMS-LD. Cependant appliquer la démarche d'ODP-RM aux EIAHs montre l'absence d'une réflexion sur la définition de métamodèle permettant de décrire le cycle de vie des systèmes de formation, les langages employés dans chaque point de vue, et les interrelations entre les activités de chaque acteur du processus.

Pour mettre en œuvre des activités de réingénierie et de rétro conception en EIAH dans le cadre de la démarche ODP-RM, Alain Corbiere a dû définir pour les EIAH les points de vue **métier** et **technologique** (Corbière, 2006). Cette mise en œuvre dans un contexte IDM a consisté à utiliser le profil UML ECA (Enterprise Collaboration Architecture) de l'OMG (ECA). La démarche ODP-RM permet alors d'explicitier les mécanismes de transformation sur des modèles ayant pour origine les différents métamodèles correspondant aux points de vue impliqués par la démarche. La démarche permet de fédérer ces différents métamodèles.

L'intérêt de la démarche d'Alain Corbière est de considérer tous les points de vue que peuvent prendre la réalisation d'une activité pédagogique sur un EIAH, et de ne pas se limiter au simple scénario, au dispositif ou aux ressources. Pour la réalisation d'une telle activité, la démarche qu'il suit permet non seulement d'établir des modèles descriptifs mais d'explicitier les relations entre ces différents modèles. Cette démarche nécessite comme nous le montrons dans notre recherche, d'étudier d'un point de vue génie logiciel, les relations entre une activité pédagogique prévue et l'application qui le supportera. Ce travail qui à notre sens permet réellement de juger de l'intérêt de la démarche n'a pas été abordé dans le cadre d'une application d'ODP-RM aux EIAHs.

7.1.6 Conclusion

Notre approche est basée sur l'IDM, elle constitue un cadre conceptuel dans lequel les différentes approches que nous venons de présenter peuvent s'inscrire même si elles ne le revendiquent pas. C'est-à-dire que les différentes méthodes décrites dans les chapitres précédents, approche cognitive (Misa), approche à base d'inférence (Cepiah), etc., sont envisageables dans le cadre de notre approche pour peu que l'on définisse avec soin le métamodèle dont elles s'inspirent. Décrire de tels métamodèles peut permettre à terme l'interopérabilité de ces méthodes en explicitant les règles de transformation qu'elles induisent. Exprimer ces méthodes dans le cadre formel de l'IDM peut autoriser une automatisation et une opérationnalisation accrue des modèles produits. Notre approche que nous allons maintenant décrire exploite ce cadre formel, et en constitue un exemple d'exploitation. Elle se démarque, des méthodes que nous venons de décrire, par son positionnement qu'elle revendique clairement : l'application des concepts de L'IDM aux EIAHs; et par les éléments qu'elle entend construire : des dispositifs pédagogiques adaptés à des petites structures pédagogiques, sur des plateformes de formations.

7.2 Une première conclusion, notre solution au regard de la modélisation, introduction à nos travaux de développement.

7.2.1 Positionnement de notre recherche en EIAH, une première synthèse

Notre problématique en EIAH concerne l'aide que nous pouvons apporter à des petites équipes d'enseignants pour construire un enseignement de type médiationnel sur des plateformes de formation. Cette problématique résulte des recherches que nous avons menées dans cette première partie :

Une première recherche s'est attachée à définir les différents sens que peuvent prendre une plateforme de formation et les différents rôles que des intervenants peuvent y jouer, cette recherche nous a permis de préciser notre approche centrée sur l'enseignant parmi d'autres approches. Elle a permis aussi de définir les différentes fonctionnalités d'une plateforme de formation ainsi que le cadre social dans lequel nous envisageons son utilisation.

Une deuxième recherche a ensuite tenté de définir une caractérisation des usages enseignants d'une plateforme de formation. Cette caractérisation nous a permis de présenter le cadre dans lequel nous nous situons, il s'agit de petites organisations enseignantes pratiquant du "bricolage pédagogique" sur des plateformes de formations souvent modulaire et open source ou détournée à partir d'application de type Web 2.0.

Nous avons ensuite établi un tableau de correspondance entre les fonctionnalités des plateformes et les différents modèles pédagogiques. Avec ce tableau nous avons pu situer plus précisément notre recherche autour de l'activité de construction de cours, de dispositif ou de scénario.

Cette approche par construction peut être documentaire, cognitive ou par activité, ces trois approches offrent la possibilité à des enseignants de construire des entités très différentes. Ce sont ces différentes entités que nous avons alors voulu caractériser par rapport aux différents standards des EIAHs, LOM, SCORM et IMS-LD et par rapport aux modèles pédagogiques. Il

est alors apparu que pour mener un enseignement constructiviste ou socio constructiviste, ces différentes approches, et les standards qu'elles supportent ne sont pas satisfaisants. En particulier l'approche par scénario, ne correspond pas bien à une pratique enseignante, elle ne permet pas l'improvisation pédagogique et la modélisation d'activités situées.

Notre avons alors présenté notre approche destinée à favoriser la construction de dispositifs pédagogique. Un dispositif correspond à un cadre pré-structuré de travail et de pensée. Nous avons explicité les différents sens que le mot dispositif peut prendre, c'est une offre de sens constituant un cadre de pensée, c'est un objet frontière et enfin c'est un dispositif technique dans le contexte d'une plateforme de formation. Coordonner ces différents sens peut se faire dans le cadre d'un processus de concrétisation. Or ce processus est proche de ce qui ce fait en modélisation et en particulier en MDA.

7.2.2 L'ingénierie dirigée par les modèles

Dans un premier temps nous avons alors décrit les différents concepts sur lesquels repose l'ingénierie dirigée par les modèles et nous avons présenté quelques plateformes de modélisation. Notre plateforme, ModX, repose sur une architecture quatre niveaux, implémentant pour le niveau supérieur le méta méta modèle MOF. C'est avec cette plateforme que nous définissons des modeleurs spécifiques pour encadrer les activités de modélisation que nous menons en EIAH.

Nous avons présenté ensuite l'approche MDA. Cette approche permet de définir à partir d'un modèle exprimé en langage naturel, un modèle spécifique automatisable et opérationnalisable. Le processus de concrétisation qu'elle implique est mené entre différents modèles, il est instrumenté par des transformations dont les règles sont définies entre les métamodèles. Cependant la difficulté pour des enseignants à manipuler le standard UML, nous pousse à redéfinir l'approche MDA dans le cadre plus général de l'IDM, au travers une série de langage spécifique de domaine (DSL). Ces différents DSL peuvent être coordonnés au sein de notre plateforme ModX.

Cette plateforme permet la définition d'un ou de plusieurs métamodèles métier, sur lesquels repose la construction des DSL que nous utilisons. Au sein de ces DSL, opérationnaliser un modèle de dispositif dans le contexte d'une plateforme de formation constitue un enjeu véritable dans la mesure où il entraîne la définition d'une relation de conformité et une opérationnalisation propre au domaine envisagé. Nous proposons de mener cette opérationnalisation à l'aide de la création d'un générateur de constructeurs spécifiques. L'outil que nous avons élaboré GenDep est décrit dans le chapitre 9 et le chapitre 10. Nous verrons dans ces chapitres qu'il est possible de construire des constructeurs spécifiques à partir du métamodèle décrivant une plateforme de formation. Cette solution envisagée pour la construction de machine virtuelle peut ici être étendue à la construction d'un générateur.

7.2.3 Les différentes problématiques abordées par nos travaux

Notre problématique affinée au regard de cet état de l'art, nous conduit à vouloir instrumenter la construction de dispositifs pédagogiques en utilisant une approche IDM. Une telle approche, reposant sur un processus de concrétisation soulève l'interrogation suivante :

L'approche de concrétisation MDA appliquée à des dispositifs pédagogiques, est elle proche ou au moins compatible avec la manière créer enseignante?

Dans un cadre qui ne conduit pas à la production de code sur une plateforme d'exécution parfaitement définie, la démarche IDM que nous souhaitons mettre en place, implique quelques interrogations qu'il nous faut résoudre :

Comment définir le métamodèle correspondant à chaque plateforme d'exécution lorsqu'il est inexistant ? En quoi consistent les interactions avec la plateforme ?

Sur le plan technique et conceptuel, les solutions de métamodélisation de transformation et d'opérationnalisation que nous abordons soulèvent l'interrogation suivante :

Notre approche peut elle être instrumentalisée?

Enfin notre approche est illustrée par trois études de cas que nous avons menées, semble répondre favorablement à la question suivante.

Notre approche peut elle être expérimentée?

Ces quatre interrogations seront respectivement traitées dans le chapitre 8, le chapitre 9, le chapitre 10, et le chapitre 11.

chapitre 8 Processus de Bricolage et création participative pour une approche par construction de dispositif

8.1 Introduction

Construire des dispositifs pédagogiques automatisables et opérationnalisables, nécessite de devoir concilier deux cycles de production différents, un cycle d'ingénierie logicielle permettant de créer l'environnement informatique support de l'apprentissage, un cycle de création pédagogique permettant la réalisation de dispositifs pédagogiques. Concilier ces deux cycles en y impliquant les enseignants peut se faire au travers de deux approches :

- La première consiste à s'interroger, indépendamment du domaine traité, sur la possibilité d'introduire de la conception participative dans un cycle de production. C'est ce que nous faisons dans la première partie de ce chapitre.
- La seconde, plus spécifique au domaine enseignant, consiste à étudier un mode de création très particulier le "*bricolage*", et les implications de ce mode de création sur les outils à utiliser et les objets à produire. Un tel mode de création permet de mettre en place un processus de conception participative adapté au monde enseignant.

Dans la seconde partie de ce chapitre, nous proposons d'outiller cette dernière approche en mettant en œuvre une ingénierie de type IDM. L'équipe NOCE du laboratoire Trigone ainsi que le laboratoire LIFL auquel notre équipe va maintenant être rattachée possède en effet des compétences et une expérience acquise en modélisation et en transformation de modèle. Outre ces compétences que nous désirons valoriser dans cette thèse, le recours à l'IDM est justifié par la similitude entre les problèmes à traiter. Il s'agit de concilier 2 cycles de production : cycles d'ingénierie logicielle et de création pédagogique d'un côté, cycles de métamodélisation et modélisation de l'autre. Nous allons montrer que les concepts de l'ingénierie dirigée par les modèles et plus particulièrement les concepts des langages spécifiques de domaines permettent de produire et manipuler des métamodèles et des modèles facilement appropriables. Pour notre ingénierie des dispositifs, un modèle de dispositif peut tenir le rôle d'objet frontière pour accompagner les activités de création des différents acteurs impliqués. La notion d'objet frontière au cœur des travaux de Susan Star (Star, 1989), sera abordée dans la section suivante.

Dans les chapitres précédents nous avons expliqué que nous considérons les plateformes, et les éléments à construire sur celles-ci, selon une approche centrée sur les enseignants dans le cadre de petites communautés éducatives utilisant des plateformes de formation classiques, modulaires ou simple application de type Web 2.0 pour y pratiquer du bricolage pédagogique. Dans le cadre de cette approche, nous suggérons qu'il y a trois activités fortement dépendantes : 1) outiller les enseignants, 2) utiliser cet outillage pour définir et construire un dispositif, 3) utiliser ce dispositif au sein d'une plateforme de formation. Nous suggérons aussi qu'il est possible de les mener en impliquant fortement les enseignants. Nous pensons que l'usage de dispositifs pédagogiques encourage, par leur déterminisme technique et leur incomplétude scénaristique, une finalisation dans l'usage des dispositifs et des outils qui ont permis de les créer.

Faire participer les usagers, ici les enseignants, à la construction des outils et des dispositifs qu'ils manipulent est un des points qui différencient notre approche des différentes méthodes d'ingénierie que nous avons listées dans le chapitre 4.

Nous désirons dans la section suivante rapprocher notre approche de la notion de conception participative. Réintroduire l'utilisateur, au sein d'un processus de conception fait en effet partie d'une problématique plus générale qui dépasse le champ de l'informatique et de l'enseignement, c'est cette problématique que nous présentons maintenant.

8.2 Réintroduire l'utilisateur dans un processus industriel de création, place de la modélisation et recherche d'objets frontières.

8.2.1 Usage et création participative

Nous tenons dans cette section à expliciter les rapports qui lient la création d'un objet à son usage. La recherche que nous présentons dans ce chapitre a été en partie élaborée dans le cadre de l'équipe de recherche technologique éducation (ERTé) Apprentissages en Réseaux auquel notre laboratoire a participé, et du projet de Pratique Collective Distribuée d'Apprentissage sur Internet (PCDAI). De ce dernier nous avons extrait une étude de cas menée pendant deux ans et relatée dans le chapitre 11.

En EIAH, le paradigme du "Share and Reuse", repose par exemple sur un modèle économique de style workflow (voir les exemples portés par TELOS (Paquette, 2005), SCENARI (Bachimont et al., 2002) etc....), minimisant les modifications apportées par l'usage d'un objet. Selon ce paradigme, des objets pédagogiques, ou des séquences d'enseignements, sont conçus par des enseignants, ou mieux, des ingénieurs pédagogiques, et sont mis à disposition de formateurs pour construire des cours. Un tel paradigme suppose la non altération des objets primaires, et ceci dans le but de minimiser les différentes versions d'un même objet. Or cette idée qu'il est possible de concevoir des objets utilisables mais inaltérables semble occulter chez les usagers le pouvoir de création qu'ils exercent par leur usage. Les EIAHs comme tous les autres domaines, sont sensibles aux détournements d'objets et d'applications. Cette activité explique d'ailleurs partiellement les processus d'appropriation individuels et collectifs.

"A récuser la consommation telle qu'elle a été conçue et (naturellement) confirmée par ces entreprises d'auteur, on se donne la chance de découvrir une activité créatrice là où elle a été déniée, et de relativiser l'exorbitante prétention qu'a une production (réelle mais particulière) de faire l'histoire en informant l'ensemble du pays" (Certeau, Giard, & Mayol, 1980).

La sociologie de l'usage définit la relation d'usage comme étant la relation qui lie l'individu à l'objet, elle considère que tout comme l'analyse d'un objet est située par rapport à l'individu qui l'utilise, l'analyse d'un individu est également située par rapport à l'objet qu'il manipule. Cette approche s'oppose d'une part à des approches d'ergonomie cognitive (étudier l'utilité d'un objet créé par rapport au profil cognitif d'un utilisateur abstrait) d'autre part à une approche ergonomique (étudier l'utilisabilité d'un objet créé) (Mallein & Toussaint, 1994). La sociologie de l'usage prône l'idée qu'on ne peut concevoir un objet sans étudier la relation d'usage qui le lie à ses utilisateurs, comme nous le rappelions précédemment, elle explique que les utilisateurs interviennent tout autant que les créateurs initiaux dans le processus de création d'un objet. Non seulement les utilisateurs achèvent la conception de l'interface dans

l'usage qu'ils en ont (Vicente & Rasmussen, 1992), mais plus généralement ils sont à l'initiative même de toute innovation technologique.

"L'innovation technique est fondamentalement sociale, et l'imaginaire est bien une composante centrale du développement des techniques" (Flichy, 1995)

D'où l'idée de faire directement intervenir l'utilisateur dans la phase même de conception d'un produit. Développé en Suède à partir des années 70, la conception participative a reposé sur un cadre législatif donnant le droit aux employés de participer aux décisions concernant leur milieu de travail. Dans les années 80, elle s'est appuyée sur le recueil du savoir des usagers (Granath, Lindah, & Rehal, 1996). Cette approche a néanmoins vite montrée des limites en établissant des rôles trop différenciés entre les différents acteurs et en n'offrant pas une véritable égalité d'expression et de pouvoir entre les différents intervenants. Depuis les années 90, dans un processus de conception collective, les acteurs sont considérés comme des experts. Idéalement, dans un tel processus, le travail de conception est pluridisciplinaire et pluri-hierarchique (Caelen & Jambon, 2004). La création participative (Schuler & Namioka, 1993) peut intervenir dans différentes activités humaines par exemple en architecture (Towers, 1995) ou en génie logiciel (O'Coill & Doughty, 2004), (Détienne, 2006). Mettre en place un processus de création participative consiste à s'assurer que les utilisateurs finaux seront impliqués dans les phases de conception et de décision menant à la création de l'objet envisagé.

8.2.2 Objets frontières : modèles

En informatique, un des écueils posé par la mise en place d'un tel processus en ingénierie est qu'il passe par une série de réifications sur des artefacts secondaires au sens qu'en donne Wartofsky (Wartofsky, 1973) Celui-ci définit trois types d'artefacts, les artefacts primaires sont utilisés dans la vie de tous les jours, les artefacts secondaires sont utilisés pour décrire des processus et procédures permettant par exemple la transmission de connaissances, les artefacts tertiaires sont utilisés pour diagnostiquer ou expliquer les propriétés et les comportements des objets. En enseignement, une telle classification conduit par exemple à classer les artefacts, utilisés pour enseigner les mathématiques, comme artefact secondaire, les mathématiques : les règles, les théorèmes et les axiomes constituent des artefacts tertiaires. "Mathematical theories are examples of tertiary artifacts, organizing the models constructed as secondary artifacts." (Bartolini Bussi, Mariotti, & Ferri, 2005). Des modèles de dispositifs pédagogiques peuvent ainsi être situés parmi les artefacts secondaires, les métamodèles auxquels ils sont conformes tenant lieu d'artefacts tertiaires.

Idéalement l'utilisation de modèle (artefact secondaire) peut permettre une participation de chaque intervenant (ingénieur, usager, designer) au sein d'un processus d'ingénierie. Les modèles et l'acte de modélisation jouent alors le rôle d'objet frontière, c'est-à-dire étant perçu et abordé par tous mais compris différemment (Star, 1989). Un objet frontière est alors un objet qui permet de coordonner les perspectives des différentes composantes d'une communauté. Ce lieu d'échange peut ainsi permettre de mettre en place un processus de conception participative.

Cependant l'expérience de la modélisation a montré que de tels artefacts, souvent trop tôt techniques, ne permettaient pas une réelle participation des usagers (Schank & Hamel, 2004). Les principales critiques formulées à leur encontre portent sur :

- La participation finalement limitée qu'ils engendrent;
- Les difficultés pour éditer et s'approprier les modèles;
- L'ambiguïté qu'ils véhiculent, s'ils ne sont pas assez situés;

- Le parasitage engendré par l'apprentissage nécessaire de leur notation.

L'utilisation de la notation UML nécessite par exemple un apprentissage conséquent et est difficile à maîtriser pour un non spécialiste (Tilley & Huang, 2003).

8.2.3 Objets frontières : prototypes

Nous venons de voir, que l'utilisation de modèles, peut poser des difficultés pour mettre en place un processus de création participative. Dans la mesure où ils ne sont pas partagés par l'ensemble de la communauté qui participe à la création, ces artefacts ne jouent pas le rôle d'objets frontières. De tels objets sont par exemple en informatique plutôt à chercher du côté des prototypes, ceux-ci jouant alors un rôle similaire aux scénarios et aux maquettes.

"Une difficulté majeure de la phase de spécification provient du fait que les différents partenaires ne parlent généralement pas le même langage :

** le client s'exprime dans le langage du domaine de l'application, un jargon qui fait souvent appel à des termes techniques très spécialisés, mais qui utilise pour support le langage naturel (le français, l'anglais). De plus, le client ne sait pas toujours précisément ce qu'il veut : sa demande peut varier en fonction de l'offre. De fait, les désirs du client sont généralement ambigus et incomplets.*

** l'informaticien fait des petits schémas, utilise des langages formels pour représenter sa perception du problème. Le client non informaticien n'est généralement pas capable de comprendre la spécification résultante, et ne peut donc la valider en connaissance de cause.*

Une solution, pour vérifier la conformité de la spécification avec les besoins du client, est de construire rapidement un prototype de l'application." (Solnon, 1997)

L'utilisation de modèles pose également le problème de la validation de ceux-ci, cette validation peut être effectuée dans le cadre d'action de prototypage. *"Formal models are useful for developing high quality software, especially at the early design stage. An important topic of using formal models is the validation of model specifications. One way to do this task is prototyping, which can also be used in other activities of software development."* (Chen & Dong, 2002) Les auteurs proposent ainsi l'élaboration d'un langage formel permettant de lier modèles et prototypes.

Pour considérer les prototypes comme des objets frontières, permettant de supporter les activités de création participative, il nous faut nous interroger sur les éléments dont ils facilitent réellement l'appréhension par les utilisateurs finaux. S'agit-il du processus de création ou bien plutôt du produit final ? De tels objets frontières donnent en effet une place à l'utilisateur dans la **définition du produit à créer** et non dans la **définition du processus de création** qui mène à terme à sa production. C'est ce que relève Olav Bertelsen :

"the 'shared understanding' constructed during, (a rapid prototyping session) in general only exists in a form crystallised into the prototype; the prototype in turn being a boundary object tying the incommensurable praxises of designers and users together, allowing them to design together but still perceive the situation and the new artefact in different ways." (Bertelsen, 1997)

Mettre en place un processus de conception participative, nécessite d'identifier les objets frontières utilisés pour supporter les interactions et la collaboration entre les différentes communautés de pratique (Wenger, 1998). Or si les modèles peuvent sembler a priori adaptés à une telle tâche, nous avons montré les limites posées par leur utilisation. En ingénierie logicielle, les prototypes sont souvent le support pour mettre en place un processus de création participative. Cependant leur utilisation nous invite à porter notre attention sur le processus de création des objets que nous souhaitons créer. C'est pourquoi dans les sections suivantes nous tentons de caractériser le processus de création enseignant.

8.3 Processus de création enseignante

Dans cette section nous établissons un rapprochement entre la création enseignante et la notion de bricolage.

Le processus de création en EIAH se distingue des autres processus de création dans la mesure où pour un enseignant participer au processus de création d'un objet pédagogique, d'une séquence, est un acte commun qu'il réalise pour s'en faciliter l'usage. Dans le domaine plus général de la création enseignante, de nombreux auteurs ont ainsi stigmatisé l'individualisme des enseignants et leur tendance à réinventer personnellement ce que d'autres ont probablement déjà expérimenté ailleurs (Huberman, 1980).

"Ce serait très mal comprendre la logique de la pratique enseignante que de combattre de front cette tendance (...), la réinvention d'activités et du matériel correspondant est un élément important dans l'enrichissement et l'appropriation personnelle du rôle professionnel" (Perrenoud, 1983)

Pour des enseignants, participer au processus permettant la production de produit pédagogique est fondamental. Mais le fait que ce processus mène à la production d'un produit fini réutilisable et stockable n'est que subsidiaire. L'enseignant est conscient du caractère incomplet et perfectible des objets qu'il produit, d'où l'importance pour lui de participer à la création et à la recréation de ses activités pédagogiques. Cette caractéristique enseignante se reflète également en EIAH, nous l'avons abordée lorsque, dans le chapitre 4, nous avons décrit les limites de l'approche documentaire, en relatant les différentes enquêtes menées et le peu de succès rencontré par les dépôts d'objets pédagogiques. Le caractère perfectible des objets produits a été abordé dans chapitre 4, plus précisément dans les sections présentant l'approche par scénario.

Ce désir d'être des acteurs majeurs du processus de création trouve son écho dans les récents succès des Blogs, Wikis, CMS, e-Portfolios, Forums utilisés par des enseignants comme plateforme légère de formation. Ces usages particuliers sont autant la preuve de l'opportunisme des enseignants qui s'emparent d'une technologie pour mieux la détourner, que du désir d'investir des outils permettant une pratique plus créative de leur métier.

Nous venons de le voir, la création enseignante possède un cycle et une finalité qui lui est propre. Cette spécificité entraîne en enseignement une redéfinition nécessaire des objets frontières participant au processus de création. Pour un enseignant la démarche de création pédagogique ne correspond pas à un cycle sans fin de recréation. Dans la section suivante nous confrontons la démarche des enseignants à la notion de bricolage. Il est en effet possible de définir quelques pointeurs scientifiques qui permettent d'encadrer cette notion de Bricolage scientifique en EIAH et plus généralement en enseignement.

8.4 Bricolage et Enseignement

Nous étudions dans cette section, les rapports que l'enseignement entretient avec la notion de bricolage.

La notion de bricolage a été popularisée par Claude Levi Strauss dans "la pensée sauvage" :

"Le bricoleur est apte à exécuter un grand nombre de tâches diversifiées ; mais, à la différence de l'ingénieur, il ne subordonne pas chacune d'entre elles à l'obtention de matières premières et d'outils, conçus et procurés à la mesure de son projet : son univers instrumental est clos, et la règle de son jeu est de toujours s'arranger avec les 'moyens du bord'" (Lévi-Strauss, 1962).

Pour un enseignant, bricoler consiste à parvenir à ses fins avec les moyens du bord. *"Le bricolage ne se définit pas par son produit, mais par le mode de production : faire avec les moyens du bord, réutiliser, au besoin en les détournant de leur destination première, des textes, des situations, des matériaux."* (Perrenoud, 1983). Cette remarque est à rapprocher en EIAH des pratiques de détournement constatées sur des applications Web que nous rappelions dans la section précédente.

Cependant ce n'est pas la nécessité économique qui pousse au bricolage, mais la part de création qu'il permet (Perrenoud, 1983). En EIAH, l'utilisation de portail léger ou de CMS modulaire peut être ainsi expliqué par deux motivations complémentaires :

- La première est d'utiliser une application simple à maîtriser, pour laquelle un enseignant est capable de développer un sentiment d'appropriation.
- La seconde est le désir d'utiliser une application suffisamment ouverte, pouvant être détournée, pour être adaptée aux activités que l'enseignant veut mettre en œuvre.

En effet la principale raison qui motive l'enseignant à recourir au bricolage est qu'il a conscience de la nécessaire adaptation qu'il devra pratiquer sur les dispositifs qu'il prévoit (Blandin, 1990). Il est à la recherche d'une voie moyenne " *entre l'artisan et l'ingénieur* " (Simondon, 1989). Le bricolage est pour l'enseignant un compromis s'inscrivant dans le provisoire lui permettant de construire et déconstruire les dispositifs qu'il avait prévus.

*"Il est donc indispensable dans ce cas de quitter un idéal de maîtrise (concernant un ensemble de petites choses réalisées à la perfection, avec pour dessein l'apprentissage parfait), et d'accepter de se lancer dans ce qui peut paraître "l'aventure", en mettant néanmoins toutes les chances de son côté."
(Audran, 2005)*

La notion de bricolage est donc particulièrement adaptée à la description de l'activité enseignante caractérisée par le désir de maîtriser et d'adapter son enseignement.

Le bricolage est non seulement un moyen utilisé par les enseignants pour créer et recréer ses activités, mais il est aussi une manière de ne pas tout ordonnancer et de laisser aux usagers finaux (les élèves) un espace d'intervention : *"La question qui taraude le pédagogue en permanence est donc de laisser à l'autre la place qui lui revient pour qu'il construise son propre savoir. Pour cela, le pédagogue en est réduit au bricolage"* (Merrieu, 1999). D'autres auteurs ont eux aussi souligné cette façon de procéder qui permet de laisser les élèves libres de construire leur propre représentation mentale. La référence au Bricolage a ainsi été utilisée pour décrire une méthode exploratoire de la programmation en logo (Papert, 1991). Plusieurs approches pédagogiques dont le Bricolage ont été comparées. Ceci a permis de mettre en évidence les libertés qu'une telle pratique permettait en la rapprochant en enseignement de la théorie constructiviste (Ben Ari, 2001) et du constructionisme (Dougiamas & Taylor, 2002). Certains courants actuels, comme Learnativity (Learnativity), lient création à enseignement, présentant l'enseignement comme un processus continu de création dans lequel il s'agit d'impliquer les élèves.

En EIAH, la plateforme Moodle (cf. chapitre 11) illustre assez bien les divers aspects de la relation qui lie le bricolage et l'enseignement. Son interface intuitive facilite l'appropriation des enseignants : *"Moodle is an excellent example of software designed for bricolage. A naïve (or even techno phobic) instructor can start doing useful things in Moodle with five minutes of instruction"* (Berggren et al., 2005). La plateforme privilégie ainsi le bricolage enseignant dans la mesure où le matériel de cours peut être défini et redéfini de façon itérative. Il est prévu d'intégrer à la plateforme des outils offrant une ingénierie reposant sur la modélisation pour favoriser ce bricolage. Cette intégration a pour but de rendre compatible la plateforme

avec le standard IMS-LD (de fait la version 1.6 de Moodle est compatible avec l'éditeur LAMS). Techniquement rien n'empêche en effet un fichier XML IMS-LD d'être modifié dynamiquement à l'exécution (Berggren et al., 2005).

La plateforme Moodle illustre par ces caractéristiques le lien fort qui existe entre le bricolage qui sous-tend l'activité de création menée par les enseignants et le bricolage pratiqué par les élèves sur la plateforme de formation.

Dans cette section et la section précédente, nous avons cherché à établir une spécification de la création enseignante. Celle-ci, par son cycle et sa finalité, semble proche de la notion de bricolage. La notion de bricolage s'appuie sur la réutilisation et le détournement d'application ou de ressources, elle permet la maîtrise du processus de création des objets manipulés et réserve aux usagers finaux un espace d'intervention. Nous avons montré ainsi qu'il existe un lien entre la pratique d'une activité et les modalités de sa création. La Plateforme Moodle illustre en EIAH cette notion. Dans la section suivante nous souhaitons établir un lien entre conception participative et bricolage.

8.5 Bricolage et conception participative

Nous allons voir dans cette section les relations qui lient le bricolage à la conception participative.

8.5.1 Communication et pragmatisme

Dans "Les arts de faire", Michel Certeau met en lumière *"des formes subreptices que prend la créativité dispersée, tactique et bricoleuse"*, (Certeau et al., 1980). L'auteur montre comment cette pratique permet à l'utilisateur de participer lui aussi à la finalisation d'un objet. De nombreux auteurs ont ainsi montré comment la pratique du bricolage, par les détournements qu'elle encourage, constitue une opportunité pour l'utilisateur d'intervenir dans le processus de conception d'un objet. Des études ont montré également que la pratique du bricolage favorise la conception participative. Le bricolage *"Involves users, participatory designers and ethnographers in a continuing cycle of design and revised work practice."* (Monika Büscher, Gill, Mogensen, & Shapiro, 2001). Des recherches menées au sein de trois ateliers du projet WorkSPACE⁵ ont permis de mettre en relation les différentes manières de supporter la création participative (Monika Büscher, Mogensen, Agger Eriksen, & Friis Kristensen, 2004). Les trois ateliers de ce projet : le laboratoire du futur, l'expérience in situ de prototypage et l'atelier bricolage, mettent en œuvre différentes méthodes de conception. Ces méthodes sont complémentaires et permettent lorsqu'elles sont mises en œuvre conjointement de créer les conditions pour l'apparition et l'analyse de nouvelles pratiques de création. En particulier :

- Le bricolage ancre l'action de prototyper dans une démarche réelle, favorisant ainsi une meilleur appréhension du prototype, en en complétant la partie usage.
- Le bricolage permet par les multiples interrogations que suscite sa mise en oeuvre d'établir des échanges presque journaliers entre les bricoleurs.

Cette augmentation de la communication est également soulevée par Mark Hartswood qui présente comment la coréalisation favorise le bricolage et comment réciproquement le bricolage permet plus facilement de mettre en place des mécanismes de communication, dans la mesure où *"the facilitator/bricoleur is able to show how to use the system while the members (...) are able to envisage more fully ways to integrate it into their everyday work tasks"*(Hartswood et al., 2002).

⁵ <http://www.daimi.au.dk/workspace/index.htm>

8.5.2 Conclusion

Dans ces trois dernières sections, nous avons vu qu'il est possible de relier conception participative et conception enseignante par le biais du bricolage. Nous avons vu également que la conception enseignante, proche du bricolage, se démarquait des démarches de conceptions plus industrielles dans la mesure où la notion de prototype n'y avait pas la même importance. En particulier, les prototypes pouvaient difficilement tenir le rôle d'objet frontière en enseignement. Dans la première section de ce chapitre nous avons vu que ce rôle pouvait être tenu par des modèles, mais que ceux-ci n'étaient pas facilement abordables par des non spécialistes.

Permettre à des enseignants de s'approprier des modèles et les outils pour les produire et les manipuler constitue donc un des objectifs de nos travaux.

Dans la section suivante nous présentons notre proposition. Celle-ci constitue un exemple d'ingénierie pédagogique adaptée au bricolage enseignant et reposant sur des tâches de modélisation. Notre proposition s'inscrit du point de vue éducatif, dans la démarche de "logique dispositif" telle que décrite au chapitre 5. Notre proposition s'inspire également de la démarche MDA. Cette démarche appliquée dans le cadre conceptuel de l'IDM, et plus spécifiquement dans le cadre conceptuel des DSL (cf. chapitre 6), permet de mettre en place deux ateliers logiciels que nous décrivons ci-dessous.

8.6 Conclusion et Proposition

8.6.1 Tensions

Nous avons vu dans la première partie plusieurs processus de type ingénieuriste s'appuyant sur les standards définis en EIAH, et permettant d'adresser de diverses manières les plateformes de formation. Nous avons ainsi décrit l'approche documentaire reposant sur les standards LOM et SCORM, l'approche par scénario reposant sur le standard IMS-LD, pouvant être abordée via une méthode d'ingénierie pédagogique cognitiviste (méthode MISA) ou par exemple via une méthode heuristique (CEPIAH).

Nous avons montré les limites que présentaient ces deux approches. Elles concernent essentiellement les difficultés d'adapter les séquences d'enseignement, ainsi préparées, à un contexte d'enseignement spécifique. Ces difficultés tiennent aux objets manipulés (documents, documents simplement séquencés, séquences d'activité), et au désir de rendre indépendant ces objets de tout type d'enseignements et de tout type de plateforme (cf. Figure 32). L'insertion d'une telle standardisation au sein d'un cycle de production pédagogique, qui n'est pas standardisé à son origine et qui ne débouche pas sur des activités standardisées, ne peut que produire les tensions que l'on constate actuellement en EIAH.

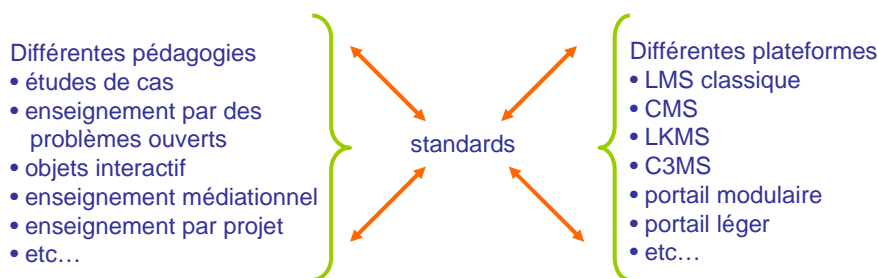


Figure 32 Tensions résultant de la difficulté d'adopter des standards en EIAH

Ces tensions ont essentiellement deux types de conséquences :

En amont elles privilégient des travaux sur l'appréhension de ces standards via des méthodes heuristiques (Giacomini-Pacurar et al., 2005) ou ingénieristes s'inspirant par exemple de théories cognitivistes (De la Teja et al., 2006) ou constructivistes (Laforcade et al., 2005).

Ces travaux révélant la réelle difficulté pour des enseignants à définir des objets ou des séquences pédagogiques réutilisables, partageables, mais finalement peu utilisables.

En aval elles contribuent à faire évoluer les plateformes vers une meilleure prise en compte de ces standards. Les évolutions récentes et prévues des plateformes "Ganesha", "Claroline" et "Moodle" illustrent ce mouvement (Burgos et al., 2005). La plateforme Moodle propose par exemple, la possibilité d'importer dans son environnement des activités LAMS⁶ dont le formalisme est proche du standard IMS-LD. Cependant une telle intégration reste superficielle, les activités LAMS utilisant leurs propres services et n'ayant aucune interaction avec les services proposés par Moodle.

Evoluer vers une meilleure prise en charge des standards en EIAH nécessite pour une plateforme de formation une transformation conceptuelle que toutes les plateformes ne sont pas prêtes à mettre en œuvre. D'autant que paradoxalement les enseignants privilégient souvent dans leur choix des plateformes plutôt atypiques, portails légers détournés ou portails modulaires.

8.6.2 Un nouveau paradigme pour les EIAHs

La situation insatisfaisante décrite dans la section précédente ne peut être résolue que par un changement de paradigme en EIAH. En effet les précédents chapitres ont montré la diversité de l'objet "plateforme de formation", la richesse des approches pédagogiques possibles, et la difficulté de s'appuyer sur les standards actuels des EIAH pour relier plateformes et intentions. Nous avons présenté de nombreux exemples illustrant cette inadéquation (bricolage enseignant, notion de dispositif, activité socio constructiviste, utilisation de plateforme modulaire, utilisation d'application Web 2.0 pour enseigner).

Considérer les EIAHs dans leur diversité en les définissant par l'acte pédagogique qu'ils favorisent constitue le changement de paradigme évoqué par cette thèse.

Pour prendre en compte ce nouveau paradigme nous proposons de mettre en place une ingénierie spécifique. Cette ingénierie permet de produire des infrastructures adaptées à la diversité des EIAHs et à la diversité des approches pédagogiques. Cette ingénierie repose sur des techniques IDM. Elle est décrite dans les sections suivantes.

8.6.3 Notre proposition = deux ateliers : Outillage et Construction.

Notre proposition repose sur une démarche IDM pour instrumenter la construction de dispositifs pédagogiques sur des plateformes de formation. Cette démarche implique la réalisation d'une infrastructure composée de deux ateliers logiciels (cf. Figure 33).

Le premier atelier "Outillage" permet de construire les outils (métamodèles, modeleur, transformation, générateur spécifique, greffon de service Web) nécessaire à la construction de dispositifs et tente de résoudre les problèmes d'interopérabilité par l'usage de l'ingénierie dirigée par les modèles.

⁶ <http://www.lamsinternational.com>

Le second atelier, nommé "**Construction**" concerne l'utilisation par des enseignants de cet outillage dans le but de concevoir puis de construire un dispositif pédagogique dans le contexte d'une plateforme de formation. Il est lié au premier dans la mesure où il en est l'usage. Au sein de cet atelier, nous proposons la manipulation par des enseignants de modèle de dispositifs.

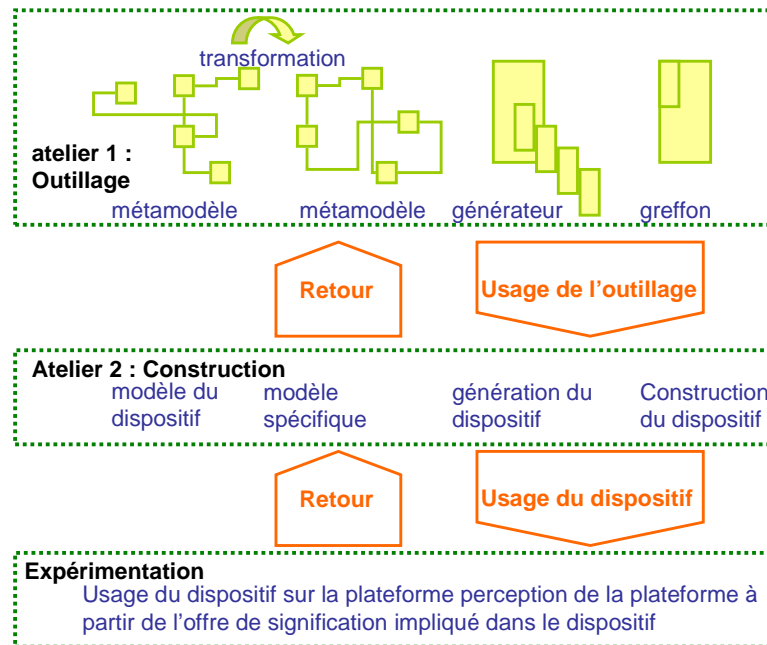


Figure 33 Présentation de la démarche

8.6.4 Le domaine d'application et d'expérimentation de notre ingénierie

Notre proposition repose sur le désir de prendre en compte les EIAHs dans leur diversité et donc de proposer une ingénierie capable de produire des infrastructures facilement adaptables. Nous avons pour cela testé nos ateliers pour accompagner le travail d'enseignant :

- sur des plateformes de formation classique, Ganesha, Claroline,
- sur des plateformes modulaires : PostNuke, Moodle,
- sur des applications de type Web 2, WikiniMST, Elg.

Nous avons privilégié, sur ces plateformes, la construction de dispositifs. En effet parmi les éléments manipulés susceptibles de faire l'objet de standard, nous avons montré qu'il était peut être possible de porter une attention plus grande à la notion de dispositifs plutôt qu'aux documents pédagogiques et à leur séquençement individualisé ou non. L'utilisation de la notion de dispositifs (cf. chapitre 5) offre un cadre normatif permettant la construction de dispositifs techniques précis sous-tendus d'intentions pédagogiques suggérés et pouvant accueillir un enseignement de type ouvert. Porter une attention aux dispositifs, c'est suggérer un cadre conceptuel de pensée, une offre d'intention pouvant être traduit par un dispositif générique pouvant lui même être transformé en un dispositif spécifique dans le contexte d'une plateforme de formation.

Dans le chapitre 9 nous présentons les problématiques IDM que la mise en œuvre de notre proposition nous ont permis d'aborder. La réalisation concrète de notre proposition est présentée dans le chapitre 10.

chapitre 9 Problématiques IDM abordées par notre recherche

Dans ce chapitre nous abordons les problématiques IDM rencontrées lors de notre recherche. Ce chapitre est divisé en trois sections : la première présente la place de l'IDM dans notre proposition, ainsi que les spécificités IDM de notre domaine de recherche. La deuxième section aborde la problématique posée par la définition d'un modèle de la plateforme. Enfin la dernière section pose le problème de l'opérationnalisation d'un modèle sur une plateforme ne disposant pas de générateur de code.

9.1 Utiliser l'Ingénierie dirigée par les modèles

9.1.1 Comment utiliser l'IDM dans le cadre de notre proposition?

Expérience acquise par l'équipe NOCE

L'équipe NOCE, a commencé à mener des recherches sur l'IDM en 1998. La thèse de Xavier Le Pallec, *"Des services d'adaptation de modèles pour la coopération de méta systèmes : application aux groupware flexibles"* (Le Pallec, 2002) propose une solution d'interopérabilité appliquée au travail coopératif assisté par ordinateur. Cette solution destinée à rendre flexible un système informatique repose sur l'approche par métamodèle. A cet effet une plateforme de métamodélisation et de modélisation a été produite. Cette plateforme nommée successivement Ram3 puis ModX est un environnement réflexif permettant de mener des activités de prototypage. Il est ainsi possible dans l'environnement ModX de manipuler des métamodèles et d'étudier instantanément les répercussions de ces modifications sur les modèles produits. Ces facilités ont été appliquées au travail coopératif pour produire des services d'adaptation permettant d'aborder l'interopérabilité entre deux groupwares.

Un mémoire rédigé dans le cadre d'un master recherche au sein de l'équipe NOCE (Caron, 2003) puis cette thèse ont permis de reconsidérer ModX selon une approche conceptuelle correspondant à une plateforme de modélisation permettant la production de langage spécifique de domaine.

La plateforme de modélisation ModX repose sur une implémentation du MOF (cf. chapitre 6). Dans le cadre de cette implémentation, la définition d'un métamodèle conforme au MOF représentant un domaine particulier permet de construire un modelleur spécifique à ce domaine. Il est alors possible d'utiliser ce modelleur pour mener, dans le domaine qu'il contribue à décrire, des activités de modélisation. Les modèles produits sont alors conformes, au sens MOF du terme, au métamodèle sous-jacent au modelleur.

Deux cycles de création

L'utilisation de l'ingénierie dirigée par les modèles repose sur deux cycles de création semblables aux deux ateliers que nous souhaitons mettre en œuvre (évoqués dans le chapitre 8). Le recours à l'IDM permet de définir une démarche formelle pour mettre en œuvre ces deux ateliers. Or en enseignement l'évolution des pratiques et des plateformes de formation peut amener les enseignants à redéfinir fréquemment les outils de modélisation définis dans le premier atelier. C'est la force de l'IDM et de la plateforme de modélisation que nous utilisons que de pouvoir effectuer ces changements de façon réflexive. Appliquée à notre approche dispositive des plateformes de formation, cette propriété permet d'accompagner les activités de conception dans l'usage qui, nous l'avons montré, sont particulièrement importantes en conception pédagogique (cf. sections 8.3, 8.4). Ainsi un changement effectué sur le métamodèle sous-jacent à l'atelier **Outils** se répercute sur l'atelier **Construction** et sur la perception du dispositif sur la plateforme de formation (cf. Figure 33 Présentation de la démarche). Un retour d'usage lors de l'expérimentation du dispositif peut également permettre d'interroger les processus de conception des deux ateliers. L'ensemble des deux ateliers concourt à favoriser la mise en place de bricolage pédagogique.

Expressivité et adaptabilité des métamodèles pour décrire les pratiques enseignantes et les plateformes de formation

Le recours à l'IDM permet, comme nous venons de le voir, de concevoir des modélisateurs spécifiques à un domaine particulier. Il est ainsi possible de produire des modélisateurs très simples appropriés aux pratiques des enseignants au sein de ces modélisateurs. Au sein de ces modélisateurs il est alors possible de définir des règles de transformation pour adapter les modèles de dispositifs pédagogiques aux concepts sous-jacents à la création de dispositifs sur une plateforme de formation spécifique. Dans le cadre des études que nous avons menées pour le projet PCDAI (cf. chapitre 11) nous verrons que le recours à l'IDM a permis très facilement de faire évoluer l'infrastructure mise à disposition des enseignants.

Transformations et interopérabilité

L'IDM offre la possibilité de définir, par exemple entre des métamodèles A et B, des règles de transformation qui, appliquées à un modèle conforme au métamodèle A, permettent la création semi-automatique d'un modèle conforme au métamodèle B.

Ces règles appliquées à notre domaine de recherche permettent de transformer un modèle abstrait de dispositif pédagogique en un modèle de dispositif dans le contexte d'une plateforme de formation (cf. chapitre 5). Par le biais des transformations, l'approche IDM aborde ainsi le problème de l'interopérabilité des intentions pédagogiques avec les plateformes de formation. Cette approche est une alternative logicielle et conceptuelle à l'utilisation de standards en EIAH.

Modèle expressif et fusion de métamodèles

Parce que l'IDM repose sur la manipulation de modèles conformes à des métamodèles appropriables, l'environnement que nous proposons peut permettre la manipulation de modèles exprimant les préoccupations métier des enseignants. Cet environnement permet en outre de placer les préoccupations enseignantes en amont du processus de création. Des mécanismes de fusion entre métamodèles peuvent également permettre d'exprimer, au sein d'un métamodèle dit "fusionné", les différents aspects fonctionnels et pédagogiques d'une

plateforme de formation. Ce recours à un métamodèle unique permet de définir un modèleur simple spécifique à une plateforme et à ses guides d'utilisation. Un tel métamodèle fusionné a été produit pour travailler lors de la phase deux de l'étude de cas sur l'application WikiniMST dans le cadre du projet PCDAI (cf. chapitre 11).

9.1.2 Place de l'IDM dans l'atelier Outillage

Notre proposition d'infrastructure pour une plateforme spécifique repose sur la définition de deux ateliers **Outillage** et **Construction** (décrit chapitre 8, cf. Figure 33).

L'IDM, dans le cadre du premier atelier accepte une participation de la communauté enseignante à l'appropriation ou à la définition des métamodèles à utiliser.

Pour la définition de l'atelier Outillage plusieurs solutions ont été envisagées :

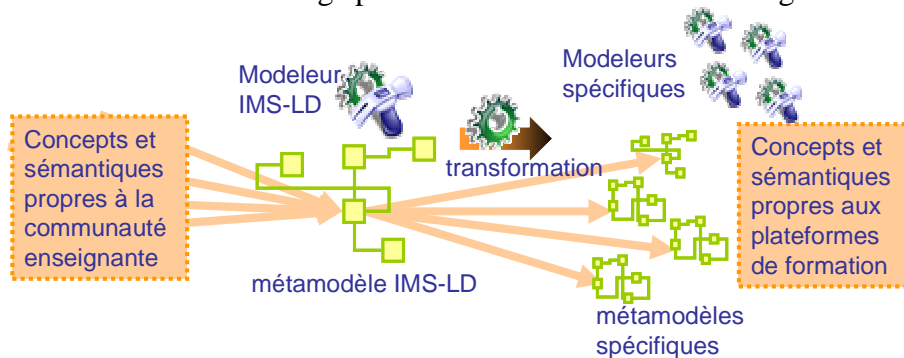


Figure 34 : Solution 1 Utiliser un métamodèle pivot

La première solution considère la notion de dispositif en la reliant à la notion de scénario pédagogique (cf. Figure 34). Pour un scénario pédagogique, il est possible d'exprimer le dispositif permettant de mettre en œuvre le scénario, cette solution exploite le standard IMS-LD pour une expression standardisée d'un dispositif générique. Un tel dispositif doit alors être spécialisé pour être construit sur une plateforme de formation spécifique. Une telle spécialisation peut être instruite par une démarche reposant sur l'ingénierie dirigée par les modèles.

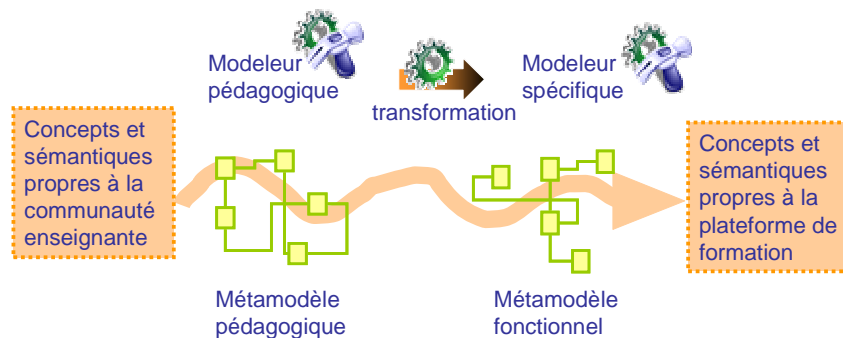


Figure 35 Solution 2 : Utiliser des métamodèles pédagogiques ad hoc

La seconde solution envisage de considérer un dispositif comme la concrétisation d'un dispositif abstrait représentant les intentions pédagogiques d'une équipe d'enseignant (cf. Figure 35). Cette nouvelle considération permet de mener une réflexion autour de la notion de dispositif sans préjuger de l'ordonnancement des activités qui vont s'y dérouler. La particularité de cette solution est de ne pas passer par un standard commun, mais par une approche commune. C'est cette approche reposant sur l'Ingénierie Dirigée par les Modèles qui permet, par la manipulation de modèles conformes à des métamodèles exprimé en MOF, d'être le garant des interopérabilités souhaitées. Des transformations automatiques permettent la définition d'un modèle fonctionnel constructible à partir du modèle pédagogique. Une des

conséquences de cette solution est cependant de devoir définir pour chaque équipe d'enseignant une partie de l'outillage spécifique nécessaire aux activités de modélisation.

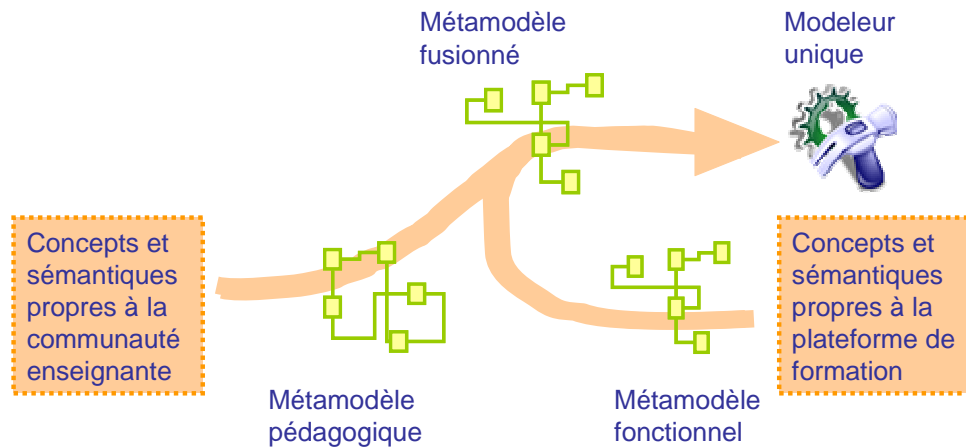


Figure 36 Solution 3 : Utiliser un métamodèle fusionné

La troisième solution, nécessite la réalisation d'un métamodèle fusionné (cf. Figure 36). Comme la solution précédente cette solution s'appuie sur la notion de dispositif, cependant elle privilégie la réalisation d'un modeleur unique adapté à l'utilisation pédagogique d'une plateforme de formation dans un cadre précis.

Ces trois solutions peuvent être mise en oeuvre par une approche IDM. Cette approche permet d'établir une transformation ou une fusion dynamique entre un modèle de dispositif indépendant d'une plateforme de formation et un modèle de dispositif dans le contexte de la plateforme. La première solution qui impose un métamodèle unique indépendant des plateformes n'est, d'un point de vue modélisation, qu'un cas particulier de la seconde solution. La troisième solution ne peut être envisagée que dans la mesure où les métamodèles pédagogiques et fonctionnels ont été définis dans un premier temps. Lors de la seconde l'étude présentée chapitre 11, cette troisième solution a permis de proposer un modeleur plus simple à utiliser pour les enseignants. Sur le plan de la participation des enseignants à la définition de l'atelier "**Outillage**" ces trois solutions se situent sur des échelles différentes. La première solution implique un travail mené par l'ensemble de la communauté EIAH à la définition de méthodes permettant l'appropriation du standard. La seconde solution nécessite une réflexion au sein de l'équipe enseignante pour définir les concepts par lesquels elle va exprimer ses dispositifs. La troisième solution nécessite une pratique permettant d'élaborer pour chaque plateforme les guides d'utilisation les plus communs.

L'approche IDM que nous venons de décrire permet de définir une partie de l'infrastructure nécessaire à la définition de modèles de dispositifs pédagogiques constructibles sur une plateforme de formation. La seconde partie de cette infrastructure sera décrite dans la section 9.3 de ce chapitre.

9.1.3 En guise de conclusion : "sur l'adéquation de l'IDM à notre proposition"

Adaptabilité

L'IDM permet de placer l'enseignant au cœur des deux ateliers permettant de définir les outils de production et de produire des dispositifs pédagogiques. Les mécanismes de d'intercession

que propose la plateforme ModX, trouvent leur application au sein de ces deux ateliers : ils permettent la modification dans l'usage de chaque atelier. Enfin le bricolage enseignant est favorisé par ces mécanismes.

Utilité

L'IDM permet d'automatiser en partie le passage d'un dispositif abstrait tel qu'il est conçu par une équipe d'enseignant, à un dispositif concret dans le contexte d'une plateforme de formation. Une solution reposant sur l'IDM permet de s'affranchir de la notion de standard pédagogique en EIAH et elle offre une solution permettant d'aborder le problème difficile de l'interopérabilité des plateformes de formation.

Notre proposition et nos exigences initiales

Nous pensons que notre proposition est particulièrement bien adaptée aux exigences que nous énonçons dans l'introduction de cette thèse. Ces exigences reposaient sur le double fossé séparant un EIAH des enseignants souhaitant l'utiliser (cf. chapitre 1). Notre proposition tente en effet d'automatiser les tâches (création, persistance, partage, modification) résultant du désir de rationaliser la création de dispositifs pédagogiques sur une plateforme de formation, en préservant les spécificités des plateformes et des enseignants.

9.1.4 Problématiques IDM évoquées par notre proposition

Dans les sous-sections précédentes nous avons montré les apports potentiels de l'IDM à notre problématique en EIAH. Nous nous plaçons dans les sections suivantes dans une perspective différente : celle d'étudier des verrous scientifiques propres à l'IDM. Ces verrous sont illustrés par la mise en oeuvre d'un cas de cycle de vie de type IDM adapté aux EIAHs.

La Figure 37 présente les deux verrous que nos travaux abordent. Ces verrous sont identifiés par l' Action Spécifique "*Model Driven Architecture*" (Bézivin, Blay-Fornarino et al., 2005).

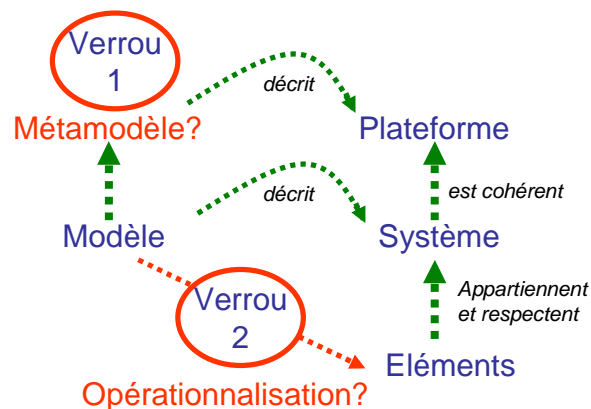


Figure 37 Problématiques spécifiques à l'IDM

Le premier verrou abordé est :

Comment définir le métamodèle correspondant à une plateforme lorsqu'il est inexistant ?

Nous verrons dans la section suivante qu'aborder ce verrou dans le domaine des EIAHs est particulièrement justifié. Le domaine des EIAH est en effet caractérisé par un grand nombre

de plateformes ayant des fonctionnalités différentes. Ces plateformes peuvent être considérées comme des systèmes d'information dont il est possible de modéliser certains aspects. Cela nous amène à interroger le concept de plateforme propre à l'IDM et de porter une attention au processus de définition d'un modèle de certaines fonctionnalités d'une plateforme de formation. Ces fonctionnalités sont principalement celles permettant la construction de systèmes. Un tel modèle peut alors être considéré comme un métamodèle permettant l'expression de modèle décrivant des systèmes dans le contexte d'une plateforme.

Le second verrou concerne ces modèles décrivant des systèmes :

En quoi consistent les interactions entre un modèle et son opérationnalisation dans le contexte d'une plateforme ?

Notre problématique en EIAH, construire des dispositifs de formation sur des plateformes de formation, nous permet d'aborder des systèmes d'informations dont les fonctionnalités de construction ne reposent pas sur la génération de code. La construction dans notre domaine d'application consiste à interagir avec une instance donnée de la plateforme correspondant à un site pédagogique afin d'y construire le dispositif pédagogique. La section 9.2 aborde le problème de la définition du modèle d'une plateforme et la section 9.3 aborde le problème de l'opérationnalisation d'un modèle de dispositif.

9.2 Modèle de plateforme et métamodèle de langage

En IDM, la définition du métamodèle "technologique" d'une plateforme a déjà été qualifiée de problématique (Marvie et al., 2005). Dans le contexte des EIAHs ce problème devient particulièrement critique à cause de la grande diversité des plateformes.

9.2.1 Qu'est ce qu'une plateforme de formation au sens de l'IDM

L'IDM aborde de façon générale la notion de plateforme d'exécution : une plateforme n'est pas exclusivement un environnement de programmation dans lequel est généré du code à partir d'un modèle (Czarnecki & Helsen, 2003). L'IDM définit une plate-forme d'exécution comme un interpréteur pour l'exécution de systèmes.

"Une plate-forme propose un modèle d'exécution ainsi qu'un ensemble de fonctionnalités de base utilisées par une majorité de systèmes. Le modèle d'exécution de la plate-forme d'exécution définit l'interprétation des systèmes logiciels. Les fonctions, qualifiées de non fonctionnelles au regard des systèmes construits, ont pour objectif principal de faciliter la construction de ces systèmes en s'abstrayant de la complexité et de la technicité de leur mise en oeuvre." (Marvie et al., 2005)

Une plateforme d'exécution établit le modèle des structures de données qu'elle manipule et traite, et pour lesquelles elle définit des règles de cohérence.

Dans le domaine des EIAHs, il est possible de considérer une plateforme de formation comme une plateforme d'exécution permettant l'interprétation de dispositifs pédagogiques. Construire un dispositif sur une telle plateforme consiste alors à créer sur celle-ci des éléments en respectant la cohérence du système qu'ils forment.

En cela notre démarche est proche des travaux concernant le déploiement de composants (Estublier et al., 2005), elle peut être également rapprochée des recherches portant sur le déploiement d'application (Merle, 2004). Ces approches ne proposent pas de générer du code

mais de construire des systèmes dans le contexte d'exécution de la plateforme. De tels systèmes doivent alors respecter les règles de cohérence et d'utilisation de la plateforme.

9.2.2 Pourquoi réaliser des modèles de plateformes d'exécution en EIAH?

Réaliser un modèle de plateforme d'exécution c'est être capable de factoriser et de décrire les traitements qu'elle réalise par des règles ou des objets. Nous proposons d'exprimer ce modèle en MOF au sein d'un environnement de modélisation. Une telle expression permet alors de considérer le modèle de la plateforme comme un métamodèle.

En faisant reposer un modèleur, spécifique à la plateforme, sur la définition de ce métamodèle, il est possible de modéliser des modèles de systèmes conformes au métamodèle induit par le modèle de la plateforme d'exécution.

Cette double interprétation du modèle de la plateforme comme un métamodèle conformément auquel il est possible de définir des modèles de système, est évoqué dans (Marvie et al., 2005). L'exemple présenté par Raphael Marvie est présenté Figure 38 ci-dessous.

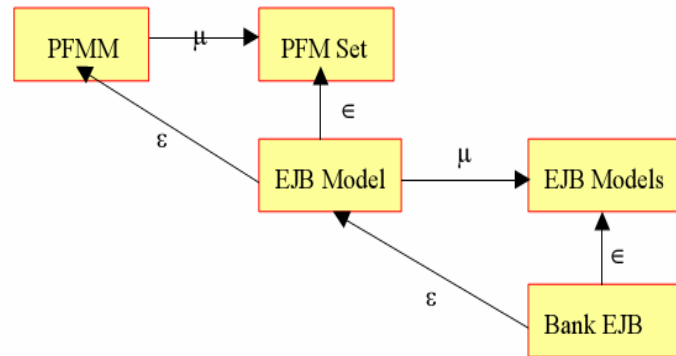


Figure 38 Relations entre modèles de plate-forme et métamodèle de système

La Figure 38 présente cette relation entre un modèle de plate-forme et un métamodèle de système dans le contexte de cette plate-forme. Un système bancaire, dans le contexte de la plateforme EJBLike,⁷ doit être conforme par rapport à la définition de cette plate-forme. Le métamodèle du système bancaire, dans le cadre de la plate-forme EJBLike, est donc le modèle de cette plate-forme EJBLike (Marvie et al., 2005).

Dans le cadre de notre recherche en EIAH, nous obtenons la Figure 39 suivante :

⁷ Enterprise JavaBeans (EJB) est une architecture de composants logiciels côté serveur pour la plateforme de développement J2EE

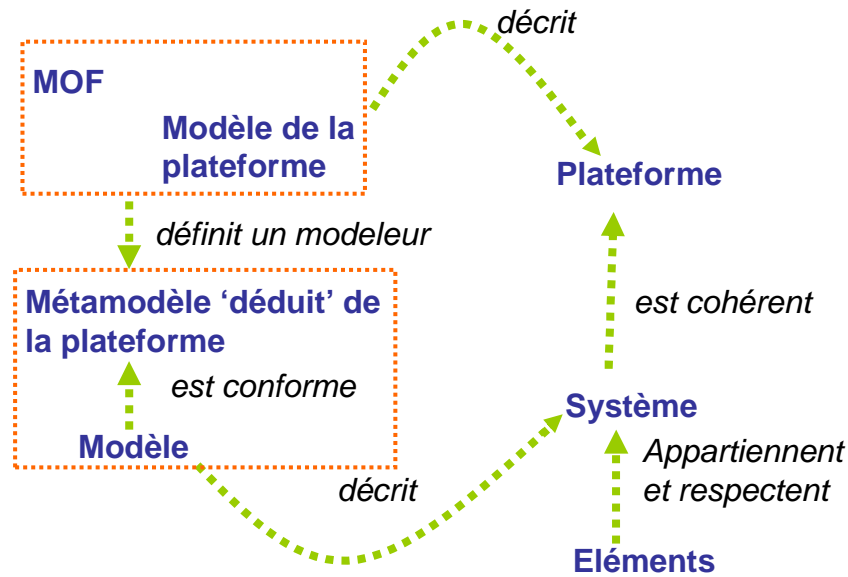


Figure 39 Modèle de plateforme = Métamodèle du modelleur

Cette conformité MOF du modèle (décrivant un système) au métamodèle (dédit du modèle de la plateforme d'exécution) est pragmatiquement ce qui permet d'espérer que les modèles de systèmes produits décrivent bien les systèmes fonctionnant sur la plateforme.

Il est donc possible de définir un modèle de la plateforme d'exécution pour en déduire un métamodèle permettant de réaliser un modelleur adapté à la plateforme. Le domaine des EIAHs est caractérisé par un nombre important de plateformes et par des pratiques pédagogiques variées (cf. 8.6.1). Disposer de tels modelleurs, c'est-à-dire spécifiques à chaque plateforme, et pouvoir réaliser des transformations automatiques de modèles, permet d'adapter les intentions dispositives d'enseignants sans les dénaturer. Notre proposition basée sur la réalisation de modelleurs et l'utilisation de transformations se révèle être, pour ce domaine, une solution aux problèmes posés par l'hétérogénéité des plateformes et des pratiques.

Dans les sections suivantes, nous allons voir que bien que notre proposition conduise à une infrastructure simple à utiliser pour un enseignant, celle-ci n'est pas simple à mettre en œuvre, et requiert une compréhension des mécanismes élémentaires de l'ingénierie dirigée par les modèles.

Notre proposition capitalise certains résultats obtenus en IDM, en particulier dans le domaine des langages spécifiques de domaine (DSL). Elle présente aussi plusieurs avancées spécifiques au domaine IDM. Ces avancées concernent la définition d'un modèle de plateforme, ainsi que l'opérationnalisation de modèles dans le contexte d'une plateforme. Ces avancées sont abordées dans les sections suivantes, la réalisation concrète de ces avancées est décrite dans le chapitre 10 suivant, des exemples d'utilisation sont ensuite relatés. Dans la suite de ce chapitre nous utiliserons indifféremment les termes métamodèle du domaine (ou du modelleur) et modèle de la plateforme pour désigner une description MOF de notre plateforme d'exécution.

9.2.3 Comment définir un modèle de plateforme

Le modèle d'une plateforme doit à la fois abstraire les différents éléments constituant les systèmes pouvant opérer sur la plateforme, et capturer les règles qui sous-tendent la

construction de ces systèmes. L'élaboration d'un tel modèle de plateforme est rendue difficile par plusieurs points. Comme souvent en IDM, la plateforme étudiée ne dispose pas d'un modèle la décrivant (Bézivin, Blay-Fornarino et al., 2005). Un tel modèle peut être alors être en partie déduit par rétro-ingénierie.

Un exemple d'utilisation des concepts de rétro ingénierie est par exemple décrit dans le cadre de la réingénierie des applications patrimoniales (Estublier, Favre, & Sanlaville, 2002). A l'occasion de ce travail de réingénierie, une méthode a été élaborée, elle est décrite dans (Favre & Musset, 2006). Cette méthode définit trois étapes : l'inventaire des outils de génie logiciel, la rétro-ingénierie des métamodèles et l'intégration des métamodèles. La rétro-ingénierie des métamodèles consiste à partir de l'analyse des bases de données, des bugs, des schémas d'échange, de Document Type Definition (DTD), d'un ensemble de fichier xml (etc...) à définir un métamodèle permettant d'encadrer la réalisation de modèle décrivant les différents aspects de l'application. Si les éléments objectifs à prendre en compte sont assez clairs, (en gros tout ce que l'on peut trouver), il est intéressant de comprendre comment intégrer toutes ces données pour définir un métamodèle.

Pour Vega Baez, un domaine est une construction créée dans une intention. Dans le cadre de ses recherches, Vega Baez aborde la création de lignes de produit logiciel. Une des caractéristiques principales des lignes de produit logiciel est de créer un cadre logiciel pouvant être décliné dans des environnements différents. L'intention qui sous-tend le domaine abordé par l'auteur est donc le "réutilisation" (Vega Baez, 2005). Une fois identifiée l'intention qui sous-tend le domaine, il faut délimiter son périmètre : *"la délimitation du périmètre du domaine est un pas indispensable pour entreprendre sa conceptualisation"* (Vega Baez, 2005, p. 47). Cette délimitation consiste pour l'auteur à définir plusieurs sous-ensembles des fonctionnalités spécifiques identifiées lors de l'analyse du domaine, *"On remarque aussi qu'on a volontairement omis certaines caractéristiques très spécifiques à cette ligne de produits, qui empêcheraient ce domaine d'être réutilisé dans un contexte plus vaste"* (Vega Baez, 2005, p. 53). A partir de là, il est possible d'entreprendre la conceptualisation du domaine, afin d'aboutir à la définition de son métamodèle global. L'auteur préconise donc une démarche consistant à identifier les fonctionnalités des différents aspects de son domaine contribuant au but qu'il s'est fixé. Pour chaque aspect, il restreint ensuite ces fonctionnalités pour augmenter le champ d'application de son métamodèle. Le métamodèle global que l'auteur obtient in fine est défini par réunion de chaque métamodèle.

La démarche que nous préconisons dans cette thèse, est proche de celle de Vega Baez, nous la décrivons dans le cadre des environnements informatiques pour l'apprentissage humain. Notre démarche s'appuie sur des pratiques que nous recensons dans la sous section suivante.

9.2.4 De bonnes pratiques pour définir un modèle de plateforme en EIAH

Définir le métamodèle global d'une plateforme existante est une démarche ascendante, dont la première étape consiste à analyser les fonctionnalités de la plateforme. Nous avons pu constater durant notre recherche que la définition d'un métamodèle est une activité subjective, inspirée par de nombreux choix conceptuels. Nous allons voir dans cette sous-section et dans la sous section suivante comment malgré ce caractère subjectif, il est possible de proposer de bonnes pratiques permettant de faciliter la définition d'un tel métamodèle.

Ces bonnes pratiques s'appuient sur une série d'actions qui, bien qu'elles soient présentées de façon séquentielle, sont, dans la pratique, menées de façon concurrente.

Elles peuvent être résumées par le tableau suivant :

Définir les intentions du domaine	
Séparer intentions opératoires et intentions pédagogiques	
Restreindre le domaine de modélisation fonctionnel	Restreindre le domaine de modélisation pédagogique, en appliquant la logique dispositive
Définir les différents aspects du métamodèle fonctionnel	Définir les différents aspects du métamodèle pédagogique
Définir les services encapsulant les fonctionnalités de la plateforme	Pour chaque métamodèle, identifier les éléments de factorisation
Pour chaque métamodèle identifier les fabriques d'éléments	Mettre en place les mécanismes de factorisation
Fusionner les différents métamodèles fonctionnels	Fusionner les différents métamodèles pédagogiques
Etablir des transformations ou fusionner métamodèle fonctionnel et pédagogique	
Définir pour le métamodèle fusionné les règles d'implication pédagogiques	

Description de ce tableau

Définir un métamodèle global c'est avant tout préciser les intentions qui sous-tendent le domaine. Ces intentions peuvent être de deux types : intentions opératoires, intentions pédagogiques. Dans cette section nous allons montrer comment la mise en exergue des intentions opératoires du domaine permet la définition de différents aspects du domaine et leur métamodèles associés. Nous verrons dans la section suivante comment les intentions pédagogiques permettent de compléter le métamodèle par des mécanismes de fusion ou de transformation.

Pour définir différents métamodèles opératoires, il nous faut dans un premier temps **restreindre** le domaine de modélisation en excluant par exemple le paramétrage global (sécurité, patron d'affichage...). Cette restriction est guidée par les intentions du domaine que nous avons identifiées. Notre objectif est de construire sur des plateformes de formation des dispositifs pédagogiques. Les fonctionnalités du domaine que nous souhaitons retenir correspondent donc à des fonctionnalités de construction. Ces fonctionnalités nous renvoient à la définition des objets qu'elles adressent, donc aux liens qu'ils entretiennent entre eux, et à la possibilité d'interroger leurs états. **Construction, interrogation** et **liaison** sont les fonctionnalités que nous retenons pour notre domaine. Il faut ensuite définir et analyser les différents aspects permettant de décrire notre domaine. La définition de ces différents aspects s'établit en parallèle avec l'encapsulation des fonctionnalités de construction, d'interrogation et de liaison de la plateforme. Dans l'exemple décrit au chapitre 11.2, nous avons défini sur la plateforme Ganesha (Ganesha) des services dont la liste suivante présente un résumé (cf. Figure 40).

about	listAllLearners	addSequenceToCourse
auth	listAllTeachers	addModuleToCourse
createMatter	listAllAdmins	listSequencesOfGroup
createDomain	listAllSequences	listModulesOfGroup
createGroup	listAllModules	listCoursesOfGroup
createLearner	listAllCourses	listLearnersOfGroup

createTeacher	addSequenceToGroup	listTeachersOfGroup
createAdmin	addModuleToGroup	listSequencesOfLearner
createSequence	addCourseToGroup	listModulesOfLearner
createModule	addLearnerToGroup	listCoursesOfLearner
createCourse	addTeacherToGroup	listMattersOfDomain
createTeacherOptions	addSequenceToLearner	listSequencesOfModule
listAllDomains	addModuleToLearner	listSequencesOfCourse
listAllMatters	addCourseToLearner	listModulesOfCourse
listAllGroups	addMatterToDomain	addSequenceToModule

Figure 40 fichier de description des services encapsulant certaines fonctionnalités de Ganesha

En établissant cette liste, on cherche à les regrouper selon plusieurs aspects, il est par exemple possible de s'intéresser à l'aspect organisationnel humain et pédagogique, cet aspect se traduit par la mise en exergue des services suivants :

`createLearner`, `createTeacher`, `createGroup`, `listAllLearners`, `listAllTeachers`, `listAllGroups`, `listLearnersOfGroup` et `listTeachersOfGroup`.

Ces services peuvent être décrits pour la plateforme Ganesha (Ganesha) par un métamodèle décrivant l'organisation humaine et pédagogique que la plateforme propose. Ce métamodèle met en jeu les relations entre enseignants, apprenants et groupes, comme le montre le schéma suivant :

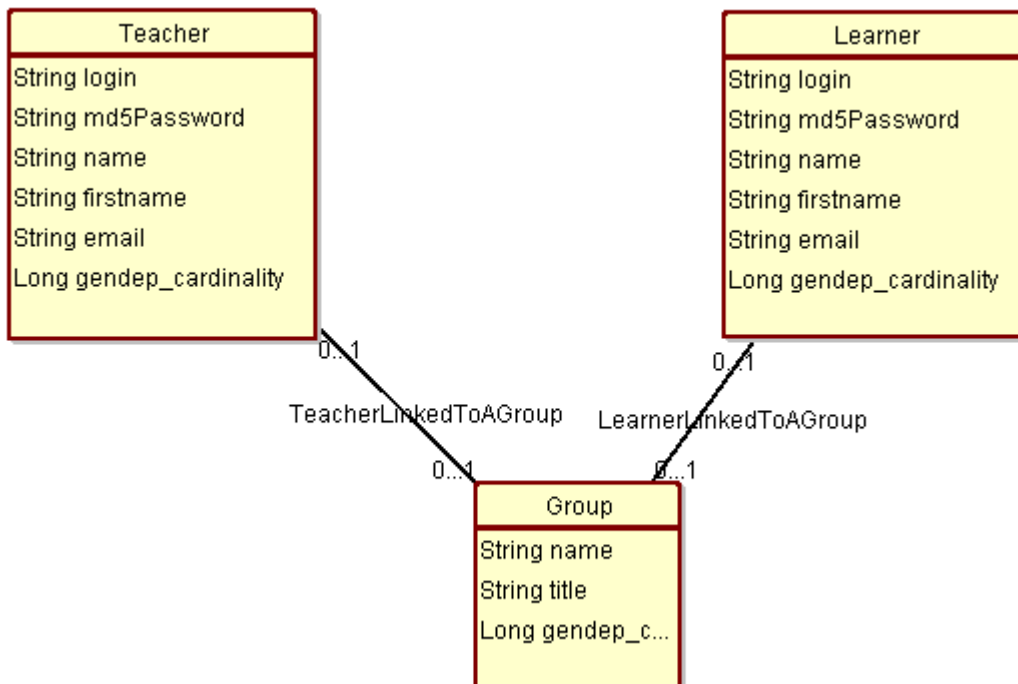


Figure 41 : détails du métamodèle Ganesha

Chaque classe, chaque association dispose de fonctionnalités de construction et d'interrogation. Dans l'exemple ci-dessus, un enseignant est défini par son identifiant, son mot de passe et son email etc..., tout comme un apprenant, il peut être relié à un groupe. Le métamodèle est complété par des aspects permettant de mettre en œuvre au sein des modèles des mécanismes de construction. Cet aspect se traduit sur la Figure 41 par la présence de l'attribut "**gendep_cardinality**", l'utilisation de cet attribut sera explicité dans la section 9.3.

Ce métamodèle doit être ensuite fusionné avec les différents métamodèles décrivant d'autres aspects du domaine relevant des intentions de construction, d'interrogation et de liaison spécifiées.

Nous identifions ensuite les **fabriques d'éléments** disponibles et utiles ainsi que leur capacité à paramétrer des éléments créés et à les relier entre eux. Cette identification est faite à travers l'analyse de systèmes existants sur des instances de la plateforme concernée, du code de la plateforme ou des interfaces de services Web (WSDL) quand elles existent. Ce sont ces éléments qui constitueront les concepts du métamodèle. Dans l'exemple Ganesha précédent, ces fabriques correspondent aux classes suivantes :

Learner, Teacher, Group, Admin, Sequence, Module, Course, Matter, Domain.

La définition d'un **mécanisme de factorisation** est quant à lui directement issue des problématiques posées par la modélisation pédagogique. En effet, créer un type d'étudiants dans un dispositif pédagogique pour indiquer le rôle qu'il tiendra au sein d'un dispositif n'est pas possible au sein d'une plateforme (tout comme créer un type de cours, de ressources...). Cette capacité à définir les différents rôles qu'un étudiant va devoir mener au sein du dispositif, ne peut-être déduite de l'analyse de la plateforme. Il n'est souvent pas possible non plus de le définir directement par un type hérité dans le métamodèle. En effet l'expérience nous a montré que c'est souvent au cours de la modélisation pédagogique, que ces rôles sont définis.

L'environnement spécifique de modélisation doit donc permettre de définir des processus de factorisation lors de la construction d'un dispositif.

Le mécanisme que nous proposons pour pallier à cette absence part du principe qu'un modèle est une vue simplifiée d'un système existant ou qui existera (Bézivin et al., 2004) et donc qu'un élément de modèle peut factoriser une collection d'éléments du système modélisé. Nous en déduisons qu'un élément de modèle peut profiler des éléments des futurs systèmes correspondants en indiquant un nombre d'éléments possibles (cardinalité) et des patrons de valeurs pour les attributs (on peut y voir quelques similitudes avec la modélisation incomplète ou modélisation tardive (Papavassiliou, Mentzas, & Abecker, 2002), mais le principe sous-jacent est différent). Ce mécanisme de profil est plus général que le mécanisme de typage, fréquent dans les plateformes plus généralistes (structures, classes, composant...).

Les règles de causalité d'un modèle constitue le dernier point de notre approche, elles sont à définir dans le métamodèle, et elles permettent de traiter les problèmes de dépendances de construction, par exemple, lors de la mise en œuvre du Wiki, la création d'un utilisateur implique la création de son espace personnel.

Ce travail mené en plusieurs étapes concurremment à la définition des services encapsulant la plateforme, permet de définir des métamodèles décrivant plusieurs aspects du domaine tant dans le domaine pédagogique que fonctionnel. Il est alors possible de fusionner ces métamodèles pour obtenir un métamodèle unique représentant les intentions du domaine. Dans la sous-section suivante nous allons aborder cet aspect ainsi que les mécanismes de transformation que nous mettons en œuvre.

9.2.5 De la transformation à la fusion de métamodèle

Nous avons expérimenté au cours de cette thèse deux méthodes pour définir un métamodèle utilisable au sein d'un modeleur. Parce que ces deux méthodes ont des répercussions sur le type d'infrastructure dans le cadre des ateliers **Outillage** et **Construction**, nous présentons ci-dessous nos différentes propositions.

Première méthode : proposition exploitant les transformations entre métamodèles.

La première méthode "par transformation" est décrite à la fin du chapitre 5, pour l'atelier **Outillage**, cette méthode propose de définir un processus de concrétisation proche du MDA (cf 6.4). Cette méthode peut être illustrée par la Figure 42, où dans un souci de clarté, l'atelier outillage a été scindé en deux parties.

La Figure 42 illustre les rapports que peuvent entretenir l'atelier **Outillage** (deux premiers diagrammes) et l'atelier **Construction**. Ce dernier, à destination d'une équipe d'enseignants, propose la modélisation et la construction de dispositif pédagogique. Il est supporté par l'élaboration en deux étapes de l'atelier Outillage. Dans les paragraphes suivants nous décrivons la finalisation de ces ateliers.

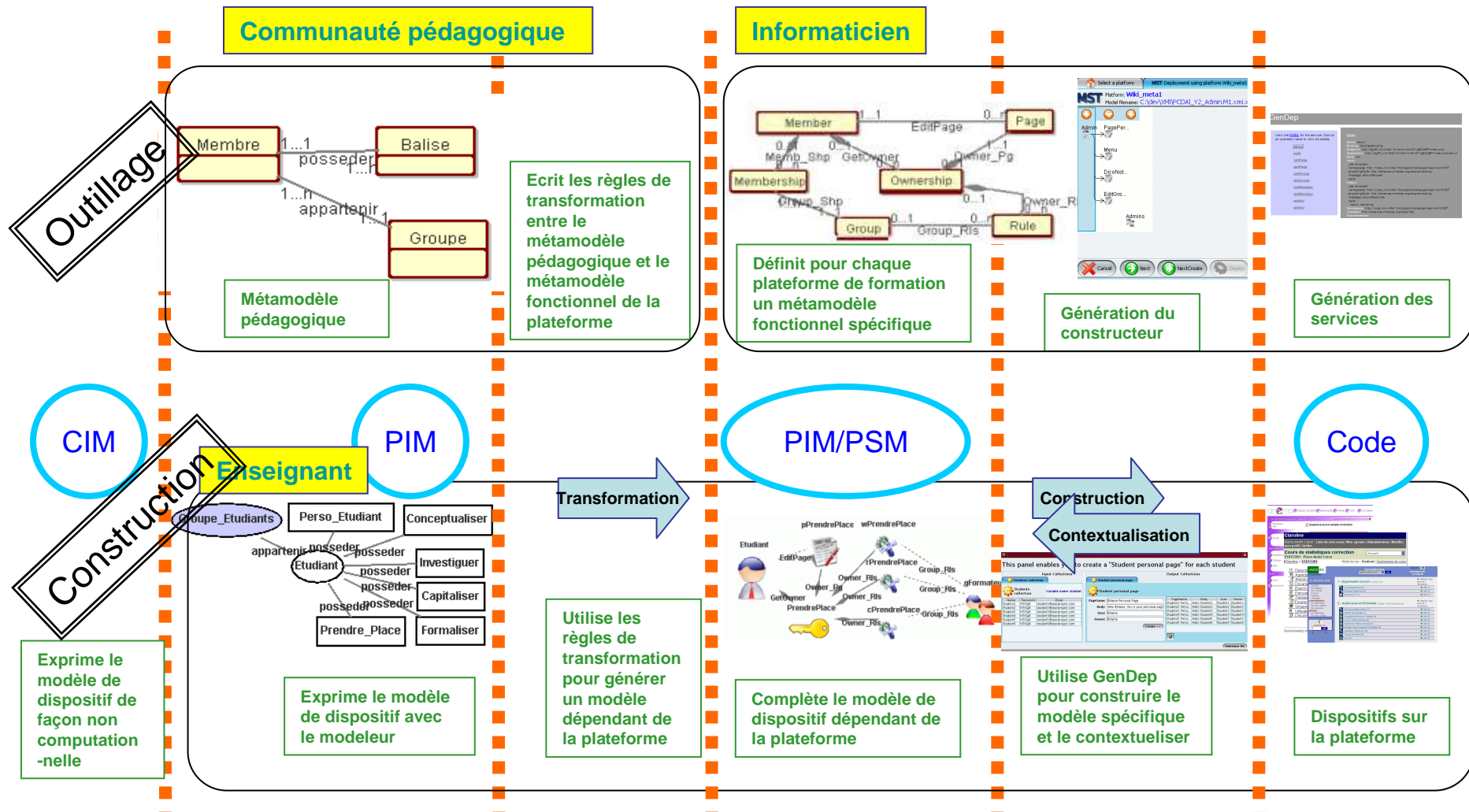


Figure 42 Processus de concrétisation d'un modèle de dispositif, en utilisant des transformations de métamodèles

Finalisation de l'atelier Outillage

Pour simplifier la compréhension de cet atelier, nous l'avons scindé en deux parties. Nous décrivons, dans un premier temps, la partie spécifique à la plateforme de formation (partie de droite sur la Figure 42), puis nous décrivons la partie métier de cet atelier (partie de gauche sur la Figure 42)

Cette première partie de l'atelier **Outillage** est mise en œuvre par un informaticien, elle est caractérisée par la définition conjointe des services encapsulant l'API de la plateforme et des différentes facettes composant le métamodèle fonctionnel de la plateforme. Le cycle de vie de cette partie de l'atelier **Outillage** est synchronisé sur celui de la plateforme de formation. Lors de l'étude menée sur la plateforme Ganesha, cette partie a été mise en œuvre en collaboration avec les développeurs de la plateforme de formation (cf. 11.2).

La seconde partie de l'atelier **Outillage** nécessite l'aide de la communauté pédagogique. Ce terme, un peu flou, désigne les différentes compétences qu'une petite communauté d'enseignants pratiquant le bricolage pédagogique, peut mettre en œuvre. Cet atelier a été expérimenté dans le cadre du projet PCDAI (cf 11.3). Pour ce projet une équipe d'enseignant a défini un vocabulaire commun ainsi que des concepts permettant d'encadrer la création de dispositifs pédagogiques (PCDAI, 2006). Des modèles de dispositifs ont été produits et construits manuellement sur des plateformes de formation (applications Accel (D'Halluin & Delache, 2005a), PostNuke (Varga, 2006), et WikiniMST (Caron et al., 2006b)). A partir de ces exemples, plusieurs réunions ont permis de définir le métamodèle pédagogique de l'équipe pour le projet concernant l'application WikiniMST Des transformations permettant de passer de ce métamodèle au métamodèle fonctionnel de la plateforme ont été réalisées. Pour achever la réalisation de cet atelier un modelleur a été défini. Ce modelleur dispose de deux métamodèles (pédagogique et fonctionnel) ainsi que de règles de transformations automatiques permettant de définir un modèle spécifique à la plateforme de formation à partir de la définition d'un modèle pédagogique. Nous verrons, dans le chapitre 10, le détail de réalisation de ce modelleur.

Finalisation de l'atelier Construction

L'atelier **Construction** est destiné aux enseignants. Il repose sur l'utilisation du modelleur et du constructeur spécifiques définis lors de l'atelier "**Outillage**". Un enseignant modélise son dispositif au travers des termes et des concepts propres à sa communauté pédagogique. Il transforme et complète ce dispositif en un modèle de dispositif pouvant être construit automatiquement sur la plateforme.

En guise de conclusion

Cette méthode par transformation est intéressante dans la mesure où elle encourage des processus réflexifs au sein des ateliers. Nous décrivons dans le chapitre 11 l'étude menée dans le cadre du projet PCDAI. Lors de la première année d'étude, cette méthode a permis de fixer les différents métamodèles à utiliser, et par l'élaboration des règles de transformation elle a permis d'exprimer les relations que les deux métamodèles entretiennent l'un par rapport à l'autre.

Un des bénéfices de la méthode par transformation est donc de permettre un ajustement souple des différents métamodèles.
--

Cependant cette solution s'est révélée être très difficile à utiliser pour des enseignants, car elle nécessitait la définition d'un modèle pédagogique simple et une contextualisation sur un modèle fonctionnel complexe généré automatiquement (cf. Figure 43).

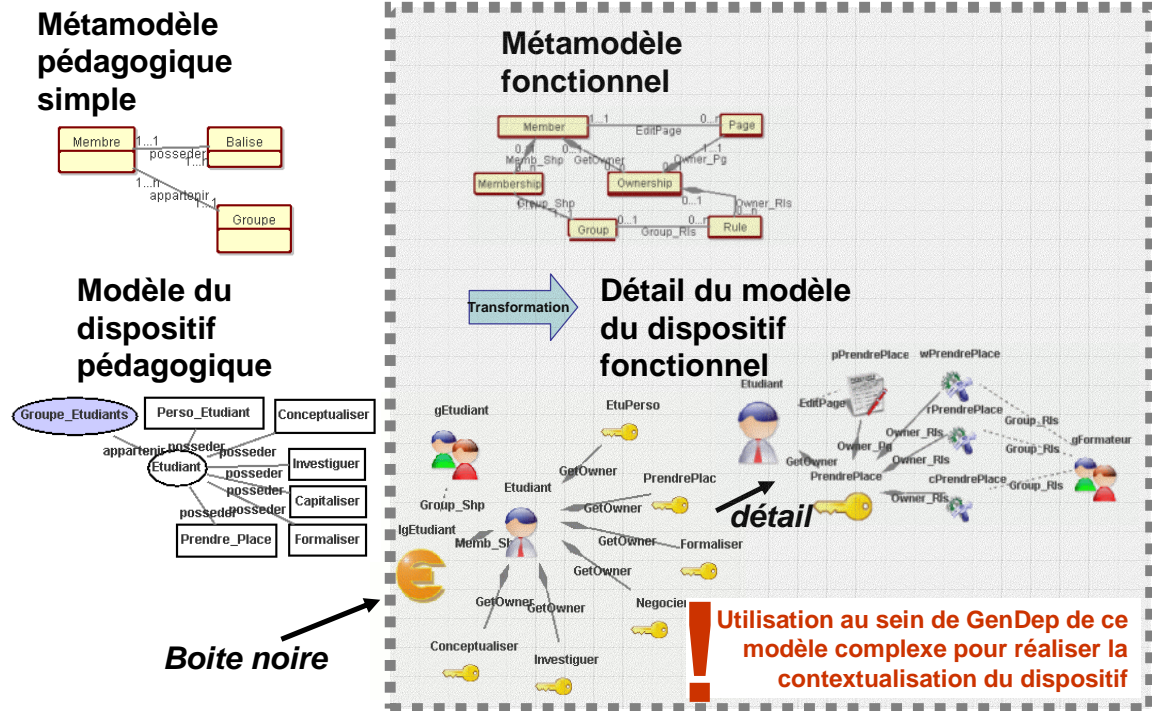


Figure 43 Problème posé, par l'approche par transformation

Aussi avons-nous poursuivi notre recherche pour définir un métamodèle unique fusionné des deux métamodèles précédents. Cette fusion est l'objet de la deuxième méthode, celle-ci a été expérimentée pour le projet PCDAI dans le cadre d'une deuxième année d'étude.

Deuxième méthode : proposition exploitant les fusions de métamodèles.

La méthode par fusion est décrite au chapitre 6 (section 6.5) : elle repose sur des mécanismes de fusion, que notre plateforme de modélisation n'implémente pas. Notre plateforme permet certes de décrire différents aspects d'un modèle en définissant pour un métamodèle donné un ensemble de vues partielles. Elle n'intègre donc pas à proprement parlé de mécanisme de fusion, dans la mesure où actuellement il n'est pas possible de fusionner deux métamodèles développés indépendamment. Pour définir un métamodèle global, il faut, au sein de ModX, le redéfinir à partir des métamodèles partiels. Le mécanisme de vue, de ModX, permet alors de choisir pour chaque métamodèle partiel une vue adaptée. Ces vues permettent dans le modèleur spécifique, de mener des activités de modélisation adaptées à chaque aspect de la tâche à modéliser.

La méthode par fusion illustrée dans la Figure 44 présente l'avantage de permettre une contextualisation directe du modèle défini par l'enseignant. Pour celui-ci, l'atelier de modélisation est plus simple à utiliser. Les différentes étapes de métamodélisation qui ont conduit à la définition d'un métamodèle global sont ainsi cachées à l'utilisateur. Un des intérêts de la méthode par fusion est également de confronter les préoccupations pédagogiques aux préoccupations fonctionnelles et donc de faire évoluer la représentation de chacune. Cette méthode a été expérimentée dans le cadre de la deuxième phase de notre étude de cas portant sur l'application de type Web 2.0 WikiniMST (Cf. chapitre 11.3).

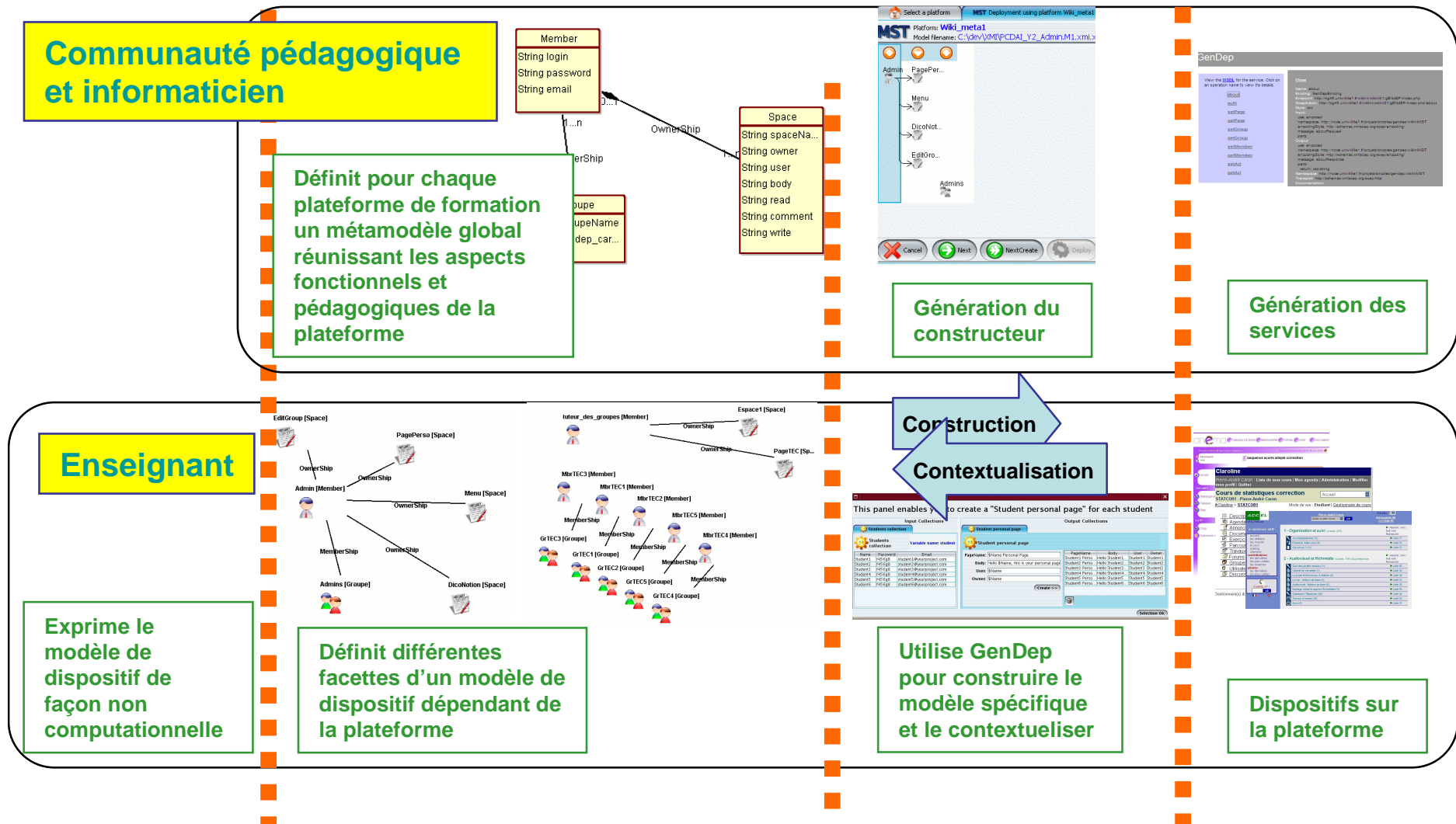


Figure 44 : Utilisation d'un métamodèle global

9.2.6 Conclusion à la problématique posée par la réalisation d'un modèle de plateforme

Nous avons montré dans les précédentes sections que générer un modèle de plateforme spécifique est réalisable et réaliste. Ce travail réputé essentiel (Estublier & Sanlaville, 2005), est nécessaire, dans le contexte de l'IDM, pour un développement non utopique de systèmes indépendamment des plateformes cibles. Nous suggérons une marche à suivre pour définir le modèle fonctionnel d'une plateforme. Cependant, un tel modèle doit, pour être utilisé, dans le cadre de notre proposition, permettre la définition d'un modèleur utilisable par des enseignants. Il doit donc être complété.

La difficulté réside dans le fait qu'un tel modèle ne doit pas uniquement représenter les fonctionnalités de la plateforme, mais aussi les utilisations de celles-ci. En d'autres termes le modèle ne doit pas décrire uniquement un artefact, mais il doit également décrire les schèmes d'utilisation qui conjugués à cet artefact en font un instrument (Béguin & Rabardel, 1999).

Pour parvenir à ce but deux méthodes ont été évoquées : la première propose d'établir des transformations semi automatique entre un métamodèle orienté utilisateur et un métamodèle orienté plateforme. La deuxième méthode propose de concevoir un métamodèle fusionnant l'activité métier propre à l'utilisateur/concepteur de modèle avec le modèle fonctionnel de la plateforme. Ce métamodèle capture les notions nécessaires à l'expression des intentions utilisateur lors de la modélisation d'un système. Il n'est pas donc ni neutre ni unique et son élaboration requiert non seulement des spécialistes de la plateforme mais également des spécialistes des usages de cette plateforme.

Quelque soit la méthode mise en œuvre, notre proposition implique la définition d'un modèleur spécifique paramétré par le ou les métamodèles définis et reposant pour sa relation de conformité sur une implémentation du MOF. Le détail de la réalisation et de l'usage de ce modèleur sera relaté dans le chapitre 10. Conceptuellement, et de façon très simplifiée, la définition d'un modèle de plateforme, va se traduire en un métamodèle de langage au sein de ModX. Ce métamodèle permet la mise en œuvre de la chaîne de modélisation illustrée par la Figure 45.

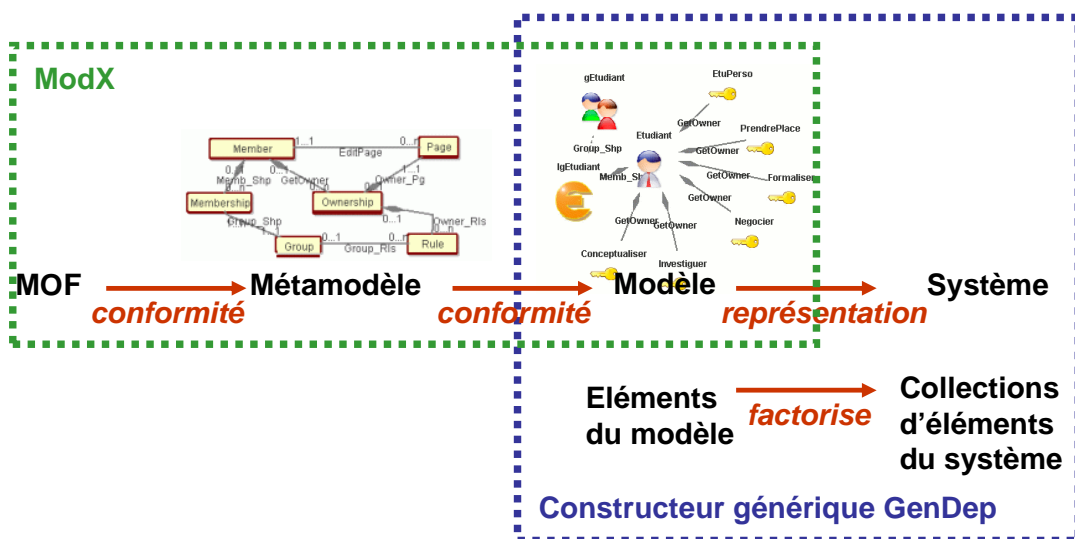


Figure 45 Différent niveaux de modélisation

ModX (encadrée sur la figure) permet la définition d'un modèleur spécifique dont le métamodèle est conforme au MOF. Grâce à ce modèleur il est possible de définir des modèles de dispositif eux même conformes au métamodèle paramétrant le modèleur. L'objet de la section suivante est d'explicitier les mécanismes qui permettent à partir d'un modèle de construire les éléments constituant le système qu'il représente sur la plateforme. Sur la Figure 45 cette problématique est illustrée par les mots "**représente**" et "**factorise**", il est mise en œuvre au sein de notre environnement de construction GenDep. Un modèle "**représente**" un système: quel sens informatique donner à ce mot dans un contexte qui ne conduit pas à de la génération de code? Un modèle "**factorise**" une collection d'élément, quel sens informatique donner à ce mot?

Ces questions constituent la problématique IDM abordée par la section 9.3, cette problématique concerne de façon plus générale la construction d'un système sur la plateforme à partir du modèle le décrivant. Notre proposition pour aborder cette problématique repose sur l'établissement d'un mécanisme de factorisation et sur la conception pour chaque plateforme de formation d'un constructeur spécifique. Dans la section suivante (9.3), nous présentons les difficultés conceptuelles entraînées par la réalisation d'un tel constructeur. Notre proposition, **GenDep** : un générateur de constructeur spécifique, est détaillée dans cette section.

9.3 Construire un modèle sur une plateforme, approche conceptuelle.

9.3.1 Introduction

Notre démarche suit le processus de création d'un DSL dont les étapes sont : (Bhanot et al., 2005) (Tolvanen, 2006) (cf chapitre 6 : opérationnalisation)

- Isoler les abstractions de la plateforme applicative, étudier son fonctionnement.
- Spécifier les concepts du langage et leurs règles (créer le métamodèle du domaine).
- Créer une représentation graphique associée ce métamodèle (notation).
- Définir les générateurs de code.

Il s'agit donc maintenant de définir pour chaque plateforme un générateur de code adapté. Or la construction d'un système à partir de son modèle n'est pas, dans notre domaine d'application, une génération de code mais l'enrichissement dynamique d'une application qui s'exécute. Ce type de construction nécessite de relier un modèle au système qu'il décrit et donc de pouvoir définir un lien informatique entre la relation de conformité (modèle/métamodèle) et la cohérence d'un système avec la plateforme qui l'héberge.

Nous avons vu dans la section précédente que notre problématique IDM peut être affinée dans la mesure où relier un modèle à un système dans le contexte d'une plateforme pose deux problèmes.

- Comment mettre en œuvre le mécanisme de factorisation qu'un modèle recèle, sur un plateforme?
- Comment lier informatiquement les instances générées à partir d'un modèle avec les éléments d'une plateforme?

9.3.2 Choix conceptuels pour réaliser un générateur de constructeurs spécifiques?

Ces problèmes conceptuels sont à mettre en relation avec d'autres problèmes spécifiques à notre domaine d'application : les EIAHs. Celui-ci, nous l'avons vu chapitre 8, est caractérisé

par un grand nombre de plateformes et l'absence de mécanisme de génération. La spécificité de notre domaine d'application, les EIAH, entraîne plusieurs conséquences pour l'élaboration de notre proposition :

Il faut mettre en place un processus de capitalisation permettant de minimiser le travail d'implémentation. Cette caractéristique dicte notre choix de réaliser un générateur de constructeur spécifique de systèmes.

La phase de génération des éléments constitutifs du système doit être accompagnée d'un dialogue permettant de faciliter leur intégration (contextualisation, résolution de conflits...) avec les éléments préexistants dans le système en cours d'exécution sur la plateforme. Cette caractéristique nécessite de mettre en place un mécanisme simple de dialogue entre une application externe (le constructeur spécifique) et la plateforme. Nous avons fait le choix de favoriser ces interactions par la mise en place d'une relation client/serveur. Dans le domaine des applications Web, nous avons privilégié une solution reposant sur les Services Web.

Notre proposition repose ainsi sur la construction d'un générateur spécifique, celui-ci aura essentiellement deux tâches à réaliser, **mettre en œuvre** le processus de factorisation qu'un modèle recèle, et **relier les instances** ainsi générées à la construction d'éléments composant un système dans le contexte d'une plateforme.

9.3.3 Construire un générateur de constructeur spécifique

Vue générale

Notre proposition et notre problématique IDM est illustrée par la Figure 46. Cette sous-section explicite chaque choix, établis pour la réalisation de ce générateur

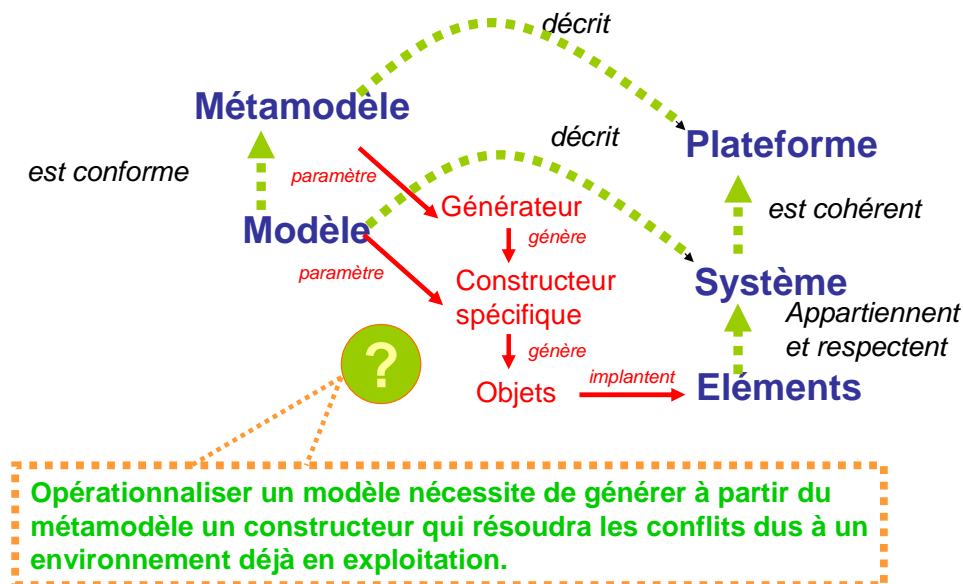


Figure 46 Problématique et proposition pour relier modèle et système.

Notre démarche consiste à produire un générateur de constructeur spécifique. Conceptuellement un tel générateur traduit une spécification en une application, et suit le schéma suivant (Cleaveland, 1988) :

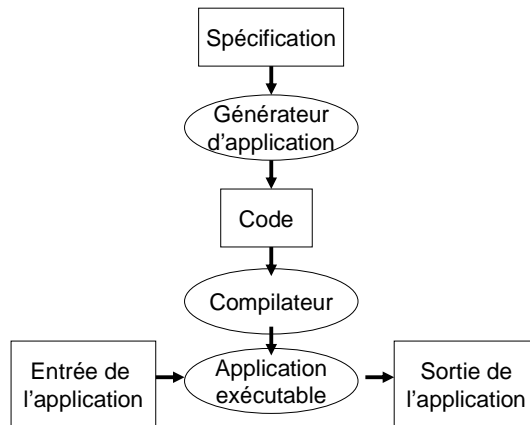


Figure 47 Construire un générateur d'application

La solution conceptuelle que nous proposons s’inspire de ce schéma :

Notre générateur est alimenté par le métamodèle décrivant la plateforme. Techniquement nous verrons que cette spécification est complétée par la description des services disponibles sur la plateforme. Cette spécification est également complétée par une interface abstraite encapsulant le comportement nécessaire à une application permettant de générer les éléments d’un système.

Notre générateur produit, à partir de ces spécifications, du code que nous compilons pour implémenter un constructeur spécifique. Ce constructeur, spécifique à la plateforme, permet de générer à partir de la lecture du modèle des objets capables de se déployer sur la plateforme (via ses services) et de construire ainsi le système décrit par le modèle.

Détail de ce générateur

La réalisation d'un générateur de code nécessite de minimiser au maximum les implémentations de bas niveau (cf. chapitre 6.6). Une partie de notre générateur est donc construite à partir du métamodèle du langage, une autre partie est directement déduite des interfaces décrivant les services disponibles sur la plateforme. La Figure 48 résume notre proposition conceptuelle d'ingénierie.

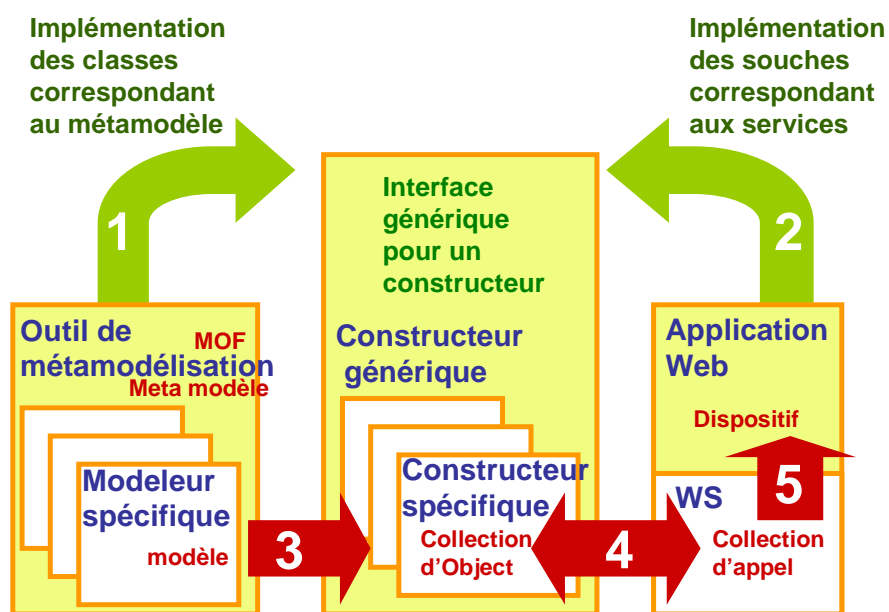


Figure 48 Proposition conceptuelle de générateur de constructeur spécifique.

Les 5 points indiqués dans cette figure peuvent être explicités dans le cadre des deux ateliers qui composent notre infrastructure :

Dans le cadre de l'atelier **Outillage** destiné à l'informaticien.

Point 1 : à partir du métamodèle défini dans notre environnement de modélisation, il est possible de générer automatiquement des squelettes de classe dont les objets construits à l'atelier **Construction** seront les instances.

Point 2 : à partir de l'interface représentant les services encapsulant les fonctionnalités de la plateforme, il est possible de générer des classes permettant d'établir un dialogue avec cette dernière, cette tâche peut être confiée à des outils traditionnels de gestion de Web Services.

Ce travail est complété par les interfaces génériques mises à disposition par le moteur de notre générateur. Ces deux points permettent très simplement de créer un constructeur adapté à une plateforme de formation.

Dans le cadre de l'atelier **Construction** destiné à l'enseignant :

Point 3 : à partir d'un modèle conforme au métamodèle de langage il est possible de paramétrer le constructeur spécifique.

A partir de la lecture du métamodèle de langage le constructeur spécifique produit une interface adaptée au modèle.

Point 4 : chaque élément du modèle est potentiellement la source d'une collection d'objets instances des classes du métamodèle. Pour construire ou pour contextualiser ces collections d'objets avec les instances s'exécutant sur la plateforme, un dialogue utilisant les services disponibles est possible. Chaque objet communique ainsi avec la plateforme de formation via des appels de services.

Point 5 : le greffon de service construit sur la plateforme les éléments constituant le système décrit par le modèle.

Le choix de réaliser un générateur de constructeur spécifique a des conséquences sur les concepts IDM manipulés par notre recherche, l'objet de la sous-section suivante est de les clarifier.

9.3.4 Décrire et factoriser une collection d'éléments?

Dans cette sous-section nous explicitons comment un élément du modèle peut décrire une collection d'éléments d'un système. Nous proposons pour cela de définir un mécanisme de factorisation. Ce mécanisme de factorisation est rendu possible par notre proposition conceptuelle. En effet la réalisation du constructeur spécifique permettra d'interpréter et de mettre en œuvre ce mécanisme.

Créer un type d'étudiants, dans un dispositif pédagogique, pour indiquer le type d'étudiants concernés par un cours particulier est rarement possible au sein d'une plateforme spécifique (tout comme créer un type de cours, de ressources...). Cette capacité à définir des types n'est donc pas présente dans les métamodèles que nous décrivons, or c'est ce qui permet de factoriser les actions lors de la construction d'un dispositif. Nous proposons de mettre en œuvre un mécanisme simple de factorisation, en considérant que chaque élément d'un modèle est capable de décrire une collection d'éléments de la plateforme. Ce mécanisme de profilage repose sur des indications interprétées par le constructeur spécifique que nous réalisons pour chaque plateforme. Ces indications nécessitent de définir des mécanismes traduisant ces factorisations, au sein du constructeur chargé d'interpréter le modèle et de générer les collections d'éléments qu'il décrit. Ces indications sont de deux sortes :

Les premières permettent la génération d'une collection d'éléments à partir de la définition d'un élément du modèle. Ce travail peut être mené en indiquant un nombre d'éléments possibles (cardinalité) et des patterns de valeurs pour les attributs. Ce mécanisme de 'profil' est plus général que le mécanisme de typage, fréquent dans les plateformes plus 'généralistes' (structures, classes, composant...), nous verrons dans le chapitre suivant le détail de son implémentation.

Les secondes permettent la génération automatique de collections d'éléments du système à partir de la lecture d'un lien reliant un élément du modèle à un autre. Cette génération nécessite de pouvoir définir dans le modèle des cardinalités. Ces cardinalités permettent d'indiquer par exemple que la génération de soixante étudiants entraîne la génération de soixante pages qui leur sont associées.

9.3.5 Conséquences IDM, deux interprétations

La chaîne de modélisation explicitée dans le chapitre 6 (modèle CIM, modèle PIM, modèle PSM) donne finalement lieu à deux interprétations. Ces interprétations sont motivées par le caractère spécifique de notre problématique liée à notre domaine d'application nécessitant de définir informatiquement les relations "décrire" et "factoriser" (cf. 9.1.3)

La première de ces interprétations consiste à interpréter le métamodèle fonctionnel de la plateforme selon le paradigme Classe / Objet (Manolescu & Johnson, 1999). Cette interprétation simple permet le paramétrage du générateur de constructeur spécifique. La description des classes dans le métamodèle permet ainsi de générer automatiquement, via l'emploi de tâches programmées lors de notre recherche, le squelette des classes nécessaires au constructeur spécifique. La Figure 49 et la Figure 50 illustrent cette correspondance.

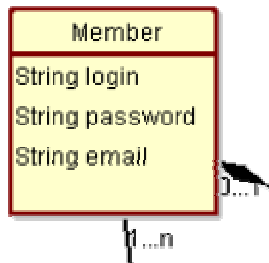


Figure 49 Classe Member

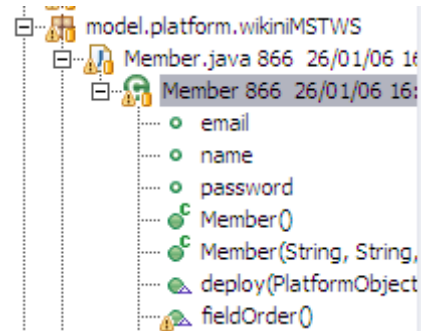


Figure 50 génération des squelettes de classe

La seconde interprétation repose sur un paradigme IDM plus général. Selon ce paradigme la relation qui lit le modèle au métamodèle est une relation de conformité MOF : le modèle décrit le système que l'on désire construire. C'est à cette relation de description que nous donnons un sens opérationnalisable informatiquement en décidant que :

Les masques des attributs décrits dans un modèle peuvent être interprétés par le constructeur spécifique de la plateforme pour attribuer des valeurs aux attributs des objets constitués en collection d'instances des squelettes des classes construites précédemment.

5 Les indications de profilage d'un élément du modèle sont interprétées par le constructeur spécifique, celui-ci établit un dialogue avec la plateforme et construit pour chaque élément une collection d'objet.

6 Chaque objet de chaque collection construit, par des appels de service, des éléments dans le contexte de la plateforme via le greffon de service Web.

La Figure 51 présente la construction et l'usage de GenDep, notre générateur de constructeur spécifique. Nous abordons dans le chapitre 10 suivant, les détails de la réalisation technique d'un modeleur spécifique et d'un constructeur spécifique.

chapitre 10 Réalisation technique : Modeleurs et Constructeurs spécifiques de dispositifs

Introduction

Nous proposons une adaptation de l'IDM au contexte des EIAHs. Pour réaliser cette adaptation, un processus de "*construction*" a été défini, ce processus apporte des solutions aux différents verrous scientifiques que nos travaux ont abordés dans le domaine spécifique des EIAHs.

Le premier verrou concerne "*l'opérationnalisation et la communication avec des plateformes de formation*". Ce verrou a été abordé au travers de la définition de services d'implantation sur une plateforme. Une déclinaison de cette solution s'est faite sous forme d'un greffon composé de service Web encapsulant l'API de la plateforme visée et proposant les services demandés.

Le second verrou constitue le verrou principal abordé par cette thèse, il concerne "*l'adaptation d'un dispositif pensé par un enseignant à un dispositif dans le contexte d'une plateforme*". Pour aborder ce verrou nous proposons deux ateliers de génie logiciel :

- Le premier atelier, à destination de l'ingénieur pédagogique, permet par des outils de métamodélisation de définir un environnement de modélisation utilisable par les enseignants.
- Le second atelier, à destination des enseignants, consiste en un outil de modélisation et de construction spécifique à une plateforme, mettant en œuvre des indications de profilage et utilisant des transformations automatiques facilitant le travail de modélisation et l'opérationnalisation de ces modèles sur une plateforme.

Avertissements

L'objet de ce chapitre est d'explicitier les techniques utilisées pour concevoir ces deux ateliers. Ce chapitre est donc essentiellement technique, il constitue un guide de bonnes pratiques pour adresser une plateforme de formation avec notre ingénierie. Les lecteurs ne désirant pas entrer dans les détails de cette ingénierie peuvent directement aborder le chapitre 11 qui relate les études de cas que nous avons menées.

Trois solutions différentes pour adresser une plateforme de formation ont été expérimentées, ces solutions sont relatées dans la section 9.2.5. La première repose sur un métamodèle pivot transformé en un métamodèle spécifique, la deuxième sur deux métamodèles (pédagogique et fonctionnel) la troisième sur un métamodèle fusionné. Nous présentons ici la deuxième proposition, celle-ci emploie toutes les techniques abordées par les autres solutions. Cette méthode a fait l'objet d'une étude relatée dans (Caron et al., 2006b).

Au cours de notre recherche chaque étude menée au cours de notre recherche a été l'occasion de travailler sur un point précis de notre infrastructure. C'est pourquoi nous utiliserons dans les sections qui suivent des exemples différents pour en illustrer le détail.

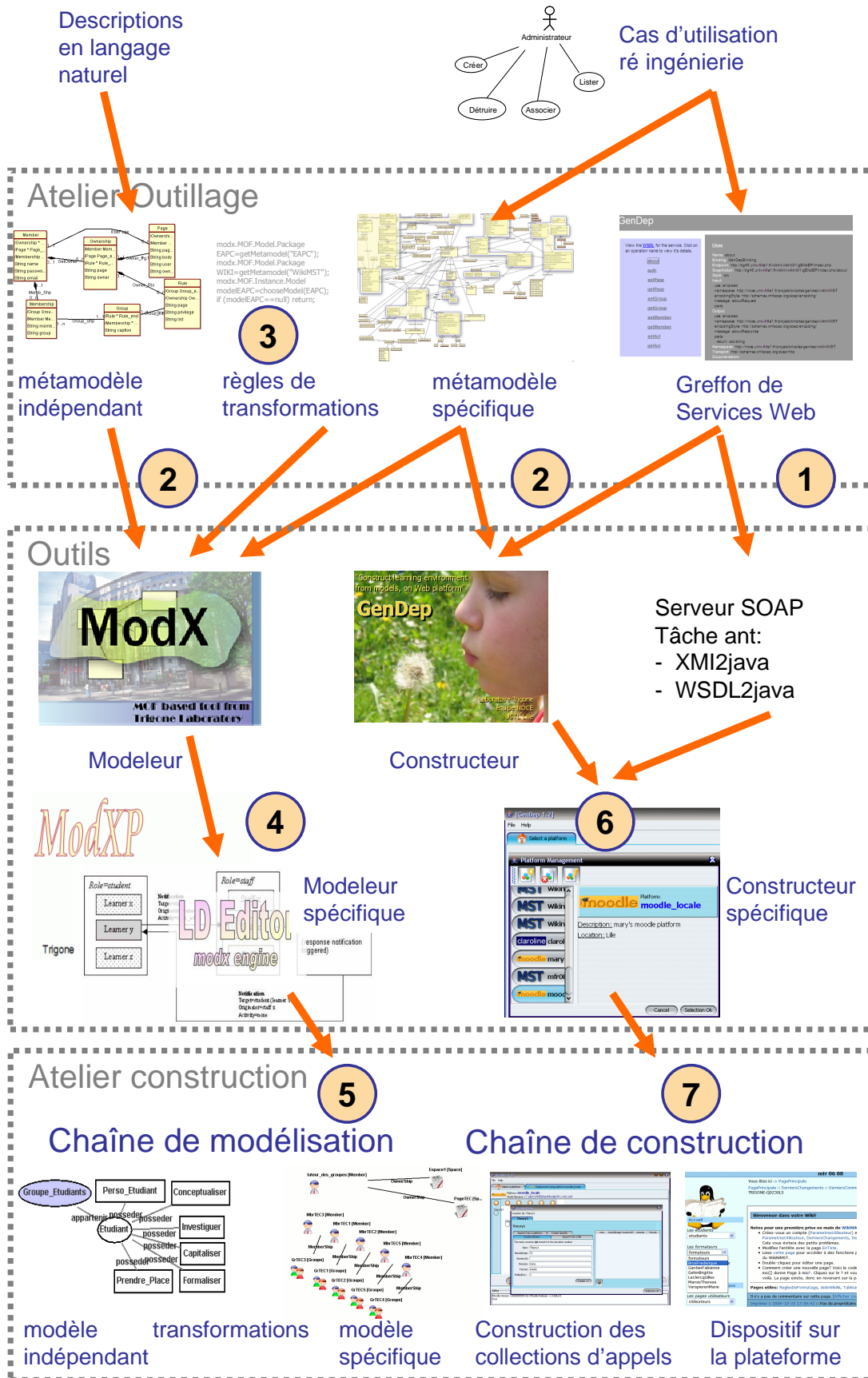


Figure 52 Réalisation technique pour adresser une plateforme de formation

La Figure 52 illustre notre proposition d'ingénierie, elle décrit les différentes étapes à réaliser pour adresser une plateforme avec notre infrastructure. La Figure 53 constitue tout autant la légende de la figure précédente qu'un fil conducteur pour la lecture de ce chapitre.

1. Implémenter, s'il n'existe pas un serveur, de Web Services
 - a. Définir un scénario d'utilisation
 - b. Définir les cas d'utilisation,
 - c. Mener une réingénierie de la plateforme de formation (étudier base de données, documentation, IHM, code sources.
 - d. Définir les types complexes et les services associés.
 - e. Réaliser un premier service about et un service d'authentification
 - f. Réaliser les services de créations, listage, liaisons, destruction
 - g. Tester ces services (par exemple avec un client Web)
2. A partir de la description textuelle, réaliser le métamodèle indépendant de la plateforme. Parallèlement, à partir des services et des types complexes, réaliser le métamodèle spécifique
 - a. Définir les classes et les attributs pertinents
 - b. Définir les associations (cardinalité et type d'association)
 - c. Organisation par vue, aspect graphique et choix des noms
3. Définir les transformations
 - a. Dialogue de création
 - b. Sens de transformation
 - c. Création du modèle cible
 - d. Création des vues
 - e. Placement dans les vues
4. Génération d'un modeleur spécifique
 - a. Exemple d'usage
 - b. Chargement des métamodèles et des scripts de transformations
5. Tester la chaîne d'édition en créant et transformant un modèle
 - a. Création du modèle indépendant
 - b. Transformation du modèle indépendant en un modèle spécifique
 - c. Ecriture des attributs
6. Création du constructeur spécifique
 - a. Création des packages dans "**model.platform**" et "**model.connection**"
 - b. Lancement de la tâche "**Ant**" construction du package spécifique "**platform**" (à partir du métamodèle) "**xmi2java**"
 - c. Lancement de la tâche **Ant** construction du package spécifique "**connection**" (à partir du Web Services) "**wsdl2java**"
 - d. Ecriture de la classe "**PlatformSpecific**", de la classe "**ExecuteSpecific**", des fichiers de configuration type et plateforme
 - e. Test de connexion et d'authentification sans déploiement.
 - f. Ecriture des méthodes "**deploys**" et "**fetchs**" de chaque objet non abstract
7. Tester la chaîne de construction sur divers exemples

Figure 53 Description du schéma de production d'une infrastructure adaptée à une plateforme spécifique.

A l'aide du modeleur généré (4) il est possible de définir pour l'atelier "**Construction**", un modèle automatisable de dispositif. Cette modélisation manuelle est effectuée à partir de la description en langage naturel du dispositif envisagé. Ce modèle de dispositif peut alors être transformé, en appliquant les règles de transformations définies à l'atelier "**Outillage**" (3), en un modèle spécifique à une plateforme de formation (5). Un tel modèle est défini dans la sémantique de la plateforme de formation, il est constructible sur cette plateforme. Ce mécanisme d'opérationnalisation est décrit ci-dessous.

Une abstraction des fonctionnalités de construction la plateforme de formation est réalisée, cette abstraction conduit à la définition d'une interface de services réalisée sous forme d'un greffon de services Web (1). En utilisant les services proposés par la plateforme, il est possible de déduire le métamodèle spécifique à la plateforme de formation (2). Celui-ci permet de définir le constructeur spécifique à la plateforme de formation (6). L'opérationnalisation d'un modèle est effectuée par ce constructeur spécifique. Celui-ci parcourt le modèle dans le sens indiqué par le métamodèle (7). Pour chaque élément de modèle présentant des éléments de profilage, il interroge l'utilisateur. Tous ces éléments, virtuels au départ, sont ensuite créés sur l'instance de la plateforme, désignée via une url, au travers d'invocations d'appels de services Web.

10.1 Implémenter, s'il n'existe pas, un serveur de Web Services

Notre ingénierie repose sur l'utilisation d'un greffon de service Web encapsulant l'API des applications ciblées. Pour illustrer cette section, concernant l'implémentation d'un serveur de services Web nous utiliserons une étude de cas que nous avons réalisée sur la plateforme **Claroline**. Claroline est une plateforme Open Source, modulaire, d'apprentissage à distance et de travail collaboratif développée par l'Université de Louvain, (Claroline). L'étude que nous avons réalisée n'est pas encore achevée. Cependant les premiers travaux réalisés, nous semblent avoir valeur d'exemples et peuvent illustrer cette section.

10.1.1 Adresser une plateforme par des services Web

Certaines applications utilisables en formation fournissent déjà de tels services. Ainsi les blogs créés avec l'outil Blogger (hébergé sur les serveurs Google) permettent un accès à leurs fonctionnalités sous forme de services Web. L'API Google utilisable pour cet accès, définit des types de données spécifiques (Gdata) utilisables pour l'ensemble de Google Services. Via l'utilisation de ces services, une application Google est directement utilisable par notre infrastructure.

Dans le domaine plus spécifique des EIAHs, le site ELF (The E-Learning Framework) propose de référencer les initiatives permettant de spécifier les services utilisables pour adresser les plateformes de formation. Parmi les initiatives recensées, le consortium IMS Global learning développe de nombreuses spécifications permettant l'échange de donnée via des services Web. La spécification IMS Enterprise Service (IMS-ES) propose ainsi de décrire les services permettant d'échanger des données de type organisationnel. Une plateforme respectant une telle spécification d'échange serait ainsi facilement adressable par notre infrastructure.

L'initiative SWEET (Enterprise Web Services with Timetable Extensions for .NET) (SWEET) propose une approche dans le domaine du e-learning, d'applications Web via les Services Web. Ce kit de développement permet de mettre en œuvre la spécification IMS Enterprise Services dans le cadre d'échange de données (étudiants, administration, classes, groupes) entre applications. Ce kit propose des extensions pour permettre l'échange d'agendas et d'emploi du

temps. Cette initiative diffère de la notre dans la mesure où d'une part, elle ne repose pas sur l'utilisation de l'IDM pour générer efficacement des outils adaptés aux services définis et d'autre part, elle reste limitée au domaine des agendas et emploi du temps.

L'initiative Open Knowledge Initiative (OKI) a pour objectif de recenser, développer et promouvoir les communications entre applications dans le domaine des EIAHs. OKI développe et publie les " Open Service Interface Definitions" (OSIDs). Ces interfaces réalisées en Java et PHP définissent les objets et services utilisables dans le cadre de l'interopérabilité des plateformes de formations. Des plateformes implémentant ces spécifications peuvent aisément être adressable par notre infrastructure.

Les exemples précédents décrivent assez bien la multiplicité des solutions reposant sur des architectures orientées service et adressant le problème de l'interopérabilité des plateforme de formation. Obtenir une spécification unique des services et objets utilisables dans le domaine des EIAHs est illusoire et sans doute à terme trop réducteur.

10.1.2 Implémenter un serveur de Web Service

Parce qu'il définit un standard couramment utilisé sur le Web, nous avons décidé d'implémenter sous forme de Web Services les services encapsulant les fonctionnalités de construction des plateformes de formation considérées. "*Web Services are a useful technology for connecting a learning environment to organizational data and processes and for integrating various kinds of common functionality*" (Wilson & Lowry, 2001). Dans le cadre des études de cas que nous avons menées (cf. chapitre 11), trois des principales plateformes Open Source de E-learning (Dooley, 2006), Ganesha, Claroline, Moodle ont été abordées. Notre travail a été également réalisé sur les applications Web, WikiniMST et Elg.

Scénario

Le premier travail à réaliser, lorsque l'on veut adresser une plateforme de formation par des Web services consiste, à définir un scénario simple d'utilisation. Le scénario suivant a par exemple été établi pour la plateforme Claroline.

Mr Le Pallec, enseignant, tuteur, et administrateur de la plateforme Claroline souhaite mettre en place un dispositif sur celle-ci.

Il commence par créer un utilisateur (Mr Laurent Grisoni) ayant les droits de gestionnaire de cours.

Puis il décide de créer deux classes d'utilisateurs, l'une dédiée aux étudiants d' « ima2i », l'autre aux étudiants d' « ima3i ». Il ajoute ensuite l'ensemble des utilisateurs, qu'il répartit dans les classes correspondantes. Dans la classe « ima2i » nous pouvons trouver Ismaël, Patrick... et dans la classe « ima3i » sont présents Benoit, Samir...

Vient alors la création des cours : il crée d'abord une catégorie principale « Imagerie » puis il crée les cours suivants : 'Synthèse d'images' et 'Interface et réalité virtuelle' qu'il place dans la catégorie élaborée précédemment.

Mr Laurent Grisoni est gestionnaire des deux cours.

Cas d'utilisation

On définit ensuite différents cas d'utilisation de la plateforme. Pour notre exemple, trois cas ont été étudiés.

Cas d'utilisation 1 (cf. Figure 54).

Un utilisateur s'inscrit à un cours. Ce cours est administré par l'enseignant. Ce cours dépend d'une catégorie.

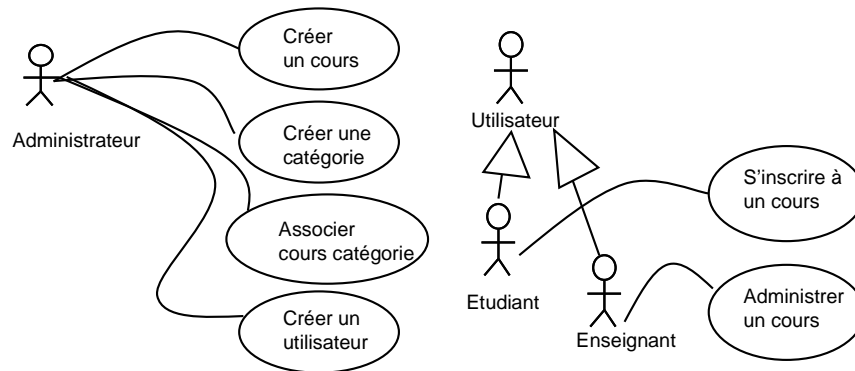


Figure 54 Cas d'utilisation simple

Cas d'utilisation 2 (cf. Figure 55, Figure 56).

Un utilisateur peut appartenir à une classe. Celle-ci peut être affectée à un cours, qui dépend d'une catégorie. Un enseignant peut aussi s'inscrire à un cours en tant qu'élève.

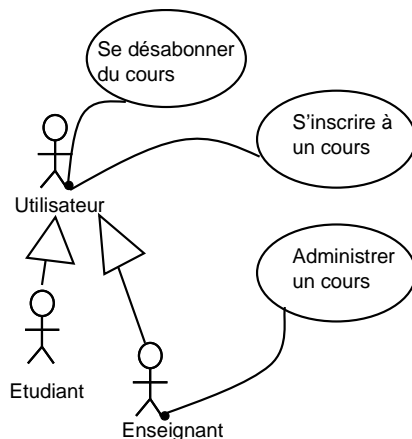


Figure 55 Cas d'utilisation 2 acteur : user

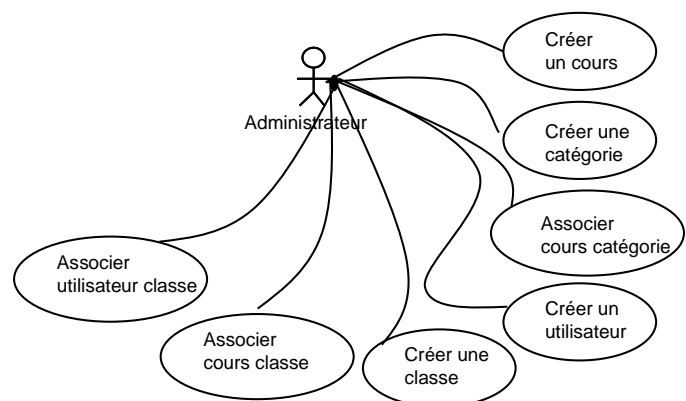


Figure 56 Cas d'utilisation 2 acteur : Admin

Ce cas d'utilisation complète et précise le précédent.

Cas d'utilisation 3 (cf. Figure 57, Figure 58, Figure 59, Figure 60).

Un utilisateur peut être enregistré dans plusieurs classes en même temps. Il peut être directement mis dans un cours sans être associé à une classe. Ces classes peuvent être inscrites à plusieurs cours, mais chacun de ces cours ne dépend que d'une catégorie. Les cas d'utilisations qui suivent, précisent les actions que l'administrateur de la plateforme peut mener.

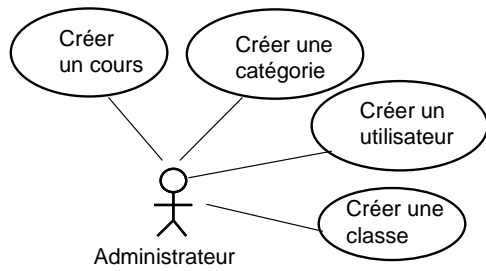


Figure 57 Créer

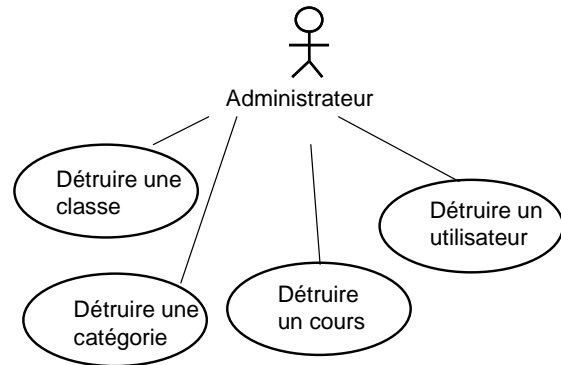


Figure 58 Détruire

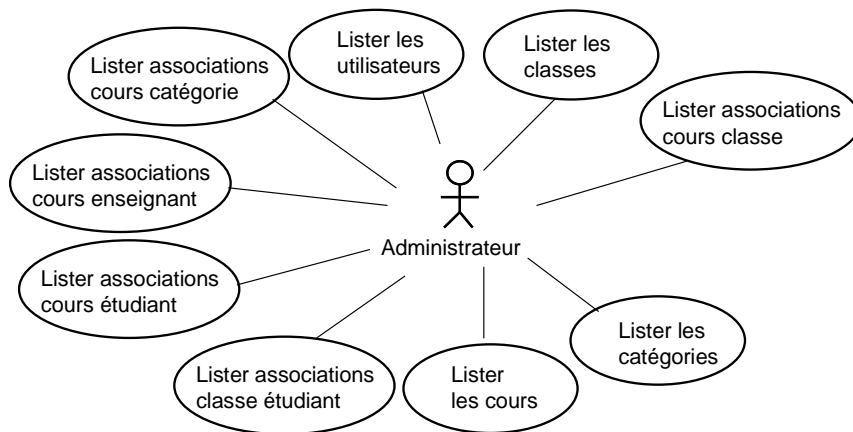


Figure 59 Lister

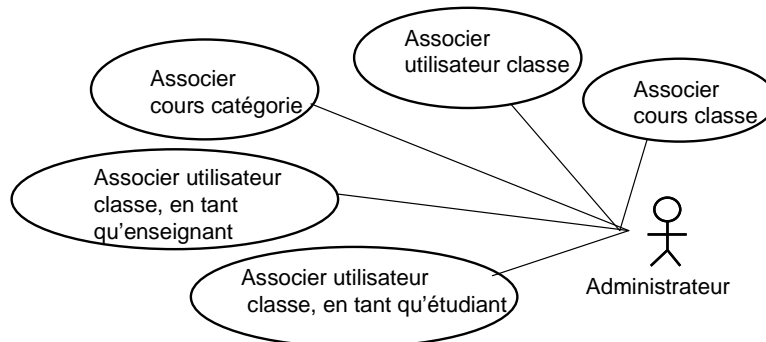


Figure 60 Associer

C'est quatre aspects peuvent être résumés par le cas d'utilisation simplifié suivant. Celui-ci explicite les actions que peut mener l'administrateur (cf. Figure 61).

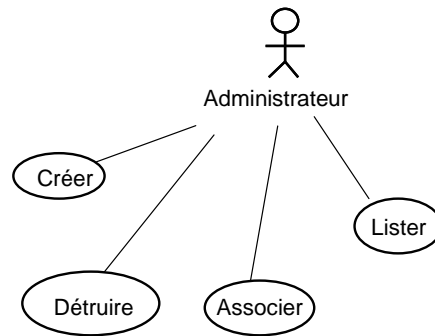


Figure 61 Quatre types d'activité d'administration

Quelques remarques portant sur une approche différente des fonctionnalités d'une plateforme de formation.

Les cas d'utilisations précédents sont adaptés à l'usage de la plateforme de formation Claroline et non à l'usage de cette plateforme via notre infrastructure. Or notre infrastructure n'a pas pour ambition de reprendre toutes les fonctionnalités de la plateforme, mais de pouvoir mettre en œuvres les plus importantes. A partir des cas d'utilisation, il faut déduire les cas d'utilisation de notre infrastructure. Les services que notre infrastructure utilise, peuvent être associés aux différents rôles définis par la plateforme. Cependant dans cet exemple nous avons privilégié un utilisateur unique : l'administrateur. Des catégories non décrites précédemment ont ainsi été implémentées; elles concernent les aspects Paramétrage, Information et Modification (cf sous section suivante)

Sémantiquement, la construction d'un dispositif au sein d'une plateforme est différente de la construction d'un dispositif par des invocations de service. En effet, au sein d'une plateforme, les actions sont contextualisées par les actions précédemment menées. Sur une plateforme, il est possible de créer une catégorie et dans le contexte de cette catégorie il est possible de créer un cours, cette possibilité est offerte par la présence de session garantissant la prise en compte du contexte. Nous avons choisi d'implémenter nos services différemment, en choisissant de ne pas implémenter de mécanisme de session. Ce choix nous permet de mettre en œuvre des services indépendants respectant strictement une architecture orientée service et présentant une granularité fine (Beryozkin, 2003). Ce choix a des répercussions sur la définition des actions complexes par exemple : les actions consistant à **créer une catégorie** puis à **créer un cours dans le contexte de cette catégorie**, sont implémentées dans notre approche par trois services : **créer un cours, créer une catégorie, associer cours à catégorie**.

Ce changement d'approche, nécessite d'adapter les fonctionnalités de la plateforme. Pour la plateforme Claroline, l'étude de la base de données, de la documentation, de l'IHM, des codes sources a permis de mener des actions de réingénierie. Ainsi les fichiers suivants ont été clonés et modifiés pour les rendre compatibles avec notre approche :

Répertoire : /claroline/claroline/admin

- admincats2.php
- admincourses2.php
- adminusers2.php

Répertoire : /claroline/claroline/inc

- claro_init_global.inc2.php

Répertoire : /claroline/claroline/inc/lib

- admin2.lib.inc.php
- claroCourseBis.class.php
- class_user.php
- class2.lib.php
- course_category.lib.php
- course_home_bis.php
- courseDescription2.lib.php
- user_course.php
- user2.lib.php

Ecriture des services Web

A partir des cas d'utilisation et en adoptant les spécificités de l'approche orienté service (rappelées précédemment), en rajoutant les aspects décrit précédemment, il est possible de définir les services suivants (exemple de la plateforme Claroline).

<p>AUTHENTIFICATION : about authentification</p>	<p>LIEN inscrire un étudiant à un cours inscrire un enseignant à un cours inscrire un cours à une catégorie inscrire un étudiant à une classe inscrire une classe à un cours</p>
<p>CREATION : créer un nouvel utilisateur créer un nouveau cours créer une nouvelle catégorie créer une classe</p>	<p>INFORMATION : informations d'une catégorie informations d'un cours informations d'un user informations d'une classe</p>
<p>LISTE lister cours lister catégorie lister classes lister utilisateurs lister association cours classe lister association cours enseignants lister association cours étudiants lister association cours catégories</p>	<p>MODIFICATION : modification d'une catégorie modification d'un cours modification d'un user modification d'une classe</p>
<p>DESTRUCTION détruire un utilisateur détruire un cours détruire une catégorie détruire une classe</p>	<p>PARAMETRAGE paramétrer cours agenda paramétrer cours annonce paramétrer cours travaux paramétrer cours exercice paramétrer cours catégorie de forum paramétrer cours forum</p>

Définir les types complexes

Les types complexes déterminent les différents paramètres utilisables par les services. Ces types préfigurent les classes constituant un des métamodèles fonctionnels de la plateforme.

Par exemple le type complexe (cf. Figure 62)

```
$server->wsdl->addComplexType(
    'UserIn',
    'complexType',
    'struct',
    'all',
    '',
    array('lastname' => array ('name'=>'lastname', 'type'=>'xsd:string'),
        'firstname' => array ('name'=>'firstname', 'type'=>'xsd:string'),
        'username' => array ('name'=>'username', 'type'=>'xsd:string'),
        'password' => array ('name'=>'password', 'type'=>'xsd:string'),
        'isCourseCreator' => array ('name'=>'isCourseCreator', 'type'=>'xsd:string'),
        'isPlatformAdmin' => array ('name'=>'isPlatformAdmin', 'type'=>'xsd:string'))
);
```

Figure 62 Exemple de type complexe

...correspond dans le métamodèle à la classe **user**, (cf. Figure 63)

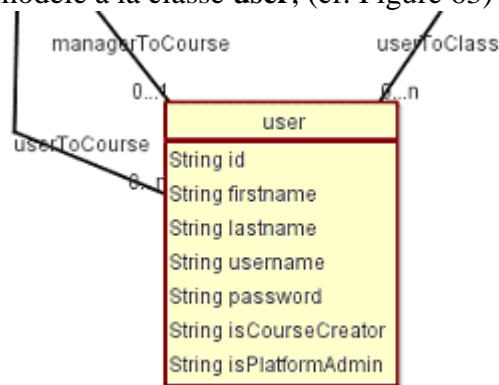


Figure 63 Exemple de classe définie par le Métamodèle

Une fois que ce travail réalisé, il faut appliquer un paradigme objet, à ces types complexes et les redéfinir en faisant apparaître les héritages possibles, les éléments de factorisation. Ce travail prépare la partie modélisation.

Après avoir écrit les services et les types complexes ceux-ci doivent être enregistrés auprès du serveur SOAP.

```
$serveur->register('createUser',
    array ('user_data' => 'tns :UserIn', 'auth_data' => 'tns :AuthDataIn'),
    array('return' => 'xsd:string'),
    $nameSpace);
```

On enregistre le service "**createUser**" auprès du serveur SOAP d'adresse \$nameSpace et de nom \$serveur. Ce service a pour paramètres d'entrée un UserIn et un AuthDataIn (types complexes), et retourne une chaîne de caractère. Ce travail est à réaliser pour chaque service.

Une lecture des différents services créés permet de réorganiser les services selon des aspects orientés objets. On voit alors se dessiner les classes du métamodèle fonctionnel de la plateforme. Les attributs de celles-ci correspondent aux champs des types complexes. Ces classes implémentent des méthodes de création, de destruction, d'information et de paramétrage. Les associations qui lient les classes entre elles correspondent aux services

recensés dans la partie Lien. Le détail du métamodèle fonctionnel de l'application Claroline est présenté Figure 64. Sur ce métamodèle, on voit en particulier que la classe "**User**" est associée de trois manières à la classe "**Course**". La première association est réalisée par l'intermédiaire de la classe "**Class**", un utilisateur peut être associé à une classe elle-même associée à un cours. Les deux autres associations "**UserToCourse**" et "**ManagerToCourse**" correspondent aux deux rôles qu'un utilisateur peut jouer au sein d'un cours.

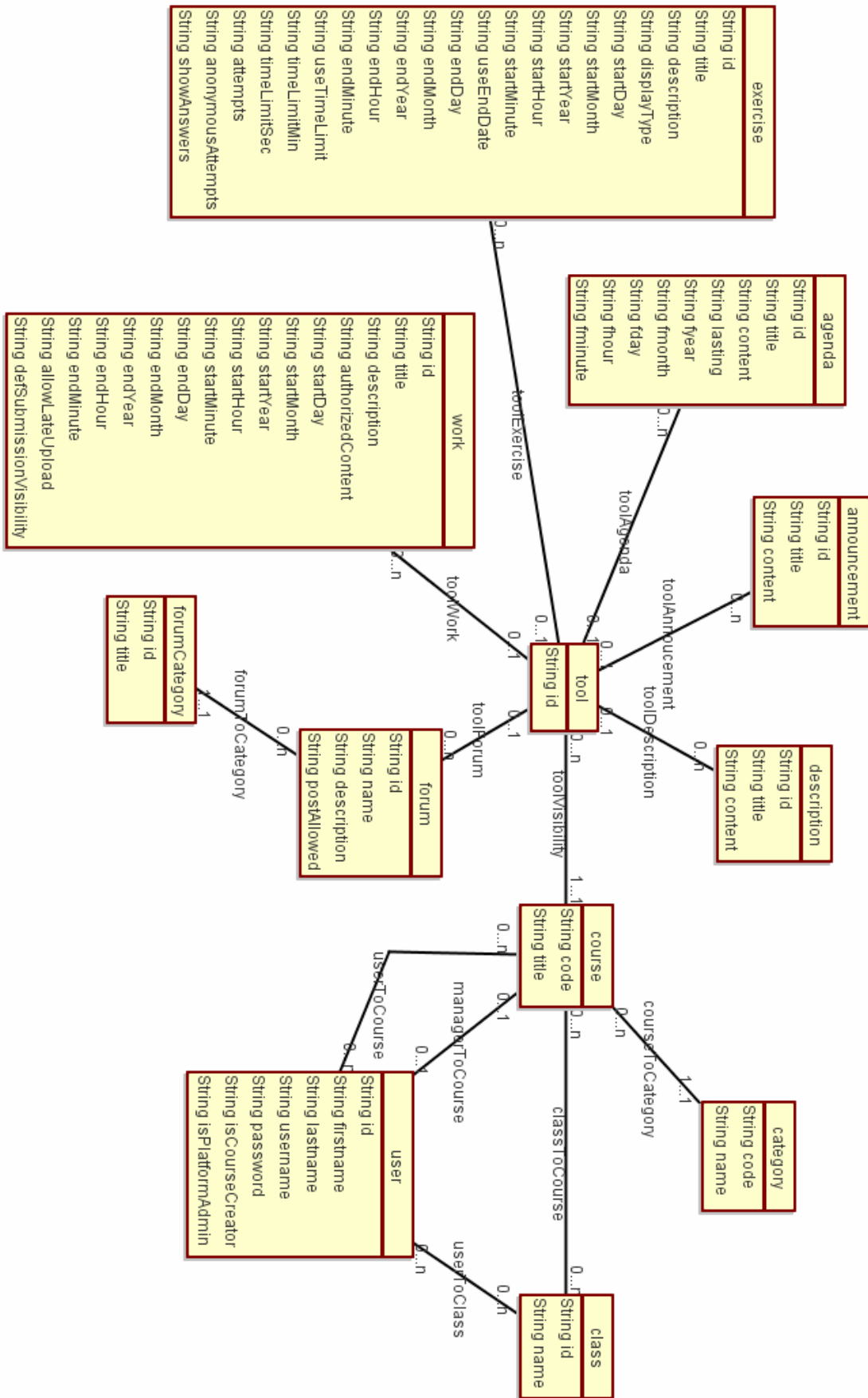


Figure 64 Métamodèle de Claroline

10.2 A partir de la description textuelle, réaliser le métamodèle indépendant de la plateforme, à partir des services et des types complexes réaliser le Métamodèle spécifique

Nous présentons dans cette section les détails de la création de différents métamodèles. Pour illustrer les notions de métamodèle pédagogique et fonctionnel ainsi que pour détailler l'élaboration des règles de transformation, nous nous basons cette fois sur une étude de cas menée sur l'application Web WikiniMST dans le cadre du projet PCDAI (cf 11.3). Cette étude, menée sur deux ans, a permis de révéler une partie de notre problématique autour de la métamodélisation (problématique transformation versus fusion) et de la construction de dispositifs (problématique posée par les dépendances de construction).

10.2.1 Une plateforme de modélisation

Manipuler informatiquement des modèles au sein d'un modeleur, puis construire un générateur permettant de les opérationnaliser sont des actions qui peuvent être menées efficacement dans le cadre de la relation de conformité induite par le MOF (cf 6.3) L'IDM propose par son architecture, un cadre formel permettant ces actions, il est par exemple possible, en respectant les règles du MOF, de construire à partir d'un métamodèle défini en MOF, un modèle conforme. En appliquant l'IDM aux problématiques traitées par langages spécifiques de domaine (DSL) il est possible de définir des plateformes de modélisation générique. De telles plateformes reposant sur une l'implémentation d'un méta méta modèle (MOF, eCore) permettent la définition efficace de différents modeleur. GMF (GMF), GME (GME), MerlinGenerator (Merlin), GEMS (GEMS), MetaEdit (MetaEdit), ModX (ModX) sont des exemples de plateforme de modélisation utilisant les concepts de l'IDM.

Dans le cadre de nos travaux nous utilisons la plateforme de modélisation ModX, cette plateforme permet une vérification de conformité par "spécifications exécutables" (Blay-Fornarino & Franchi, 2005). Des travaux parallèles sont également menés au sein de notre équipe de recherche, autour de eCore (eCore), ces travaux utilisent l'outillage EMF (EMF), pour générer des modeleurs (Moura & Derycke, 2007).

La sous-section suivante présente plus précisément la plateforme de modélisation ModX.

10.2.2 Présentation de ModX

ModX une plateforme de modélisation réflexive, intégrant une implémentation du MOF 1.4. La plateforme permet d'éditer des métamodèles conformes au MOF, elle utilise une notation graphique proche de l'UML et propose un schéma de persistance selon le format XMI 1.1 Ce schéma de persistance est complété par des extensions propres à ModX permettant de configurer le modeleur graphique engendré par la plateforme à partir du métamodèle défini.

La configuration du modeleur graphique, spécifique à un domaine, est effectuée par la définition pour un métamodèle donné d'un ensemble de vue de modélisation, ces vues permettent de distinguer les différents aspects d'un domaine de modélisation. Trois éléments caractérisent une vue.

- Une vue est composée d'une liste de classes, et de leurs caractéristiques, telles que les droits de création, suppression, visualisation et modification. La représentation graphique de ces classes peut être paramétrée (couleurs, icônes, texte) ...
- Une vue est composée d'une liste d'associations. Au sein d'une vue il est possible d'afficher ou non un type d'association, et de paramétrer leur représentation graphique (simple lien, ou relation de type conteneur).

- Une vue est composée d'une liste d'opérations dont on peut définir les paramètres, c'est-à-dire les informations qui seront demandées à l'utilisateur lors de l'exécution de l'opération. Une opération permet de créer des instances de classes et des associations entre deux instances, elle offre la possibilité de générer automatiquement du code java et de le compiler, puis le sauvegarder dans le métamodèle. Lors de la création d'un modèle, l'utilisateur pourra alors accéder à cette opération, grâce au menu contextuel de la vue à laquelle l'opération appartient. Lors de son exécution, les instances spécifiées seront créées et paramétrées par l'utilisateur.

ModX possède un mécanisme de transformation simple, basé sur l'exécution de scripts java utilisant l'API XMI de ModX. Entre deux métamodèles A et B chargés par ModX il est possible de définir des règles permettant la création de modèles conformes au métamodèle B à partir d'un modèle conforme au métamodèle A. Le script ainsi défini inclut l'appel à des fonctions de dialogue avec l'utilisateur pour résoudre les alternatives ou la nécessité de nommer un objet. Ce script est directement compilé par ModX lors de son appel par le modeleur.

La sous-section suivante détaille sur un exemple réalisé lors de l'étude de cas menée sur la plateforme WikiniMST dans le cadre du projet PCDAI (cf. section 11.3), l'utilisation de la plateforme ModX.

10.2.3 Exemple de réalisation d'un modeleur spécifique

Dans le cadre du projet PCDAI, (cf. section 11.3), une des études a porté sur la construction de dispositif pour l'application WikiniMST. La réalisation d'un modeleur spécifique (cf le script de configuration Annexe) pour ce projet a nécessité de mener les actions suivantes :

- Création de deux métamodèles (pédagogique et spécifique)
- Création d'un script de transformation.
- Habillage et paramétrage de la plateforme de modélisation, pour en faire graphiquement un artefact spécifique au projet.

Le premier métamodèle pédagogique (cf 11.3) exprime en MOF les concepts de modélisation utilisé par l'équipe pédagogique impliquée dans le projet la description en langage naturel des concepts impliqués par le projet est consultable en annexe. La représentation graphique de ces concepts est illustrée Figure 65 :

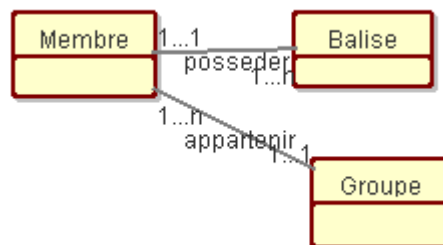


Figure 65 Représentation graphique des concepts de modélisation dans le cadre du projet PCDAI

- Les cardinalités exprimées permettent d'encadrer dans le modeleur spécifique la réalisation des modèles conformes au métamodèle.
- Un membre possède une ou plusieurs balises mais appartient à un et un seul groupe, un groupe est composé d'au moins un membre, une balise possède un et un seul propriétaire.

- Le métamodèle très simple est utilisable à partir d'une seule vue.

Le second métamodèle correspond au métamodèle spécifique à la plateforme WikiniMST, il est illustré par la Figure 66 :

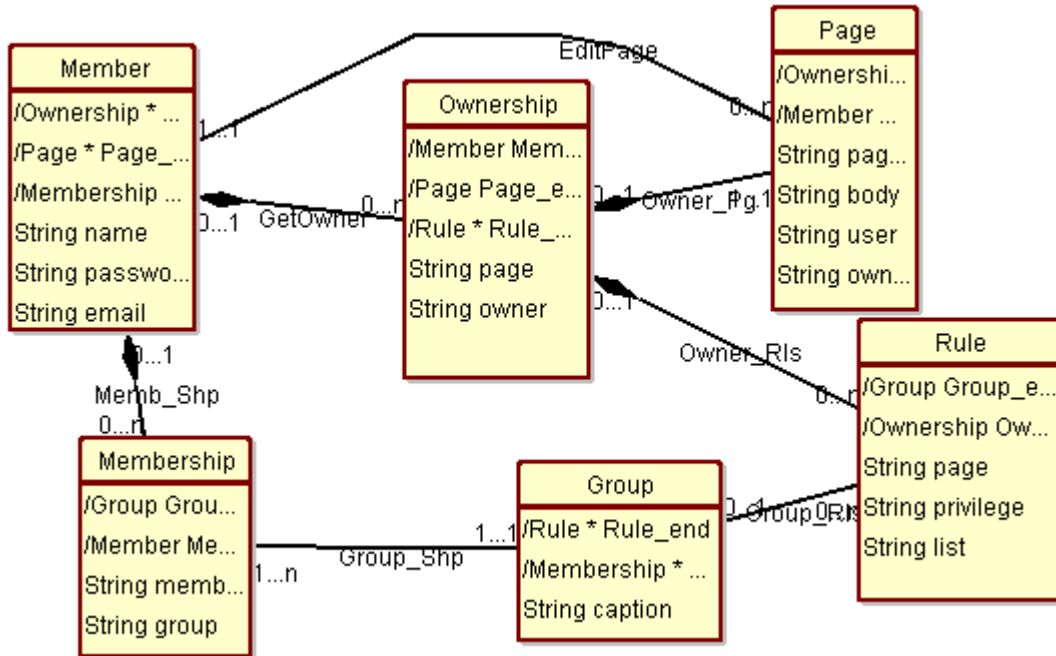


Figure 66 Métamodèle du WikiniMST

- Le métamodèle du WikiniMST définit pour un membre les notions de propriété ("Ownership") et d'appartenance ("Membership"),
- Chaque notion de propriété met en relation une personne, une page et des règles.
- Chaque notion d'appartenance met en relation un membre avec le groupe auquel il appartient.
- Les associations de composition permettent de spécifier pour le modèle des comportements qui sont exploités lors de la construction des éléments du dispositif (cf. 10.6).

Trois types de vues (Cf. annexe pour le détail des vues) sont décrits pour exploiter ce métamodèle dans un modeleur. La première "Basic" explore les relations entre "Member", "Membership" et "Ownership". La seconde "Owner" regroupe pour chaque notion de type "Ownership" les relations avec "Page" "Rules" et "Group". La troisième la vue "Element" permet de regrouper tous les éléments impliqués par un modèle de dispositif, le détail de sa configuration est décrit par la Figure 67 :



Figure 67 détail de la vue "élément"

Dans cette vue les associations entre un membre et ses propriétés sont représentées sous forme d'inclusion, de même les associations entre une propriété et les pages et les règles qu'elle implique. Le paramétrage de ces associations est détaillé par la Figure 68 :

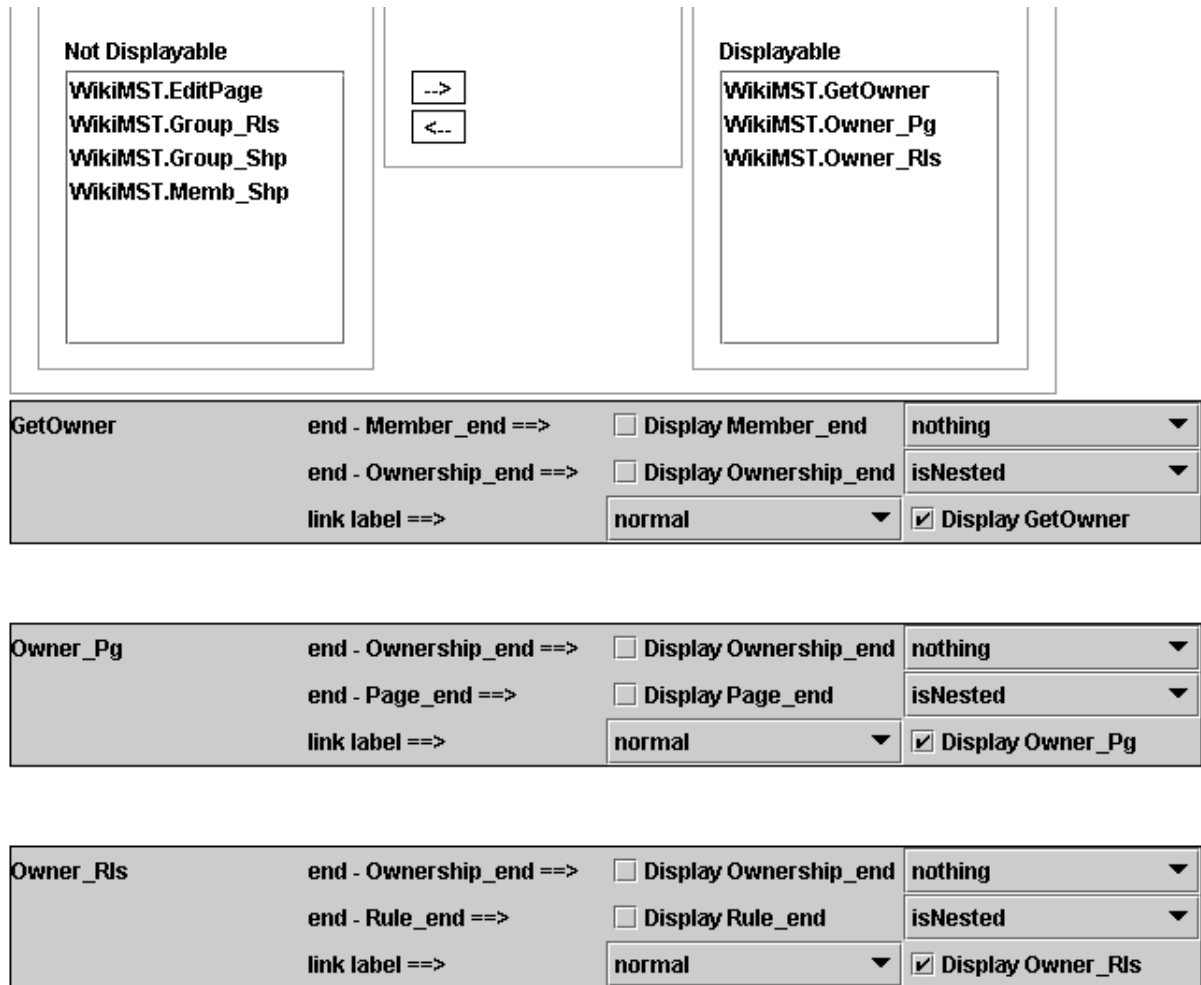


Figure 68 Détail du paramétrage des associations d'une vue

Un modèle dispositif exprimé avec le métamodèle peut donc être décrit à l'aide des trois types de vue "Basic", "Owner" et "Element".

10.3 Définir les transformations

10.3.1 Les transformations dans ModX

Une des caractéristiques de l'environnement de modélisation ModX, est de proposer, au sein de sa plateforme de modélisation, la possibilité de définir sous forme de script java, interprété lors de l'exécution du modeleur, des règles de transformations mettant directement en œuvre des macros de l'environnement ModX. Cette solution ad hoc au problème des transformations de modèles permet de résoudre de simples transformations impliquant non seulement les métamodèles mais les vues qu'ils définissent. Nous illustrons notre approche des transformations par l'étude de cas PCDAI, (utilisant l'application **WikiniMST**) (cf. section 11.3).

L'étude de cas s'est déroulée sur deux années, la première année, des règles permettant des transformations entre le métamodèle pédagogique et le métamodèle spécifique ont été définies et utilisées. Lors de cette première année de l'étude PCDAI, nous avons privilégié une démarche de concrétisation permettant de déduire le métamodèle fonctionnel spécifique à la

plateforme WikiniMST du métamodèle pédagogique. Le tableau suivant présente un détail commenté du script de transformation. Dans les cas limités que nous traitons, et parce que de tels scripts peuvent faire appel aux fonctionnalités de la plateforme de modélisation, la réalisation d'un fichier de transformation entre deux métamodèles très simple n'est pas une chose difficile à réaliser.

Identifier les deux métamodèles	<pre>modx.MOF.Model.Package EAPC=getMetamodel("EAPC"); modx.MOF.Model.Package WIKI=getMetamodel("WikiMST");</pre>
Identifier le modèle source	<pre>modx.MOF.Instance.Model modeleAPC=chooseModel(EAPC);</pre>
S'il existe on continu	<pre>if (modeleAPC==null) return;</pre>
Interagir avec l'utilisateur du modeleur, créer une instance du modèle cible	<pre>java.lang.String WikiName=askName("Nom pour le modèle wiki",modeleAPC.getName()+"_wiki"); modx.MOF.Instance.Model modeleWIKI=modx.MOF.Instance.Models.newModel(WikiName,WIKI);</pre>
On charge les membres du modèle sources	<pre>modx.MOF.Instance.Object[] members=getObjects(getClass("EAPC.contents.Membre"),modeleAPC);</pre>
Pour chaque membre du modèle source on crée un membre dans le modèle cible	<pre>for (int i=0;i<members.length;i++){ modx.MOF.Instance.Object EAPCmember=members[i]; modx.MOF.Instance.Object WIKImember=createObjectInModel(EAPCmember.getName(),getClass("WikiMST.contents.Member"),modeleWIKI);</pre>
on cherche s'il existe des groupes avec lesquels il serait relié	<pre>modx.MOF.Instance.Object[] EAPCgroups=EAPCmember.getsThroughLinks("Groupe_end");</pre>
Si c'est le cas, pour chaque groupe avec lequel il est relié, on crée une instance de Membership et de groupe dans le modèle cible, on complète les attributs name	<pre>if (EAPCgroups!=null) for (int j=0;j<EAPCgroups.length;j++){ modx.MOF.Instance.Object WIKIgroup= getObject(EAPCgroups[j].getName(),getClass("WikiMST.contents.Group"),modeleWIKI); modx.MOF.Instance.Object WIKImembership= getObject(EAPCgroups[j].getName()+"ship",getClass("WikiMST.contents.Membership"),modeleWIKI);</pre>
etc...	

10.3.2 Génération d'un modeleur spécifique

La dernière étape de la réalisation d'un modeleur consiste à définir un script pour en définir l'habillage. Ce script inhibe certaines fonctions de la plateforme de modélisation pour simplifier son apparence, il effectue également le pré-chargement du métamodèle pédagogique, du métamodèle spécifique, et du fichier de transformation.

Dans la section suivante nous allons étudier l'usage d'un modeleur ainsi produit. Cette étude nous permet d'aborder le deuxième point de notre problématique IDM :

En quoi consistent les interactions entre un modèle et son opérationnalisation dans le contexte d'une plateforme ?

Pour répondre à cette question, nous introduisons, lors de la tâche de modélisation, des éléments de profilage qui permettent lors de l'opérationnalisation du modèle de factoriser une partie de la construction des éléments qu'il décrit. Notre proposition est détaillée dans la section 10.4 suivante.

10.4 Tester la chaîne d'édition en créant et transformant un modèle

La Figure 69 présente un modèle de dispositif réalisé lors de la première année de l'étude de cas PCDAI (cf. section 11.3) sur la plateforme WikiniMST, la modélisation de ce dispositif a été réalisée à l'aide d'un modeleur spécifique défini à partir du métamodèle (cf. Figure 65, page 160).

Figure 69 Exemple de modèle de dispositif PCDAI 1 conforme au métamodèle pédagogique

Dans ce modèle de dispositif, deux types de membre sont définis : les formateurs et les étudiants, nous retrouvons pour chaque type de membre les balises et les groupes qui leur sont reliés dans le cadre de ce modèle de dispositif.

Ce modèle est transformé en un modèle fonctionnel en appliquant les règles de transformation définie entre les deux métamodèles (cf. page 164).

On obtient ainsi un modèle décrivant le même dispositif que précédemment mais exprimé cette fois dans le métamodèle spécifique WikiniMST (cf. Figure 66, page 161). Ce modèle est organisé selon un système de Vue. Ces vues définies au sein du métamodèle appartiennent à trois types : Basic, Owner et Element (cf. Figure 70).

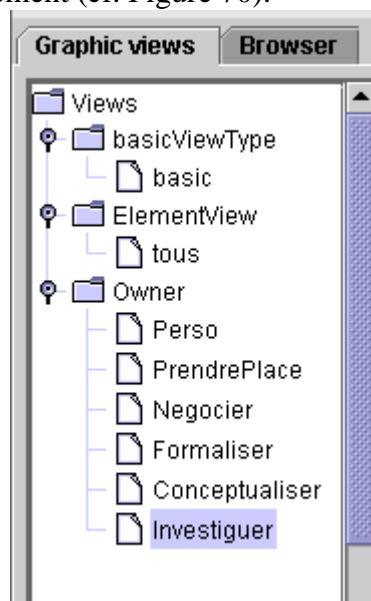


Figure 70 Détails des vues créés après transformation

Pour la vue "Basic" on obtient la Figure 71 :

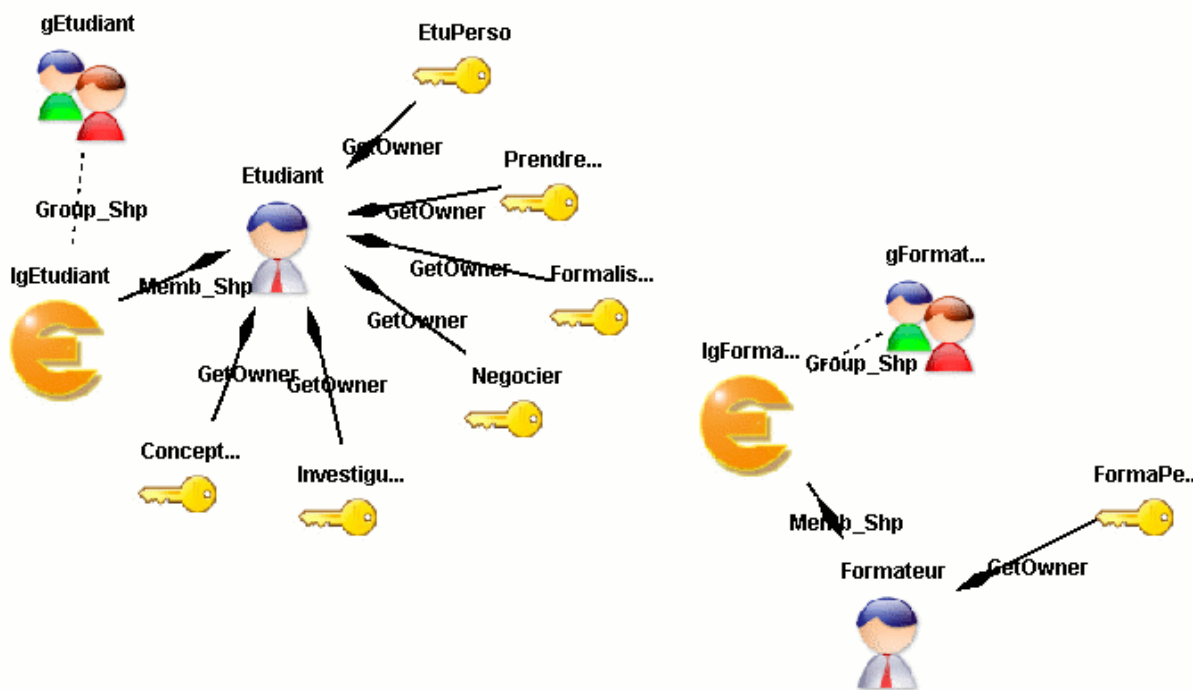


Figure 71 Vue "Basic" du modèle de dispositif, PCDAI

Cette vue est très proche du modèle de dispositif (cf. Figure 69) tel qu'il est exprimé avec le métamodèle pédagogique, chaque notion de propriété est explicitée dans une vue séparée. Ainsi la notion "Négociateur" est détaillée dans la vue de type "Owner" (cf. Figure 72).

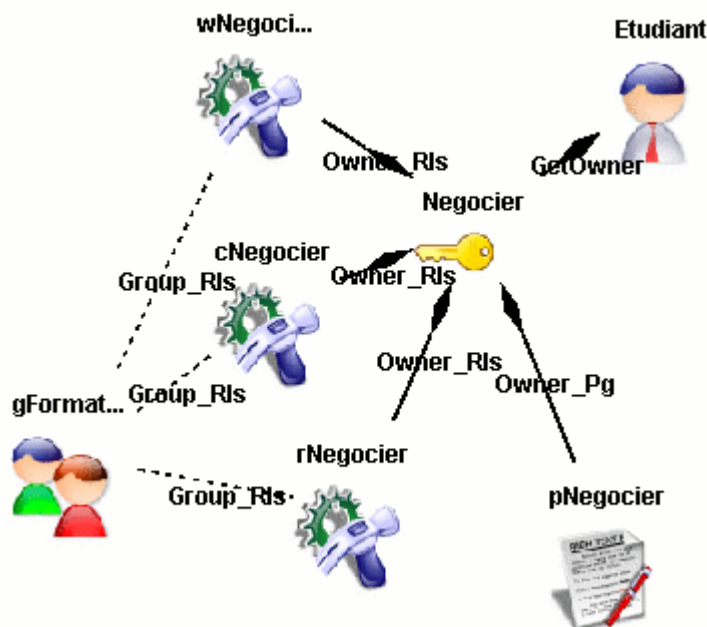


Figure 72 La vue de type "Owner" représentant la notion "Négociateur" du modèle de dispositif, PCDAI 1

Lors de la construction des éléments constituant le dispositif, pour chaque élément de type Etudiant le modèle indique qu'il faut générer différentes notions de propriétés. La notion "Négociateur" engendre une page Wiki et trois règles d'utilisation. Ces trois règles font référence

au groupe des formateurs. Ces implications de constructions ainsi que le système de profilage que nous avons développé sont exploitées lors de la construction d'un dispositif.

Par exemple Figure 73, les indications de profilage permettent la génération d'étudiants dont l'attribut "password" prendra la valeur etu01, etu02 etc.

Member Etudiant

Name : Etudiant

String password etu\$\$

Page Page_end [Add a Page](#)

String email etu\$\$@rien.com

Membership Membership_end [Add a Membership](#)

Figure 73 Profilage des attributs pour étudiant

Figure 74, le pré-formatage des règles d'écriture au sein du Wiki permet d'illustrer le type d'indication que nous pouvons mettre dans le modeleur spécifique :

L'indications @formateur est une indication textuelle interprétée directement par le Wiki, elle suit les règles d'écriture du Wiki et suppose la présence d'un groupe ayant pour nom formateur. L'indication "write" est interprétée par le constructeur pour identifier les services à appeler. L'indication "\$page.Name" référence l'attribut "Name" de l'objet de type Page auquel les objets de type Rule sont reliés.

Rule wEtuPerso

Name : wEtuPerso

String list @formateur

String privilege write

String page \$page.Name

Ownership Ownership_end [obj15](#)

Group Group_end [obj14](#)

Figure 74 Exemple d'indications à mettre en attribut

Le tableau suivant issue de la documentation permet de comprendre cette notation.

NomUtilisateur	Droit d'accès (lecture/écriture/commentaires) accordé à NomUtilisateur
@legroupe	Droit d'accès accordé au groupe legroupe
*	Droit d'accès accordé à tout le monde
+	Droit d'accès accordé uniquement aux personnes identifiées
!NomUtilisateur	Droit d'accès refusé à NomUtilisateur
!@legroupe	Droit d'accès refusé au groupe legroupe
!*	Droit d'accès refusé à tout le monde
!+	Droit d'accès refusé aux personnes identifiées

L'exploitation de ces éléments de profilage est décrite dans les sections suivantes.

10.5 Création du constructeur spécifique

10.5.1 Introduction

Le constructeur spécifique GenDep d'une plateforme de formation repose sur le principe illustré par la Figure 75.

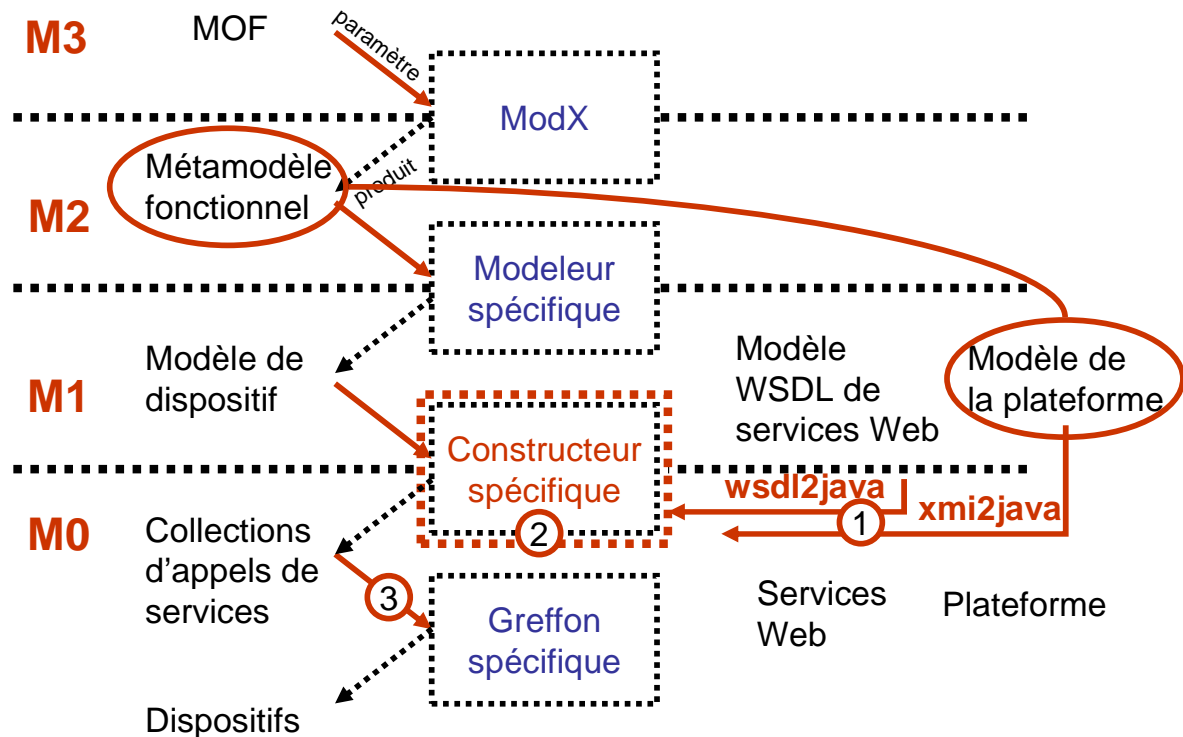


Figure 75 Schémas de situation du constructeur spécifique

Sur la gauche de cette figure nous faisons référence aux indications relatives à l'architecture de modélisation de l'OMG : M3 (méta méta modélisation), M2 (méta modélisation), M1 (modélisation), M0 (implémentation) (cf. 6.2.1). Le constructeur spécifique se situe entre les niveaux M1 et M0, il utilise en paramètre un modèle pour produire une collection d'appel de service. Sa construction est cependant réalisée à partir d'éléments de type M1 : Un modèle de la plateforme, un modèle des services Web. Rappelons (cf. chapitre 9) que le métamodèle fonctionnel, qui paramètre le modeleur correspond, à un modèle de la plateforme; c'est cette correspondance qui garantit que le constructeur saura manipuler les modèles de dispositif.

Cette Figure 75, présente les éléments nécessaires à la compréhension du plan que nous allons suivre dans cette section :

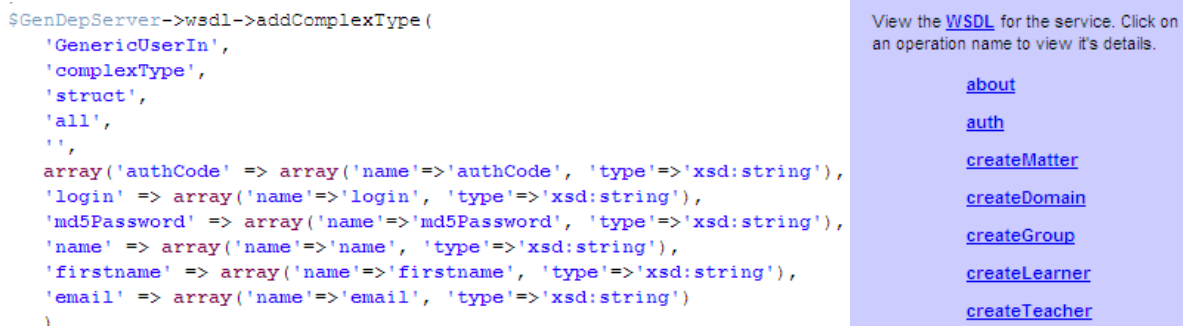
- 1/ Dans la sous section 10.5.2, nous présenterons les deux tâches ("**wsdl2java**" et "**xmi2java**") permettant la génération du squelette d'un constructeur spécifique.
- 2/ Dans la sous-section 10.5.3, nous établirons un premier dialogue avec la plateforme
- 3/ Enfin dans la sous section 10.5.4, nous montrerons comment interpréter un modèle et construire par des collections d'appels de service, le système qu'il décrit.

Pour présenter cette partie de nos travaux nous allons utiliser l'étude que nous avons réalisée sur la plateforme **Ganesha**, cette étude est détaillée dans le chapitre 11.2, et a permis de finaliser le mode de communication entre GenDep et une plateforme de formation.

10.5.2 Deux tâches pour implémenter un constructeur spécifique

Pour engendrer le constructeur spécifique d'une plateforme de formation, deux tâches ("wsdl2java" et "xmi2java") ont été réalisées. Ces tâches ont été pour la première fois utilisées dans le cadre de notre thèse, lors de l'étude portant sur la plateforme **Ganesha**, aussi utiliserons nous cet exemple pour illustrer nos propos.

A partir de l'interface du greffon de services Web (WSDL), la première tâche ("wsdl2java") crée en Java, au sein du constructeur, les souches des services Web. Cette tâche a été implémentée à l'aide de l'outil Ant (Ant), elle utilise classiquement la bibliothèque Axis (Axis), son intérêt réside dans les facilités qu'elle offre lui permettant d'être intégrée dans notre environnement de développement (Eclipse).



```

$GenDepServer->wsdl->addComplexType (
    'GenericUserIn',
    'complexType',
    'struct',
    'all',
    '',
    array('authCode' => array('name'=>'authCode', 'type'=>'xsd:string'),
    'login' => array('name'=>'login', 'type'=>'xsd:string'),
    'md5Password' => array('name'=>'md5Password', 'type'=>'xsd:string'),
    'name' => array('name'=>'name', 'type'=>'xsd:string'),
    'firstname' => array('name'=>'firstname', 'type'=>'xsd:string'),
    'email' => array('name'=>'email', 'type'=>'xsd:string')
)
    
```

View the [WSDL](#) for the service. Click on an operation name to view it's details.

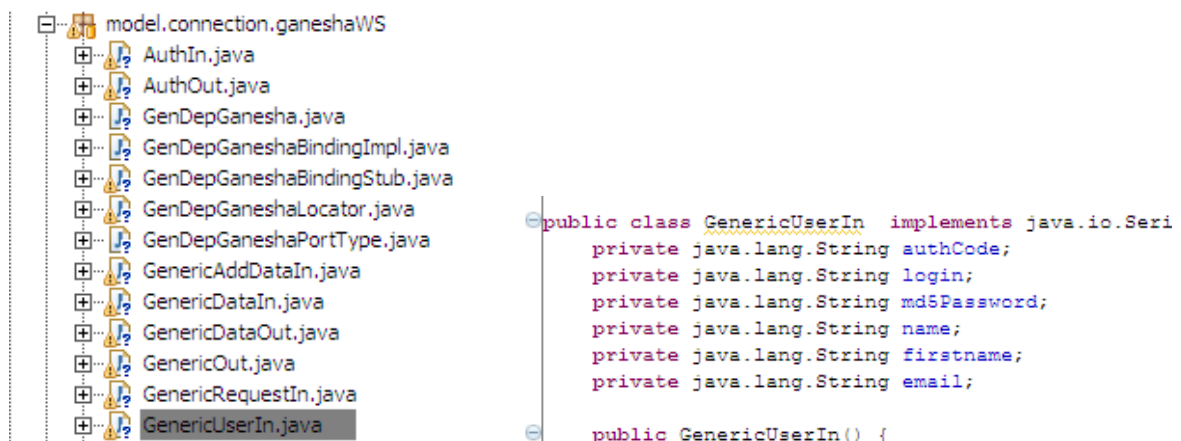
- [about](#)
- [auth](#)
- [createMatter](#)
- [createDomain](#)
- [createGroup](#)
- [createLearner](#)
- [createTeacher](#)

Figure 76 : Serveur de services Web et WSDL pour l'application Ganesha

La Figure 76 présente un détail du serveur de services Web et la partie correspondante du fichier WSDL ajouté à la plateforme Ganesha. En particulier la définition du type complexe "**GenericUserIn**".

Il est possible de générer les souches correspondant aux services implémentés dans le greffon. Ces souches permettront la manipulation sous forme de classes java, des services implémentés eux en PHP. Pour ce faire, une tâche Ant a été définie, "**wsdl2java-ganeshaWS**"

Cette tâche permet de construire le package ("**model.connection**") spécifique à l'application à partir du fichier WSDL décrivant les services Web. Ce package permet d'établir une communication avec la plateforme en implémentant des appels de service.



model.connection.ganeshaWS

- AuthIn.java
- AuthOut.java
- GenDepGanesha.java
- GenDepGaneshaBindingImpl.java
- GenDepGaneshaBindingStub.java
- GenDepGaneshaLocator.java
- GenDepGaneshaPortType.java
- GenericAddDataIn.java
- GenericDataIn.java
- GenericDataOut.java
- GenericOut.java
- GenericRequestIn.java
- GenericUserIn.java

```

public class GenericUserIn implements java.io.Seri
    private java.lang.String authCode;
    private java.lang.String login;
    private java.lang.String md5Password;
    private java.lang.String name;
    private java.lang.String firstname;
    private java.lang.String email;

    public GenericUserIn() {
    
```

Figure 77 : Détail de l'implémentation des souches de services Web

Pour chaque type complexe présent dans le fichier WSDL, une classe java est générée. Sur l'exemple Figure 77, la classe "**GenericUserIn**" est générée, elle correspond au type complexe défini Figure 76.

A partir du métamodèle de la plateforme modélisé à l'aide le l'éditeur ModX, la deuxième tâche ("**xmi2java**") crée au sein du constructeur spécifique les classes java correspondantes.

Pour appliquer cette tâche, il faut commencer par définir, au sein de ModX, le métamodèle Ganesha comprenant une classe : la classe "**User**" (cf. Figure 78). Il faut faire en sorte que les attributs de cette classe soient identiques aux attributs de la classe "**GenericUserIn**" (créée par la tâche précédente Figure 77).

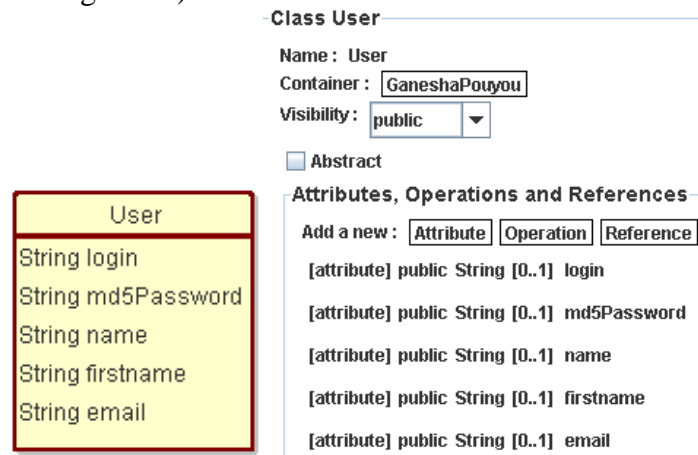


Figure 78 Class User dans ModX

Cette classe (cf. Figure 78) est interprétée par la tâche "**xmi2java-GabeshaWS**" pour construire les classes correspondantes (cf. Figure 79) dans le constructeur spécifique.

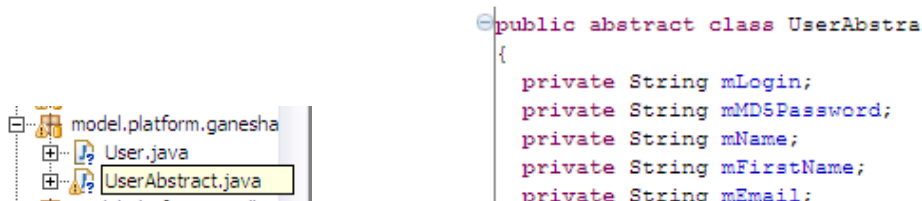


Figure 79 Implémentation de la classe User

Pour chaque classe dans ModX, deux classes java sont ainsi créées : "**User**" et "**UserAbstract**". Seule la classe "**User**" peut être modifiée. Cette modification concerne pour chaque classe non abstraite l'écriture des méthodes "**deploy**" et "**fetch**".

10.5.3 Etablir une première communication entre le constructeur et la plateforme.

Avant d'écrire les méthodes "**deploy**" et "**fetch**" de chaque classe, il faut créer le constructeur spécifique de la plateforme Ganesha. L'outil GenDep propose pour cela des interfaces et un moteur de construction générique qu'il faut spécifier pour chaque plateforme.

Deux classes et un fichier doivent être écrits :

- "**typeGaneshaWS.xml**" dans le répertoire de configuration de GenDep, ce fichier indique la structure des informations nécessaire à l'authentification sur une plateforme de type Ganesha. Il indique également le fichier spécifique "**ExecuteSpecific**" de

GenDep relatif à ce type de plateforme. Des paramètres iconographiques sont également donnés (cf. Figure 80).

- **"ExecuteSpecific"** du package **"controler.deploy"** spécifique à l'application visée, permet de récupérer les paramètres de localisation et d'authentification d'une instance de plateforme de type Ganesha.
- **"PlatformSpecific"** du package **"model.platform"** spécifique à l'application visée est la classe chargée d'établir la connexion avec la plateforme de formation et de vérifier que la personne désirant construire un dispositif dispose des droits suffisants pour réaliser cette construction.

Ces trois fichiers sont dépendants et doivent être implémentés ensemble.

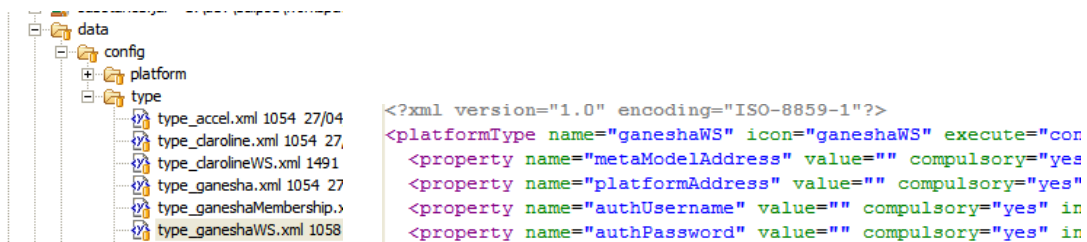


Figure 80 Fichier de configuration de ce type de plateforme

Dès lors un premier essai de communication est possible avec une instance spécifique de la plateforme. Pour effectuer cette communication, il faut placer, au côté de l'instance de la plateforme, le greffon de services Web ainsi que le métamodèle. Il faut également créer un modèle de dispositif conforme au métamodèle de Ganesha. Ce modèle n'a pas besoin d'être abouti dans la mesure où, à ce stade de réalisation, les fonctionnalités de construction et de dialogue n'ont pas été implémentées pour les objets. Enfin, il faut créer un paramétrage spécifique correspondant à l'instance spécifique de la plateforme de type Ganesha. Ce dernier point peut être réalisé directement lors de l'exécution du constructeur spécifique (cf. Figure 81).

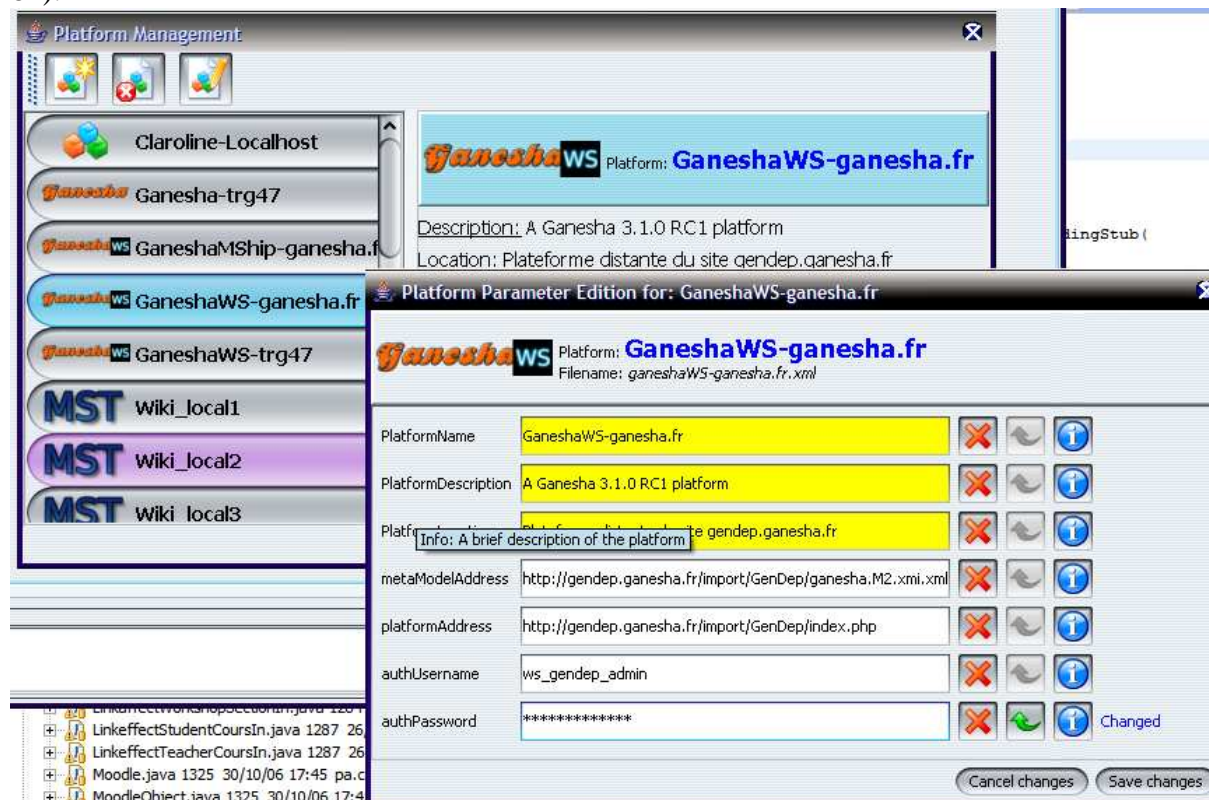


Figure 81 Paramétrage de l'instance de la plateforme Ganesha.

Une première communication avec une instance de la plateforme permet de vérifier, si celle-ci est accessible et si on dispose des droits nécessaires (cf. Figure 82).



Figure 82 Une première communication avec une instance de la plateforme

Un fois cette communication établie, il faut s'assurer que les objets et les liens auront la possibilité d'être construits. Cette possibilité repose sur l'implémentation de deux méthodes ("**deploy**" et "**fetch**"), ce travail est l'objet de la sous-section suivante.

10.5.4 Construire, Recevoir des informations.

Au sein de GenDep l'échange d'information s'effectue à l'aide de deux méthodes qu'il faut implémenter dans chaque classe non abstraite. Ces méthodes doivent appeler les services correspondants sur la plateforme. Il s'agit des méthodes "**deploy**" et "**fetch**". L'utilisation de ces méthodes est illustrée par la Figure 83.

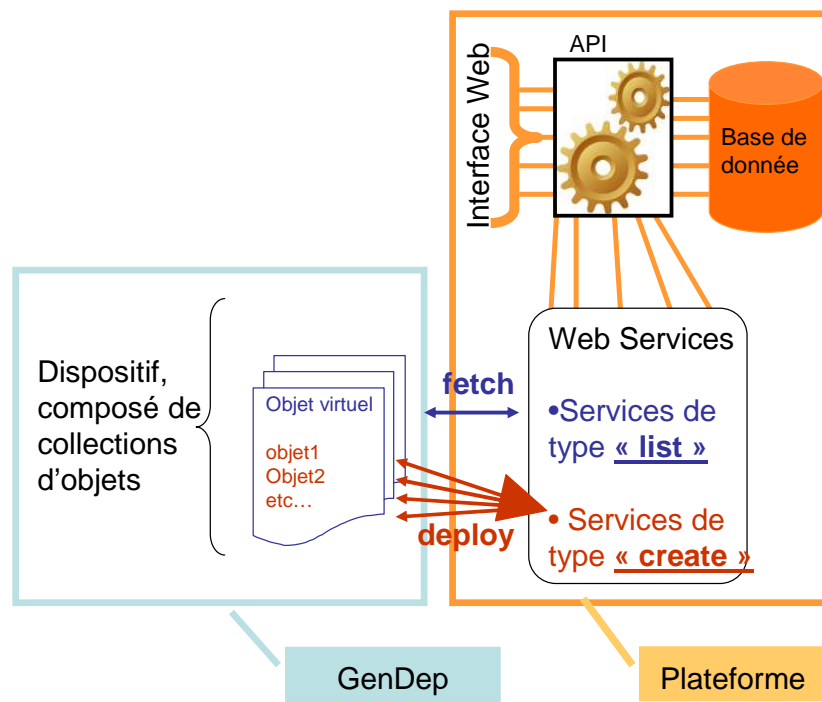


Figure 83 Utilisation des méthodes "fetch" et "deploy"

- Pour créer un objet dans genDep à partir de sa construction sur la plateforme, la collection qui le contiendra interroge la plateforme. Cette interrogation est rendue possible en plaçant par défaut dans toutes les collections manipulées par GenDep un objet virtuel, que la collection peut introspecter. Cette introspection permet de générer les interfaces de contextualisation, elle permet également d'établir un dialogue entre plateforme et la collection par l'intermédiaire des méthodes de cet objet.
- Pour qu'un objet puisse se construire via des appels de service il faut implémenter pour la classe non abstraite dont il est une instance, une méthode de construction "**deploy**".

La méthode "**fetch**" s'est avérée très utile lors des constructions effectives de dispositifs, cette méthode permet des communications avec la plateforme qui seront illustrées dans la section 10.6 suivante. Le détail de l'implémentation de ces deux méthodes est expliqué en annexe.

10.6 Dépendance de construction

Un des problèmes rencontrés au cours de nos études concerne les dépendances de construction, ce problème relativement complexe est illustré par l'étude de cas portant sur l'application **WikiniMST** lors du projet PCDAI, ce problème est apparu lors de la deuxième année de l'étude.

Pour comprendre le problème, il nous faut présenter le métamodèle fusionné (cf. Figure 84) expérimenté lors de cette seconde année.

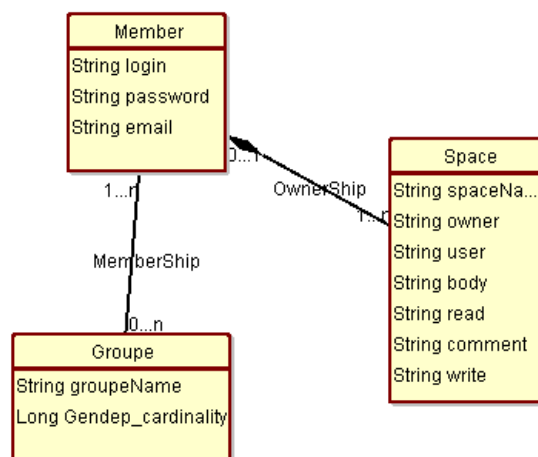


Figure 84 Métamodèle fusionné de l'application WikiniMST

Présentation du problème.

Ce métamodèle illustre un problème de dépendance que nous avons résolu de deux manières lors de nos travaux. Pour plusieurs membres du Wiki appartenant au groupe des administrateurs nous souhaitons créer quatre espaces personnalisés, un espace personnel (PagePerso), un espace d'édition des groupes qu'il administre (EditGroup), un espace correspondant au menu qu'il administre (Menu), et un dictionnaire de notion qui lui soit propre (DicoNotion). Le modèle de dispositif correspondant à cette description est illustré par la Figure 85.

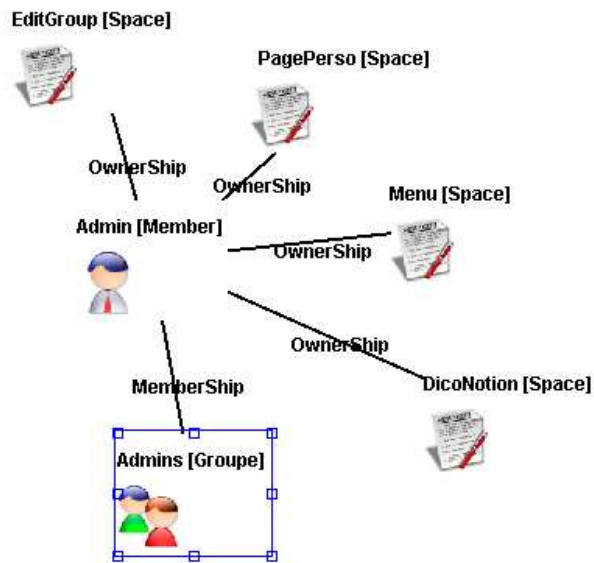


Figure 85 Exemple de modèle illustrant le problème de dépendance.

Lors de la génération des collections d'appel de service, il faut pouvoir générer la construction de 3 administrateurs, de 1 groupe, de 3x4 espaces personnels, de 3 associations de type member-groupe, de 3x4 associations de type space-member.

Le premier problème et sa solution : utiliser les associations de compositions

Le premier problème est posé par la génération des espaces personnels : Comment indiquer au constructeur que ces espaces dépendent itérativement de leurs utilisateurs?

Au sein du constructeur nous avons choisi d'établir une distinction entre les associations simple et les associations de composition. Ces dernières sont interprétées par le constructeur en terme de cycle de vie et d'appartenance. Ainsi dans l'exemple (cf. Figure 85) la relation qui lie les quatre espaces à un des administrateurs est une composition (cf. Figure 84). Cela signifie que la vie de ces quatre espaces est directement liée à celle de l'administrateur. La destruction de celui-ci entraîne la destruction de ses espaces, la création de celui-ci entraîne la création de ces espaces. La création de 3 administrateurs entraîne ainsi la création de 3x4 espaces.

Il faut remarquer que ce problème de dépendance est un problème récurrent en EIAH, ainsi le standard IMS-LD aborde le problème des activités pédagogiques de deux manières : "**learning activity**" est une activité qui s'effectue une fois, "**support activity**" est une activité qui s'effectue autant de fois qu'il y a d'élèves. Le standard apporte ainsi une solution simple en définissant deux types de classe aux comportements différents. Notre solution exploite, quant à elle, la sémantique du MOF, elle explicite l'interprétation de la relation d'association composite lors de la construction d'un modèle.

Le deuxième problème et sa solution : persistance des identifiants renvoyés par la plateforme, et déploiement des liens.

Le deuxième problème concerne la création des administrateurs et du groupe auquel ils appartiennent. En effet la plateforme WikiniMST ne suit pas la logique de notre constructeur. Celui-ci est basé sur la logique suivante : créer les administrateurs, créer le groupe, lier les administrateurs au groupe. Or la plateforme WikiniMST ne possède pas a proprement parlé de

notion de groupe. Une table recensant les utilisateurs et le groupe, auquel ils appartiennent, permet après analyse de déduire les groupes existants.

Pour résoudre ce problème, nous avons implémenté différentes fonctionnalités dans notre constructeur. Un analyseur de graphe permet de choisir l'ordre de création des collections d'objets. Cet analyseur se base sur les associations de composition et définit un ordre de construction. Les liens joignant les objets sont créés dès que cela est possible, c'est-à-dire dès que les deux extrémités sont disponibles sur la plateforme. La Figure 86, illustre notre solution :

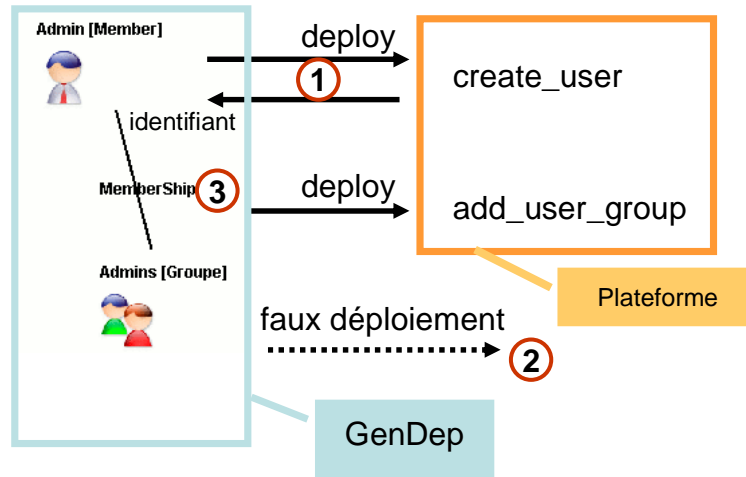


Figure 86 Ecriture d'une méthode *deploy* différé, réalisée par le lien.

Point 1 : Dans notre exemple la classe "**Member**" possède une méthode "*deploy*", cette méthode fait appel à un service Web qui retourne au constructeur l'identificateur de cet élément sur le Wiki; dans notre cas il s'agit simplement de son attribut "*name*". Dans le cas d'autres plateformes comme Ganesha, Moodle, Claroline il s'agit d'un identifiant unique numérique.

Point 2 : La classe "**Groupe**" ne possède pas de méthode "*deploy*" car créer un groupe n'a pas de signification sur la plateforme WikiniMST. Cependant bien que non construit sur la plateforme une collection d'objet de type "**Group**", contenant un seul objet est bien instancié au sein du constructeur pendant le déploiement.

Point 3 : Une fois la non construction de l'objet de type "**Group**" prise en compte par le constructeur, celui-ci est libre de construire les liens liant cet objet à la collection de trois administrateur. Cette construction s'effectue alors via la méthode "*deploy*" propre au lien. Ainsi à partir d'un lien, il est possible de remonter aux deux objets marquant ses extrémités et de déployer ensuite le lien via le service "*add_user_group*".

10.7 En guise de conclusion à l'implémentation de GenDep

10.7.1 Un identifiant

Le mécanisme précédent permet des constructions différées. Ce mécanisme peut être généralisé, pour cela nous avons directement inséré dans tout objet manipulé par GenDep un identifiant de type numérique permettant de retrouver l'objet construit sur la plateforme.

Cet ajout, complété par l'utilisation de la méthode "*fetch*" permet de décomposer les dispositifs en sous dispositifs plus simples. Il est par exemple possible de construire un premier dispositif où chaque utilisateur appartient au groupe des utilisateurs et possède un espace personnel puis de compléter avec un autre dispositif précisant, pour certains,

l'appartenance à un groupe particulier et l'accès à un espace collectif. Lors du second déploiement, la méthode "*fetch*" permet d'interroger la plateforme et de récupérer les identifiants des objets déjà construits en vue de les réinsérer dans le sous-dispositif.

10.7.2 Les applications Web 2.0

La méthode "*fetch*" s'est avérée particulièrement performante pour aborder des constructions de dispositifs pédagogiques sur des applications de type Web 2.0 (O'Reilly, 2005). L'application de l'ingénierie Bricoles à ce type de plateforme fait partie de nos perspectives de recherche que nous détaillerons chapitre 12, section 12.3.2. Le Web 2.0 est caractérisé par des applications dont la principale caractéristique est d'offrir à leurs utilisateurs des mécanismes de création individuels et collectifs portant sur les contenus et les moyens d'accès à ces contenus. Blog, Wiki, Réseaux sociaux, Mots clefs collaboratifs, Forum sont des exemples d'application Web 2.0. Les technologies utilisées par ces applications (Ajax, Web Services etc ...) permettent, par les fonctionnalités qu'elles promeuvent, une co-construction des espaces collectifs. Par exemple, l'utilisation de mots-clefs collectifs au sein d'un forum permet de dessiner une carte de ces mots clefs, illustrant les principales préoccupations du forum et offrant des moyens pour s'y déplacer. L'utilisation de ces applications à des fins éducatives entraîne de devoir prendre en compte un contexte auto-produit par les activités de la plateforme. Lors de la construction d'un dispositif, cette prise en compte nécessite de pouvoir interroger la plateforme et de pouvoir analyser la structure des dispositifs qui s'y trouvent. Les fonctionnalités que nous avons implémentées sont une première étape vers la prise en compte de ce contexte.

10.7.3 Des fonctions prévues mais pas encore implémentées.

Tant au niveau des services, que du constructeur, des fonctions ont été prévues mais n'ont pas été implémentées. En particulier les fonctions de destruction seraient utiles pour implémenter dans le constructeur une fonction d'annulation, il serait ainsi possible de détruire un dispositif construit. Cette fonction permettrait en outre de mettre en œuvre plus facilement des activités de bricolage s'appuyant sur des mécanismes cognitifs de type essai-erreur.

10.8 Tester le constructeur spécifique

10.8.1 Modèle de dispositif

Dans le cadre du projet PCDAI (cf 11.3 et annexe) le métamodèle et les dispositifs à construire ont été légèrement modifiés lors de la deuxième phase d'étude menée sur le **WikiniMST**. Pour cette phase de l'étude un métamodèle fusionné, a été réalisé (cf. Figure 84). Cette fusion a permis de simplifier l'usage des outils produits. Il est possible maintenant d'affiner et de contextualiser directement un modèle produit. Ce changement a nécessité de rendre invisible les transformations automatiques permettant de générer un modèle opérationnel. Le dispositif prévu a été amendé, seule une balise correspondant à un espace personnel a été conservée (cf. Figure 87), des espaces pour des projets de groupe ont été définis. L'aspect administration a été modélisé (cf. Figure 85).

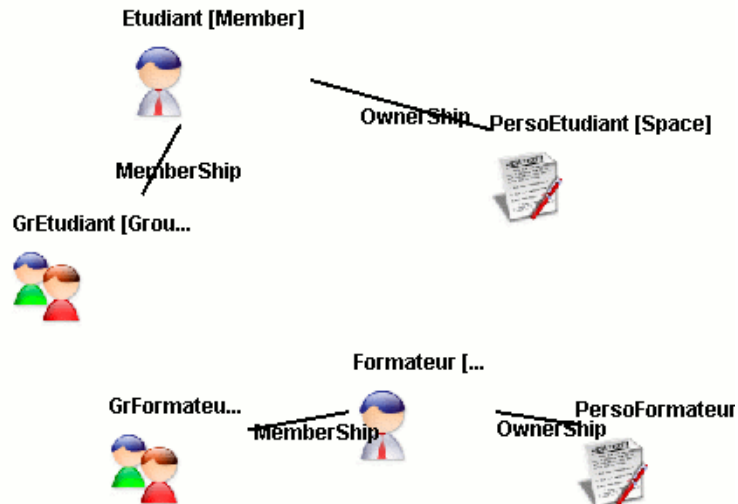


Figure 87 Dispositif espace personnel

10.8.2 Paramétrage de GenDep et première communication

Nous illustrons l'usage de notre infrastructure en explicitant la construction du modèle du premier dispositif (cf. Figure 87). Cette construction nécessite de paramétrer dans GenDep une instance spécifique de la plateforme WikiniMST.



Figure 88 Paramétrage d'une instance de la plateforme

La Figure 88 illustre le paramétrage d'une instance de l'application WikiniMST sur un serveur local : l'adresse du serveur de services Web et l'adresse du métamodèle doit être spécifiée.

Le constructeur spécifique charge le métamodèle ainsi que son environnement iconographique; il organise la construction du modèle de dispositif en analysant les dépendances impliquées par le métamodèle (cf. Figure 89).

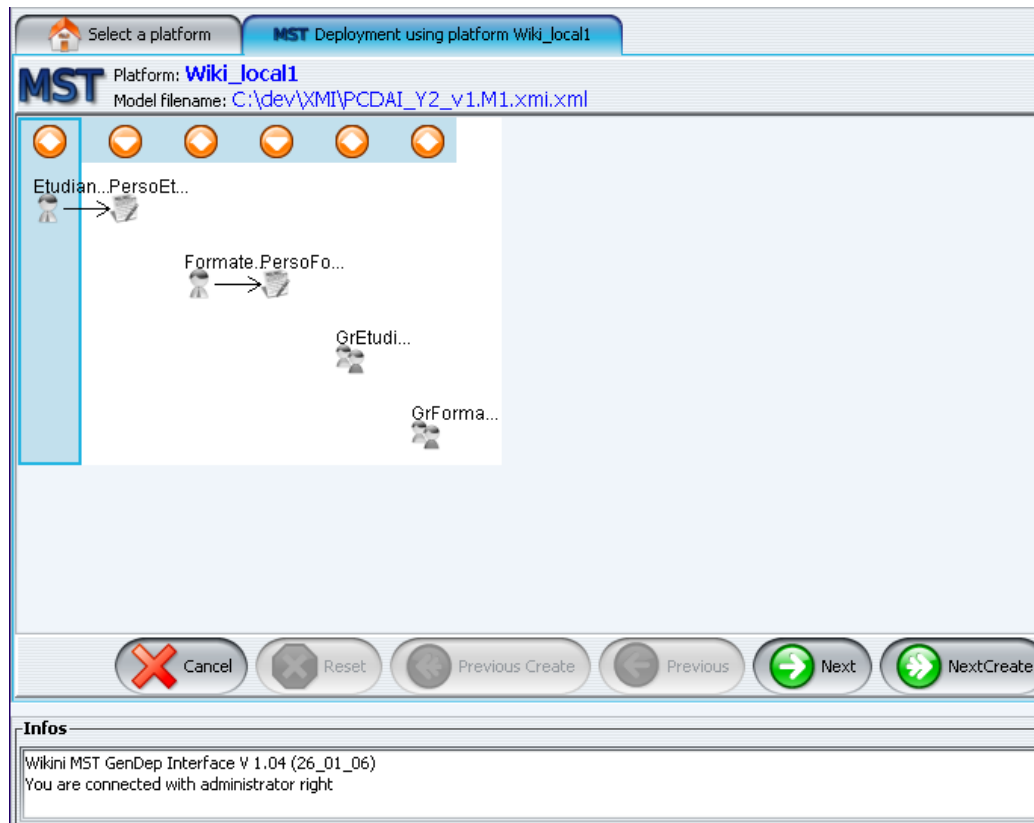


Figure 89 Lecture du métamodèle, interprétation du modèle.

Le constructeur effectue un premier essai de communication avec la plateforme, et récupère de celle-ci les indications concernant ses droits de construction.

10.8.3 Créer des collections primaires

La première collection d'objets à créer est une collection d'étudiants. Le constructeur spécifique analyse à partir du métamodèle la classe Member et construit dynamiquement la fenêtre de dialogue permettant la génération de la collection d'objet factorisée par l'élément étudiant.

Pour générer cette collection plusieurs solutions sont possibles. La Figure 90 illustre une génération de dix étudiants en utilisant un masque de profilage.

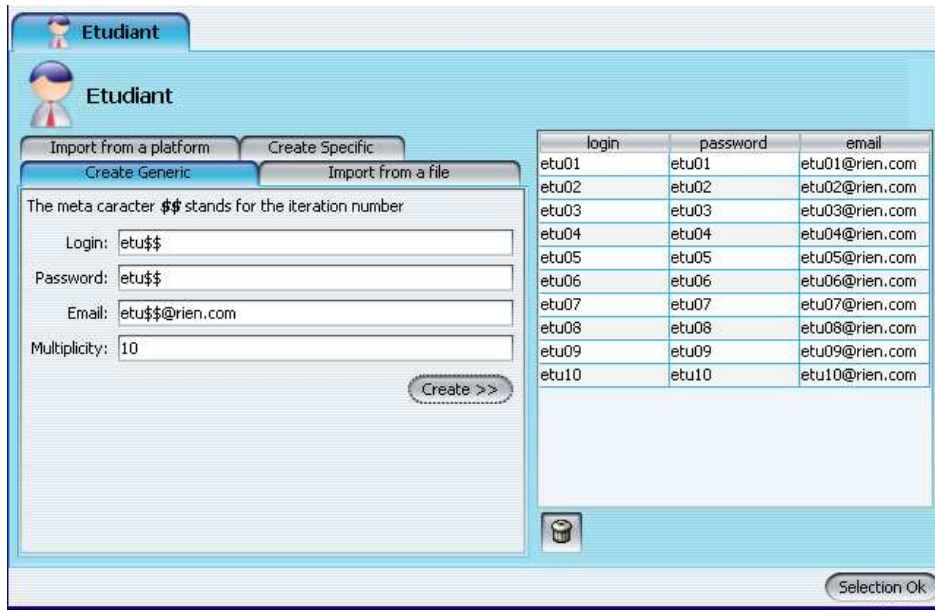


Figure 90 Génération itérative utilisant un masque de profilage

La Figure 91 illustre la récupération de 5 étudiants déjà sur la plateforme à partir de la fonction "fetch" (cf. section 10.5.4).

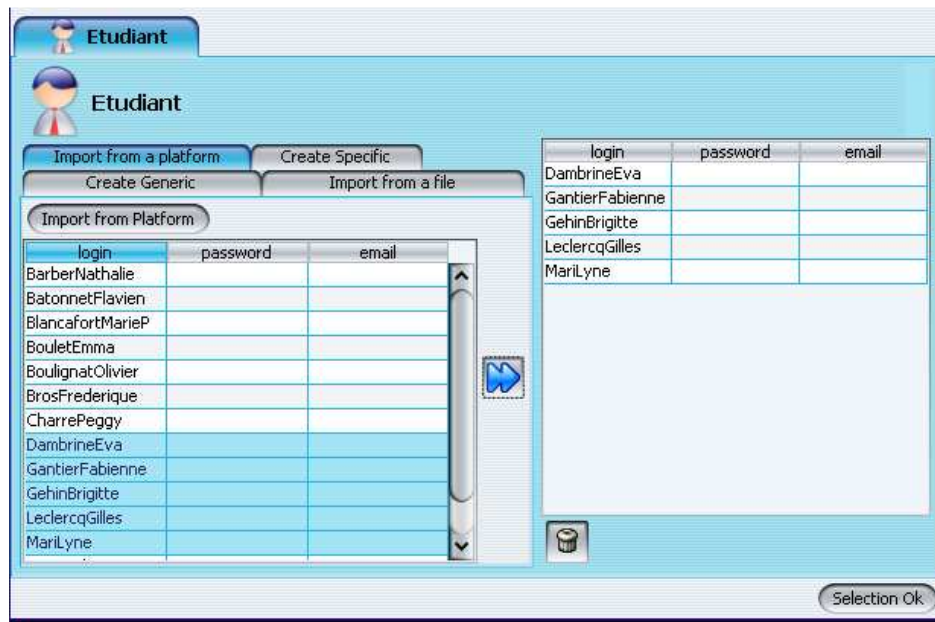


Figure 91 Communication avec la plateforme

La Figure 92 illustre la création de 7 étudiants à partir d'un fichier de type csv.

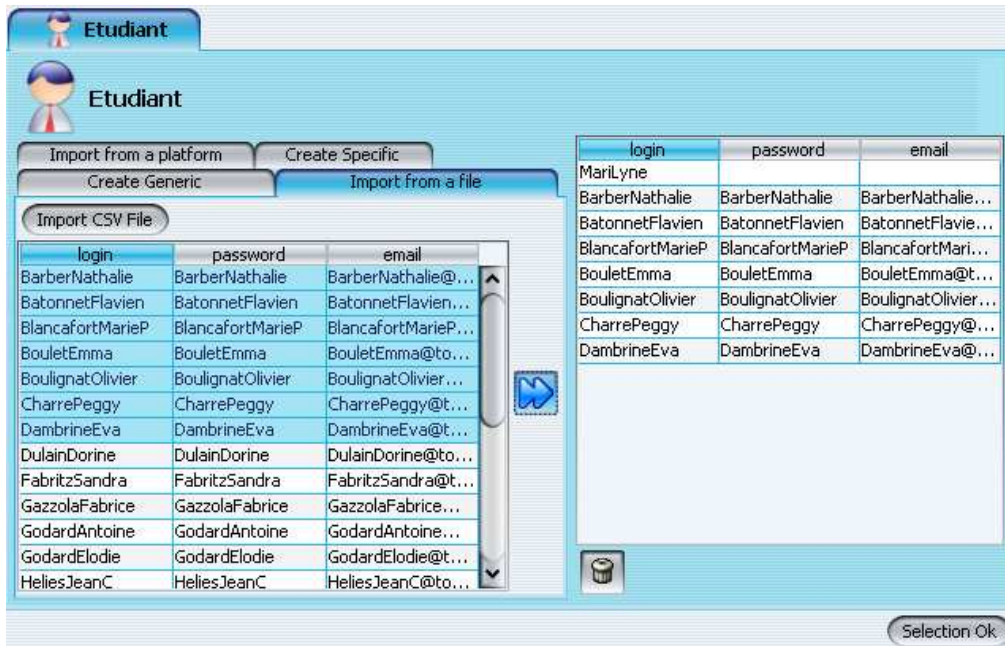


Figure 92 Génération à partir de fichier CSV

10.8.4 Créer des collections dépendantes

Une fois la collection d'étudiants créée, il faut créer les collections qui lui sont dépendantes. Dans cet exemple une seule collection dépend de la collection précédente, elle correspond à la création de la collection d'espace personnel, chaque étudiant dispose ainsi d'un espace précis, paramétré par la valeur des attributs de son propriétaire. Sur la Figure 93 le nom de l'espace correspond au nom Wiki de l'étudiant, des indications wiki permettent d'éditer les règles de lecture, commentaire et écriture. GenDep comporte un mécanisme permettant à un objet de faire référence aux attributs d'un objet auquel il est relié par un lien de composition. Par exemple sur la Figure 93 la valeur *\$login*, (ou *\$Member.login*) fait référence à l'attribut login de la classe **Member**.

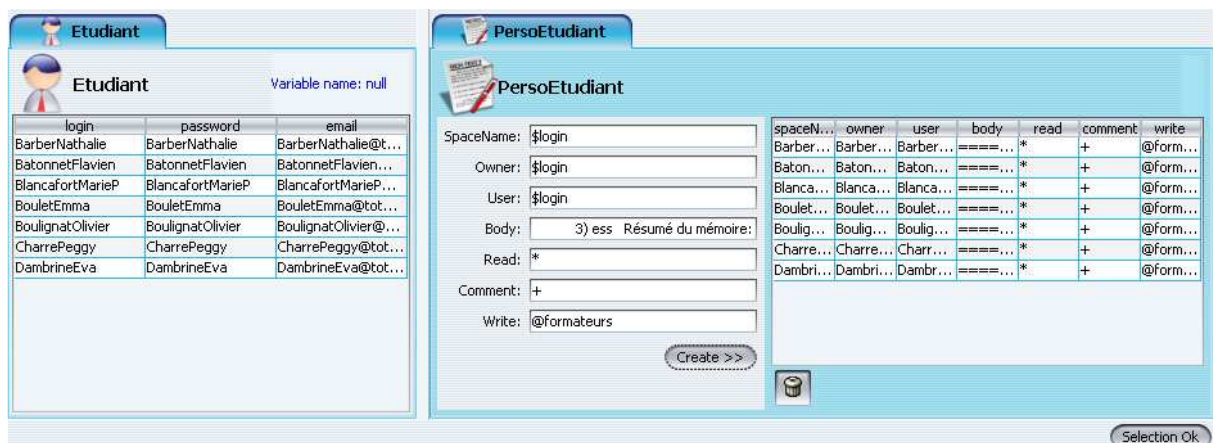


Figure 93 Création d'une collection dépendante

10.8.5 Comportement spécifique de GenDep, et construction

La création des autres collections est similaire, nous pouvons cependant nous arrêter pour commenter la création de la collection correspondant à la notion de groupe. Des indications de cardinalité (**GenDepCardinality**) ont été introduites dans notre métamodèle, ces indications sont interprétées par notre constructeur. Ainsi, une cardinalité indiquée, et tous les champs

déjà remplis dans le modèle, offre la possibilité d'effectuer automatiquement l'étape de génération des collections, (cf. Figure 95), bouton "**NextCreate**". La Figure 94 illustre la création d'une collection comportant un objet unique.

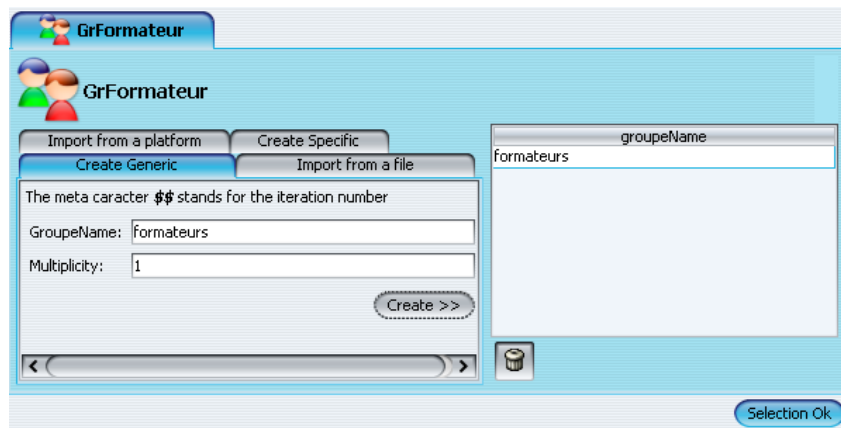


Figure 94 Création d'un objet unique

Cette création est d'ailleurs virtuelle comme cela est expliqué dans la section précédente, la notion de groupe n'étant réellement construite que lorsque les liens sont construits. La construction effective des collections intervient dans la dernière étape (cf. Figure 96).



Figure 95 Construction effective

Le constructeur, réordonne encore une nouvelle fois les collections d'objets pour optimiser le déploiement.



Figure 96 Résultat de la construction

Finalement, les dispositifs sont créés sur la plateforme (cf. Figure 96).

10.9 Conclusion

Ce chapitre évoque les bonnes pratiques liées à la création d'une infrastructure Bricoles spécifiques à une plateforme de formation. Une telle infrastructure permet la modélisation et la construction de dispositifs pédagogiques.

10.9.1 Résumé de notre ingénierie

Pour mettre en place cette infrastructure, la plateforme de formation doit disposer d'une interface de communication adressable par un outil informatique. Nous montrons que lorsque cette interface est absente il est possible de la réaliser. Cette interface peut par exemple prendre la forme d'un serveur de services Web à greffer sur la plateforme. Nous montrons dans ce chapitre un exemple de réalisation d'un tel greffon. Cette réalisation s'appuie sur l'élaboration de cas d'utilisation décrivant les différentes interactions entre utilisateurs et plateforme de formation. De ces interactions, les interactions possibles entre notre constructeur et la plateforme de formation sont déduites. Ce travail permet d'implémenter le greffon souhaité.

A partir de ce greffon et comme décrit dans la section 9.1.1, il est possible de définir par fusion ou par transformation le métamodèle, qui servira de support de conformité à la modélisation des dispositifs pédagogique.

Ce métamodèle et ce greffon peuvent être interprétés par deux tâches que nous avons implémentées et qui permettent de générer automatiquement 95% du code du constructeur spécifique à la plateforme de formation.

Ce constructeur doit être complété pour implémenter le service d'authentification et pour chaque objet les méthodes de déploiement et d'interrogation.

10.9.2 Usage de notre ingénierie

Nous illustrons l'usage de notre ingénierie par de nombreux exemples extraits d'études de cas menées lors de notre thèse (cf. chapitre 11). Ces études de cas mettent en œuvre :

- L'implémentation d'un greffon
- L'élaboration d'un métamodèle
- La définition des règles de transformations
- La spécification d'un modeleur spécifique
- La génération d'un constructeur spécifique et sa finalisation.

Le travail mené lors de ces études montre la faisabilité technique de notre approche conceptuelle. L'ingénierie Bricoles permet de réaliser facilement pour une plateforme de formation un modeleur et un constructeur de modèle de dispositif. Les choix techniques que nous avons effectués dès le début de cette thèse se sont avérés particulièrement bien adaptés aux évolutions des plateformes de formation, en particulier ils permettent d'adresser grâce à notre ingénierie des applications de type Web 2.0 et de faciliter leur utilisation pédagogique. En particulier, notre infrastructure favorise pour ce type d'application la mise en œuvre de tâches de contextualisation. A l'aide de l'infrastructure Bricoles, l'enseignant modélise son dispositif, éventuellement en plusieurs parties. Lors de la génération des éléments constituant le dispositif, il le complète avec des indications contextuelles propres à la promotion (liste d'étudiants etc...), ainsi qu'avec des éléments déjà construits sur la plateforme (groupe etc...). Cette possibilité permet non seulement de contextualiser finement un dispositif avec un usage en cours d'émergence, mais aussi de composer les dispositifs à construire, en séparant différents aspects d'un dispositif global.

10.9.3 La complexité du travail réalisé

L'outil GenDep compte actuellement environ 80 000 lignes de codes. Cependant, ce nombre à lui seul, est un très mauvais indicateur du travail réalisé, en effet l'outil est composé d'un noyau (40 000 lignes de codes) et d'une série de modules correspondant aux plateformes abordées. Ces modules sont générés à 90% par les tâches que nous avons décrites section 10.5. La complexité du travail réalisé se situe donc plutôt dans la réalisation des métamodèles et des serveurs de services Web permettant de générer le code de GenDep. La complexité de notre travail réside également dans la volonté de créer une architecture facilement extensible, utilisable pour adresser tout type de plateforme. Si la tâche transformant un fichier WSDL est classique, par contre le travail réalisé sur la tâche générant du code à partir des métamodèles exprimé en XMI est plus novateur. Ce travail a nécessité de mener une réflexion sur le sens à donner en IDM à la relation de description. Cette réflexion a conduit à notre proposition : **générer directement, à partir du métamodèle, l'implémentation des interfaces permettant de définir une collection à partir d'un élément du modèle passé en paramètre.**

Outre la réalisation de GenDep, du travail de réingénierie a été mené sur des plateformes de formations, (Ganesha, Claroline, Moodle, WikiniMST, Elg). Ce travail a permis de créer, pour chacune de ces plateformes, un serveur de services Web. Malgré la diversité des langages de programmation, les difficultés rencontrées ont essentiellement concerné les protocoles d'authentification, et le problème posé par le changement de logique dû à l'absence de session côté serveur. Enfin nous avons participé à la finalisation de la plateforme de modélisation ModX et à son adaptation aux domaines des EIAHs, (transformation, patron, placement automatique, script d'habillage etc...).

chapitre 11 Détail des études de cas menées

11.1 Introduction

11.1.1 Préambule

Notre proposition d'une nouvelle démarche d'ingénierie des EIAHs est difficile à prouver scientifiquement, comme d'ailleurs bon nombre de propositions méthodologiques. Dans l'absolu, on pourrait dire que seul un usage installé et durable serait la preuve pratique du bien fondé de notre proposition. Mais cela n'est pas compatible avec la durée d'une thèse. Aussi avons-nous essayé de prouver la validité de nos propositions de deux manières :

- Une étude assez exhaustive de la faisabilité (implémentation) de nos propositions.
- Le traitement, au cours de cette thèse, d'études de cas significatives, en lien avec des terrains opérationnels de formation.

C'est à cette seconde manière que le chapitre 11 est consacré, au travers de 3 études de cas bien distinctes qui se sont déroulées à des périodes différentes de la thèse.

11.1.2 Présentation des études par ordre chronologique

La première étude de cas a permis d'aborder deux points évoqués par notre thèse :

- L'adaptation du standard IMS-LD, par la définition de règles de transformation permettant de transformer un modèle de cours IMS-LD en un modèle de cours spécifique à la plateforme Ganesha.
- La définition d'un système de communication entre la plateforme Ganesha et notre constructeur. Cette adaptation a permis d'étudier la faisabilité de notre démarche quant aux communications à établir pour l'atelier "Outillage", elle a été menée en collaboration avec les concepteurs de la plateforme Ganesha.

L'expérience a permis de faire évoluer nos choix techniques et sociaux et de remettre en cause l'interface de communication que nous souhaitions développer. Il est en effet apparu au cours de l'étude de cas que les choix conceptuels sur lesquels était conçu notre atelier outillage pouvaient ou non favoriser une implication des concepteurs de la plateforme. Notre choix s'est de fait tourné vers une solution accroissant leur implication.

La seconde étude, s'est déroulée sur deux ans, et a adressé l'application Web 2.0 WikiniMST. Cette étude est intervenue dans le cadre du Projet PCDAI décrit au chapitre 5.3.3. Elle a permis de faire intervenir des enseignants dans le cadre de deux ateliers, le premier concerne la mise en place d'une infrastructure spécifique à l'application, le second concerne son usage par des enseignants. L'étude dans le cadre du premier atelier a essentiellement permis d'explorer les modalités de définition, par des enseignants, d'un ou de plusieurs métamodèles nécessaires à notre ingénierie. Cette étude a été l'occasion pour nous de d'aborder plusieurs solutions évoquées dans la section 9.2.5, concernant les problèmes de fusion ou de transformation de modèle. La spécificité des applications Web 2.0, a permis également de compléter notre modèle d'interaction (entre constructeur et application Web) en abordant le

problème posé par la contextualisation des dispositifs à construire. Dans le cadre du second atelier, l'usage de notre infrastructure par des enseignants pour modéliser de véritables dispositifs a permis de remettre en cause notre travail précédent. Les améliorations ont principalement porté sur la nécessité de proposer un métamodèle simple et fusionné, et de proposer un mécanisme permettant de composer et de contextualiser les dispositifs à construire.

La troisième étude, encore en cours à cet instant, prend la forme d'une recherche action portant sur l'adaptation de la méthode de suivi de projet étudiant *Mepulco-Université* (Talon, Toffolon, & Warin, 2005.), (Talon, Toffolon, & Warin, 2007) dans le cadre de sa mise en œuvre sur une plateforme de formation. Cette étude adresse la plateforme de formation Open Source Moodle et porte essentiellement sur la mise en place de dispositifs requis par une méthode. L'étude s'appuie sur un travail mené depuis 5 ans par le Laboratoire Informatique du Littoral (LIL), les étapes de cette étude prévoient :

- Une adaptation manuelle des dispositifs pour leur médiatisation dans le cadre de la plateforme Moodle.
- Une définition des concepts propres à la méthode et à la plateforme Moodle
- La création d'une infrastructure Bricoles adaptée à l'usage de la plateforme *Moodle* par la méthode *Mepulco-Université*.

Nous espérons par ce travail, réinterroger l'ingénierie Bricoles, en révélant ses insuffisances pour prendre en compte une méthode produisant des dispositifs. Certes, les dispositifs produits sont conformes au métamodèle fonctionnel de Moodle et au métamodèle pédagogique de la méthode *Mepulco-Université*. Mais cette conformité ne suffit pas à décrire le cadre de création des dispositifs impliqués. Nous détaillerons, à la fin du compte rendu de notre étude, les pistes que nous explorons actuellement pour palier à cette insuffisance.

11.2 La première étude : le travail réalisé avec la plateforme Ganesha

11.2.1 Résumé et liens avec la thèse

Notre étude porte sur la modélisation de cours au sens large, scénario, ressource, dispositif, dans une optique de déploiement ou de transfert sur une plateforme de formation. Cette étude aborde le problème de l'interopérabilité des plateformes de formation au travers l'adaptation par une ingénierie de type IDM, du standard IMS-LD. Sur le plan conceptuel, notre proposition met en place une démarche de type MDA, et propose diverses méthodes pour l'opérationnaliser.

11.2.2 Problématique de l'étude

La problématique initiale de l'étude est illustrée par la Figure 97, elle concerne la réutilisation, d'un cours, son transfert et son partage.



Figure 97 Ré-utilisation d'un cours sur diverses plateformes

Dans le cadre des enseignements que nous donnons au Centre Universitaire pour l'Education Permanente (CUEEP) (centre qui accueille notre laboratoire), il faut pour chaque promotion, mettre en place les structures (groupe, atelier, outils) et les objets pédagogiques requis. Ce travail intéressant pour un enseignant lors de sa première approche des plateformes de formations se révèle vite rébarbatif. Dans la mesure où d'années en années nous constatons un appauvrissement des structures mise en place par les enseignants, cette construction manuelle des structures et objets constitue un obstacle au suivi pédagogique de la formation.

Cette difficulté à réutiliser un cours est aggravée lorsque l'enseignant doit migrer ses enseignements sur une autre plateforme. Cette situation intervient assez naturellement dans un institut participant à de nombreux projets pluri-partenaires. Il n'est pas rare alors que les enseignants doivent intervenir dans des formations qui ne sont pas exclusivement hébergées par le CUEEP.

Transférer un cours sur une autre plateforme nécessite de connaître les concepts attachés à la nouvelle plateforme et de traduire manuellement ce cours. Ce travail est facile à réaliser grossièrement, mais conduit en pratique très fréquemment à un appauvrissement des activités prévues par le cours. Cet appauvrissement est un obstacle à l'évolution technologique d'une formation.

Enfin, partager un cours entre enseignants nécessite une compréhension mutuelle du cours impliquant une compréhension de la plateforme l'hébergeant. Cette compréhension est difficile à obtenir entre des usagers de plateformes différentes. La partie pédagogique est en effet souvent sémantiquement exprimée dans les termes introduits par la plateforme. Cette dépendance entraînant un parasitage des échanges en mêlant préoccupation pédagogique et préoccupation technique.

Ces trois problèmes, réutilisation, transfert et partage pourraient être résolus si le cours pouvait être exprimé dans une sémantique indépendante de la plateforme de formation.

Une solution consiste donc à exprimer les cours dans un langage de modélisation indépendant de la plateforme de formation. Choisir un langage proche des enseignants, leur permettraient de s'exprimer dans un langage qui ne soit pas assujéti aux limitations sémantiques de la plateforme de formation qu'ils utilisent. Le cours exprimé par des concepts "métiers" plutôt que "techniques" serait, de plus, compréhensible pour un autre enseignant et donc échangeable et réutilisable. Mais modéliser dans un langage adapté à l'univers enseignant nécessite non seulement d'identifier un tel langage mais aussi de définir un mode de persistance, une démarche et un outillage permettant de transformer l'expression métier du dispositif en une instance informatique de dispositif sur la plateforme de formation.

Comme langage indépendant, il semble pertinent d'utiliser IMS LD (IMS-Learning Design). Cependant, ce langage n'est qu'une partie de la solution, c'est ce que nous montrons dans la partie suivante.

11.2.3 Présentation d'IMS-LD et intérêt pour notre problématique

IMS Learning Design (IMS LD) est une spécification du consortium IMS qui permet la description de processus pédagogiques essentiellement électroniques (cf. chapitre 4.2.2).

Dans notre contexte d'implantation de dispositif pédagogique, IMS-LD peut être utile pour permettre aux enseignants de définir ces dispositifs de manière précise, indépendamment de toutes technologies/plateformes et de la pédagogie sous-jacente au cours. Il peut constituer un langage pivot pour décrire, sauvegarder, échanger, composer, déplacer de tels dispositifs (Pernin & Lejeune, 2004a). Cependant IMS-LD se révèle être un mauvais candidat pour conduire concrètement, à partir du fichier XML généré, une implantation sur une plateforme de formation. Nous rappelons succinctement ici les arguments développés au chapitre 4.2.3.

D'une part il est difficile pour un enseignant d'aborder ce langage de modélisation (F. De Vries, Tattersall, & Koper, 2005), (Santos, Boticario, & Barrera, 2004). Bien que des outils comme (Reload), ou (MOT Plus) permettent l'édition de fichier IMS-LD, les recherches actuelles se placent plutôt dans l'hypothèse d'un langage orienté métier permettant d'assister cette édition (De la Teja et al., 2006). D'autre part, il est difficile d'exprimer en IMS-LD les spécificités d'une plateforme donnée, et d'effectuer informatiquement la projection entre les fonctionnalités du scénario exprimé en IMS-LD et les fonctionnalités réellement disponibles (Berggren et al., 2005).

11.2.4 Notre proposition pour cette étude de cas n°1

Au vu de ces deux points, il ne semble pas envisageable de prévoir l'implantation automatique de scénario pédagogique, au sens où nous les entendons, à partir du langage IMS-LD. Notre démarche consiste donc à considérer IMS-LD comme une étape intermédiaire entre la description en langage naturel du scénario et l'expression de son modèle dans un langage reprenant les spécifications de la plateforme. Ce langage qui décrit les spécifications de la plateforme constitue dans notre proposition un métamodèle (métamodèle spécifique à la plateforme). Ce métamodèle permet de générer l'infrastructure qui en se nourrissant du modèle élaboré par l'enseignant, procède la construction du dispositif pédagogique. Cette démarche est inspirée du MDA elle permet de considérer le langage IMS-LD comme un langage pivot sur lequel fonder l'interopérabilité des cours entre les plateformes de formation. (cf. Figure 98)

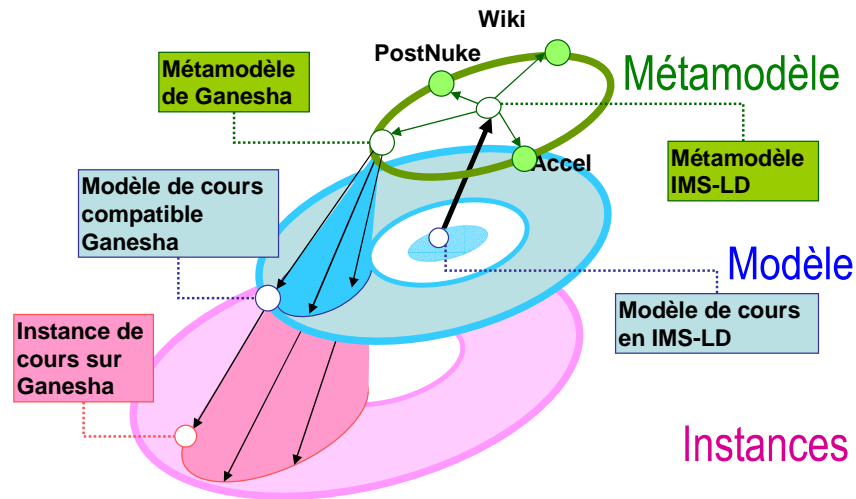


Figure 98 IMS-LD un langage pivot pour l'interopérabilité des cours sur des plateformes de formation.

Nous proposons dans cette étude de modéliser un cours en IMS-LD, puis d'utiliser au sein de ModX, les règles de transformation établies entre le métamodèle IMS-LD et le métamodèle de la plateforme Ganesha. Appliqué au modèle IMS-LD ces règles permettent de générer un modèle de cours compatible avec la plateforme Ganesha. L'utilisation de GenDep notre constructeur de modèle permet alors de créer une instance de cours sur la plateforme Ganesha. Pour cette étude, notre proposition est aussi illustrée par la Figure 34 au chapitre 9.1.2.

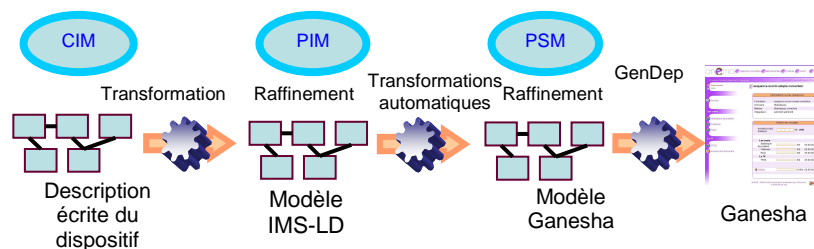


Figure 99 Démarche de l'étude Ganesha

Notre démarche est composée de quatre phases illustrées par la Figure 99. Ces phases correspondent à : la description du dispositif dans un langage non computationnel (CIM), la modélisation en langage IMS-LD (PIM), la traduction de ce modèle pédagogique vers la plateformes Ganesha (PSM) et enfin l'écriture, par l'utilisateur, des éléments contextuels et donc non automatisables pour obtenir des instances utilisables.

L'usage de cette démarche correspond au scénario suivant :

L'enseignant choisit la plateforme ciblée (parmi les métamodèles définis par l'informaticien et l'ingénieur pédagogique). Avec **l'ingénieur pédagogique**, ils expriment le scénario dans le métamodèle IMS-LD. Pour transformer ce modèle de cours IMS-LD en un modèle de cours pour la plateforme ciblée, l'enseignant choisit des règles de transformation. La génération du modèle dans le métamodèle associé à la plateforme ne peut pas être une génération sans perte. En particulier, les différences, les incohérences qu'elle souligne entre les deux métamodèles sont véritablement porteur de sens et font la richesse de la démarche. Une fois le modèle généré, complété et exprimé dans le métamodèle associé à la plateforme ciblée, l'enseignant procède à sa construction pour cela il indique au dépoyeur GenDep : le modèle, l'url de la plateforme et lance les requêtes nécessaires à la création d'une instance du

scénario pédagogique.

11.2.5 Réalisation autour du standard IMS-LD

Le métamodèle IMS-LD a été défini dans ModX (cf. Figure 101).

Ce métamodèle a fait l'objet de nombreux ajouts pour le rendre compatible avec l'outil Reload⁸. Reload est un outil de modélisation pédagogique dédié aux standards SCORM et IMS-LD. Réalisé par l'université de Bolton, il constitue dans le domaine des EIAHs, une référence.



Figure 100 Compatibilité ModX Reload

Des outils de transformation entre un fichier XML IMS-LD et modèle de cours conforme au métamodèle ont été définis (cf. Figure 100).

⁸ <http://www.reload.ac.uk/>

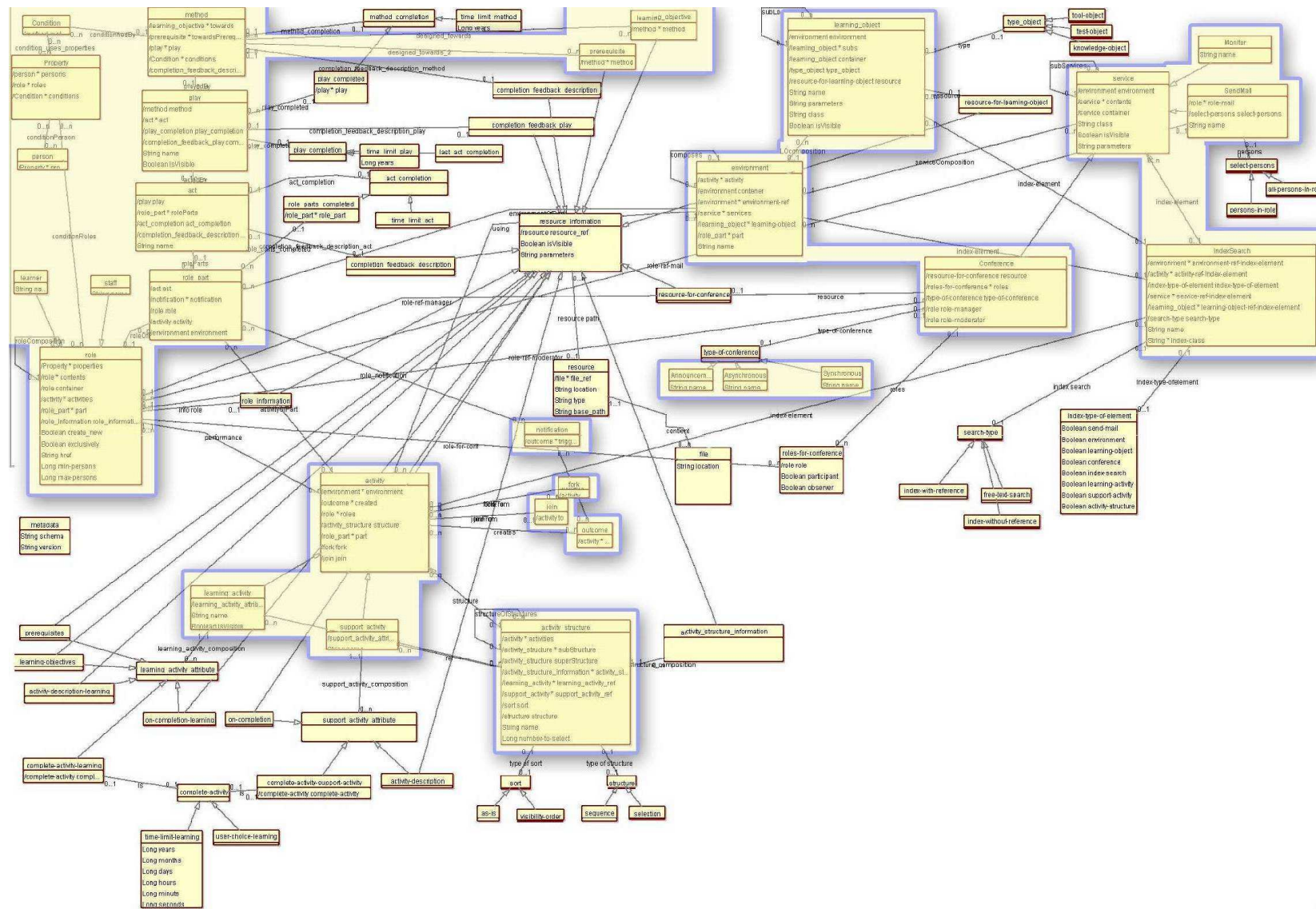


Figure 101 Métamodèle IMS-LD avec grisés les ajouts motivés par la compatibilité avec Reload

Des patrons, facilitant au sein de ModX, l'appréhension par des enseignants du standard IMS-LD, ont été définis.

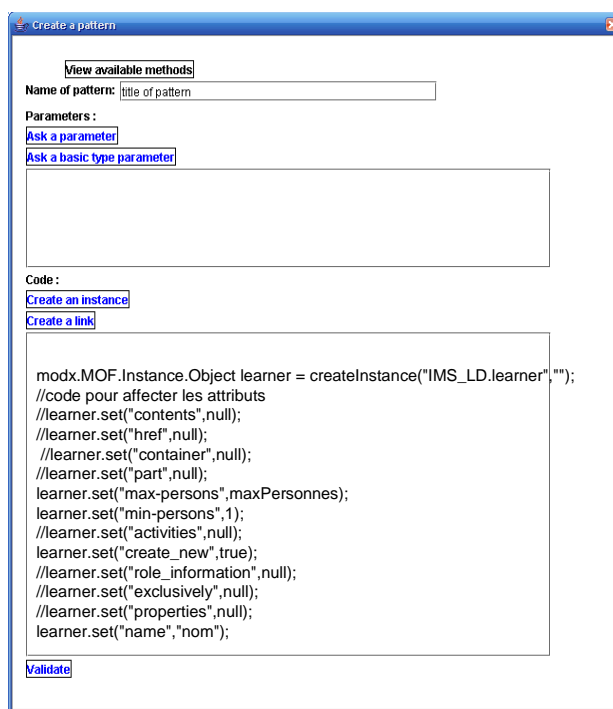


Figure 102 Editeur de pattern facilitant la modélisation IMS-LD

La Figure 102 présente notre éditeur de patrons. Au sein de ModX un patron facilite la modélisation, en organisant les différentes phases requises par cette activité. Différents patrons ont ainsi été définis parmi lesquels un patron correspondant à la mise en œuvre IMS-LD d'activités pédagogiques de type Jigsaw (Aronson & Patnoe, 1997). Pour un métamodèle aussi complexe que le métamodèle IMS-LD, ce mécanisme semble prometteur. Cependant le travail fourni reste très expérimental, en particulier il conviendrait d'offrir la possibilité de composer les patrons. Notre proposition a été complétée par l'établissement de règles de transformation entre IMS-LD et le métamodèle de la plateforme Ganesha, ce travail a été décrit précédemment au chapitre 9.3.

11.2.6 Construire un modèle, sur Ganesha

Cette étude a permis de faire évoluer notre approche, la Figure 103, présente deux solutions que nous avons explorées.

La première solution s'est appuyée sur un générateur d'appel de requête http. Dans GenDep on définit un constructeur spécifique comme expliqué au chapitre 10.5. La création des éléments du dispositif à construire est réalisée via des méthodes qui simulent sur l'interface Web de la plateforme les interactions d'un utilisateur, (appui de bouton, envoi de formulaire etc.).

Cette méthode est pertinente car elle permet d'adresser tout type de plateforme, sans participation de l'équipe des développeurs de celle-ci. Cependant, d'une part la fonction d'authentification des plateformes c'est avérée être souvent très difficile à simuler et a nécessité d'employer des techniques plus proches du *"hacking"* que de la réingénierie (analyse des datagrammes lors des session d'authentification). D'autre part les constantes évolutions auxquelles sont soumis les plateformes ne permettent pas d'assurer facilement une maintenance du code du constructeur spécifique.

La deuxième solution est née de la collaboration que nous avons menée avec les développeurs de la plateforme Ganesha⁹. Cette collaboration nous a permis de définir une approche plus pérenne basée sur l'emploi de Web Services, cette approche est décrite au chapitre 10.1. Cette solution nécessite la présence d'une interface de Web Service encapsulant l'API de la plateforme de formation. Cette interface qui existe pour certaines applications peut être développée sous forme d'un greffon à ajouter à la plateforme de formation. Lorsque le code de la plateforme est ouvert son implémentation est très aisée. Sinon un travail de coopération avec des développeurs de la plateforme de formation, ou un travail de réingénierie important permet de la réaliser.

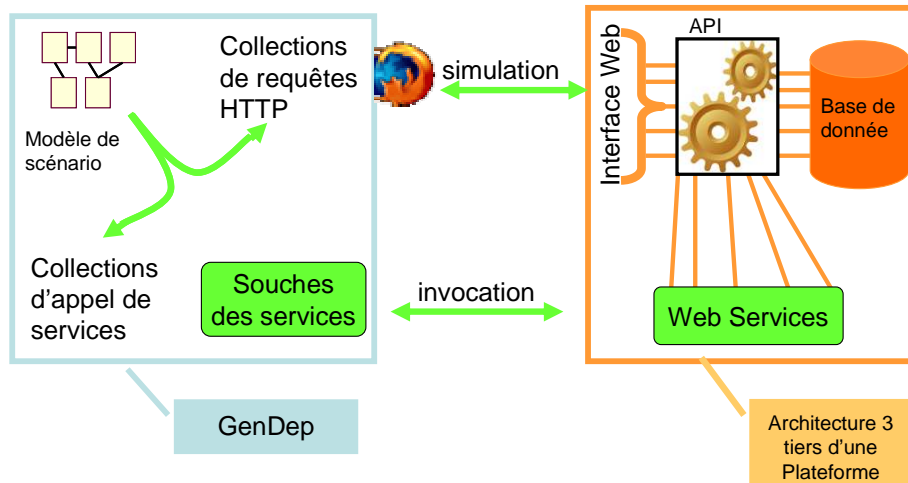


Figure 103 Les deux approches explorées par cette étude

Pour l'application Ganesha, l'interface de Web Service a été réalisée par notre équipe, l'implémentation des services a quant à elle été réalisée par les développeurs de la plateforme.

11.2.7 Exemple de Mise en œuvre

Cet exemple de mise en œuvre s'appuie sur un scénario et un dispositif impliqué par un "cours de remise à niveau java" décrit dans le chapitre 5.2, (cf. Figure 14). A l'aide de ModX, l'enseignant décrit le scénario et définit les intervenants (les étudiants et le tuteur) et les différentes phases se déroulant en parallèle : cours, exercices et projet. La Figure 104 indique ce qu'il obtient pour la phase/pièce cours.

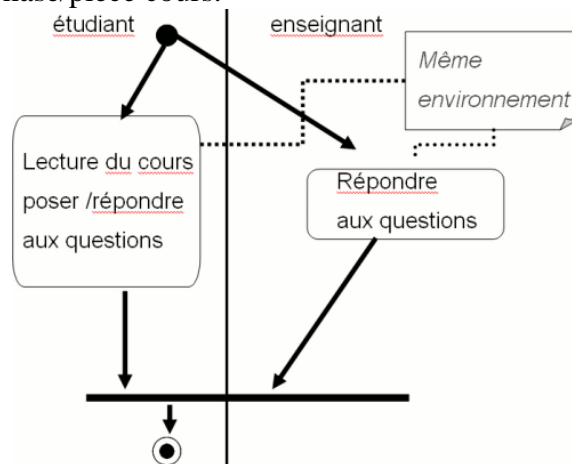


Figure 104 Diagramme d'activité simplifié de la phase/pièce 'Cours'

⁹ Société Anema, George Caldeira, Marc Vincent.

En s'appuyant sur le métamodèle IMS LD définie dans ModX et l'enseignant établit le modèle du dispositif, exprimé sous forme d'un modèle conforme à IMS-LD (cf. Figure 105).

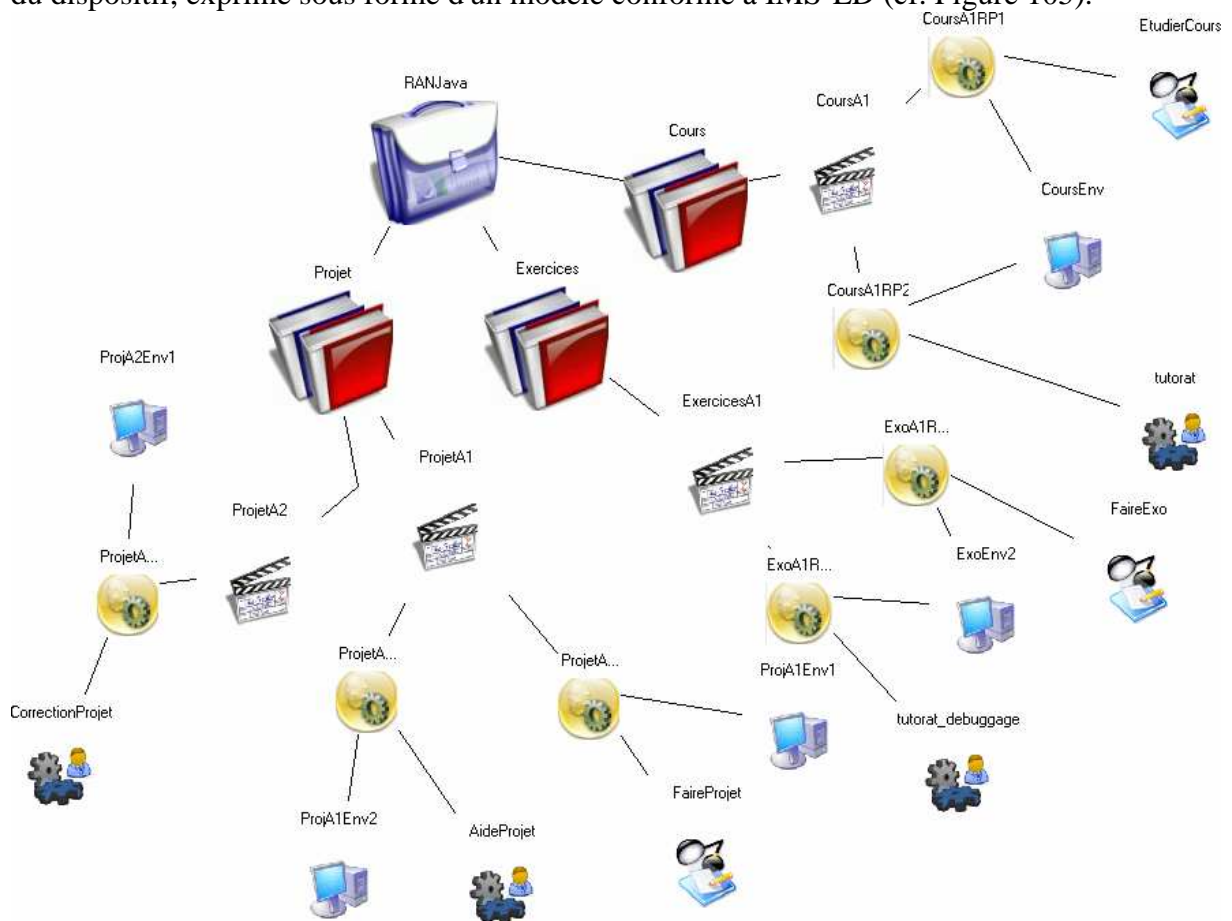


Figure 105 Diagramme de classe du cours

Dans le modèle ci-dessus, le modèle du dispositif "cours de remise à niveau java" a été représenté. La plateforme visée pour construction du dispositif est Ganesha. Cette plateforme est très différente de la plateforme Accel dont est issu le dispositif. En particulier le métamodèle de la plateforme Ganesha implique lorsqu'un groupe est défini, que lui soient associés automatiquement un forum et un espace de messagerie unique, une zone unique de dépôt de documents, un espace de discussion synchrone ne comportant qu'une seule salle. L'ensemble des services est attaché au groupe d'étudiants, plutôt qu'à une activité précise. La Figure 106 présente le premier métamodèle de l'application Ganesha que nous avons défini.

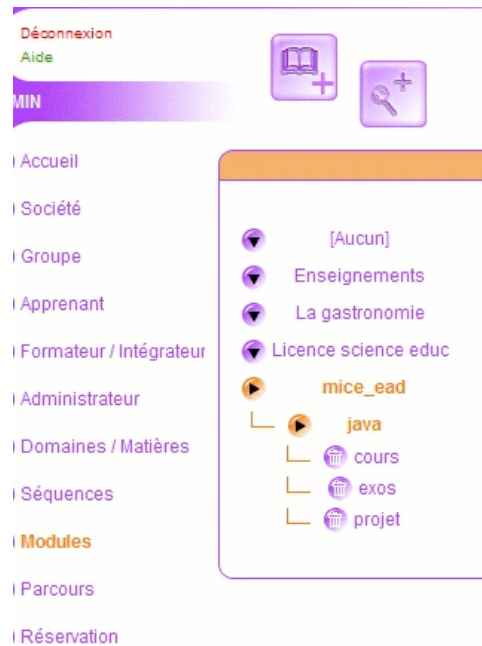


Figure 108 Dispositif construit sur la plateforme Ganesha

11.2.8 Conclusion et constat de l'étude

Cette étude a permis d'explorer une proposition d'ingénierie permettant la construction de dispositifs exprimés dans le formalisme IMS-LD, sur la plateforme Ganesha. Cette mise en œuvre débute par la description de la partie pédagogique d'un scénario. Cette partie pédagogique est exprimée sous la forme d'un modèle de scénario conforme au métamodèle IMS-LD. Ce modèle est ensuite transformé en un modèle de dispositif spécifique à la plateforme de formation puis construit sur celle-ci. Les avantages théoriques de notre proposition sont :

- de partager avec d'autres tuteurs le cœur même d'un scénario pédagogique
 - pas de réduction sémantique dûe au pouvoir limité d'expression d'une plateforme
 - pas de "pollution" technique.
- d'accélérer la mise en place d'un dispositif au sein d'une plateforme. Le déploiement automatique à partir d'un parcours logique (avec d'éventuelles modifications) sera plus rapide que la retranscription/redéploiement manuel.
- De favoriser une compréhension plus rapide des plateformes.

Cependant force est de constater que cette étude a mis en évidence bien plus de problèmes qu'elle n'en a résolus:

- La modélisation IMS-LD s'est avéré être difficile à mettre en œuvre par un enseignant, d'où notre soucis de maintenir une compatibilité avec l'outil de modélisation IMS-LD Reload, et de mettre en œuvre une approche par Pattern.
- Le mécanisme de transformation, comporte par essence un paradoxe difficile à surmonter:
 - automatique, il ne favorise pas la compréhension du métamodèle spécifique à la plateforme,
 - manuel, il est difficile et laborieux à mettre en œuvre.
- Le mécanisme de construction, par simple simulation d'un explorateur Web ne permet de mettre en place une solution pérenne.

Sur ce dernier point cette étude a permis de tester un partenariat avec les développeurs de la plateforme Ganesha, notre proposition de construction en a été changée. Plus qu'une simple

alternative technique, l'utilisation de greffons de Web Services permet de construire de l'intérieur même de la plateforme les dispositifs envisagés. Cette solution permet de prendre en compte l'évolution des plateformes en leur proposant une interface qu'il suffit de maintenir inchangée. Cette étude ouvre de nouveaux chantiers en suggérant par exemple d'établir avec la plateforme des dialogues impossibles à mettre en œuvre à partir de son interface Web. Ce chantier est abordé dans l'étude suivante. Celle-ci tente de remédier, par les solutions qu'elle explore, aux obstacles mis en évidence par notre première étude.

Diffusion de l'étude de cas n°1 portant sur le mécanisme de communication avec le plateforme Ganesha.

Conférences nationales avec comité de lecture : Rencontre jeune chercheur 2006(Caron, 2006)

Conférence internationale avec comité de lecture : E Learn 2005 (Caron et al., 2005a), AIED 2005 (Caron, Derycke, & Le Pallec, 2005b).

Chapitre de livre (Laforcade, Nodenot, Caron, & Choquet, 2007)

Participation à Ecole d'été des EIAHs 2005, animation d'un module SCORM et interopérabilité

11.3 La seconde étude : le travail réalisé avec l'application Web, WikiniMST

11.3.1 Résumé et liens avec la thèse

Cette étude de cas n°2, présente une application de l'ingénierie Bricoles à la modélisation et à la construction de dispositif sur une application de type Web 2.0. Elle intervient dans le cadre d'un projet de recherche portant sur l'usage de l'internet. Cette étude s'est déroulée en deux phases d'une année chacune. Pour cette thèse, les apports de cette étude sont les suivants :

Première phase de l'étude : (première année)

- L'utilisation et la formalisation de la notion de dispositif,
- L'utilisation, la distinction et la formalisation de la notion de métamodèle pédagogique et de métamodèle fonctionnel.

Deuxième phase de l'étude : (deuxième année)

- L'utilisation de mécanisme de fusion et de transformation entre métamodèles.
- La mise en place de services permettant la contextualisation des dispositifs à construire.

Enfin, cette étude, menée sur deux ans, a permis d'évaluer auprès d'enseignants l'usage réel de l'infrastructure produite par l'ingénierie Bricoles.

11.3.2 Cadre scientifique de l'étude

La recherche Pratiques Collectives Distribuées d'Apprentissage sur Internet (PCDAI) est un projet de recherche projets pluri partenaires financé par la Direction de la Technologie (du MENRT) dans le cadre de l'appel à projet "usages de l'Internet" (2004-2006). Durant cette recherche, quatre projets de mise en place d'un environnement numérique ont été menés en parallèle par des chercheurs du laboratoire TRIGONE et du laboratoire GERIICO. Ces projets peuvent être regroupés en deux sous-ensembles : d'une part, l'accompagnement de la rédaction d'un mémoire professionnel et d'autre part, la réalisation d'un événement professionnel. Gilles Leclercq, Professeur des universités en sciences de l'éducation et Dominique Delache, enseignant chercheur en sciences de l'éducation à l'Institut CUEEP, tous deux membres du laboratoire TRIGONE sont les responsables du sous-chantier : "*Co-construction et évolution d'une infrastructure, déploiement et mise en exploitation*". Ce sous

chantier relève de la thématique "*Accompagnement de mémoires professionnels*". L'expérimentation menée par ces derniers concerne l'accompagnement collaboratif à distance de stage et de rédaction de mémoire professionnel destiné à des étudiants suivant une licence professionnelle.

11.3.3 Les prémisses de l'étude

Dans un article introductif (Leclercq, 2005), Gilles Leclercq présente une étude de cas menée précédemment au projet PCDAI, à partir de laquelle il imagine un cadre de travail, mêlant formation et recherche, qu'il souhaite mettre en place. Cette étude de cas concerne l'écriture de mémoire professionnel et a été hébergé sur une plateforme de formation. Elle lui a permis d'enquêter sur l'émergence et le développement d'une communauté d'apprentissage. Cette enquête l'a conduit à poser quelques conclusions partielles qui conditionnent notre étude actuelle. Ainsi dans cet article Gilles Leclercq, propose d'envisager un dispositif de formation comme une offre de signification. Cela permet selon lui de spécifier pour l'étudiant des zones de développement potentielles. En observant la genèse de l'écriture des mémoires professionnels, il montre que pour organiser "*le processus d'écriture et la nidification de l'action*" il suffirait de structurer le dispositif pédagogique par quatre balises :

- prendre place dans une organisation, construire son tutorat ;
- formaliser sa mission et la questionner, raconter comment elle évolue ;
- mener des investigations de terrain ;
- mener des investigations conceptuelles.

Il montre également qu'au terme de l'écriture de leur mémoire, les étudiants pourraient participer à la production d'une mémoire collective. L'exploitation d'une telle mémoire pourrait devenir la clef d'une "organisation apprenante", autrement dit, d'une communauté d'expérience capable de nourrir une communauté d'apprentissage. Au cours de ce processus, les balises identifiées dans la première phase d'écriture seraient probablement de bons repères pour organiser la navigation.

En conclusion de ce texte, Gilles Leclercq évoque un cahier de charge qu'un environnement structuré par un tel dispositif pédagogique devrait respecter pour être adapté à l'écriture de mémoires professionnels.

11.3.4 Le cahier des charges imaginé pour la première phase de l'étude.

Dans un cours texte non publié, Gilles Leclercq nous décrit l'environnement qu'il souhaiterait mettre en place pour l'étude du projet PCDAI. Ce texte précise le cahier des charges évoqué dans l'article précédent, et explicite l'usage attendu de l'environnement (cf. annexe).

En voici un bref résumé :

L'EAPC (explorateur d'action personnel et collective) est une collection d'espace personnel ayant une structure de base commune organisée par des balises. Dans l'espace potentiel que décrit ce dispositif chacun doit pouvoir utiliser un espace personnel structuré par 5 balises. :

- *prendre place dans une organisation, construire son tutorat ;*
- *formaliser sa mission et la questionner, raconter comment elle évolue ;*
- *mener des investigations de terrain ;*
- *mener des investigations conceptuelles ;*
- *capitaliser les références bibliographiques.*

Ce texte initie notre étude. Dans notre sémantique empruntée à l'IDM, il constitue une description CIM (Computer Independant Model) du modèle de dispositif envisagé. La

difficulté qu'un informaticien rencontre avec le texte, repose sur l'entremêlement des indications portant sur les fonctionnalités de l'EAPC, son usage, la sémantique du dispositif, le dispositif lui-même, l'usage du dispositif.

11.3.5 Les motivations de l'approche que nous avons choisie pour la première phase du projet

Nous souhaitons rendre opérationnelle technologiquement l'approche pédagogique décrite dans le texte précédent. Nous proposons pour cela d'offrir, aux enseignants, un atelier d'opérationnalisation de leurs intentions didactiques. Or pour un exemple aussi simple, il nous semble difficile de demander à des enseignants de maîtriser un standard de scénarisation comme IMS-LD (cf chapitre 4.2.3 Modéliser avec IMS-LD?). En ce sens l'approche suivie par cette étude est donc différente de l'approche précédente. Notre démarche consiste à impliquer les enseignants à la création de l'atelier spécifique qui correspondent à leurs exigences pédagogiques. Leur réserver un rôle d'expert au sein du processus de création de leurs outils renvoie à une problématique plus générale : faire intervenir l'utilisateur dans la phase de conception d'un produit. Cette problématique est décrite dans la section 8.2.

Nous proposons de créer des ateliers où la spécification des intentions se fait au travers de modèles dont le métamodèle sous-jacent provient de la conciliation entre préoccupations pédagogiques et "ingénieuristes". Pour opérationnaliser les modèles didactiques, nous adoptons les principes de l'Architecture Dirigée par les Modèles MDA (chapitre 6.4 Le MDA).

MDA propose de concevoir une application au travers d'une chaîne logicielle qui se décline en quatre étapes :

- L'élaboration d'un modèle sans préoccupation informatique
- Sa transformation manuelle en un modèle dans un contexte technologique particulier
- Sa transformation automatique en un modèle associé à la plateforme de réalisation cible, modèle qui doit être raffiné ;
- Sa réalisation dans la plateforme cible.

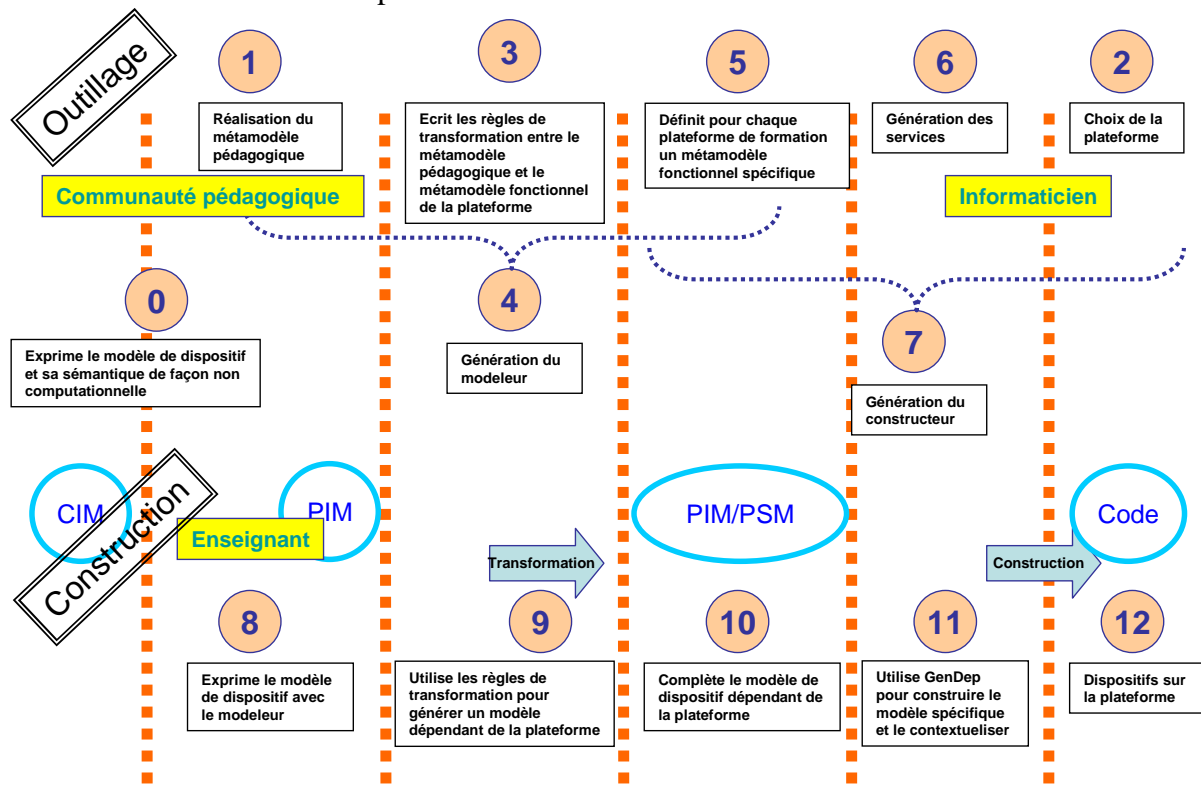


Figure 109 Cadre conceptuel, éléments à définir

Le cadre conceptuel imaginé pour cette première phase du projet correspond à la Figure 109. La pertinence de cette approche repose sur une série d'arguments que nous avons détaillés tout au long de cette thèse. Cependant dans le cadre de cette étude de cas n°2 nous en rappelons les plus importants.

Le choix de l'équipe enseignante s'est tourné vers une application Web WikiniMST. L'application WikiniMST est une application de type Web 2.0 permettant aux utilisateurs de créer collectivement ou non des documents. Des applications de ce type peuvent être utilisées pour la réalisation d'activités d'enseignement. Le grand nombre d'applications Web et leur genericité permet aux enseignants de concevoir un environnement proche de leurs intentions didactiques. L'inconvénient de ce choix est que de tels environnements ne procurent pas de facilités concernant le métier d'enseignant (schéma de persistance des dispositifs, installateur de dispositif, standard facilitant le partage etc..). Associer ces facilités tout en conciliant perspectives pédagogiques et perspectives ingénieuristes constituent la problématique abordée par cette étude. Nous proposons, au travers du cadre conceptuel (cf. Figure 109), d'associer les enseignants à la création d'ateliers spécifiques leur permettant d'opérationnaliser sur l'application WikiniMST, les dispositifs conformes à leurs préoccupations.

Parmi les différentes approches concernant la spécification des intentions pédagogiques en EIAH, nous retenons, avec les enseignants engagés dans l'étude, l'approche basée sur la description et la construction de dispositifs. Cette approche dispositive est au coeur des travaux de l'équipe d'enseignant chercheur, dans un premier temps dans le cadre de l'ERTE ("Apprentissages en Réseaux.") dans un second temps dans le cadre du présent projet PCDAI. Les principes de notre proposition, reposent sur la modélisation et la métamodélisation, ceux ci ont pour première ambition de pouvoir exprimer les préoccupations des enseignants et de les relier aux concepts véhiculés par une plateforme. La deuxième ambition de notre proposition est de générer avec économie l'infrastructure nécessaire à la modélisation et à l'opérationnalisation de dispositifs.

11.3.6 Description de notre approche

Ce cadre conceptuel de notre proposition, (cf. Figure 109), est organisé en deux ateliers, Outillage et Construction, il respecte la chaîne de contextualisation de l'approche MDA et renvoie aux différentes figures explicitées chapitre 9.2.5.

Les acteurs du premier atelier sont les enseignants, un ingénieur pédagogique et un informaticien. La première étape de cet atelier voit la description des types des préoccupations pédagogiques (i.e. métamodèle PIM) par les enseignants et l'ingénieur pédagogique. La seconde étape est la définition, par tous les acteurs, des règles de transformation de modèles PIM en modèles PSM. La mise en œuvre d'intentions pédagogiques par un enseignant est l'objet du deuxième atelier. Son usage peut être décrit ainsi : définition des intentions dans un modèle CIM, transcription dans un modèle PIM lequel sera transformé automatiquement en un modèle PSM opérationnalisable.

Le support logiciel de ces deux ateliers est constitué de deux outils : ModX pour la définition des métamodèles et la création des modèles, et GenDep pour leurs réalisations, leur mise en œuvre est visible sur la Figure 110 qui reprend la même légende numérotée. Sur cette dernière figure les éléments ont été regroupés par type d'activité.

Le premier groupe d'activité correspond au travail initial à mener par la communauté pédagogique et l'ingénieur pédagogique.

0 : expression initiale des besoins, description CIM mêlant dispositif, sémantique usage supposé, etc.

1 : description de la sémantique utilisée par les enseignants sous forme d'un métamodèle, utilisant les notions de Membre, Balise et Groupe.

2 : choix de la plateforme WikiniMST.

3 : écriture des règles de transformation entre le métamodèle pédagogique et le métamodèle fonctionnel.

4 : paramétrage du modeleur spécifique

Le second groupe d'activité correspond au travail informatique à réaliser pour adresser la plateforme avec l'ingénierie Bricoles ce travail peut très bien avoir déjà été réalisé pour un autre projet ou utiliser des services proposés par la plateforme.

5 : écriture des services encapsulant l'API de la plateforme WikiniMST

6 : écriture du métamodèle fonctionnel de WikiniMST

7 : écriture du constructeur spécifique à l'application WikiniMST

Le troisième groupe d'activité correspond à l'usage enseignant, ce travail intuitif ne comporte qu'un seul point

8 : la modélisation du dispositif constitue une activité remarquable à elle toute seule.

Le quatrième et dernier groupe d'activité concerne des activités de paramétrage à réaliser par l'enseignant

9 : mise en œuvre des règles de transformation, répondre aux fenêtres de dialogue générées par l'interpréteur de règles.

10 : profilage du modèle fonctionnel

11 : génération des collections d'appel

12 : usage du dispositif sur la plateforme

11.3.7 Les réalisations

La Figure 110 illustre le travail réalisé pour mettre en œuvre l'explorateur d'action personnel et collective. Seuls les groupes d'activité (0,1,2,3,4) et (8), en rouge sur la figure, correspondent à un travail pédagogique.

0 Descriptions en langage naturel

3

```

modx.MOF.Model.Package
EAPC=getMetamodel("EAPC");
modx.MOF.Model.Package
WIKI=getMetamodel("WikiMST");
modx.MOF.Instance.Model
modelEAPC=chooseModel(EAPC);
if (modelEAPC==null) return;
                    
```

1

2 WikiniMST

4

5

6

7

9

8

10

11

12

Figure 110 Eléments à réaliser ou à identifier pour le projet PCDAI phase 1

11.3.8 Usage de l'infrastructure, phase 1

Le métamodèle spécifique à l'application WikiniMST et le modèle PSM correspondant au dispositif pédagogique sont chargés dans GenDep. Le logiciel parcourt le métamodèle, l'analyse et génère l'interface qui va permettre le déploiement. L'enseignant ou l'ingénieur pédagogique remplit alors les formulaires permettant la génération des objets à implanter. Pour 40 étudiants et 5 formateurs, le dispositif implique la génération de 1315 éléments. Pour faciliter ce travail, l'utilisation de masque de saisie permet la création générique d'éléments, un champ peut être utilisé pour générer un autre champ via une syntaxe très simple. Une fois le travail de création virtuelle réalisé, tous les éléments sont déployés sur l'application cible. GenDep utilise pour cela un ensemble de services Web que nous avons défini pour l'application WikiniMST.

11.3.9 Résultats à l'issue de la première phase :

L'expérience réalisée lors de la première phase de l'étude a permis l'implantation d'un dispositif compliqué (par le nombre d'éléments à créer) sur l'application WikiniMST. Elle a prouvé la faisabilité technique de notre proposition. Elle a permis également de souligner l'importance des relations qu'il est possible d'établir entre enseignant et ingénieur pédagogique lors des quatre activités que nous avons relatées. Le travail de modélisation et d'implantation peut être mené en quelques heures une fois que la plateforme est adressable par GenDep. L'ensemble des éléments nécessaires à la reproduction de notre expérience sont disponibles sur le site de ModX / GenDep¹⁰. Le recours à l'ingénierie dirigée par les modèles a permis non seulement d'exhiber des objets frontières (les modèles et métamodèles utilisés) qui supportent alors les nécessaires négociations entre enseignants, ingénieurs pédagogues et informaticiens, mais a offert un cadre formel permettant l'opérationnalisation réelle d'un dispositif pédagogique sur l'application WikiniMST.

11.3.10 Une évolution de notre approche

L'usage de notre proposition initiale s'est révélé être difficile. En effet le choix de conserver au sein de l'atelier de **Construction** des règles de transformation a profondément compliqué la nature de l'acte de contextualisation. La présence de ces règles nécessitait la modélisation d'un modèle pédagogique simple et la contextualisation sur un modèle fonctionnel complexe généré automatiquement. Les enseignants, usagers de notre atelier de construction, ont expliqué la difficulté pour eux d'identifier au sein du modèle fonctionnel les éléments qu'ils avaient modélisés.

Dans le même temps, le travail mené sur d'autres applications Web (forum et e-portfolio), réalisé entre la phase 1 et 2 de notre étude de cas n°2, a montré l'importance d'une véritable phase de contextualisation. Composer un EIAH, en agglomérant différentes fonctionnalités proposées par des applications de type Web 2.0, a pour conséquence de banaliser des activités d'enseignement de type socioconstructiviste, celles-ci ne nécessitant plus, pour être mises en oeuvre, le développement de prototypes expérimentaux (chapitre 12, section 12.3.2). Or de telles activités socioconstructivistes impliquent de devoir prendre en compte, dans le cadre d'un dispositif à construire, les structurations et les objets produits spontanément par les apprenants.

La deuxième phase de notre étude est donc caractérisée par la prise en compte des mécanismes de structuration émergents présents dans les applications de type Web 2.0. Nous

¹⁰ <http://noce.univ-lille1.fr/projets/ModX/>

avons introduit dans l'infrastructure dédiée à l'enseignant la possibilité pour un modèle d'interroger, via des services, l'application à laquelle il est destiné. Cette possibilité permet de contextualiser un dispositif avec un usage en cours d'émergence, et de composer les modèles à construire.

Pour faciliter ce travail de contextualisation, nous avons défini un métamodèle unique fusionné des deux métamodèles précédents. Cette fusion a permis de simplifier l'usage du modelleur. La modélisation et la construction se réfèrent au même métamodèle unique.

11.3.11 La phase 2 de notre étude.

Pour la seconde phase de l'expérimentation un métamodèle fusionné, a été réalisé (cf. Figure 41 : détails du métamodèle). Cette fusion a permis de simplifier l'usage des outils produits. Il est possible maintenant d'affiner et de contextualiser directement un modèle produit, ce changement a nécessité de rendre invisible les transformations automatiques permettant de générer un modèle opérationnel. Le dispositif prévu dans l'EAPC a été modifié, seule une balise correspondant à un espace personnel a été conservée, des espaces pour des projets de groupe ont été définis. L'aspect administration a été modélisé.

La Figure 111 résume à la fois l'usage de notre ingénierie pour générer une infrastructure dédiée à l'application WikiniMST et l'usage de cette infrastructure pour générer les trois modèles de dispositifs utilisés dans cette deuxième phase.

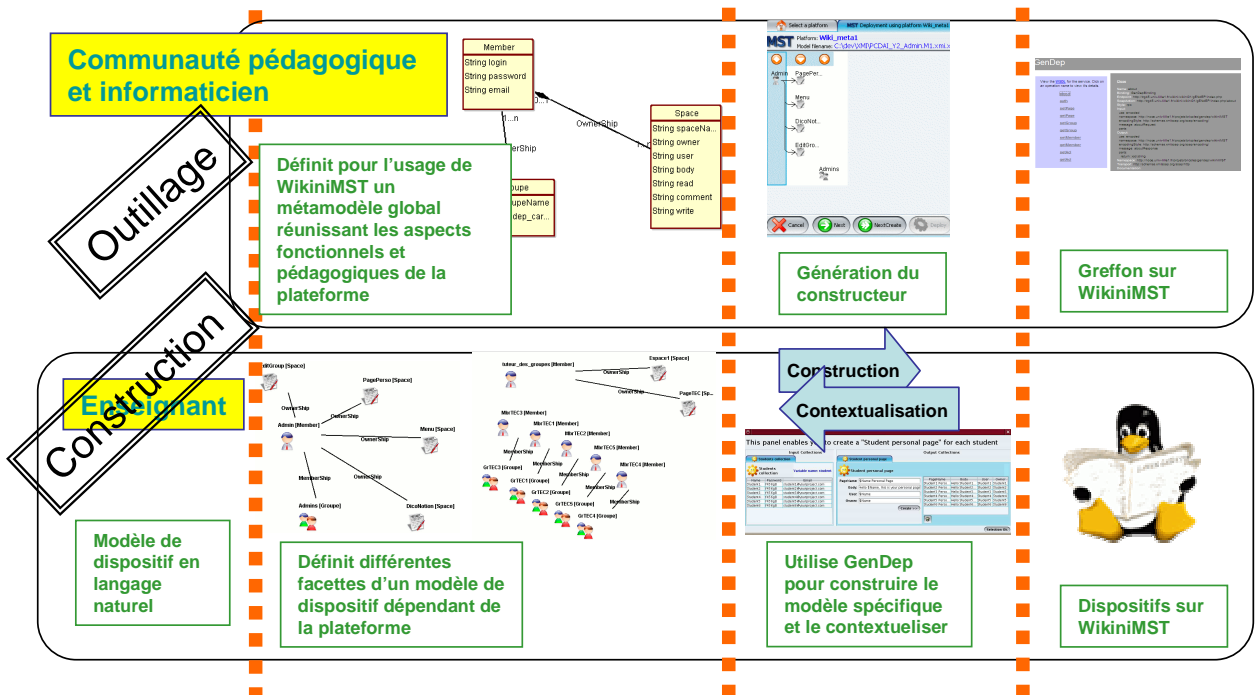


Figure 111 Notre proposition pour la phase 2

La communauté pédagogique et l'informaticien définissent le métamodèle décrivant l'application WikiniMST. Ce travail permet de générer modelleur et constructeur spécifique, la génération des services permettant de dialoguer avec l'application complète l'infrastructure dédiée. Cette infrastructure est à réaliser une fois pour chaque application. A l'aide de cette infrastructure, L'enseignant modélise son dispositif, éventuellement en plusieurs parties. Lors de la génération des éléments constituant le dispositif, il le complète avec des indications contextuelles propres à la promotion (liste d'étudiants etc...), ainsi qu'avec des éléments déjà construits sur la plateforme (groupe etc...).

11.3.12 Les dispositifs de la phase 2

Trois dispositifs ont été modélisés pour la phase 2, ces dispositifs sont en cours d'utilisation (cf. Figure 112, Figure 113, Figure 114).

Le premier crée pour chaque utilisateur, un espace personnel. Le second concerne l'administration de l'application, il crée les droits et les espaces des administrateurs. Enfin le troisième permet de construire les espaces requis pour des activités de groupe. La possibilité de contextualiser le deuxième et le troisième dispositif avec les éléments du premier permet d'ajouter de la souplesse l'usage de notre infrastructure.

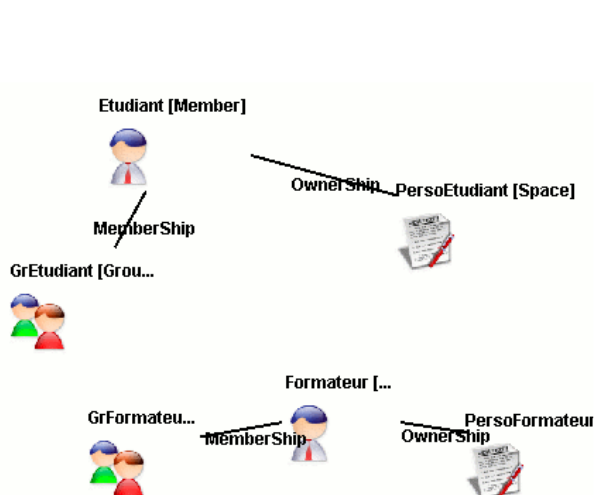


Figure 112 Espaces personnels

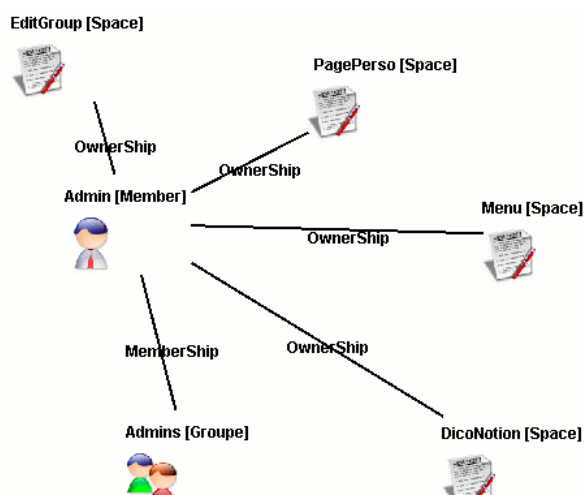


Figure 113 Espace administratif

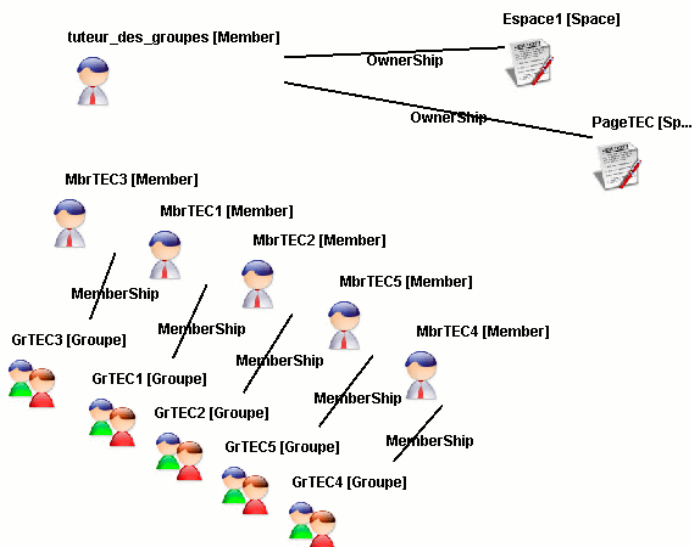


Figure 114 Espaces collectifs

11.3.13 Conclusion de l'étude de cas

L'étude de cas n°2 nous a permis d'explorer les diverses relations entre les acteurs impliqués par notre ingénierie. Nous avons montré qu'il est possible pour une communauté d'enseignants de définir à partir de leurs intentions pédagogiques une infrastructure permettant la modélisation et la construction des dispositifs impliqués par celle-ci. Cette étude se démarque ainsi de la précédente étude de cas n°1 qui explorait l'usage et l'accompagnement de la norme IMS-LD sur une plateforme de formation. L'étude a permis de compléter conceptuellement la

phase de construction des dispositifs, en démontrant en particulier pour la phase 2, la nécessité de définir un dialogue, plutôt qu'un monologue mis en œuvre par une liste d'appel de service. L'étude de cas n°2 généralise et complète l'usage des services Web pour adresser une plateforme de formation.

Sur le plan expérimental, les diverses expérimentations menées ont permis de faire évoluer notre ingénierie en la simplifiant et en privilégiant la fusion de métamodèles aux mécanismes de transformations explicites. Nous nous engageons actuellement dans une nouvelle phase d'expérimentation sur les plateformes Claroline et Moodle, ainsi que sur les applications Web 2.0 de type e-Portfolio. Ces diverses expérimentations visent à étudier les usages de notre ingénierie, ainsi qu'à explorer de nouvelles voies portant la possibilité d'exprimer des méthodologies et de lier ces méthodologies aux modèles générés.

L'étude suivante illustre cette orientation de nos travaux.

Diffusion de l'étude

Conférence nationale avec comité de lecture : Colloque Scénario 2006, (Caron et al., 2006b), EIAH 2007 (Caron, 2007)

Conférence internationale avec comité de lecture : IADIS 2006, (Caron et al., 2006a)

Monographie du projet PCDAI : <http://edutice.archives-ouvertes.fr/PCDAI>

11.4 La troisième étude : le travail réalisé avec la plateforme modulaire Moodle

11.4.1 Résumé

Nous présentons dans cette étude de cas n°3 un exemple, de scénarisation dispositive d'activité pédagogique, s'appuyant la méthode Mepulco-Université, médiatisé sur la plateforme Moodle. Cette étude a été réalisée dans le cadre du projet MetaWep. Le projet s'inscrit dans une collaboration menée au sein de la fédération régionale FERNAHTI entre les laboratoires Trigone et LIL (laboratoire informatique du littoral).

Indépendamment de tout support informatique la méthode Mepulco-Université permet de guider étudiants et tuteurs pour la création et l'utilisation de dispositifs pédagogiques favorisant la réalisation de projets. L'objet de notre étude est d'adapter cette méthode à une utilisation sur la plateforme de formation Moodle. Nous relatons pour cela la recherche-action que nous menons. Cette recherche action se décline en trois étapes. Tout d'abord, nous expérimentons la plateforme pour le suivi de projet en respectant les principes de Mepulco-Université. Ensuite nous extrayons de ces expérimentations des règles de construction d'environnement Mepulco-Université pour Moodle. Enfin nous formalisons l'architecture de Moodle et les précédentes règles au sein de notre plateforme de modélisation ModX. Ce travail nous permet de générer un modèleur et un constructeur spécifique à la plateforme Moodle respectant la méthode Mepulco-Université. L'infrastructure que nous proposons permet ainsi une modélisation et une construction sur Moodle de dispositifs requis par la méthode Mepulco-Université. En conclusion à notre étude nous décrivons cette infrastructure ainsi que son utilisation pour le contexte de Mepulco-Université/Moodle.

11.4.2 Introduction

Dans le cadre des formations universitaires, les projets d'étudiants constituent un élément clé de leur apprentissage. Pour améliorer l'encadrement de ces projets, le groupe Mepulco¹¹ du LIL (laboratoire informatique du littoral) a défini en 2001 la méthode Mepulco-Université. Cette méthode consiste en un ensemble de principes à respecter concernant les échanges entre étudiants et le support, si possible électronique, à ces échanges. Cette méthode a été utilisée sur plusieurs années et un ressenti positif a été dégagé tant par les étudiants que par les enseignants (Talon et al., 2005,). A travers l'étude que nous relatons dans cette section nous souhaitons étudier l'adaptation de la méthode, à la plateforme Moodle. Nous avons pour ambition d'extraire des résultats de cette étude, un ensemble de règles de construction afin de fournir un outil de modélisation et de construction automatique adapté à la plateforme. Nous espérons par ce travail, réinterroger la méthode elle-même, et l'insérer dans une démarche de co-évolution, par rapport aux pratiques enseignantes ainsi que par rapport aux évolutions des médias utilisées (plateforme de formation ou simple application Web). Une telle perspective semble possible grâce à l'adoption de la démarche Bricoles et de l'utilisation de son ingénierie logicielle. Cette démarche permet de définir rapidement modéleur et constructeur adapté à une approche dispositives préconisée par la méthode Mepulco-Université et adaptée à l'orientation conceptuelle socioconstructiviste de la plateforme Moodle.

La présentation de la méthode Mepulco-Université est l'objet de la première partie de cette étude. La seconde partie concerne la construction manuelle des dispositifs requis par la méthode Mepulco-Université sur une plateforme Moodle, cette expérience ayant été menée à son terme. Dans la dernière partie, nous abordons l'automatisation de la construction sur Moodle, des dispositifs requis par Mepulco-Université. Cette automatisation nécessite la génération d'une infrastructure permettant modélisation et construction. Cette infrastructure est générée en appliquant l'ingénierie Bricoles dont nous explicitons dans ce cas d'étude les principes

11.4.3 Mepulco-Université

Définition

Mepulco-Université propose aux étudiants une démarche de réalisation de projets et aux tuteurs une assistance à l'encadrement de projets. Cette méthode repose sur cinq principes et plusieurs moyens qui complètent ou permettent de mettre en œuvre la démarche initiée par les principes. Les moyens se présentent sous forme de ressources comme des dossiers pilotes, des exemples de réalisation, une norme documentaire de rédaction de rapport technique à respecter, etc. Les principes préconisés par la méthode et exposés dans (Talon et al., 2005,) sont les suivants :

1. La répartition des responsabilités.
2. La sollicitation régulière de l'équipe.
3. Le respect d'une méthode qualité de type PDCA (Plan Do Check Act) (Deming, 1982), le respect d'une telle méthode est courant en pédagogie par projet (Bru, 2006).
4. La concrétisation d'un espace de communication et de gestion des contenus produits ou utiles.
5. La capitalisation finale.

¹¹ <http://www.mepulco.net>

Un premier dispositif d'instrumentation technique de Mepulco-Université

Le site de suivi de projet est le principal vecteur de communication du projet. Il aide aussi à gérer les contenus produits ou utiles au projet. Les sites d'un certain nombre de projets encadrés avec la méthode Mepulco-Université peuvent être consultés à l'adresse <http://mepulco.net>. Chaque site doit comporter au minimum six sections, ces six sections préfigurant les dispositifs à construire :

1. Une section "Accueil" souhaite la bienvenue et indique au minimum le nom, la nature et la période du projet.
2. Une section "Présentation" précise le but et le cadre du projet.
3. Une section "Membres" présente les différents acteurs du projet, leurs rôles respectifs dans le projet et leurs coordonnées.
4. Une section "Réunions" retrace la vie du projet. Elle fournit un récapitulatif des différentes réunions et offre des liens vers leurs comptes-rendus.
5. Une section "Documents" ou "Bibliothèque" regroupe toutes les versions des documents produits lors du projet.
6. Une section "Liens" offre des liens d'intérêt pour le projet vers d'autres sites.

Les droits d'accès de ces sites sont basés sur trois rôles :

- L'administrateur qui peut mettre à jour à distance toutes les pages du site,
- Les membres qui peuvent consulter ou déposer les documents qu'ils ont produits,
- Les visiteurs qui ont accès seulement aux sections "Accueil" et "Présentation".

Moodle pour un dispositif socioconstructiviste modulaire

La plateforme Moodle est conçue pour privilégier un modèle pédagogique socio constructiviste. Plus précisément, les activités qu'elle médiatise permettent de mettre en œuvre des apprentissages constructivistes (exploitation du conflit cognitif), socioconstructivistes (exploitation du conflit sociocognitif) et constructionnistes (apprendre en construisant pour que d'autres expérimentent.) et suscitent un comportement équilibré entre lié (impliqué) et détaché (objectif) (Dougiamas & Taylor, 2003). Or une des caractéristiques des activités de ce type est de réinterroger l'usage des objets pédagogiques (cf. 4.4). Nous avons vu que dans le cadre d'activités socioconstructivistes, l'enseignant n'a pas pour unique ambition de favoriser l'assimilation par des étudiants des objets qu'il diffuse et de préciser le séquençage de ces objets. Il souhaite favoriser également des activités de construction, de commentaire, de partage. La pédagogie par projet, telle qu'elle est mise en œuvre par la méthode Mepulco-Université, repose sur le développement de l'autonomie et la créativité des étudiants au sein de dispositif. Les principes de la méthode contribuent à dessiner un cadre très précis pour définir un dispositif pouvant accueillir la méthode; les principes de la méthode explicitent également comment utiliser le dispositif. Cette pédagogie peut donc être concrétisée à partir d'une approche dispositive (cf. section 5.3.2). Cette caractéristique nous a semblé être en adéquation avec les principes sur lesquels la plateforme Moodle repose. Pour tester cette adéquation nous avons mené une expérience, dans un premier temps manuelle, de construction sur Moodle de dispositifs préconisés par la méthode Mepulco-Université. Cette expérience est relatée dans la deuxième partie de cette étude.

11.4.4 Construction d'un dispositif sur Moodle respectant la méthode Mepulco-Université

Adaptation Moodle

L'adaptation Moodle a pour but de fournir une assistance médiatisée par une plateforme de formation équivalente à celle préconisée par la méthode Mepulco-Université et prendre en compte les particularités du déroulement des projets menés dans le cadre d'expérimentation de notre étude. Les principales particularités de cette adaptation étant : un sujet unique pour toutes les équipes plutôt que des sujets différents, des séances communes aux équipes, planifiées à l'emploi du temps, au lieu de réunions séparées et à l'initiative du tuteur ou de l'équipe projet, l'existence de quatre tuteurs communs pour toutes les équipes, chacun étant spécialiste dans un des types d'apprentissages à faire acquérir aux étudiants.

Nous avons choisi de regrouper l'ensemble de l'assistance sous forme d'un seul "cours" Moodle, pour traiter aussi bien l'assistance à la méthode Mepulco-Université que l'aide aux apprentissages métier. Ce cours a été créé au format "thématique" et a donné lieu à la création de cinq thèmes.

- Thème 0 : Accueil commun dont le but est de recevoir les éléments généraux. Une partie importante des activités de ce thème est consacrée à l'accompagnement de la prise en main du site et au bon usage des TICE et de Moodle.
- Thème 1 : Espace de suivi Mepulco-Université dont le but est de recevoir les éléments concernant le suivi pédagogique au sens prescrit par la méthode Mepulco-Université.

Les trois derniers thèmes sont liés au contenu du sujet de projet et portent sur les trois domaines de compétence à acquérir : gestion de projet ; analyse et conception UML et méthodologie en processus de développement logiciel. Parallèlement à l'aspect adaptation Moodle, cette étude de cas fait l'objet d'une expérimentation pour évaluer l'applicabilité et la pertinence pédagogique de la méthode Mepulco-Université médiatisée par Moodle. Dans ce contexte, un sixième thème a été créé pour permettre aux étudiants volontaires de participer à cette expérimentation.

Traduction en Moodle de l'assistance TIC pour la méthode

Nous avons traité les six sections du site Web de suivi de projet dans le thème Espace de suivi Mepulco-Université comme expliqué ci- après.

- Un Wiki regroupe les sections *Accueil*, *Présentation* et *Membres*
- Un deuxième Wiki regroupe les ordres du jour et comptes-rendus des Réunions
- Un troisième Wiki regroupe les fiches d'activité hebdomadaire d'équipe.

La rédaction des fiches d'activité individuelle est sollicitée au moyen d'un événement périodique dans le calendrier du cours. Un forum en groupes séparés constitue la section *Bibliothèque-Documents*.

Le suivi des activités individuelles est réalisé au moyen de l'outil "*devoirs Moodle*". La fiche d'activité individuelle hebdomadaire et le journal de bord personnel que nous avons souhaités confidentiels sont rendus sous forme d'un fichier. Les bilans de compréhension du sujet et de la méthode sont quant à eux des travaux individuels mais doivent être partagés au sein de chaque équipe. Chaque étudiant dépose dans le forum "*La bibliothèque des documents d'équipe*" un lien vers son bilan de compréhension. La Figure 115 présente les dispositifs mis en œuvre par le thème dédié au suivi pédagogique du projet.

Organisation de Moodle

- Thème n°0: Accueil commun
- Thème n°1: Espace de suivi Mepulco-Université
- Thème n°2: Gestion de projet
- Thème n°3: Analyse
- Thème n°4: Développement
- Thème n°5: Recherche action

Détail du thème

Espace de suivi Mepulco-Université

Activités en commun

- Apprentissage de Mepulco-Université et du suivi du projet pédagogique **3 messages non lus**

Activités par équipe

- L'accueil et la présentation du projet
- Les ordres du jour et comptes-rendus des réunions
- Les rapports d'activités hebdo d'équipe
- La bibliothèque des Documents d'équipe
- Forum de coordination de l'équipe

Activités individuelles

- devoir semaine 1 : Fiche d'activité individuelle
- Bilan de compréhension du sujet et de la méthode MEPULCO
- devoir semaine 2 - Fiche individuelle d'activité
- devoir semaine 3 - Fiche individuelle d'activité
- Journal de bord personnel de déroulement du projet pédagogique

Dispositif requis par la spécificité des enseignements

Dispositif requis par la méthode

Fonctionnalités proposées par la plateforme:

- Travail de commun, groupe individuel.
- Wiki, Forum, Devoir.

Figure 115 La partie du cours dédiée au suivi pédagogique

Premiers retours sur l'expérimentation et problématique posée par l'automatisation

Les types d'outils proposés et la malléabilité offerte par Moodle ont permis aux encadrants d'adapter le dispositif à leurs besoins. La notification utilisée dans les forums et les devoirs, la gestion des événements et les fonctions de suivi des activités proposées par Moodle, permettent aux tuteurs d'assurer un suivi réactif nécessaire à l'apprentissage socioconstructiviste. La possibilité pour les tuteurs, lorsque le besoin s'en fait sentir, de proposer facilement de nouvelles activités aux équipes permet de donner des jalons communs et par là une dynamique collective aux équipes. L'expérience d'adaptation manuelle de la méthode Mepulco- Université a permis de vérifier l'adéquation de la méthode à la plateforme Moodle. Cette adéquation repose principalement sur deux points :

- La plateforme Moodle promeut un enseignement socioconstructiviste mis en œuvre par la méthode Mepulco-Université.
- Les dispositifs requis par la méthode peuvent être facilement construits à partir des fonctionnalités modulaires de Moodle.

Cependant la traduction du site de suivi de projet dans Moodle occasionne un travail important d'appropriation et de combinaison des outils par les concepteurs et animateurs du dispositif pédagogique. Cette tâche se heurte principalement à trois obstacles :

Le premier concerne la nécessité d'offrir un cadre de réflexion indépendant des fonctionnalités de la plateforme. Les tâches de co-conception, de co-construction, d'adaptation et d'exploitation des dispositifs nécessitent de nombreux échanges entre les membres de l'équipe pédagogique et avec les étudiants. Ces échanges sont nécessaires pour construire et maintenir une compréhension commune du dispositif et de l'articulation des différentes activités. Ces échanges devraient pouvoir s'exprimer dans un langage ne mêlant pas fonctionnalités techniques et fonctionnalité conceptuelle de la plateforme Moodle.

Le second concerne, dans un but de diffusion de la méthode, la difficulté d'offrir un cadre de travail garantissant la conformité des dispositifs conçus avec la méthode Mepulco-Université.

Le troisième pose, dans un but de réutilisation, le problème de la persistance des dispositifs pédagogiques construits. La première solution repose sur l'importation de fichier SCORM et la deuxième sur la modélisation et l'utilisation de séquences pédagogiques LAM's (exploitant le formalisme IMS-LD). Ces deux solutions ne permettent pas la contextualisation des objets

importés avec ceux qui auraient été construits préalablement par l'enseignant ou par les étudiants sur la plateforme (Allert, 2004), (Nodenot, 2006a). Dans le cadre d'un enseignement socioconstructiviste manipulant des objets co-construits et commentés, ces solutions ne sont pas satisfaisantes. La troisième solution repose sur un système de sauvegarde permettant la restauration de la structure d'un dispositif indépendamment ou non de son contenu. D'une part cette solution ne permet pas la composition, la modification, la modélisation simple de dispositifs pédagogiques. D'autre part dans le cadre de cette solution, la conception d'un dispositif ne peut être guidée que par la sémantique de la plateforme Moodle et ne peut pas s'appuyer de façon formelle sur la méthode Mepulco-Université.

La troisième partie de cette étude de cas n°3 aborde les étapes de l'adaptation de l'ingénierie Bricoles à l'étude de cas Mepulco-Université / Moodle.

11.4.5 Automatiser la construction de dispositifs sur Moodle

Bricoles : une approche par dispositif

L'ingénierie Bricoles s'appuie en premier lieu sur la notion de dispositif pédagogique comme objet de persistance à privilégier (cf. chapitre 5). Rappelons qu'un tel objet se distingue d'un scénario pédagogique dont il peut être le complémentaire. Cependant dans le cadre du projet Bricoles, les dispositifs construits constituent des entités indépendantes de tout scénario, ils représentent une entité duale : offre de signification et organisation de moyen technique. Sur la plateforme Moodle cette dualité s'exprime sous forme d'une pré-structuration de la plateforme permettant d'accueillir les activités prévues par l'enseignant, mais aussi sous forme d'une offre de signification offerte aux enseignants, leur permettant de mettre en œuvre facilement une pédagogie par projet. L'expérience relatée précédemment est à la fois un exemple de construction manuelle de dispositifs techniques et une offre conceptuelle permettant l'encadrement de projet étudiant.

Pour modéliser ces dispositifs et pour faciliter leur construction nous proposons de définir une infrastructure spécifique à la plateforme et à la méthode Mepulco.

Vue générale de l'infrastructure pour un utilisateur

Notre proposition identifie trois phases pour cet environnement : la phase de modélisation, la phase de génération/contextualisation et la phase de construction (cf. Figure 116).

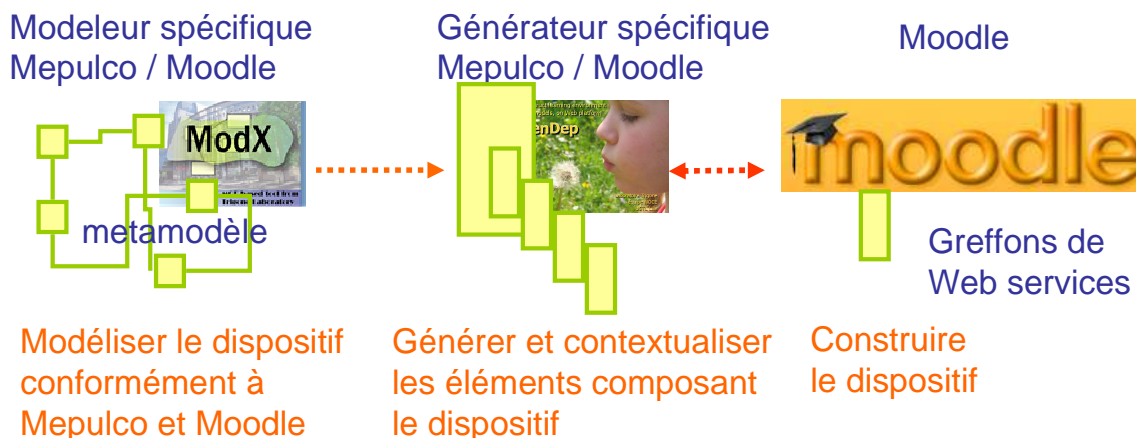


Figure 116 Vue générale de l'infrastructure pour un utilisateur

L'utilisation de l'infrastructure Bricoles commence d'abord par la définition du dispositif en langage naturel. Le dispositif est ensuite modélisé au sein de l'infrastructure dans une sémantique fusionnant les concepts de Mepulco-Université et de Moodle. Il s'agit ici de définir un modèle graphique qui se focalise sur les intentions pédagogiques de la méthode tout en les projetant vers l'espace technologique Moodle. Cette définition terminée, l'enseignant peut analyser son modèle pour comprendre la projection de ses intentions dans la plateforme Moodle. Une fois l'analyse et les modifications effectuées, il peut utiliser le constructeur spécifique à la plateforme Moodle afin de construire ce modèle sur celle-ci. Lors de cette phase de construction, il a la possibilité de lier les éléments de son dispositif avec des éléments déjà construits sur la plateforme. Une fois la contextualisation et la construction terminées, il peut utiliser son dispositif sur la plateforme.

Choix technologiques : vue concepteur d'une ligne de production

L'environnement logiciel sur lequel s'appuie notre ingénierie est décrit dans le chapitre 10 (Réalisation technique : Modeleurs et Constructeurs spécifiques de dispositifs). La démarche suivie par l'ingénierie Bricole appliquée à notre expérimentation consiste à définir un métamodèle décrivant les concepts pédagogiques de la méthode Mepulco-Université (cf. Figure 117) et à les fusionner avec le métamodèle décrivant les fonctionnalités de la plateforme Moodle (cf. Figure 118).

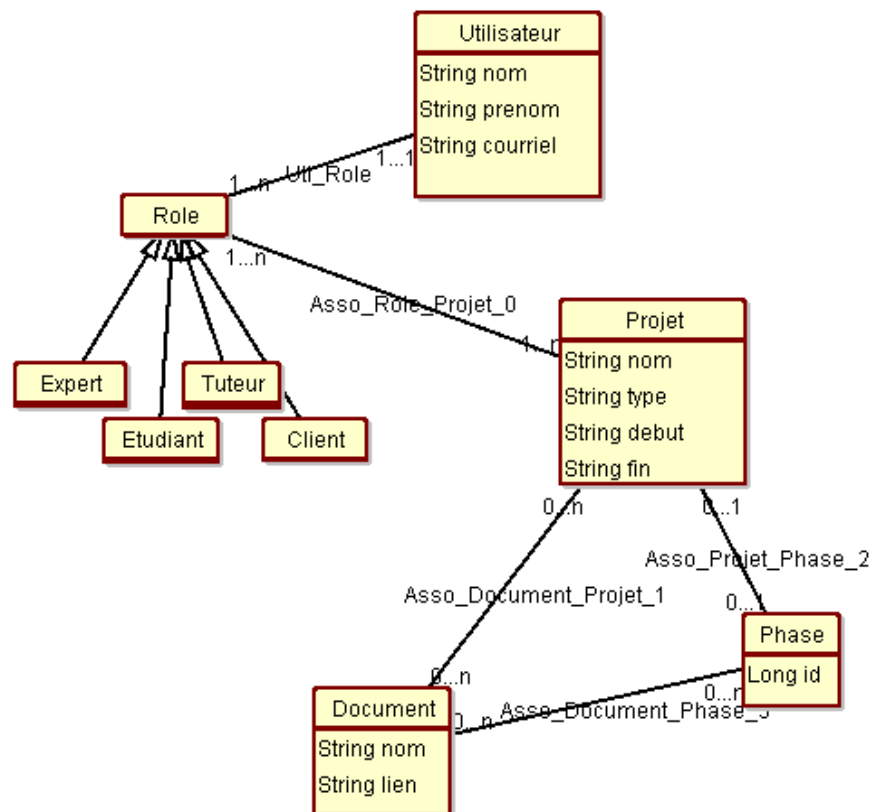


Figure 117 Concepts de la méthode Mepulco-Université

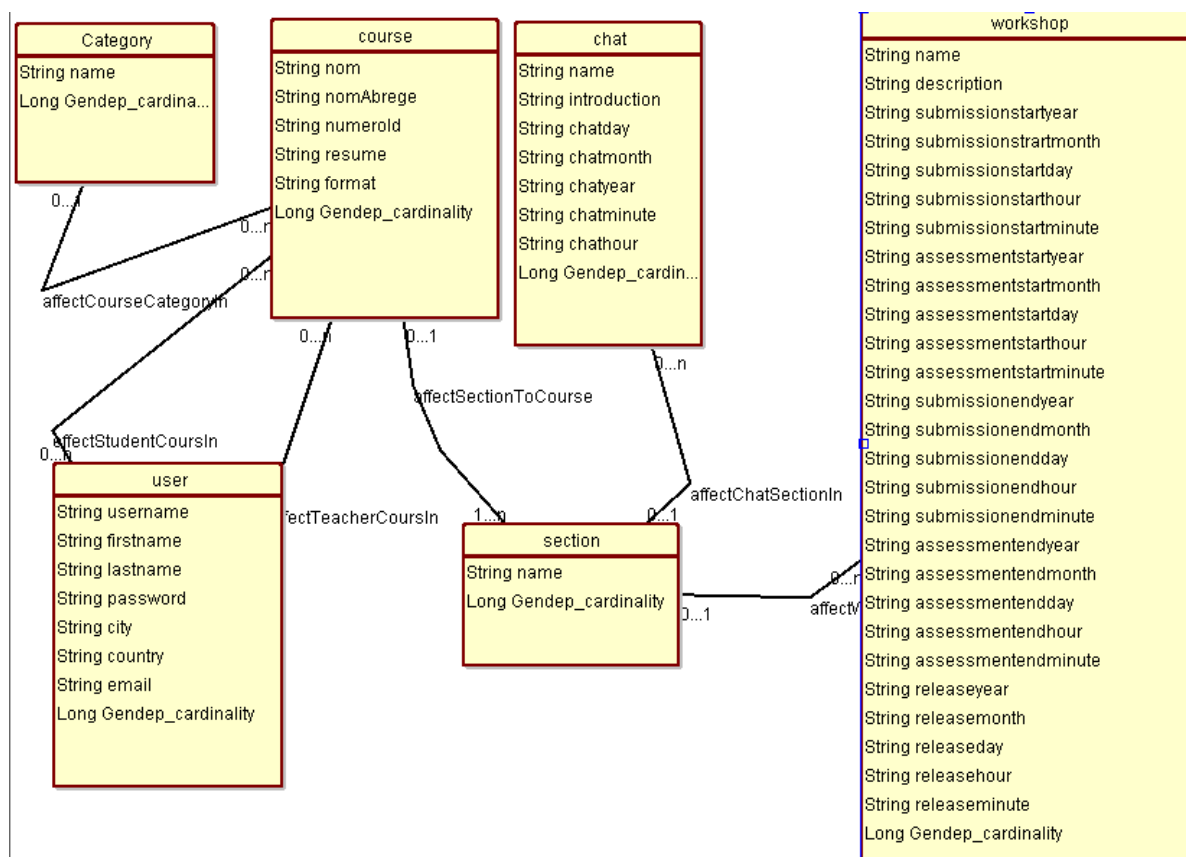


Figure 118 Détail du métamodèle de Moodle

Le métamodèle fusionné n'a pas encore été défini, il permettra d'abstraire les notions nécessaires à l'expression des modèles de dispositifs conformes à la fois à Mepulco-Université mais également à la plateforme Moodle. Cette conformité sera le garant que les éléments composant un dispositif s'intégreront facilement dans la plateforme Moodle et respecteront la méthode. Nous comptons utiliser ce métamodèle dans notre environnement pour produire un modeleur et un générateur spécifique à Mepulco-Université/ Moodle.

Pour instaurer un dialogue entre le constructeur spécifique et la plateforme Moodle, (dialogue permettant la construction et la contextualisation des dispositifs modélisés) nous suivons une approche orientée service. Moodle ne dispose pas d'une telle interface permettant sa manipulation par des services. Notre proposition consiste donc à encapsuler certaines des fonctionnalités de la plateforme Moodle par un greffon de services Web. De tels services facilitent la phase de contextualisation en ouvrant un dialogue entre dispositif à construire et structurations émergentes des activités pédagogiques menées sur la plateforme Moodle.

Les services suivants ont été définis, ils correspondent au fonctionnement minimum de Moodle.

```

'about',
'auth',
'createNewUser',
'createNewCourse',
'affectStudentToTheCourse',
'affectTeacherToTheCourse',
'getAllCategories',
'getAllCourses',
'getAllUsers',
'getAllTeachers',
'getAllStudents',
'addCategory',
'removeCategory',
'renameCourse',
'renameCategory',
'removeStudentFromCourse',
'removeTeacherFromCourse',
'removeCourse',
'removeUser',
'createNewSection',
'affectSectionToCourse',
'getAllSectionsOfACourse',
'affectCourseToCategory'
    
```

La particularité de Moodle étant son aspect modulaire, certains modules ont commencé à être adressés : le module Chat, le module Workshop, et le module Wiki.

```
'affectChatToCourse',      'createNewWorkshop',      'affectWikiToSection',
'createNewChat',          'affectWorkshopToSection', 'addPageToWiki'
'affectChatToSection',    'createWiki',
```

11.4.6 Conclusions et perspectives de cette étude de cas n°3

L'étude de cas n°3, Mepulco-Université/Moodle, s'inscrit dans le projet MetaWep dans le cadre d'une collaboration entre Trigone et le LIL au sein de la fédération FERNAHTI. Au sein du projet MetaWep, cette étude de cas n°3 explore l'adéquation de la méthode aux plateformes de formation actuelles. Notre étude a débuté par la plateforme Moodle car elle présente une orientation socioconstructiviste et une certaine modularité, deux propriétés sous-jacentes à l'approche dispositif requise par la méthode Mepulco-Université. Nos premières expériences de mise en situations, nous ont permis effectivement de vérifier que les fonctionnalités proposées par Moodle permettent de construire les dispositifs requis par la méthode Mepulco-Université. La deuxième partie du projet MetaWep consiste à élaborer une ligne de production logicielle (à partir de l'architecture Bricoles) afin de permettre aux enseignants de modéliser leur démarche de type Mepulco-Université et de construire automatiquement l'environnement correspondant sur la plateforme de leur choix. Nous venons d'entamer cette deuxième étape pour l'étude de la plateforme Moodle. Grâce à la précédente expérimentation, nous avons commencé à formaliser l'architecture de la plateforme et à en définir les règles de construction. Le greffon de service Web adapté à Moodle a été en partie implémenté. Il nous reste à compléter dans notre environnement de modélisation le métamodèle fusionné de la méthode et de la plateforme pour générer le modeleur et le constructeur spécifique à la plateforme et à la méthode.

Enseignement de cette étude :

En appuyant notre étude sur une méthode éprouvée préconisant une approche dispositif de l'accompagnement de projets étudiants ainsi que sur une plateforme non prototypale, nous avons voulu placer dans le cadre du paradoxe évoqué dans l'introduction de cette thèse (cf. Figure 119). Nous montrons que ce paradoxe se vérifie dans des cas précis dont notre étude fournit un parfait exemple. Nous vérifions que malgré la nature profondément différente des objets (Mepulco/Moodle) considéré, la méthode Mepulco peut être manuellement adaptée à la plateforme Moodle sans être fondamentalement dénaturée.

Respecter le cadre théorique du "bricolage pédagogique" c'est aussi accepter les exigences induites par ce cadre :

- La première en amont du processus de création enseignant conclue au respect de cette création enseignante à partir de laquelle tout cycle automatisé de construction de dispositif doit débiter. Ce n'est pas aux enseignants de s'adapter aux standards mais au cycle de création de s'adapter aux enseignants.
- La seconde, en aval du cycle de création, consiste à étudier les plateformes de formation comme des objets naturels et non comme des prototypes dont on peut faire évoluer les propriétés en fonction des fonctionnalités que l'on veut utiliser.

Figure 119 Paradoxes de notre thèse

La dernière étape de notre travail consiste à étudier l'adéquation de l'ingénierie Bricoles pour fournir une infrastructure pouvant reproduire et faciliter cette adaptation manuelle. Si le travail manuel réalisé lors des autres études de cas (Ganesha, WikiniMST) a pu être reproduit, informatiquement par une infrastructure Bricoles, nous nous interrogeons actuellement sur la

nécessité de prendre en compte l'essence d'une méthode produisant des dispositifs. Certes les dispositifs produits sont conformes au métamodèle fonctionnel de Moodle et au métamodèle pédagogique de la méthode *Mepulco-Université*. Mais cette conformité ne semble pas suffire pour décrire le cadre de création des dispositifs impliqués.

Nouvelle problématiques et perspectives

La particularité de la méthode Mepulco-Université est qu'elle ne constitue pas à proprement parler un domaine. Il y a deux différences notables. Tout d'abord, la méthode ne vise qu'à définir des démarches qui doivent respecter un ensemble de principes. La définition d'une démarche de type Mepulco doit a priori s'appuyer sur un cadre d'analyse afin de pouvoir définir le dispositif global. Ensuite la méthode se caractérise surtout par des règles dynamiques à respecter à la différence des domaines que nous avons abordés dans Bricoles où la description structurelle est l'activité principale. Les spécifications IMS-LD (cf. 4.2.2 Une présentation d'IMS-LD) en sont un exemple typique : la dynamique sous-jacente aux concepts est majoritairement laissée libre d'interprétation aux concepteurs de moteurs d'exécution.

Plusieurs pistes d'application de l'approche Bricoles à la méthode Mepulco-Université apparaissent. Nous envisageons pour l'instant trois possibilités :

1. **Basique**. Une démarche de type Mepulco est modélisée à partir d'un métamodèle dédié à Mepulco. Le modèle résultant est ensuite projeté vers un espace technologique et peut ainsi servir à la construction de dispositifs correspondants. Chaque construction prendra toujours en compte l'environnement existant dans lequel elle agit. Cette solution correspond à la première solution mise en œuvre dans le cadre de la phase 1 du projet PCDAI.
2. **Fusion**. Avant la projection vers un espace technologique, le modèle de la démarche est d'abord fusionné avec un modèle du dispositif global dans lequel la méthode sera appliquée. Cette solution correspond à la deuxième solution mise en œuvre dans le cadre de la phase 2 du projet PCDAI.
3. **Règles de modélisation**. La méthode Mepulco n'est pas formalisée sous forme de concepts de modélisation mais plutôt sous forme d'opérations génériques de modélisation applicables sur des métamodèles de domaine.

Chaque possibilité semble présenter pour l'instant un rapport avantages/inconvénients équivalent. L'application basique apparaît comme simple à mettre en œuvre, simple pour la modélisation car elle se focalise sur la démarche à décrire et elle dispose toujours des capacités de GenDep pour la contextualisation finale. L'application fusionnelle apporte un plus évident car elle conserve la possibilité de se focaliser sur la méthode et permet en même temps une intégration au niveau abstrait. Par contre, elle implique l'écriture de règles de fusion avec des métamodèles de domaine et celles-ci semblent plus difficiles à écrire que les règles de transformation. Enfin l'application par règles de modélisation semble une solution élégante et intuitive pour le concepteur de modèle. Toutefois, elle nécessite de rationaliser la création des opérations de modélisation, ce qui pourrait se faire au travers d'une mise en équivalence entre les concepts de Mepulco et ceux du métamodèle de domaine cible. Ce type d'opération apparaît proche des règles de fusion et donc renvoie à un haut de degré de complexité.

Diffusion de l'étude

Conférence nationale avec comité de lecture : Colloque EPAL (Caron, Warin, Hoogstoel, & Le Pallec, 2007), EIAH 2007 (poster) (Warin, Caron, Le Pallec, & Hoogstoel, 2007).

Conférence internationale avec comité de lecture : Colloque Scénario 2007, (Caron, Derycke, Hoogstoel, Le Pallec, & Warin, 2007).

11.5 Conclusion

Les trois études de cas, que nous avons présentées dans ce chapitre, illustre de façon chronologique chaque étape que notre ingénierie a dû franchir : établir une communication avec une plateforme de formation par l'intermédiaire d'un greffon, proposer à des enseignants un modèle et un constructeur adapté à leur pratiques, prendre en compte dans l'acte de modélisation les principes d'une méthode.

Les études de cas que nous avons menées adressent également des plateformes très différentes, Ganesha est une plateforme de formation classique, WikiniMST est une application de type Web 2.0, Moodle est une plateforme modulaire dont l'orientation pédagogique est socioconstructiviste. Ces différences contribuent à rendre significatives les études que nous avons menées.

Enfin ces trois études adressent des problèmes très différents rencontrés au cours de notre thèse. La première étude aborde essentiellement des problèmes d'ingénierie. La seconde étude pose le problème d'adapter une infrastructure à une équipe d'enseignant. La troisième étude pose le problème conceptuel, des rapports que peuvent entretenir méthode et modèle.

Les deux dernières études ne sont pas encore terminées. Nous pensons pouvoir mener dans le cadre de l'étude n°2 (application WikiniMST, projet PCDAI), des enquêtes pour estimer l'adéquation de notre infrastructure au travail des enseignants.

De l'étude n°3, nous attendons plusieurs bénéfices :

Le premier bénéfice attendu de cette étude n°3 est de nous permettre d'élargir le spectre d'application du projet Bricoles aux méthodes d'apprentissage. Nous pouvons ainsi étudier la pertinence et l'adéquation de l'Ingénierie Dirigée par les Modèles aux "méthodes", et particulièrement les méthodes pédagogiques qui sont en général peu ou pas formalisées.

Le second bénéfice de cette étude est permettre une formalisation des principes de la méthode Mepulco-Université. Cette formalisation permettrait dans un premier temps de réduire les ambiguïtés inhérentes à la création de dispositif respectant la méthode et dans un second temps de définir des opérateurs formels pour vérifier conceptuellement qu'un dispositif respecte la méthode.

D'autres travaux sont engagés qui pourraient donner lieu à une étude plus approfondie, ces travaux sont décrits dans le chapitre 12 suivant.

chapitre 12 Conclusion, perspectives

Cette thèse aborde les problématiques posées par l'opérationnalisation d'une approche dispositive des plateformes de formation. Nous concluons nos travaux en présentant dans ce chapitre : le bilan de notre recherche, les études de cas que nous avons menées et la diffusion qu'elles ont connue. Quelques travaux, encore à faire font partie de notre horizon immédiat, ces travaux seront décrits dans ce chapitre. Nous abordons ensuite les apports de notre recherche. Enfin, nous terminons en indiquant les perspectives de recherches et l'intégration de nos travaux dans les recherches actuelles menées dans notre laboratoire.

12.1 Bilan des travaux réalisés

Notre contribution se présente sous la forme de deux propositions complémentaires : Une **approche dispositive des plateformes de formation**, opérationnalisée par une **ingénierie dirigée par les modèles**.

12.1.1 Une approche dispositive des plateformes de formation

Notre proposition repose sur l'instrumentation d'enseignants pour les aider à construire des dispositifs pédagogiques sur les plateformes de formation. Ce positionnement a nécessité, dans un premier temps, de définir et de caractériser l'acte de construction qu'il est possible de mener sur une plateforme de formation.

Définir notre cadre de recherche

Pour cela notre recherche a débuté par l'étude des plateformes de formation. Cette étude s'est attachée à explorer les relations entre les fonctionnalités qu'une plateforme de formation propose, les différents rôles que des enseignants peuvent y jouer et l'organisation sociale au sein de laquelle la conception d'un EIAH peut s'effectuer. Cette étude a permis de spécifier le cadre social dans lequel nous envisageons l'utilisation d'une plateforme de formation : il s'agit de petites organisations enseignantes pratiquant du bricolage pédagogique sur des plateformes de formations souvent modulaire et open source ou détournée à partir d'application de type Web 2.0. Nous avons montré que la conséquence de ce positionnement est d'établir une écologie distincte entre les pratiques de création enseignantes et l'implémentation d'une plateforme de formation.

Comment lier intention socioconstructiviste et plateforme de formation

En respectant notre cadre de recherche, nous nous sommes employés à définir les liens qu'il est possible de tisser entre les intentions didactiques des enseignants et les plateformes de formation. Nous avons montré qu'il est possible de mettre en correspondance les fonctionnalités d'une plateforme de formation et les principaux modèles pédagogiques. Ce travail nous a permis de positionner notre recherche en montrant qu'il semblait difficile pour un enseignant d'utiliser une infrastructure pour concrétiser sur une plateforme de formation un enseignement de type socioconstructiviste. Il est apparu que les plateformes de formation ne semblaient pas prévoir de fonctionnalités pour accompagner la concrétisation de telles intentions pédagogiques. Nous avons étudié les principaux standards des EIAHs, ainsi que les différentes propositions en matière de construction de ressources, de séquences, d'activités

scénarisées. Cette étude a montré la difficulté pour ces approches de décrire et à fortiori de favoriser la mise en place d'un enseignement socioconstructiviste.

Notre proposition : construire des dispositifs

Nous avons alors montré que la notion de dispositif pouvait être cette entité qui permettrait d'établir un lien entre intention pédagogique de type socioconstructiviste et plateforme de formation. L'adéquation de la notion de dispositif à notre recherche repose essentiellement sur les points suivants.

- La notion de dispositif est une entité duale, elle constitue pour les enseignants un cadre de pensée, elle constitue au sein d'une plateforme de formation l'organisation des moyens techniques mise à disposition par la plateforme.
- Les potentialités offertes par cette notion, son caractère ouvert en faisait un excellent support pour le bricolage enseignant.

Pour que cette notion de dispositif soit réellement efficiente dans un cadre informatique :

- Il faut pouvoir la définir et la manipuler informatiquement.
- Il faut que l'on puisse implémenter avec économie une infrastructure permettant de la concrétiser.
- Il faut pouvoir définir un processus instrumenté permettant la transition entre l'expression d'un dispositif reflétant les intentions didactiques des enseignants et un dispositif technique encadrant les activités des apprenants.

12.1.2 Une ingénierie basée sur l'IDM

Notre proposition adresse ce triple challenge en offrant un cadre de formel pour définir un dispositif pédagogique, que celui-ci décrive des intentions pédagogiques ou bien qu'il décrive une organisation de moyens techniques dans le contexte d'une plateforme de formation. Nous appliquons une approche IDM pour implémenter une ligne de produits logiciels adaptable à chaque plateforme de formation. Enfin nos travaux établissent un parallèle entre le processus de concrétisation qu'un dispositif peut suivre et le Model Driven Architecture.

Une application de l'IDM

Après avoir décrit les différents concepts sur lesquels l'ingénierie dirigée par les modèles repose, nous avons expliqué de quelle manière ce type d'ingénierie pouvait être adapté à notre domaine de recherche. L'approche MDA peut être redéfinie dans le cadre plus général de l'IDM, en l'appliquant aux langages spécifiques de domaine. Nous avons proposé de produire une série d'infrastructures spécifiques, composées d'un modeleur et d'un constructeur reposant sur la définition d'un ou de plusieurs métamodèles. Ces infrastructures permettent de modéliser et de construire un dispositif dans le contexte d'une plateforme de formation.

Les outils développés

Deux outils ont été développés pour cette thèse :

Le premier, ModX, développé en collaboration avec Xavier Le Pallec, est une plateforme de modélisation implémentant la version 1.4 du MOF. Cette plateforme permet de définir graphiquement des métamodèles et des règles de transformation. Nous avons généré à partir de cette plateforme des modeleurs spécifiques reposant sur un ou plusieurs métamodèles et utilisant un moteur de transformation. De tels modeleurs permettent la modélisation de dispositifs pédagogiques conforme à un métamodèle pédagogique, fonctionnel ou fusionné.

Le second, GenDep, est un générateur de constructeur spécifique. Ce générateur peut être paramétré par deux modèles, l'un décrivant la plateforme de formation, l'autre son interface de service. Cette paramétrisation permet de générer un constructeur spécifique adapté à une plateforme de formation particulière. Un tel constructeur spécifique permet de construire dans le contexte d'une plateforme de formation un dispositif pédagogique.

Pour finaliser le dialogue nécessaire entre constructeur et plateforme, des greffons, serveur de services Web, ont été implémentés sur les plateformes : Ganesha, Claroline, Moodle, Elg et WikiniMST.

Les études de cas

Nous décrivons dans cette thèse trois de études de cas.

La première, consacrée à l'adaptation de Ganesha, a permis de définir les modalités du dialogue qu'il est possible d'établir avec une plateforme de formation. La seconde étude sur l'application WikiniMST, a permis de mettre en place notre proposition d'infrastructure. Cette infrastructure permet à des enseignants de modéliser un dispositif puis de le construire sur une plateforme de formation. La troisième étude, se déroulant sur la plateforme Moodle, étudie la notion de méthodologie et son couplage avec un métamodèle dans le but de définir un formalisme pour exprimer les manières de modéliser. Cette dernière étude n'est pas terminée et doit se poursuivre dans le cadre d'un projet commun MetaWep entre les laboratoires Trigone et le LIL (laboratoire Informatique du Littoral). Le projet MetaWep participe au projet d'Evaluation des Usages dans des Contextes d'E-services Ubiquitaires et Evolutifs (EUCEU).

12.2 Apports de nos travaux de thèse

12.2.1 L'approche dispositifive et sa cohérence avec le type d'enseignement que nous adressons

L'approche proposée

Les travaux de cette thèse s'inscrivent dans les travaux actuels d'ingénierie portant sur la modélisation de scénarios pédagogiques (Laforcade et al., 2005), (Villiot-Leclercq, 2006), (De la Teja et al., 2006), (Martel et al., 2006), (Burgos et al., 2005), (Dalziel, 2005), (Giacomini-Pacurar et al., 2005). L'analyse des différentes approches ingénieristes existant actuellement en EIAH, nous a permis de positionner scientifiquement notre recherche par rapport à celles-ci. Notre approche dispositifive des plateformes de formation, adresse un public, un type de pédagogie et un type de plateforme rarement traité par ces travaux .

Nous montrons qu'en considérant la notion de plateforme de formation au sens large comme tout moyen mis à la disposition des élèves par des enseignants pour encadrer un enseignement, un nouveau champ de recherche s'ouvre alors en ingénierie pédagogique. Pour adresser ce champ de recherche, il nous a fallu décrire un nouveau paradigme pour les EIAHs. Ce paradigme s'appuie sur les notions de processus de conception participative, de bricolage pédagogique, et de dispositifs pédagogiques. Notre recherche montre la cohérence de ces notions, et suggère que la mise en place d'un enseignement de type socioconstructiviste est favorisée par ce type d'organisation sociale, cette manière de concevoir des séquences pédagogiques, ce type d'entité pédagogique sur ce type de média. Cette suggestion est à la base des choix instrumentaux que nous avons proposés.

Notre choix d'ingénierie

Nous avons montré au chapitre 7 que différentes approches actuelles en ingénierie pédagogique pouvaient être reconsidérées dans une perspective IDM. L'originalité de notre approche est d'établir dès sa conception un parallèle entre l'approche de contextualisation que nous préconisons et l'approche MDA reconsidérée dans le cadre élargi de l'IDM. Ce positionnement, nous a permis de proposer des solutions d'infrastructures destinées aux enseignants, éventuellement non dépendantes des standards actuellement utilisés en EIAH (cf. chapitre 9.1.2)

Nous montrons qu'une telle prise de distance permet néanmoins de décrire et de construire des dispositifs riches (l'étude de cas PCDAI en est un exemple, cf. chapitre 11.3). Notre ingénierie permet ainsi d'adresser des applications de type Web 2.0, utilisées à des fins pédagogiques mais conçues initialement pour d'autres fins. De telles applications ne sont pas, pour le moment, adressées par les standards de l'éducation. Pourtant, l'usage de ces applications, banalise en enseignement de activités de type socioconstructiviste (Downes, 2005). De telles activités nécessitent de pouvoir contextualiser finement les dispositifs à construire, la phase dialogue que notre constructeur implémente permet cette contextualisation (cf 11.3.10).

Récolte d'indices

Un travail de validation des ateliers et des infrastructures que nous proposons reste à entreprendre auprès des enseignants : ce travail est en partie amorcé. Dans le cadre du projet PCDAI (11.3) des travaux sont actuellement menés sur l'étude de trace de co-conception d'un dispositif pédagogique et la recherche d'indicateurs de co-évolution des dispositifs, des communautés et des médias de formation. Dans le cadre de la recherche-action MetaWep (11.4), nous avons entrepris d'analyser les traces d'utilisation fournies par la plateforme Moodle dans le but de faire évoluer la méthode Mepulco, le modèle engendré par la méthode et modèle fonctionnel de la plateforme. Enfin, la récolte d'indice de Bricolage pédagogique ne peut avoir lieu, que sur des études longitudinales pouvant montrer l'évolution d'un dispositif et des pratiques pédagogiques au cours du temps. La simplification des dispositifs, entre la phase 1 et 2 du projet PCDAI, sont des exemples d'indices que nous recherchons. Une telle recherche, à réaliser à grande échelle, nécessite une finalisation et une intégration que notre prototype ne permet pas encore.

12.2.2 Modèle de plateforme et construction de dispositif

Nos travaux en IDM ont permis d'aborder dans des cas concrets les verrous constitués par la définition du modèle d'une plateforme d'exécution et par la construction sur cette plateforme d'un système modélisé. Nos recherches en IDM se situent dans le cadre des travaux adressant la problématique des langages spécifiques de domaine (Vega Baez, 2005), (Emerson, 2005), (Tolvanen & Rossi, 2003), (Balasubramanian, Gokhale, Gabor Karsai, Sztipanovits, & Neema, 2006). La particularité de notre approche consiste à adresser des domaines spécifiques nombreux et pas clairement exécutables. Le domaine que nous adressons, les EIAHs, présente en effet pour l'IDM deux caractéristiques principales : un grand nombre de plateformes, et des éléments à construire formant un système devant être cohérent avec la plateforme de formation.

Modèle de Plateforme

Nous nous sommes donc interrogés sur la signification d'une démarche IDM dans un cadre qui ne conduit pas à la production de code. Pour proposer une telle démarche dans le cadre des EIAHs, nous avons montré qu'un EIAH pouvait être considéré comme une plateforme d'exécution des systèmes que nous voulions construire. Nos travaux ont permis de proposer des bonnes pratiques pour décrire une telle plateforme, cette description a été réalisée à l'aide du logiciel ModX sous forme d'un métamodèle exprimé en MOF. Ce métamodèle correspond en partie à la description des fonctionnalités qu'une plateforme doit fournir pour opérationnaliser les dispositifs pédagogiques que nous souhaitons décrire. Une partie de ce métamodèle peut donc être simplement déduit d'une étude des services nécessaires à la construction de dispositifs sur la plateforme.

Une des originalités de notre travail est de montrer qu'il est possible d'utiliser ce métamodèle de deux façons.

- La première consiste à l'utiliser pour définir **un modeleur spécifique à la plateforme**. Pour une telle utilisation, le métamodèle décrit, par des classes et des associations, les fonctionnalités précédentes. Il traduit, par des indications de cardinalité, les spécificités des systèmes à construire. Ces contraintes peuvent être imposées par la plateforme de formation, ainsi que par les pédagogies projetées.
- La seconde consiste à l'utiliser pour définir **un constructeur spécifique à la plateforme**. Le métamodèle précédent doit être complété pour lui adjoindre des indications permettant la génération de collections.

Nos travaux expliquent donc, de façon pragmatique, comment définir le métamodèle d'une plateforme d'exécution. Ce métamodèle peut être utilisé classiquement pour définir un langage spécifique de domaine. Nous montrons qu'il peut également être utilisé pour définir un constructeur spécifique au domaine. Cette définition se fait dans le cadre de la plateforme de construction générique GenDep que nous avons réalisée pendant cette thèse.

Générer un constructeur spécifique

Cette plateforme de construction générique (GenDep) que nous proposons permet la génération presque automatique d'un constructeur spécifique à une plateforme de formation (cf. chapitre 10). Nous avons montré (cf. 9.3) que cette proposition d'ingénierie est directement déduite des principales caractéristiques du domaine que nous adressons. Celles-ci impliquent la nécessité de concevoir une ingénierie adaptable à un grand nombre de plateformes ainsi que la nécessité de définir des mécanismes permettant une contextualisation des dispositifs modélisés. Notre choix d'ingénierie s'inspire donc dans un cadre IDM des travaux portant sur les générateurs d'applications (Tolvanen, 2006) (Bhanot et al., 2005), et propose une solution adaptée à la simplicité des modèles à construire ne passant pas par des mécanismes de composition dynamique de métamodèle (Estublier et al., 2005).

Les constructeurs spécifiques, que nous avons produits, nous ont invités à questionner la relation de factorisation qui lie un modèle au système qu'il décrit. Ils nous ont amenés à définir, au niveau du métamodèle et du modèle, des éléments pour favoriser la construction des dispositifs.

Construire une collection d'élément à partir d'un modèle.

Nos travaux présentent ainsi des propositions à une problématique rarement abordée en IDM, où l'opérationnalisation d'un modèle se fait souvent par la production d'un squelette de code. Nous avons choisi de considérer que l'élément d'un modèle de dispositif, pouvait décrire, dans le contexte de la plateforme, une collection d'élément. Pour mettre en œuvre ce mécanisme, nous avons proposé des éléments de profilage et de cardinalité (cf. chapitre 9.3.4). Ces

éléments permettent, tout en respectant une compatibilité MOF des modèles modélisés, de spécifier comment doivent être générées les collections décrites.

12.2.3 Réalisation Technique

Notre thèse propose pour adresser des plateformes de formation une solution basée sur la réalisation d'un greffon de service Web spécifique. Les travaux concernant l'adressage par des services Web des plateformes de formation sont encore balbutiants et très hétéroclites, nous avons au cours de cette thèse cité de nombreuses initiatives (cf chapitre 10.1). Elles tentent de promouvoir en ce domaine un standard d'adressage de ces plateformes (IMS-ES), (OKI-OSID). Ces travaux en EIAHs se heurtent non seulement problèmes d'interopérabilité des plateformes de formation, mais également au problème de définition des usages de ces plateformes. Les applications plus généralistes de type Web 2.0, en proposant des interfaces de Services Web moins ambitieuses, semblent en ce domaine plus avancées (Google-API).

La solution que nous avons développée, consiste en un simple greffon de service Web encapsulant les principales fonctionnalités des plateformes de formation que nous avons étudiées (cf. chapitre 10.1). L'originalité de notre proposition est de représenter ces services par des métamodèles et donc de traiter les problèmes d'interopérabilité entre services, à ce niveau. Un des autres avantages de notre solution est de séparer considération métier et considération technique, cette séparation permet une meilleure abstraction des services que nous proposons. Notre solution peut être utilisée dans d'autres circonstances que celle que nous décrivons dans cette thèse; nous avons proposé dans notre laboratoire d'étendre ce principe pour l'interrogation des plateformes de formation par des périphériques mobiles (Kaddouci, Vantrois, & Chevrin, 2007)

Nous montrons dans cette thèse qu'il est techniquement possible, via l'ingénierie dirigée par les modèles, d'interopérer avec de tels greffons. De plus le logiciel que nous avons implémenté, GenDep, est en grande partie généré par l'interprétation d'une part du modèle du greffon de Service Web et d'autre part du modèle de la plateforme de formation. Cette double génération de code permet un adressage économique des applications Web.

12.3 Perspectives

Nous décrivons dans cette section, les évolutions récentes des EIAHs et montrons comment notre ingénierie (Bricoles) s'insère dans des programmes de recherches que nous menons dans notre laboratoire pour appréhender ces évolutions.

12.3.1 Le cadre de l'ingénierie Bricoles

Nous avons montré tout au long de cette thèse que l'ingénierie Bricoles était conçu pour un usage dans le cadre artisanal de petites équipe d'enseignant pratiquant le bricolage pédagogique sur des plateformes de formation au sens large. Cette adaptation à ce cadre particulier repose les facilités que notre ingénierie procure pour mettre en oeuvre des pratiques de conceptions participatives, de bricolage pédagogique, de construction de dispositif pédagogique et de pédagogie socioconstructiviste (cf. respectivement, 8.2, 5.4.1, 10.5et 3.4). Nous pensons que les évolutions actuelles de l'informatique tendent à généraliser et à banaliser ce type de pratique. En effet de nombreuses recherches en EIAH, relient les conditions d'émergences et de généralisation d'un enseignement de type socioconstructiviste, à l'apparition et à l'utilisation de nouveaux outils (Synteta et al., 2002), (George, 2005), (Bounie et al., 2005), (Gueraud et al., 2004). Ces recherches qui s'appuient sur l'usage de prototypes dressent néanmoins un tableau précis des artefacts qui pourraient être utilisés pour un tel enseignement et des usages que l'on pourrait en avoir.

12.3.2 Un nouveau paradigme en EIAH

Dans l'étude de cas portant sur l'application WikiniMST utilisée lors du projet PCDAI (cf.11.3) nous avons montré que notre ingénierie présentait des qualités pour adresser avec succès des plateformes de type Web 2.0 (O'Reilly, 2005). Sous ce vocable il est possible de désigner un ensemble d'application Web dont la principale caractéristique est de permettre à l'utilisateur de définir la forme de son environnement (par des procédés de "MashUp") et les conditions d'usage pour lui et pour la communauté à laquelle il appartient (tag, rating, geodata, etc.).

L'usage de ces applications en enseignement

Pour prendre en compte de l'usage de ces nouvelles plateformes en enseignement Stephen Downes a introduit le terme e-Learning 2.0, (Downes, 2005). L'auteur fait un bilan actuel du e-Learning, tel qu'il est pratiqué au travers des LMS. Il rappelle les différents standards sur lesquels s'appuie ce type de plateforme (Scorm, IMS QTI etc...) et conclut en affirmant qu'actuellement il est possible de recréer, sur des plateformes de formation, les conditions classiques d'un enseignement de type frontal. *"Where we are now in the online world is where we were before the beginning of e-learning"* (Downes, 2005). L'auteur décrit alors les changements que l'usage d'Internet a entraînés chez les étudiants, il dresse un tableau exhaustif des possibilités offertes actuellement par les applications de type Web 2.0. Ce type d'application place l'internaute au centre des usages via le partage et la collaboration entre individus. Les applications, blog, wiki, réseaux sociaux, tag collaboratif, e-portfolio, réunies sous ce nouveau vocable, dessinent de nouveaux usages fondés sur des utilisateurs actifs. Stephen Downes recense ces applications, l'usage que l'on peut en faire en enseignement (Scott, 2003), (Downes, 2004), (Barrett, 2006) et décrit un nouveau paradigme pour les EIAHs. Ce paradigme s'appuie sur la création individuelle de contenu, et pose la question de l'exploitation en enseignement de ces activités constructivistes. *"how learning content— whether professionally authored or created by students— can be used as the basis for learning activities rather than the conduit for learning content"* (Downes, 2005). Des exemples de ces usages sont depuis souvent recensés dans la littérature scientifique (O'Hear, 2006), ainsi l'utilisation de Blog Wiki et Podcast pour promouvoir un enseignement collaboratif en médecine (Boulos, Maramba, & Wheeler, 2006). La British Educational Communications and Technology Agency, (Becta, 2006b) dans un guide destiné aux ingénieurs pédagogiques indique que l'usage des applications de type Web 2.0 en enseignement offre la possibilité de mettre en œuvre un enseignement personnalisé, mais interpelle les concepteurs de ressources pédagogiques sur le statut des objets pédagogiques. Le guide pose la question dans le cadre de ce nouveau paradigme de l'intérêt de produire, entreposer et standardiser des ressources pédagogiques.

Application Web 2.0 et Micro Learning

Dans ce nouveau paradigme que nous décrivons pour les EIAH il est possible d'aborder le rôle et la place des contenus, dans le cadre des recherches menées actuellement sur les micro-enseignements et micro-contenus. Un micro-contenu est typiquement un article soumis dans un blog, un commentaire, une photo placée sur un site de partage etc...Ce qui caractérise un micro contenu, d'un autre type de contenu, disponible sur le Web, c'est de pouvoir être référencé, et interprété, informatiquement, via la lecture par exemple de fichier RSS. La publication individuelle de micro-contenu sur des applications Web 2.0, tend à dessiner une vue subjective du monde réel (Mosel, 2005). L'auteur montre que, par de simples mécanismes de liens, de commentaires, de renvois, une vue socialement partagée du monde réel émerge de

la toile tissée par ces micro-contenus. Il montre que la subjectivité, inhérente à ce type de contenu, favorise un apprentissage constructiviste par la confrontation des différents réels chez leurs créateurs, ou leurs lecteurs. Cependant l'usage de ces contenus, à des fins d'enseignement, soulève pour des enseignants des questions portant par exemple sur l'évaluation des activités réalisées, sur la cohérence des parcours apprentissages, lorsqu'ils sont organisés individuellement (Roth, Philipp, & Osburg, 2006), (Fiedler & Kieslinger, 2006).

L'apprentissage pervasif.

En 1991 Mark Weiser dans un article fondateur de l'informatique Ubiquitaire décrivait, à un horizon de vingt ans, l'avenir de l'informatique (Weiser, 1991). Nous vivons actuellement dans un monde en partie décrit par cet article. Des indices de cette révolution sont désormais tout à fait visibles dans notre environnement, du GPS dans les voitures, aux appareils photos numérique et communicants. Notre laboratoire mène actuellement des recherches sur cette déclinaison de notre avenir dans les domaines de l'éducation, ces recherches portent sur l'apprentissage pervasif (Derycke & Chevrin, 2007), (Chevrin et al., 2006).

L'apprentissage Pervasif fait référence à un apprentissage utilisant des réseaux pervasifs. Un réseau pervasif est un réseau dans lequel il est possible d'être connecté partout et à tout moment par l'intermédiaire d'objets communicants classiques ainsi que grâce à des objets équipés d'une capacité de mémoire et d'intelligence tels que décrit par Mark Weiser. Apprendre, dans un tel contexte, c'est utiliser l'intelligence ambiante du réseau pour favoriser l'apprentissage.

Dans le cadre du projet p-Learnnet¹² (P-learnnet), nous prévoyons de poursuivre les recherches menées lors de cette thèse pour aborder l'adressage par une ingénierie dirigée par les modèles de service d'enseignement. La problématique que nous abordons dans ce projet nécessite de piloter une approche SOA flexible et adaptable, évolutive en temps réel pour mettre en œuvre des dispositifs pédagogiques dans des contextes changeants. Cette problématique peut être abordée dans un cadre IDM en définissant classiquement le métamodèle des différents services adressables, le métamodèle des intentions pédagogiques ainsi que des règles de transformation. Cette recherche peut être abordée en couplant notre constructeur générique avec la plateforme "Ubilearn" réalisée dans le cadre de la thèse de Vincent Chevrin (Chevrin, 2006). La plateforme multi-agent "Ubilearn" propose une interface de services génériques en aval desquels selon le contexte d'enseignement, elle choisit et réalise l'orchestration de services réels. Par l'intermédiaire de GenDep, la construction ou l'interrogation de dispositifs via ces services deviendrait alors possible. Les travaux actuels menés dans notre équipe de recherche sur la notion de e-portfolio sont un exemple très simple de ce couplage (Kaddouci et al., 2007).

MashUp : Assemblage précaire et hétérogène de fonctionnalité Web

¹² Le projet p-Learnnet a reçu la "labellisation" du Pôle de Compétitivité 'Industries du Commerce' en date du 10 Mai 2006, il s'agit d'un projet ANR volet Télécommunications, dont les partenaires sont : LIFL, Auchan, France Telecom, La Poste, ENST-B, INT Evry, Université de Lille 2, UMVF (Université Médicale Virtuelle Francophone).

L'objectif principal du projet est d'explorer le potentiel offert par l'informatique ubiquitaire dans le champ d'application concernant les Environnements Informatiques pour l'Apprentissage Humain (EIAH). L'Ingénierie Dirigée par les Modèles est ici utilisée de concert avec une approche par ontologies pour générer des environnements d'apprentissage qui peuvent s'adapter à l'apparition ou la disparition de dispositifs électroniques et aux variations d'activités que cela entraîne

Les évolutions que nous venons de décrire autorisent les apprenants à créer, structurer, leur propre enseignement, et constituer un réseau reflétant les activités d'une communauté d'apprentissage (Kahnwald & Köhler, 2006). A ces évolutions dans le domaine des pratiques pédagogiques et des média utilisés s'ajoutent des évolutions logicielles, ces dernières redéfinissent en EIAH les différents rôles décrits dans le chapitre 2.4.

Par des assemblages précaires et hétérogènes de fonctionnalités Web, il est possible de composer une application Web. Cette approche exploitant des composants légers (gadget (Google-API), widget (Wikipedia, 2006)) repose sur une technologie de type Ajax (Garrett, 2005). Elle exploite les Services Web tout en se démarquant de l'approche SOA, plus lourde (l'agrégation ayant lieu chez le client plutôt que sur le serveur). Cette approche permet néanmoins de construire des applications performantes utilisables en entreprise (McAfee, 2006). Avec ce type d'approche, l'utilisateur d'un site Web peut façonner son environnement. Ce façonnage peut être réalisé par des agrégations de contenus, ou par l'ajout de fonctionnalité. L'influence de ce façonnage sur la vie d'une communauté de pratique a par exemple été étudiée par (Goodman & Moed, 2006), les auteurs ont observé l'insertion, dans les pages Web de certains internautes, des indications géographiques des visiteurs et contributeurs de leur espaces. Les intégrations dans l'espace de travail d'un internaute, de dictionnaires, de lecteur de file RSS, de carte conceptuelle présentant les mots clefs des contributions présentes, sont d'autres exemples de ce type de façonnage. Cette possibilité pour un apprenant de construire son propre environnement d'apprentissage et d'évaluation ouvre des perspectives nouvelles en EIAH. Il est par exemple possible d'imaginer un enseignement employant les outils, génériques et hétéroclites, choisis et composés par les apprenants eux-mêmes. Le challenge pour un enseignant ne consiste plus, comme dans les études de cas décrites dans le chapitre 11, à construire des dispositifs sur une même application, détournée ou non, mais destinée à tous. Il consiste à pouvoir interroger facilement les médias hétéroclites utilisés par chaque apprenant, pour pouvoir mener efficacement un travail de suivi. Ce travail réputé fastidieux (Lecllet et al., 2007) peut être instrumenté par notre ingénierie. Ce type d'assemblage logiciel que nous venons de décrire et parfois désigné par le mot français "Bricolage" dans la littérature anglaise (Rees, 2006). Ce mot introduit scientifiquement par Claude Lévi-Strauss (Lévi-Strauss, 1962), nous renvoie aux solutions que nous proposons dans notre thèse.

En guise de conclusion : un paradigme en EIAH basé sur le "Bricolage"

Nous commençons (chapitre 1) cette thèse en décrivant la difficile adaptation à laquelle un enseignant est confronté pour passer de ses intentions pédagogiques à leur concrétisation dans le cadre d'une plateforme. Nous avons montré qu'il est possible de produire une infrastructure pour accompagner cette adaptation.

Dès le début de cette thèse (Caron et al., 2005a) nous avons l'intuition que le Bricolage serait la clef en EIAH d'un nouveau paradigme. Nous vivons actuellement dans un monde où ce mot (Lévi-Strauss, 1962) peut s'appliquer en EIAHs pour décrire des pratiques pédagogiques (Papert, 1991), pour décrire des pratiques de conception de séquences pédagogiques (Perrenoud, 1983), pour décrire des usages des plateformes de formation (Berggren et al., 2005), pour décrire la création par les apprenants eux-mêmes de leurs parcours pédagogique (Mosel, 2005), pour décrire des techniques d'assemblages précaires et hétérogènes d'application Web (Floyd, 2006).

Pour accompagner ces différents bricolages, l'ingénierie dirigée par les modèles propose des solutions basées sur un cadre formel mettant en œuvre les notions de métamodèles et de transformations. Ces notions permettent comme nous l'avons montré dans cette thèse la prise en compte des spécificités techniques ou conceptuelles caractérisant une situation d'apprentissage reposant sur du bricolage. Nous pensons avoir démontré par notre recherche l'intérêt de cette voie en ingénierie pédagogique, pour accompagner les usages enseignants actuels et pour accompagner les usages à venir. Les travaux menés récemment dans notre laboratoire, intégrant nos recherches (Lecllet et al., 2007), (Derycke & Chevrin, 2007), (Kaddouci et al., 2007), (Warin et al., 2007), (Caron, Warin et al., 2007) (Caron, Derycke et al., 2007) sont les indicateurs des potentialités de notre approche à relever les nouveaux défis en EIAH.

12.4 Conclusion

Cette thèse définit une ingénierie, à l'usage des enseignants, pour accompagner la construction de dispositifs pédagogiques sur des plateformes de formation au sens large. Notre recherche s'est attachée à étudier dans un premier temps la notion de plateforme en rattachant les différentes facettes de cette notion à des pratiques pédagogiques, à des pratiques de création, à des modes d'adressage. Ce travail, essentiel pour notre recherche, a permis de dresser un bilan concret des infrastructures permettant l'apprentissage par des moyens informatiques. De ce travail il ressort que les EIAHs que nous utilisons se situent à la frontière de deux modèles.

La premier, classique, ingénieuriste, repose sur une vision industrielle de l'enseignement, soutenu par un modèle économique décrit dans le chapitre 2.4.4. Ce modèle aborde les évolutions actuelles de l'enseignement et de l'informatique en proposant, d'un point de vue ingénieuriste, la plasticité des plateformes de formation par le recours par exemple à la composition de services ou de composants. Ce travail, pour être réalisé, nécessite de définir des interfaces standard d'adressage des plateformes de formation (IMS-ES) (ELF, 2007) (OKI-OSID). Il nécessite également de définir des standards permettant de lisser l'interface homme machine, lors de la composition et l'agrégation de composants (Portlet). Ce modèle ingénieuriste aborde les évolutions de l'enseignement, en promouvant l'individualisation de l'apprentissage par la mise en oeuvre de Workflows adaptatifs pilotant des activités nourrissant des moteurs d'exécution. Les standards (SCORM_2004, 2006) et (IMS-LD) sont des exemples de ces orientations récentes.

Le second modèle, artisanal, promeut un enseignement socioconstructiviste mené par des artisans bricoleurs. Il exploite les ressources d'applications de type Web 2.0 pour les utiliser de façon opportuniste en enseignement. Les mécanismes de MashUp (Floyd, 2006), reposant sur des technologies telles que Ajax (Garrett, 2005), décrits dans ce chapitre, permettent la construction par les apprenants eux-mêmes de leur environnement d'apprentissage. Ceux-ci peuvent composer leur parcours individuel d'enseignement et contribuer à celui-ci sous la forme de micro-contenu. Nous avons montré dans cette thèse que pour un tel modèle, reposant sur la notion de bricolage sous toutes ses formes, il est néanmoins possible de définir, par le recours à l'ingénierie dirigée par les modèles, des infrastructures adaptées permettant d'accompagner le travail des enseignants.

Ces deux modèles ne doivent pas être opposés, ils constituent sur le plan pédagogique, social, économique, et ingénieuriste deux extrêmes d'un même continuum, entre lesquels les EIAHs se placent. La force de notre proposition est de les aborder également et de pouvoir constituer pour l'ensemble de ce continuum des infrastructures facilitant le travail des enseignants.

Nous avons présenté, dans ce chapitre, les diverses évolutions en cours dans les EIAHs, et comment notre recherche pouvait s'inscrire dans chacune d'elle. Certains travaux sont déjà débutés, ils permettent d'aborder la problématique posée par la contextualisation à réaliser dynamiquement dans le cadre d'un enseignement pervasif (Kaddouci et al., 2007), (Derycke & Chevrin, 2007), ils concernent également l'aide à apporter aux enseignants pour le suivi d'apprenants en milieux hétéroclites (Caron, Warin et al., 2007), (Lecllet et al., 2007).

En conclusion, l'ingénierie dirigée par les modèles permet de définir un cadre formel pour exprimer simplement les problèmes résultants du manque d'interopérabilité entre la pensée humaine et des artefacts disponibles pour l'enseignement; elle permet également de résoudre dynamiquement des problèmes posés par l'interopérabilité des plateformes de formation. La capacité de l'IDM à pouvoir définir des infrastructures adaptées à des plateformes spécifiques, en fait une ingénierie particulièrement adaptée aux EIAHs. Nous avons montré que l'IDM constitue en EIAH, une proposition techniquement réalisable. Notre recherche s'est attachée à montrer des exemples d'utilisation d'une telle ingénierie. Des recherches peuvent maintenant être entreprises autour de l'IDM pour relever les défis proposés par les évolutions actuelles des EIAHs.

Bibliographie

- Abd-Ali, J., & El Guemhioui, K. (2005, Sept. 19-23, 2005). *An MDA-Oriented .NET Metamodel*. Paper presented at the 9th International IEEE Enterprise Distributed Object Computing Conference (EDOC 2005), Enschede, The Netherlands.
- Acolad. Acolad. from <http://eppun.u-strasbg.fr/>
- Adkins. (2005). Wake-Up Call: Open Source LMS from <http://www.learningcircuits.org/2005/oct2005/adkins.htm>
- Adobe. Retrieved March 2007, from <http://www.adobe.com/fr/products>
- AFNOR. (2005, août). AFNOR Technologies de l'information pour l'éducation, la formation et l'apprentissage - Profil français d'application du LOM (LOMFR) - Métadonnées pour l'enseignement statut projet de norme. from <http://www.boutique.afnor.fr>
- AICC. AICC. from <http://www.aicc.org/>
- Alagic, S., & Berstein, P. A. (2001). A model theory for generic schema management. *Springer Lecture Notes in Computer Science*, 2397, 228-246.
- Allert, H. (2004). Coherent Social Systems for Learning: An Approach for Contextualized and Community-Centred Metadata. *Journal of Interactive Media in Education*, 2004(2).
- Amigues, R., Faïta, D., & Saujat, F. (2004). Travail enseignant et organisation de l'apprentissage scolaire. E. Gentaz, Dessus, Ph., (Ed.). *Comprendre les apprentissages : Psychologie cognitive et éducation Paris : Dunod*, 155-168.
- Ant. Ant. from <http://ant.apache.org/>
- Ariadne. Ariadne. from <http://www.ariadne-eu.org/>
- Aronson, E., & Patnoe, S. (Eds.). (1997). *The Jigsaw Classroom: Building Cooperation in the Classroom* (2nd edition (September 1996 ed.)). New York: Allyn & Bacon; .
- Audran, J. (Ed.). (2005). *Ethnologie et conception des sites Web scolaires*. Paris: Hermes Science/Lavoisier.
- Axis. Axis. from <http://ws.apache.org/axis/java/ant/axis-wsd12java.html>
- Bachimont, B., Cailleau, I., Crozat, S., Majada, M., & Spinelli, S. (2002, 13-15 nov). *Le procédé SCENARI : Une chaîne éditoriale pour la production de supports numériques de formation* Paper presented at the TICE 2002, Lyon.
- Bachimont, B., & Crozat, S. (2004, 05/07/2004). Instrumentation numérique des documents : pour une séparation fonds/forme. *Revue I3*, 4-1, 95-104.
- Balasubramanian, K., Gokhale, A., Gabor Karsai, Sztipanovits, J., & Neema, S. (2006). Developing Applications Using Model-Driven Design Environments. *Computer*, 39(2), 33-40.
- Barrett, H. C. (2006, June 2006). Using Electronic Portfolios for Formative/Classroom-based Assessment. Retrieved March 2007, from <http://electronicportfolios.org/portfolios/ConnectedNewsletter.pdf>
- Bartolini Bussi, M. G., Mariotti, M. A., & Ferri, F. (2005). Semiotic mediation in the primary school: Dürer glass.
- Basque, J. (1999, 29 octobre 1999). *L'influence du béhaviorisme, du cognitivisme et du constructivisme sur le désign pédagogique*. Paper presented at the XIIe colloque du Conseil interinstitutionnel pour le progrès de la technologie éducative, Montréal, Québec.

- Basque, J., & Doré, S. (1998). Le concept d'environnement d'apprentissage informatisé, . *Journal of Distance Education/Revue de l'enseignement à distance*, 13.
- Becta. (2006a). *Learning platform technical specifications*.
- Becta. (2006b). Web 2.0 – what might it mean for developers? . Retrieved 20 septembre, 2006, from <http://industry.becta.org.uk/display.cfm?resID=20065>
- Béguin, P., & Cerf, M. (2004). Formes et enjeux de l'analyse de l'activité pour la conception des systèmes de travail. *Activités électroniques*, 1(1).
- Béguin, P., & Rabardel, P. (1999). Concevoir pour les activités instrumentées. . *Revue d'Intelligence Artificielle*, 14, 35-54.
- Ben Ari, M. (2001). Constructivism in computer science education. *Journal of Computers in Mathematics and Science Teaching*, 20(1), 45-73.
- Berggren, A., Burgos, D., Fontana, J. M., Hinkelman, D., Vu Hung, A. H., & Tielemans, G. (2005). Practical and Pedagogical Issues for Teacher Adoption of IMS Learning Design Standards in Moodle LMS. *Journal of Interactive Media in Education*,.
- Bertelsen, O. (1997). *Understanding objects in use-oriented design*. Paper presented at the 20th Information systems Research seminar, Scandinavia, Oslo.
- Berten, A. (1998). *Dispositif, médiation, créativité: petite généalogie*. Paper presented at the GReMS Colloque Dispositifs & Médiation des Savoirs, Louvain-la-Neuve.
- Bertrand, A., & Hennequin, X. (2004, janvier 2006). *Les partenariats autour du portail documentaire de l'université de technologie de Compiègne*. Paper presented at the L'indexation des ressources pédagogiques numériques (journée d'étude du 16/11/2004).
- Beryozkin, S. (2003). Web Services and Sessions from <http://webservices.xml.com/pub/a/ws/2003/07/22/sessions.html>
- Bézivin, J. (2001, August 2001). *From Object Composition to Model Transformation with the MDA*, . Paper presented at the TOOLS'USA, Santa Barbara.
- Bézivin, J., Belaunde, M., & Marvie, R. (2004). Transformations et modèles plates-formes. In A. 30 (Ed.), *Ingénierie dirigée par les modèles: OFTA*.
- Bézivin, J., Blay-Fornarino, M., Bouzeghoub, M., Estublier, J., & Favre, J. M. (2005). Rapport de Synthèse IDM. In *Synthèse de l'AS CNRS sur l'Ingénierie Dirigée par les Modèles* (Vol. décembre 2005).
- Bézivin, J., Brunette, C., Chevrel, R., Jouault, F., & Kurtev", I. (2005). *Bridging the Generic Modeling Environment (GME) and the Eclipse Modeling Framework (EMF)*. Paper presented at the Oopsla.
- Bézivin, J., Dupé, G., Jouault, F., Pitette, G., & Rougui, J. (2003). *First experiments with the ATL model transformation language: Transforming XSLT into XQuery*. Paper presented at the OOPSLA 2003 Workshop.
- Bézivin, J., & Gerbé, O. (2001). *Towards a Precise Definition of the OMG/MDA Framework* Paper presented at the ASE'01.
- Bézivin, J., & Jouault, F. (2005). *Using ATL for Checking Models*. Paper presented at the GraMoT.
- Bhanot, V., Paniscotti, D., Roman, A., & Trask, B. (2005, 17th October, 2005). *Using Domain-Specific Modeling to Develop Software Defined Radio Components and Applications* Paper presented at the 5th OOPSLA Workshop on Domain-Specific Modeling (DSM'05), San Diego, California, USA.
- Blanc, X. (2005). *MDA en action* (Vol. 2005).
- Blandin, B. (1990). *Formateurs et formation multimédia - Les métiers, les fonctions, l'ingénierie*. Paris: Editions d'Organisation.
- Blay-Fornarino, M., & Franchi, P. (2005). Espace Technologique Langages et IDM. In *Synthèse de l'AS CNRS sur l'Ingénierie Dirigée par les Modèles* (Vol. décembre 2005).

- Bloom, B. J. (1979). *Caractéristiques individuelles et apprentissages scolaires*. Paris: Nathan Labor.
- Bondé, L., Dumoulin, C., & Dekeyser, J.-L. (2004). *Metamodels and MDA transformations for embedded systems* Paper presented at the FDL04.
- Bouillaux, E., Gardenat, P., Loeillet, G., Pellen, B., Previt, J.-P., & Tabarly, S. (2002). Analyse comparative d'après quelques solutions (classées par principaux types). from <http://www.ac-rennes.fr/tic/glossaire/AnalysePlateformes.htm>
- Boulos, M., Maramba, I., & Wheeler, S. (2006). Wikis, blogs and podcasts: a new generation of Web-based tools for virtual collaborative clinical practice and education. *BMC Med Educ*, 6(41).
- Bounie, D., Catté, M., D'Halluin, C., & Réthoré, S. (2005). *Apport d'un outil de TCAO à la démarche Projet en école d'ingénieurs* Paper presented at the Questions de pédagogie dans l'enseignement supérieur, nouveaux contextes nouvelles expériences. from <http://pedagogie2005.ec-lille.fr/resume/30.pdf>.
- Bourda, Y. (2001). Objets pédagogiques, vous avez dit objets pédagogiques ? . *Actes du congrès GUTenberg*(2001), 71-79.
- Bourdieu, P. (1972). *Esquisse d'une théorie de la pratique*. Genève: Droz.
- Bourguin, G. (2000). *Un support informatique à l'activité coopérative fondé sur la Théorie de l'Activité :Le projet DARE*. Unpublished PhD Thesis, Université des Sciences et technologies de Lille1, Lille.
- Bourguin, G., & Derycke, A. (2005). Systèmes Interactifs en Co-Evolution: Réflexions sur les Apports de la Théorie de l'Activité au support des Pratiques Collectives Distribuées. *Revue d'Interaction Homme-Machine AFIHM Europa*, 6(1), 1-31.
- Bourguin, G., & Hoogstoël, F. (1999). Activités Distribuées dans un Environnement Réflexif. *actes de la 11ème Conférence francophone Interaction Homme-Machine IHM 99*, 2, 21-24.
- Bouyer, A., Rouillard, J., Derycke, A., . (2006). *Pourquoi il n'est pas possible actuellement d'utiliser des objets pédagogiques plastiques respectant la norme SCORM*. Paper presented at the IHM 2006.
- Brassard, C., & Daele, A. (2003). *Un outil réflexif pour concevoir un scénario pédagogique intégrant les TIC*. Paper presented at the EIAH 2003.
- Bru, M. (2006). *Les méthodes en pédagogies* (Vol. 2006). Paris: Presse Universitaire de France.
- Bruner, J. S. (1967). *Toward a theory of instruction*. Cambridge, Massachusetts: The Belknap Press of the Harvard University Press.
- Brusilovsky, P., & Nijhawan, H. (2002). *A framework for adaptive e-learning based on distributed re-usable learning activities*. Paper presented at the E-Learn 2002, Montreal, Canada.
- Budinsky, F., Steinberg, D., & Ellersick, R. (2003). *Eclipse Modeling Framework : A Developer's Guide*: Addison-Wesley Professional, .
- Burgos, D., Arnaud, M., Neuhauser, P., & Koper, R. (2005). IMS Learning Design : la flexibilité pédagogique au service des besoins de la e-formation *La Revue de l'EPI, France: L'association Enseignement Public et Informatique*.
- Büscher, M., Gill, S., Mogensen, P., & Shapiro, D. (2001, 3/1/2001). *Landscapes of Practice: Bricolage as a Method for Situated Design*. Paper presented at the Computer Supported Cooperative Work (CSCW).
- Büscher, M., Mogensen, P., Agger Eriksen, M., & Friis Kristensen, J. (2004). *Ways of grounding imagination* Paper presented at the Participatory Design Conference (PDC).

- Caelen, J., & Jambon, F. (2004). *Conception participative par moments* Paper presented at the Proceedings of the 16th conference on Association Francophone d'Interaction Homme-Machine.
- CanCore. Retrieved March 2007, from <http://www.cancore.ca/fr/>
- Caron, P.-A. (2003). *Spécialisation d'un environnement de conception de systèmes flexibles aux Environnements Informatiques pour l'Apprentissage Humain*. Unpublished Master Thesis, Université des Sciences et Technologie de Lille, Lille.
- Caron, P.-A. (2006, 11-12 mai 2006,). *Implantation de scénarios pédagogiques selon une approche orientée modèle*. Paper presented at the Premières Rencontres Jeunes Chercheurs en EIAH INT, Evry.
- Caron, P.-A. (2007). *Bricoles: une approche dispositive des applications Web 2.0 utilisables pour enseigner*. Paper presented at the EIAH 07, Lausanne, Suisse.
- Caron, P.-A., Derycke, A., Hoogstoel, F., Le Pallec, X., & Warin, B. (2007). *Scénarios et Dispositifs de formations spécialisés : Application de la Démarche d'Ingénierie BRICOLE pour une Instanciation sur MOODLE*. Paper presented at the Colloque scénario 2007, Montreal.
- Caron, P.-A., Derycke, A., & Le Pallec, X. (2005a). *Bricolage and Model Driven Approach to design distant course*. Paper presented at the E learn 2005, world conference on E-learning in corporate Government, Healthcare & higher education.
- Caron, P.-A., Derycke, A., & Le Pallec, X. (2005b, 21-07-2005). *The Bricoles project: support socially informed design of learning environment*. Paper presented at the 12th International Conference on Artificial Intelligence in Education (AIED 2005), Amsterdam.
- Caron, P.-A., Le Pallec, X., & Sockeel, S. (2006a). *Configuring a web based tool through pedagogical scenarios*. Paper presented at the IADIS Virtual Multi Conference on Computer Science and Information Systems (MCCSIS 2006).
- Caron, P.-A., Le Pallec, X., & Sockeel, S. (2006b). *Paramétrage d'application Web par un dispositif pédagogique*. Paper presented at the actes du colloque, "Scénariser l'enseignement et l'apprentissage : une nouvelle compétence pour le praticien ?"
- Caron, P.-A., & Midenet, A. (2002). *Mathematics teachers initial training and collaborative work*. Paper presented at the Second International Conference on Teaching of Mathematics. from <http://www.math.uoc.gr/~ictm2/Proceedings/pap517.pdf>
- Caron, P.-A., Warin, B., Hoogstoel, F., & Le Pallec, X. (2007, June 7-9). *Instrumentation de l'apprentissage en ligne par l'échange pour le suivi de projets étudiants*. Paper presented at the Colloque Echanger Pour Apprendre en Ligne, Grenoble, France.
- Certeau, M., Giard, L., & Mayol, P. (1980). *L'Invention du quotidien*. Paris: Union générale d'éditions.
- Chakroun, M. (2003). *Conception et mise en place d'un module pédagogique pour portails communautaire: INSAT*.
- Charlier, B. (2000). *Utiliser les TIC, une occasion de changer sa pratique ?* Paper presented at the Colloque SYNTIC.
- Chartron, G. (2004, janvier 2006). *L'indexation des ressources pédagogiques numériques : questions transversales*. Paper presented at the L'indexation des ressources pédagogiques numériques (journée d'étude du 16/11/ 2004).
- Chen, H., & Dong, Y. (2002). Modeling and Prototyping of Software Systems. *Software Engineering and Applications*, 374.
- Chevrin, V. (2006). *L'Interaction Usagers/Services, multimodale et multicanale : une première proposition appliquée au domaine du e-Commerce*. Unpublished PhD Thesis, Université de Lille1, LILLE.

- Chevrin, V., Derycke, A., & Rouillard, J. (2006, 19-02-2006). *Project Ubi-Learn: an Intermediation Infrastructure for Multi-channel Accesses to Future LMS*. Paper presented at the ELETE (E-learning and mobile learning on telecommunications) / AICT 2006 (Advanced International Conference on Telecommunications), Guadeloupe, French Caribbean.
- Chewar, C. M., McCrickard, D. S., & Carroll, J. M. (2003). *Persistent Virtual Identity in Community Networks: Impact to Social Capital Value-Chains*. Paper presented at the Human Computer interaction Consortium (HCIC), 2003 Winter Workshop.
- Chikofsky, E. J., & Cross II, J. H. (1990). Reverse Engineering and Design Recovery: A Taxonomy. *IEEE Software*, 7(1), 13-17.
- Ciussi, M., Simonian, S., & Augier, M. (2004). *L'échange favorisé par l'organisation relationnelle du contenu*. Paper presented at the TICE Méditerranée.
- Clark, T., Evans, A., & Kent, S. (2002). *Engineering Modelling Languages: A Precise Meta-modelling Approach*.
- Claroline. Claroline. from <http://www.claroline.net/>
- Cleveland, J. C. (1988). Building Application Generators (Vol. 5, pp. 25-33): IEEE Computer Society Press Y2 - 23 March 2007.
- Clot, Y. (2003). *Catachrèse : éloge du détournement*. Nancy: PUN.
- CMS. CMS. from <http://www.cmsmatrix.org/>
- Competice. Competice. from <http://www.educnet.education.fr/bd/competice/superieur/competice/index.php>
- Corbière, A. (2006). *Analyses des apports du méta-standard, ODP-RM à la communauté EIAH, Instances sur un système de formation*. Unpublished Phd thesis, Université du Maine.
- Corbière, A., & Choquet, C. (2005). ODP-RM : Un cadre de réingénierie des systèmes de formation. *revue Sticef, Numéro spécial : Conceptions et usages des plates-formes de formation*, 12.
- Corbière, A., & Choquet, C. (2004, 1-3 juillet 2004). *A model driven analysis approach for the re-engineering of e-learning systems*. Paper presented at the ICICTE'04, Samos (Grèce).
- Cortés, G., & Guéraud, V. (1998). *Experimentation of an authoring tool for pedagogical simulations*. Paper presented at the International Conference on Computer Aided Learning and Instruction in Science and Engineering (CALISCE'98).
- Crawley, S., Davis, S., Indulska, J., McBride, S., & Raymond, K. (1997). *Meta-Meta is Better-Better*. Paper presented at the Proceedings IFIP TC 6.1 Working Conference DAIS'97, Cottbus, Germany.
- Crozat, S., Majada, M., & Spinelli, S. (2003). *SCENARISup : Un projet pour la gestion des contenus pédagogiques numériques dans l'enseignement supérieur*. Paper presented at the CNUR'2003.
- Czarnecki, K., & Helsen, S. (2003). *Classification of Model Transformation Approaches*. Paper presented at the OOPSLA'03 Workshop on the Generative Techniques in the Context Of Model-Driven Architecture, Anaheim, California, USA.
- D'Halluin, C., & Delache, D. (2005a). *Analyse de dispositifs utilisant un Forum 'étendu' basé sur les échanges de groupe Influences sur les pratiques pédagogiques et les dispositifs*. Paper presented at the Symposium Symfonic "Formation et nouveaux instruments de communication"
- D'Halluin, C., & Delache, D. (2005b). *Entre pratiques pédagogiques et pratiques d'apprentissage, effets induits d'un changement d'interface numérique*. Paper presented at the Colloque du SIF, Les institutions éducatives face au numérique". from <http://sif2005.mshparisnord.org/pdf/Delache.pdf>.

- D'Halluin, C., & Delache, D. (2006). *Etude d'un processus dynamique de construction d'une communauté par interactions entre dispositif et interface numérique*. Paper presented at the Colloque JOCAIR
- Dalziel, J. (2005). From reusable e-learning content to reusable learning designs: Lessons from LAMS.
- De la Teja, I., Lundgren-Cayrol, K., & Paquette, G. (2006). Transposing MISA Learning Scenarios into IMS Units of Learning. *Journal of Educational technology and Society ET&S, 2006*(Special issue on Learning Design).
- De Vries, E. (2001). les logiciels d'apprentissage : panoplie ou éventail ? *Revue Française de pédagogie, 137*, 105-116.
- De Vries, F., Tattersall, C., & Koper, R. (2005). Pre-Discussion Paper Future developments of IMS Learning Design tooling. *Educational Technology & Society, 9*(1), 9-12.
- Dehainsala, H., Jean, S., Dung, N. X., & Pierra, G. (2005, June 2005). *Ingénierie dirigée par les modèles en EXPRESS : un exemple d'application*. Paper presented at the IDM 05, Paris, France.
- Delestre, N., Frénot, S., Mottelet, S., & Vayssade, M. (2002, 13-15 novembre, 2002). *Distributed PolyTeXML Une nouvelle plateforme de partage d'items didactiques*. Paper presented at the TICE 2002, Lyon.
- Deming, W. E. (1982). *Quality, productivity, and competitive position*. Cambridge, MA: Massachusetts Institute of Technology, Center for Advanced Engineering Study.
- Derycke, A., & Chevrin, V. (2007, 12-16th June). *A flexible infrastructure for p-Learning : a first application in the field of professional training*. Paper presented at the 9th International Conference on Enterprise Information Systems, ICEIS 2007, Funchal, Madeira, Portugal.
- Derycke, A., & D'Halluin, C. (1995). Co-operative learning in the distance education of adults: Why, how, and first results from the Co-Learn Project. In B. C. G. Davies (Ed.), *Innovative adult learning with innovative technologies* (pp. 101-122). New York Elsevier.
- Derycke, A., Hoogstoel, F., Peter, Y., & Vantroys, T. (2006). Chapitre 8. In Lavoisier (Ed.), *Environnement Informatiques pour l'apprentissage Humain sous la direction de Jean-Marc Labat et Monique Grandbastien* (Vol. traité IC2, pp. 181): Hermès Sciences Publication.
- Derycke, A., Hoogstoel, F., & Vieville, C. (1997). *Campus virtuel et apprentissages coopératifs*. Paper presented at the Environnements Interactifs d'Apprentissage avec Ordinateur, EIAO'97, Paris.
- Dessus, P. (2002). *Les effets de la planification sur l'activité de l'enseignant en classe* Paper presented at the Les stratégies de l'enseignant en situation d'interaction, Grenoble, Univ. Pierre-Mendès-France.
- Dessus, P., & Schneider, D. K. (2006, 14 avril 2006). *Scénarisation de l'enseignement et contraintes de la situation*. Paper presented at the Colloque Scénariser l'enseignement et l'apprentissage : une nouvelle compétence pour le praticien ?, Lyon : INRP.
- Détienne, F. (2006). Collaborative design: managing task interdependencies and multiple perspectives. *Interacting With Computers, 18*(1), 1-20.
- Dictionnaire_OLF. Dictionnaire terminologique de l'Office de la langue française. from <http://www.granddictionnaire.com/>
- Dinh, T.-L.-A., Gerbé, O., & Sahraoui, H. (2006, July 2006). *Un méta-métamodèle pour la gestion de modèles*. Paper presented at the IDM 06 Actes des 2èmes Journées sur l'Ingénierie Dirigée par les Modèles, Lille, France.
- Dooley, B. J. (2006). *Open Source E-Learning: Alternatives to Proprietary Tools, Systems, and Courseware*.

- DotNetNuke. DotNetNuke. from <http://www.dotnetnuke.com/>
- Dougiamas, M., & Taylor, P. C. (2002). *Interpretive analysis of an internet-based course constructed using a new courseware tool called Moodle*. Paper presented at the Proceedings of the Higher Education Research and Development Society of Australasia (HERDSA) 2002 Conference, Perth, Western Australia.
- Dougiamas, M., & Taylor, P. C. (2003). *Moodle: Using Learning Communities to Create an Open Source Course Management System*. Paper presented at the Proceedings of the EDMEDIA 2003 Conference.
- Downes, S. (2004). Educational Blogging. *EDUCAUSE*, 39(5), 14-26.
- Downes, S. (2005). E-learning 2.0. *National Research Council of Canada ACM - Association for Computing Machinery, Elearn magazine*.
- Dublin-Core. Dublin Core Education. from <http://dublincore.org/>
- ECA. ECA. from <http://www.omg.org/docs/formal/04-02-01.pdf>
- ECMAScript. ECMAScript. from <http://www.ecma-international.org/publications/standards/Ecma-262.htm>
- eCore. metamodel eCore for EMF. Retrieved March 2007, from <http://download.eclipse.org/modeling/emf/emf/javadoc/2.3.0/org/eclipse/emf/ecore/package-summary.html#details>
- Ecoutin, E., & Guidon, J. (2000, novembre). Algora etude 2000 Comparatif technico-pédagogique de plates-formes pour la formation ouverte et à distance. from http://ressources.algora.org/frontblocks/news/papers.asp?id_papers=1266
- Edna. Edna. from <http://www.edna.edu.au/>
- Edutech. (1999). Etude comparative technique et pédagogique des plates-formes pour la formation ouverte et à distance. from www.educnet.education.fr/chrgt/Synt07.doc
- EduTools. EduTools. from <http://www.edutools.info/>
- ELF. (2007). The E-Learning Framework. Retrieved March 2007, from <http://www.elframework.org/>
- Emerson, M. J. (2005). *GME-MOF: an MDA metamodeling environment for GME*. Unpublished Master Thesis, Faculty of the Graduate School of Vanderbilt University.
- EMF. Eclipse Modeling Framework. Retrieved March 2007, from <http://www.eclipse.org/modeling/emf>
- Engelbrecht, J. C. (2003). SCORM Deployment Issues in an Enterprise Distributed Learning Architecture. *The learning developers journal*
- Ertoran, P. (2006). Les Learning Content Management Systems (LCMS) *Actualité de la Formation Permanente*, 200, 71-73.
- Estublier, J., Favre, J. M., & Sanlaville, R. (2002). An Industrial Experience with Dassault Systèmes' Component Model. In M. L. I. Crnkovic (Ed.), *Building Reliable Component-Based Systems* (pp. 375-386): Archtech House.
- Estublier, J., & Sanlaville, S. (2005, 22-26 Aug. 2005). *Extensible Process Support Environments for Web Services Orchestration*. Paper presented at the International IEEE Conference on Next Generation Web Services Practices (NWeSP'05), Séoul, Corée du Sud.
- Estublier, J., Vega, G., & Ionita, A. (2005). *Composing Domain-Specific Languages for Wide-scope Software Engineering Applications*. Paper presented at the MODELS 2005.
- Faerber, R. (2001). *Accompagner les apprentissages à distance et collaborer en petits groupes*, . Paper presented at the Quatrième congrès AECSE 2001.
- Farooqui, K., Logrippo, L., & Meer, J. d. (1996). The ISO Reference Model for Open Distributed Processing- An Introduction. *Computer Networks and ISDN Systems*, 27, 1215-1229.

- Favre, J.-M. (2006, July 2006). *Concept fondamentaux de l'IDM, tutoriel*. Paper presented at the IDM 06 Actes des 2èmes Journées sur l'Ingénierie Dirigée par les Modèles, Lille, France.
- Favre, J.-M., & Musset, J. (2006, July 2006). *Rétro ingénierie dirigée par les métamodèle*. Paper presented at the IDM 06 Actes des 2èmes Journées sur l'Ingénierie Dirigée par les Modèles.
- Ferraris, C., Lejeune, A., Vignollet, L., & David, J. P. (2005, 25-27 mai 2005). *Modélisation de scénarios d'apprentissage collaboratifs pour la classe*. Paper presented at the EIAH 2005, Montpellier, France.
- Fiedler, S., & Kieslinger, B. (2006, June 8-9, 2006). *Adapting to Changing Landscapes in Education (On Microlearning)* Paper presented at the Microlearning 2006, Innsbruck, Austria.
- Flichy, P. (1995). *L'innovation technique : récents développements en sciences sociales, vers une nouvelle théorie de l'innovation*. Paris: La Découverte.
- Floyd, I. R. (2006, November, 2006). *Using Mashups for End-user rapid and responsive prototyping in collaborative environments*. Paper presented at the Wokshop Web Mash-ups and CSCW: Opportunities and Issues. At the 20th ACM Conference on Computer Supported Cooperative Work. , Banff, Alberta, Canada.
- Foucault, M. (1975). *Surveiller et punir : naissance de la prison*. Paris: Gallimard.
- Frete, C., Schneider, D., & Synteta, P. (2002). Activity Design: "CSCL light" TECFA::SEED Project::Working Paper 10. from <http://tecfa.unige.ch/proj/seed/catalog/>
- Friesen, N. (2004). Three Objections to Learning Objects. . In R. McGreal (Ed.), *Online Education Using Learning Objects* (pp. 59-70). London: Routledge.
- Ganesha. Ganesha. from <http://www.anemalab.org/ganesha/>
- Garrett, J. J. (2005, February 18). Ajax: A New Approach to Web Applications. Retrieved March 2007, from <http://www.adaptivepath.com/publications/essays/archives/000385.php>
- Gebers, E. (2005, 12 et 13 décembre 2005). *Peut-on s'affranchir des standards*. Paper presented at the Colloque du SIF: Les institutions éducatives face au numérique, Carré des Sciences, Paris, France.
- GEMS. Generic Eclipse Modeling System. from <http://sourceforge.net/projects/gems/>
- George, S. (2001). *Apprentissage collectif à distance. SPLACH : un environnement informatique support d'une pédagogie de projet*. Unpublished PhD Thesis, Université du Maine, Le Mans.
- George, S. (2005). *Vers des forums dédiés au contexte éducatif*. Paper presented at the Symposium Symfonic: Formation et nouveaux instruments de communication.
- George, S., & Derycke, A. (2005). Editorial *STICEF: Numéro spécial : Conceptions et usages des plates-formes de formation, 12*.
- Giacomini-Pacurar, E., Trigano, P., & Alupoiaie, S. (2005). NetUniversité: un portail web utilisant l'IMS Learning design. *RES - ACADEMICA AIPU*, 23(1).
- Giacomini-Pacurar, E., Trigano, P., & Alupoiaie, S. (2006). Concevoir des modèles de sites webs éducatifs en utilisant IMS Learning Design. *Canadian Journal of Learning and Technology - La Revue Canadienne de l'Apprentissage et de la Technologie*, 32(1).
- Giordan, A. (1994). Le modèle allostérique et les théories contemporaines sur l'apprentissage. In (pp. 289-315). Berne: Peter Lang.
- Giordan, A., & Girault, Y. (1996). *The new learning models, their consequences for the teaching and the mediation of Biology, Health and Environment: Z' éditions* collection Penser Agir.
- Globe. Globe. from <http://globe.edna.edu.au/>
- GME. The Generic Modeling Environment from <http://www.isis.vanderbilt.edu/projects/gme/>

- GMF. Graphical Modeling Framework from <http://www.eclipse.org/gmf/>
- Goodman, E., & Moed, A. (2006, november 2006). *Community in Mashups: The Case of Personal Geodata*. Paper presented at the Wokshop Web Mash-ups and CSCW: Opportunities and Issues. At the 20th ACM Conference on Computer Supported Cooperative Work. , Banff, Alberta, Canada.
- Google-API. Google API. from <http://code.google.com/>
- Granath, J. A., Lindah, G. A., & Rehal, S. (1996). From Empowerment to Enablement. An evolution of new dimensions in participatory design. *Logistik und Arbeit*, 8.
- Greenberg, L. (2002). LMS and LCMS: What's the Difference? , from <http://www.learningcircuits.org/NR/exeres/72E3F68C-4047-4379-8454-2B88C9D38FC5.htm>
- Greenfield, J., & Short, K. (2003). *Software factories: assembling applications with patterns, models, frameworks and tools*. Paper presented at the Companion of the 18th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications.
- Guéraud, V. (2006, avril 2006). *Une approche auteur pour les scénarios d'activités*. Paper presented at the Colloque, Scénariser l'enseignement et l'apprentissage: une nouvelle compétence pour le praticien?, INRP, Lyon, France.
- Gueraud, V., Adam, J.-M., Pernin, J.-P., Calvary, G., & David, J. (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*, 11(2004).
- Guzdial, M. (2000). *A Catalog of CoWeb Uses*: Georgia Institute of Technology.
- Guzdial, M., Rick, J., & Kehoe, C. (2001). Beyond adoption to invention: Teacher-created collaborative activities in higher education. *Journal of the Learning Sciences*, 10(3), 265-279.
- Haladjian, R. (2003). *Devenez beau, riche et intelligent, avec PowerPoint, Excel et Word*.
- Hartswood, M., Procter, R., Slack, R., Voß, A., Büscher, M., Rouncefield, M., et al. (2002). Co-realisation: Towards a principled synthesis of ethnomethodology and participatory design. *Scandinavian Journal of Information Systems*, 14(2), 9-30.
- Hernández-Leo, D., Asensio-Pérez, J. I., Dimitriadis, Y., Bote-Lorenzo, M. L., Jorrín-Abellán, I. M., & Villasclaras-Fernández, E. D. (2005). Reusing IMS-LD Formalized Best Practices. *Collaborative Learning Structuring Advanced Technology for Learning*, 2(3), 223-232.
- Hoogstoël, F. (1995). *Une approche organisationnelle du travail coopératif assisté par ordinateur application au projet Co Learn*. Unpublished PhD Thesis, Université des Sciences et technologies de Lille1, Lille.
- Huberman, M. (1980). *Recipes for Busy Kitchens : A Situational Analysis of Routine Knowledge Use in Schools*. Washington, D.C: National Institute of Education, Program on Research and Educational Practice.
- Hung, D., & Chen, D.-T. (2003). A Proposed Framework for the Design of a CMC Learning Environment: Facilitating the Emergence of Authenticity. *Educational Media International*, 40(1), 7-14.
- IEEE, L. T. S. C. o. t. (2002). *Draft Standard for Learning Object Metadata*.
- IMS-ES. IMS-ES. from <http://www.imsproject.org/es/>
- IMS-LD. IMS LD. from <http://www.imsproject.org/learningdesign/>
- IMS-Packaging. IMS Packaging from <http://www.imsproject.org/content/packaging/>
- IMS-QTI. IMS QTI. from <http://www.imsproject.org/question/>
- Ionita, A. D., Estublier, J., & Vega, G. (2005, June 2005). *Domaines Réutilisables Dirigés par les Modèles*. Paper presented at the IDM 05, Paris, France.

- Jézéquel, J.-M., Gérard, S., Mraidha, C., & Baudry, B. (2005). Approche unificatrice par les modèles. In *Synthèse. de l'AS CNRS sur l'Ingénierie Dirigée par les Modèles*: décembre 2005.
- Jonassen, D. H., Peck, K. L., & Wilson, B. G. (1999). *Learning with technology : a constructivist perspective*. Upper Saddle River, N.J.: Merrill.
- Joomla. Joomla. from <http://www.joomla.org/>
- Kaddouci, S., Vantroys, T., & Chevrin, V. (2007, 3/14/2007 - 3/16/2007). *From Task Model to Multi-Channel Access: Services Integration in the Ubi-Learn Platform*. Paper presented at the Web-based Education, Chamonix France.
- Kahnwald, N., & Köhler, T. (2006). *Microlearning in Virtual Communities of Practice? An explorative analysis of changing information behaviour (Changing Patterns of Learning: Schools, Universities, Vocational Training)*. Paper presented at the MicroLearning 2006, Innsbruck, Austria.
- Kent, S. (2002). *ModelDrivenEngineering*. Paper presented at the 3rd Intl. Conf. on Integrated Formal Method-IFM 2002, Turku, Finland.
- Kiczales, G., Lamping, J., Mendhekar, A., Maeda, C., Lopes, C. V., Loingtier, J.-M., et al. (1997). *Aspect-Oriented Programming*. Paper presented at the ECOOP 97.
- Kommers, P. A. M., Grabinger, R. S., & Dunlap, J. C. (1996). *Hypermedia learning environments : instructional design and integration*. Mahwah, N.J.: L. Erlbaum Associates.
- Koper, R. (2001). Modelling units of learning from a pedagogical perspective: The pedagogical meta-model behind EML. from <http://www.learningnetworks.org/downloads/ped-metamodel.pdf>
- Koper, R., & Olivier, B. (2004). Representing the Learning Design of Units of Learning. *Educational Technology & Society*, 7(3), 97-111.
- Koper, R., & Tattersall, C. (2005). Preface to Learning Design: A Handbook on Modelling and Delivering Networked Education and Training. *Journal of Interactive Media in Education* 2005(18), 1.
- Lacourse, F. (2004). *La construction des routines professionnelles chez de futurs enseignants de l'enseignement secondaire: intervention éducative et gestion de la classe*. Unpublished PhD Thesis, Université de Sherbrooke.
- Laforcade, P. (2004). *Méta-modélisation UML pour la conception et la mise en oeuvre de situations-problèmes coopératives*. Unpublished Phd Thesis, Université de Pau.
- Laforcade, P. (2005, Juillet 5-8, 2005). *Towards a UML-based Educational Modeling Language*. Paper presented at the The 5th IEEE International Conference on Advanced Learning Technologies, Kaohsiung, Taiwan,.
- Laforcade, P., Nodenot, T., Caron, P.-A., & Choquet, C. (2007). Model-Driven Engineering (MDE) and Model-Driven Architecture (MDA) applied to the Modeling and Deployment of Technology Enhanced Learning (TEL), Systems: promises, challenges and issues. In C. Pahl (Ed.), *Architecture Solutions for E-Learning Systems*. Hershey, PA: IDEA Group Inc.
- Laforcade, P., Nodenot, T., & Sallaberry, C. (2005). Un langage de modélisation pédagogique basé sur UML. *revue STICEF Sciences et Technologies de l'Information et de la Communication pour l'Éducation et la Formation*, 12(Numéro spécial : Conceptions et usages des plates-formes de formation).
- Landon, B. (2002, June 27, 2002). *Hard Choices for Individual Situations: Selecting a Course Management System*. Paper presented at the ED-MEDIA 2002, Denver, Colorado.
- Lattion, S. (2005). Développement et implémentation d'un module d'apprentissage par investigation (inquiry-based learning) au sein d'une plateforme de type PostNuke.

- Le Guennec, A., Sunyé, G., & Jézéquel, J.-M. (2000, octobre 2000). *Precise Modeling of Design Patterns*. Paper presented at the UML'00.
- Le Pallec, X. (2001). *RAM3 : un outil dynamique pour le Meta-Object Facility*. Paper presented at the LMO2001: Langages et Modèles à Objets, Le Croisic, FRANCE.
- Le Pallec, X. (2002). *Des services d'adaptation de modèles pour la coopération de méta-systèmes : application aux groupware flexibles*. Unpublished PhD Thesis, Université des Sciences et technologies de Lille1, Lille.
- Le Pallec, X., Moura, C., Marvie, R., Nebut, M., & Tarby, J.-C. (2006, 05-07-2006). *Supporting generic methodologies to assist IMS-LD modeling*. Paper presented at the ICALT 2006.
- Learnativity. Learnativity from <http://www.learnativity.com>
- Leclercq, G. (2005). *Les dispositifs de formation professionnalisés comme situation d'auto et d'inter-confrontation. L'impact d'un environnement numérique de travail*. Paper presented at the Colloque du SIF, Les institutions éducatives face au numérique.
- Leclercq, G., Leprêtre, E., Peter, Y., Quénu-Joiron, C., Talon, B., & Vantroys, T. (2007). *Améliorer un dispositif pédagogique par l'intégration de nouveaux canaux de communication*. Paper presented at the EIAH 2007, Lausanne Suisse.
- Ledeczi, A., Bakay, A., Maroti, M., Volgyesi, P., Nordstrom, G., Sprinkle, J., et al. (2001). *Composing Domain-Specific Design Environments*. *Computer Networks and ISDN Systems*, 44-51.
- Ledeczi, A., Maroti, M., Bakay, A., Karsai, G., J. G., IV, C. T., et al. (2001, May 17,2001). *The Generic Modeling Environment*. Paper presented at the Workshop on Intelligent Signal Pro-cessing,, Budapest,Hungary.
- Legendre, R. (1993). *Dictionnaire actuel de l'éducation*. Montréal: Guérin ou Paris : ESKA.
- Legros, D., & Crinon, J. (2002). *Psychologie des apprentissages et multimédia* (Vol. 2002). Paris: Armand Colin.
- Lejeune, A., & Pernin, J.-P. (2004, dec. 2004). *A taxonomy for scenario-based engineering*. Paper presented at the Cognition and Exploratory Learning in Digital Age (CELDA 2004), Lisboa, Portugal.
- Lévi-Strauss, C. (1962). *La pensée sauvage*. [Paris]: Plon.
- Levi Strauss, C. (1962). *The Savage Mind*. Chicago, IL: University of Chicago Press.
- Lexique_Formation. Lexique de la Formation Ouverte et/ou à distance. from <http://www.univ-angers.fr>
- Linard, M. (1998). *L'écran de TIC, dispositif d'interaction et d'apprentissage : la conception des interfaces à la lumière des théories de l'action*. Paper presented at the GReMS Colloque Dispositifs & Médiation des Savoirs, Louvain-la-Neuve.
- Linard, M. (2002). *Conception de dispositifs et changement de paradigme en formation. Education permanente, Regards multiples sur les nouveaux dispositifs de formation*, 152(octobre), 43-155.
- LOM. Lom. from <http://ltsc.ieee.org/wg12/20020612-Final-LOM-Draft.html>
- Lornet. LORNET. from <http://www.lornet.org>
- m2m. m2m. from <http://www.eclipse.org/proposals/m2m/>
- Mallein, P., & Toussaint, Y. (1994). *L'intégration sociale des technologies d'information et de communication : une sociologie des usages*. *Technologies de l'information et société*, 6(4), 315-335.
- Manolescu, D. A., & Johnson, R. E. (1999). *Dynamic Object Model and Adaptive Workflow. Metadata and Active Object-Model Pattern Mining Workshop*. Paper presented at the OOPSLA'99, Denver, USA.
- Martel, C. (1998). *La modélisation des activités conjointes. Rôles, places et positions des participants*. Unpublished PhD Thesis, Université de Savoie.

- Martel, C., Ferraris, C., Caron, B., Carron, T., Chabert, G., Courtin, C., et al. (2004). *A model for CSCL allowing Tailorability: Implementation in the Electronic Schoolbag Groupware*. Paper presented at the 10th CRIWG conference, San Carlos, Costa Rica.
- Martel, C., Vignollet, L., Ferraris, C., David, J. P., & Lejeune, A. (2006). *Modeling collaborative learning activities on e-learning platforms*. Paper presented at the IEEE International Conference on Advanced Learning Technologies (ICALT 06), Kerkrade, The Netherlands.
- Marvie, R., Duchien, L., & Blay-Fornarino, M. (2005). Les plates-formes d'exécution et l'IDM. In *synthèse de l'AS CNRS sur l'Ingénierie Dirigée par les Modèles* (pp. 71-98).
- Marvie, R., & Nebut, M. (2006, Juin 2006). *Processus de modélisation incrémentaux*. Paper presented at the IDM 06, 2nd Journées sur l'Ingénierie Dirigée par les Modèles (IDM'06), Lille, France.
- McAfee, A. P. (2006). Enterprise 2.0: The Dawn of Emergent Collaboration. *Spring 2006* 47(3), 21-28.
- McAndrew, P. (2003). Can generic models be reused and shared. *LTSN Generic Center Self Session*.
- McAndrew, P., Nadolski, R., & Little, A. (2005). Developing an approach for Learning Design Players. *Journal of Interactive Media in Education, Special Issue, Advances in Learning Design, Colin Tattersall, Rob Koper, 14(2005)*.
- McCalla, G. (2004). The Ecological Approach to the Design of E-Learning Environments: Purpose-based Capture and Use of Information About Learners. *Journal of Interactive Media in Education, 2004/7*(Special Issue on the Educational Semantic Web).
- McCullough, M. (1998). *Abstracting craft: The practiced digital hand*. Cambridge MA: MIT Press.
- Merle, N. (2004, 28-29 Octobre 2004). *Un métamodèle pour l'automatisation du déploiement d'applications logicielles*. Paper presented at the Conférence Francophone sur le Déploiement et la (Re) Configuration de Logiciels (DECOR'04), Grenoble, France.
- Merlin. Merlin from <http://sourceforge.net/projects/merlingenerator/>
- Merlot. Merlot. from <http://www.merlot.org/>
- Merrieu, P. (1999). *Un nouvel art d'apprendre*. Paper presented at the Intervention aux Entretiens de la Villette. from http://www.unige.ch/fapse/SSE/groups/life/textes/Meirieu_A2001_02.html.
- MetaEdit. MetaEdit supports your modeling language from <http://www.metacase.com/mep/>
- Module_PostNuke. Modules PostNuke. from <http://tecfaseed.unige.ch>
- ModX. ModX: Modelling Everything ! . from <http://noce.univ-lille1.fr/projets/ModX/>
- Moodle. course management system. from <http://moodle.org/>
- Moolinex. Moolinex. from <http://ganesha.keonox.com/addons/moolinex2/>
- Mortensen, T., & Walker, J. (2002, 8 April 2002). *Blogging thoughts: personal publication as an online research tool*. Paper presented at the SKIKT-RESEARCHERS' CONFERENCE 2002 Researching ICTs in Context, University of Oslo.
- Mosel, S. (2005, 23-24, 2005). *Self Directed Learning With Personal Publishing and Microcontent, Constructivist Approach and Insights for Institutional Implementations*. Paper presented at the Microlearning 2005, Innsbruck, Austria.
- MOT Plus. Mot et Mot +. Retrieved March 2007, from <http://www.licef.telug.quebec.ca/fr/realisations/mot1.htm>
- Moura, C., & Derycke, A. (2007). *Concevoir des Scénarios Pédagogiques Exécutables avec des Patrons de Conception Pédagogiques*. Paper presented at the EIAH 07, Lausanne, Suisse.
- MTL. MTL. from <http://modelware.inria.fr/mtl>

- Muszynski, M., & AG, T. (2005, 17th October, 2005). *Implementing a Domain-Specific Modeling Environment For a Family of Thick-Client GUI Components*. Paper presented at the 5th Workshop on domain specific modelling, San Diego, California, USA.
- NIME. Nime. from <http://www.nime.ac.jp/index-e.html>
- Nodenot, T. (2006a). *Contribution à l'ingénierie dirigée par les modèles en EIAH: le cas des situations problèmes coopératives*. Unpublished Habilitation à diriger les recherches, Université de Pau et des Pays de l'Adour.
- Nodenot, T. (2006b, avril 2006). *Etude du potentiel du langage IMS-LD pour scénariser des situations d'apprentissage: résultats et propositions*. Paper presented at the colloque, Scénariser l'enseignement et l'apprentissage : une nouvelle compétence pour le praticien ?, INRP, Lyon, France.
- Nuke. Nuke. from <http://en.wikipedia.org/wiki/Nuke>
- O'Coill, C., & Doughty, M. (2004). *Computer game technology as a tool for participatory design*. Paper presented at the Proceedings of the eCAADe 2004 22nd conference: Architecture in the Network Society.
- O'Hear, S. (2006). e-learning 2.0 - how Web technologies are shaping education. Retrieved 28 mars 2007, from http://www.readwriteweb.com/archives/e-learning_20.php
- O'Reilly, T. (2005). What Is Web 2.0, Design Patterns and Business Models for the Next Generation of Software. Retrieved march 2007, from <http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html>
- OKI-OSID. O.K.I. Open Service Interface Definitions. from <http://www.okiproject.org/>
- OMG. OMG :Object Management Group. . from www.omg.org
- OMG. (1997). *Meta Object Facility (MOF) Specification*.
- OMG. (2001). *Request For Proposal: \MOF 2.0 Query / Views / Transformations RFP", Final Adopted Specification*
- OMG. (2003). *MDA GUIDE Version 1.0.1 document number omg/2003-06-01*.
- OMG. (2004). *Metamodel and UML Profile for Java and EJB Specification, Version 1.0, formal/04-02-02* (An Adopted Specification of the Object Management Group, Inc).
- P-learnnet. *apprentissage pervasif*.
- Papavassiliou, G., Mentzas, G., & Abecker, A. (2002). *Integrating knowledge modelling in business process management*. Paper presented at the Xth European Conference on Information Systems (ECIS-2002), Gdansk, Poland.
- Papert, S. (1991). *Situating constructionism*. Norwood, NJ: Ablex Publishing Corporation.
- Paquelin, D. (2005, 12 et 13 décembre 2005). *Planification versus Potentialisation. De la structuration des contenus à la structuration de la contenance*. Paper presented at the Colloque du SIF, "Les institutions face au numérique", Carré des Sciences, Paris.
- Paquelin, D., & Choplin, H. (2003). Se former dans un dispositif ouvert et à distance : l'enjeu des régulations. In A. B. (Ed.), *Autoformation et enseignement supérieur : au-delà de l'effet de mode une conception éducative* (pp. 166-183): Hermès éditeur.
- Paquette, G. (2001). Designing Virtual Learning Centers. In B. C. H. Adels-berger, J. Pawlowski (Ed.), *Handbook on Information Technologies for Education & Training* (pp. 249-272): Springer-Verlag
- Paquette, G. (2005). Apprentissage sur l'Internet : des plateformes aux portails à base d'objets de connaissance. In S. Pierre (Ed.), *Innovations et tendances en technologies de formation et d'apprentissage* (pp. 1-30): Presses de l'école polytechnique de Montréal.
- Paquette, G. (2006). Introduction à la spécification IMS-LD. D'une perspective d'ingénierie pédagogique. from http://helios.liceftelug.quebec.ca/residld/2/Introduction_à_IMSLD.doc

- Paquette, G., De la Teja, I., & Léonard, M. (2005). An Instructional Engineering Method and Tool for the Design of Units of Learning. In R. K. C. Tattersall (Ed.), *Learning Design: A handbook on Modelling and Delivering Networked Education and Training*. Berlin Heidelberg New York: Springer.
- Paquette, G., De la Teja, I., Lundgren-Cayrol, K., Léonard, M., & Ruelland, D. (2002). La modélisation cognitive, un outil de conception des processus et des méthodes d'un campus virtuel. *Revue de l'Éducation à distance, Edmonton, Numéro spécial RCE-Téléapprentissage, 17(3)*, 4-25.
- Paquette, G., Marino, O., De la Teja, I., Léonard, M., & Lundgren Cayrol, K. (2005). Delivery of Learning Design: the Explor@ System's Case. In R. K. C. Tattersall (Ed.), *Learning Design: A handbook on Modelling and Delivering Networked Education and Training*. Berlin Heidelberg: New York Springer.
- Paquette, G., Ricciardi-Rigault, C., de la Teja, I., & Pasquin, C. (1997). Le Campus Virtuel: un réseau d'acteurs et de moyens diversifiés. *Revue de l'Association canadienne d'éducation à distance, 12(1/2)*, 85-101.
- Paquette, G., Rosca, I., Teja, I. D. I., Léonard, M., & Lundgren-Cayrol, K. (2001, 2001). *Web-based Support for the Instructional Engineering of E-learning Systems*. Paper presented at the WebNet'01 Conference, Orlando.
- Passadière, B. d. L., & Jarraud, P. (2004). ManUeL, un profil d'application du LOM pour C@mpuSciences® *revue STICEF : Sciences et Technologies de l'Information et de la Communication pour l'Éducation et la Formation, 11(2004)*.
- PCDAI. (2006). Monographie du projet PCDAI
- Peeters, H., & Charlier, P. (1999). Contributions à une théorie du dispositif. In *Le dispositif entre usage et concept, cognition communication politique* (Vol. 25, pp. 15-23): Hermes.
- Peraya, D. (1998). Le cyberspace : un dispositif de communication et de formation médiatisées. In A. S. (Ed.), *Cyberspace et autoformation* (Vol. REF-98): De Boeck.
- Pernin, J.-P. (2003). Objets pédagogiques : unités d'apprentissage, activités ou ressources ? . *Revue "Sciences et Techniques Educatives", Hors série 2003 " Ressources numériques, XML et éducation"*, 179-210.
- Pernin, J.-P. (2004, janvier 2006). *LOM, SCORM et IMS-Learning Design : ressources, activités et scénarios*. Paper presented at the L'indexation des ressources pédagogiques numériques (journée d'étude du 16/11/ 2004).
- Pernin, J.-P., & Lejeune, A. (2004a). *Dispositifs d'apprentissage instrumentés par les technologies : vers une ingénierie centrée sur les scénarios*. Paper presented at the colloque TICE 2004.
- Pernin, J.-P., & Lejeune, A. (2004b). *Nouveaux dispositifs instrumentés et mutations du métier de l'enseignant*. Paper presented at the 7ème biennale de l'Éducation.
- Perrenoud, P. (1983). La pratique pédagogique entre l'improvisation réglée et le bricolage, Essai sur les effets indirects de la recherche en éducation. *La formation des enseignants entre théorie et pratique, Éducation & Recherche, 1983(2)*, 198-212.
- Portlet. Java Specification Requests, JSR 168: Portlet Specification. Retrieved March 2007, from <http://jcp.org/en/jsr/detail?id=168>
- PostNuke. PostNuke. from <http://www.postnuke.com/>
- Préau. Le préau Plates-formes et portails de téléformation Étude 2000.
- Preece, J. (2004). Etiquette, Empathy and Trust in Communities of Practice: Stepping-Stones to Social Capital. *UCS 10(3)*, 294-302.
- Rasseneur, D. (2004). *Saafir : un environnement support à l'appropriation d'une formation à distance par l'apprenant*. Unpublished PhD Thesis, Université du Maine.

- Rees, M. (2006, 22 January 2006). Bricolage - a better word for mashup? *Words and Phrases*, from <http://blog.mrees.biz/myblog/?p=9>
- Reload. Retrieved March 2007, from <http://www.reload.ac.uk/>
- Rosca, I. (1999). *Vers une vision systémique du processus de l'explication. Récit d'une recherche sur l'intégration de la pédagogie, de l'ingénierie et de la modélisation*. Unpublished PhD Thesis, Université de Montréal.
- Roth, M. A., Philipp, J., & Osburg, T. (2006). *Micro-learning Elements in the Intel Advanced Training Course (Changing Patterns of Learning: Schools, Universities, Vocational Training)*. Paper presented at the MicroLearning 2006, Innsbruck, Austria.
- Rouillard, J., Chevrin, V., Tarby, J.-C., & Derycke, A. (2006). La plasticité des documents numériques. *Document numérique de la Revue TSI - Revue Technique et Science Informatiques Hermes-Lavoisier Paris*, 4(2006).
- Sanlaville, S., & Estublier, J. (2005). Melusine: un environnement de modélisation et de coordination de service. *Revue d'Ingénierie des Systèmes d'information*, 12(3).
- Santos, O. C., Boticario, J. G., & Barrera, C. (2004, 26-30 juillet). *Authoring a Collaborative Task Extending the IMS LD to be performed in a standard-based Adaptive Learning Management System called ALFANET*. Paper presented at the International Conference on Web Engineering, workshop AHCW'04, Munich.
- Sceren. Sceren from <http://educasources.education.fr/>
- Schank, P., & Hamel, L. (2004). *Collaborative Modeling: Hiding UML and Promoting Data Examples in NEMO*. Paper presented at the Computer Supported Cooperative Work (CSCW), Chicago, Illinois, USA.
- Schneider, D. (2003). *Conception and implementation of rich pedagogical scenarios through collaborative portal sites*. Paper presented at the Innovative Learning & Knowledge Communities / les communautés virtuelles: apprendre, innover et travailler ensemble, ICOOL 2003 /Colloque de Guéret 2003, University of Mauritius.
- Schuler, D., & Namioka, A. (1993). *Participatory Design: Principles and Practices*. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.
- SCORM. Scorm. from <http://www.adlnet.gov/>
- SCORM_2004. (2006). SCORM 2004 3rd edition. Retrieved March 2007, from <http://www.adlnet.gov/scorm>
- Scott, L. (2003). Some Uses of Blogs in education. Retrieved march 2007, from <http://www.edtechpost.ca/gems/matrix2.gif>
- Seidewitz, E. (2003). What Models Mean. *IEEE Software*, 20(5), 26-32.
- Selic, B. (2003). The pragmatics of Model Driven Development. *IEEE software*, 20(5), 19-25.
- Sfard, A. (1998). On two metaphors for learning and the dangers of choosing just one. *Educational Researcher*, 27(2), 4-13.
- Simondon, G. (1989). *Du mode d'existence des objets techniques* (Ed. augm. / ed.). [Paris]: Aubier.
- Simplylearn. Managed Learning Environment. Retrieved March 2007, from <http://www.simplylearn.co.uk/>
- Sjunnesson, J. (2001). *Digital learning portfolios: inventory and proposal for Swedish teacher*. Uppsala University.
- Skinner, B. F. (1968). *The technology of teaching*. New York,: Appleton-Century-Crofts.
- Solnon, C. (1997). Ateliers de Génie Logiciel. Retrieved March 2007, from <http://bat710.univ-lyon1.fr/~csolnon/>
- SPIP. SPIP from <http://www.spip.net/>
- Sriplakich, P. (2003). *Techniques des transformations de modèles basées sur la méta-modélisation*. Unpublished Master Thesis, Université Pierre et Marie Curie.

- Stacey, P. (2003). *People to People, not just people to content*. Paper presented at the Presentation at the IMS Open Technical Forum. from <http://www.bctechonology.com/statics/pstacey-feb1403.html>.
- Star, S. L. (1989). The Structure of Ill-Structured Solutions: Boundary Objects and Heterogeneous Distributed Problem Solving. In L. Gasser & M. N. Huhns (Eds.), *Distributed Artificial Intelligence* (Vol. 2, pp. 37-54). London: Pitman.
- SVL. Shared Virtual Laboratory. from http://www.noe-kaleidoscope.org/pub/lastnews/symposium_2005/svl.html
- SWEET. SWEET. from http://www.elearning.ac.uk/frameworks/toolkit1projects/sweet/topic_view
- Synteta, P., Schneider, D., & Frété, C. (2002). *Activity software : C3MS bricks for socio-constructivist scenarios*.
- Talon, B., Toffolon, C., & Warin, B. (2005.). *Projet en milieu universitaire : vers une gestion collaborative assistée par le Web*. *Revue internationale des technologies en pédagogie universitaire*, 2(2), 28-33.
- Talon, B., Toffolon, C., & Warin, B. (2007, 24-26 janvier 2007). *Accompagner les projets en milieu universitaire. Présentation d'une méthodologie d'encadrement de projets collaboratifs assistée par le Web*. Paper presented at the Colloque "Questions de pédagogies dans l'enseignement supérieur", Louvain-la-Neuve, Belgique.
- Tattersall, C., & Koper, R. (2004). EML and IMS Learning Design: from LO to LA. *Educational Technology Expertise Centre The Open University of the Netherlands*.
- Tchounikine, P. (2002). *Quelques éléments sur la conception et l'ingénierie des EIAH*. Paper presented at the 2ème assises nationales su GDR I 3, Nancy, France.
- Tchounikine, P. (2004). Platon-1 : quelques dimensions pour l'analyse des travaux de recherche en conception d'EIAH, Rapport de l'Action Spécifique « Fondements théoriques et méthodologiques de la conception des EIAH », département STIC du CNRS. (2004). from <http://telearn.ccsd.cnrs.fr/telearn-00080490/en/>
- Tchounikine, P. (2006). Introduction à l'ingénierie des EIAH. In M. Grandbastien & J.-M. Labat (Eds.), *Environnements informatiques pour l'apprentissage humain* (Vol. Traité IC2 Information Commande Communication, pp. 141-160): Lavoisier Hermes.
- Tecfa. Technologies de Formation et Apprentissage. Retrieved March 2007, from <http://tecfa.unige.ch/>
- Terrasse, M.-N., Savonnet, M., Leclercq, E., Grison, T., & Becker, G. (2005). *Points de vue croisés sur les notions de modèle et métamodèle*. Paper presented at the IDM 05, Paris, France.
- Thot. Thot. from <http://thot.cursus.edu/>
- Tilley, S., & Huang, S. (2003, October 12-15, 2003). *A qualitative assessment of the efficacy of UML diagrams as a form of graphical documentation in aiding program understanding*. Paper presented at the 21st Annual International Conference on Documentation, San Francisco, CA.
- TLF. Trésor de la Langue Française, dictionnaire en ligne from <http://atilf.atilf.fr/>
- Tochon, F. V. (1996). Grammaires de l'expérience et savoirs-objets : le savoir focal dans la construction des nouveaux modèles de formation. In J.-M. Barbier (Ed.), *Savoirs théoriques et savoirs d'action* (pp. 249-273). Paris: P.U.F.
- Tolvanen, J.-P. (2004). Making model-based code generation work. *Embedded Systems Europe*, 8(60).
- Tolvanen, J.-P. (2005). Making model-based code generation work - Practical examples (Part 2). *Embedded Systems Europe*, 9(64).
- Tolvanen, J.-P. (2006). Domain-Specific Modeling: How to Start Defining Your Own Language. *DevX.com*.

- Tolvanen, J.-P., & Rossi, M. (2003). *MetaEdit+: defining and using domain-specific modeling languages and code generators*. Paper presented at the Companion of the 18th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications.
- TopClass. TopClass e-Learning Suite. from <http://www.wbtsystems.com/solutions/technology>
- Touvet, D. (2005, 30/11/2005). Learning Object Repositories: tutoriel pour gérer des repositories avec Moodle! , from <http://www.davidtouvet.com/blog/archives/2005/11/30/learning-object-repositories-tutoriel-pour-gerer-des-repositories-avec-moodle/>
- Towers, G. (1995). *Building Democracy: Community Architecture in the Inner Cities*. London: UCL Press.
- Typo3. Typo 3. from <http://www.typo3.fr/>
- Uden, L. (2003). An Engineering Approach for Online Learning. *Journal of Distance Education Technologies*, 1(1), 63-77.
- Université, B., Cailleau, I., Crozat, S., Majada, M., & Spinelli, S. (2002). *Le procédé SCENARI : Une chaîne éditoriale pour la production de supports numériques de formation*. Paper presented at the Technologies de l'Information et de la Communication dans les Enseignements d'ingénieurs et dans l'industrie, Villeurbanne : Institut National des Sciences Appliquées de Lyon.
- Vachet, C. (2006). *Scénarisation Pédagogique, Situation Interactionnelle et IDM*. Paper presented at the IDM 06. from <http://planetmde.org/idm06/posters/14.pdf>.
- Vantroys, T. (2003). *Du langage métier au langage technique, une plateforme flexible d'exécution de scénarios pédagogiques*. Unpublished PhD Thesis, Université de Lille 1, Lille.
- Vantroys, T., & Peter, Y. (2005). COW, une plate-forme de support d'exécution de scénarios pédagogiques. *Revue STICEF : Sciences et Technologies de l'Information et de la Communication pour l'Education et la Formation*, 12(numéro spécial "Conceptions et usages des plates-formes de formation).
- Vara, J. M., de Castro, V., & Marcos, E. (2005). *WSDL Automatic Generation from UML Models in a MDA Framework*. Paper presented at the Next Generation Web Services Practices, Seoul, Korea.
- Varga, R. (2006). Suivi de stages sur une plate-forme pédagogique PostNuke., from <http://edutice.archives-ouvertes.fr/edutice-00106233>
- Vega Baez, G. E. (2005). *Développement d'Applications à Grande Echelle par Composition de Méta-Modèles*. Unpublished PhD Thesis, Université Joseph Fourier.
- Vicente, K. J., & Rasmussen, J. (1992). Ecological interface design: Theoretical foundations. *IEEE Transactions on Systems, Man, and Cybernetics*(1992, ieeexplore.ieee.org).
- Viéville, C., & Passardièrre, B. d. L. (2003). Mise en œuvre des métadonnées pour les ressources éducatives et construction de parcours de formation personnalisés. In É. Bruillard & B. d. L. Passardièrre (Eds.), *Sciences et Techniques Educatives* (Vol. Hors Série- 2003, Numéro spécial, Ressources numériques, XML et éducation).
- Villiot-Leclercq, E. (2006). *Conception de scénarios pédagogiques : un dispositif d'assistance pour soutenir l'interaction entre l'enseignant et l'environnement ExploraGraph*. Paper presented at the colloque Scénarios 2006.
- Vygotski, L. S. (1985). *Pensée et langage*. Paris: Messidor, Éditions sociales.
- Wang, H., & Zhang, D. (2003, november 3-6 2003). *MDA-based Development of E-Learning System*. Paper presented at the 27th Annual International Computer Software and Applications Conference (Compsac 2003), Dallas, Texas, USA.

- Warin, B., Caron, P.-A., Le Pallec, X., & Hoogstoel, F. (2007). *Le projet Metawep, Ingénierie dirigée par les modèles de dispositifs web support à l'apprentissage par projet*. Paper presented at the EIAH 2007, Lausanne, Suisse.
- Wartofsky, M. W. (1973). *Perception, representation, and the forms of action: toward an historical epistemology*, . Dordrecht:: D. Reidel Publishing Company.
- WebCT. WebCT. from <http://www.webct.com/>
- Weiser, M. (1991). Les réseaux informatiques de l'an 2000. *Pour la science*, 169(novembre 1991).
- Wenger, E. (1998). *Communities of Practice: Learning, Meaning, and Identity*, . NewYork: Cambridge University Press.
- WfMC. The Workflow Management Coalition modèle de référence. from <http://www.wfmc.org/standards/standards.htm>
- Wikipedia. (2006). Widget. Retrieved March 2007, from <http://fr.wikipedia.org/wiki/Widget>
- Wile, D. S. (2001). Supporting the DSL Spectrum. *Journal of Computing and Information Technology CIT 9, 2001(4)*, 263-287.
- Wiley, D. (2002). Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy. In D. A. Wiley (Ed.), *The instructional use of learning objects* (pp. 3-25). Bloomington: Agency for Instructional technology.
- Wiley, D., Waters, S., Dawson, D., Lambert, B., Barclay, M., Wade, D., et al. (2004). *Overcoming the Limitations of Learning Objects* (Vol. 13, pp. 507-521). Norfolk, VA: AACE.
- Wilson, B., & Lowry, M. (2001). Constructivist Learning on the Web. In L. Burge (Ed.), *Learning Technologies: Reflective and Strategic Thinking*. San Francisco: Jossey-Bass, New Directions for Adult and Continuing Education.
- Zhang, W., Mei, H., Zhao, H., & Yang, J. (2005, October 2-7, 2005). *Transformation from CIM to PIM: A Feature-Oriented Component-Based Approach*. Paper presented at the Model Driven Engineering Languages and Systems, 8th International Conference MoDELS 2005, Montego Bay, Jamaica.

Glossaire

Affordance : La notion d'affordance perçue caractérise la capacité pour un humain d'appréhender spontanément les fonctionnalités d'un objet. Cette notion est utile pour décrire les fonctionnalités offertes par une plateforme de formation (cf. chapitre 2), pour établir le métamodèle décrivant la plateforme de formation (cf. chapitre 9.2.3), et enfin pour expliquer les assemblages précaires de fonctionnalité Web utilisables par les apprenants pour composer leur environnement d'apprentissage (cf. 12.3.2).

Application Web : Dans notre thèse nous étudions des applications Web utilisées à des fins d'enseignement. Ces applications peuvent être des plateformes de formation telles qu'elles sont définies au chapitre 3.2, mais notre ingénierie adresse également des applications plus spécifiques telles que des Forum, Wiki, Blog ou Portfolio. Enfin dans le chapitre 12.3 nous évoquons plus spécifiquement des applications Web 2.0 pouvant être recomposée par les utilisateurs par des mécanismes de "mash-up".

Approche : Différentes approches sont proposées dans cette thèse. En EIAH nous évoquons, chapitre 4, les approches documentaires, cognitivistes et par scénario, permettant à un enseignant d'adresser une plateforme de formation. Notre approche dispositif est décrite chapitre 5. En IDM, différentes approches permettent d'opérationnaliser un modèle, ces approches sont décrites chapitre 6.6.

Artefact : Cette notion est abordée au chapitre 2, pour distinguer l'objet "plateforme de formation", de l'instrument au service des enseignants. Elle est également utilisée, au chapitre 8, pour distinguer les différents types d'artefacts (cf. page 113) permettant de médiatiser la conception participative.

Atelier de génie logiciel : Nous proposons dans notre thèse deux ateliers distincts: **Outillage**, et **Construction**. (cf. page 134). L'atelier Outillage, destiné à l'informatien et à l'ingénieur pédagogue, permet de réaliser le modeleur et le constructeur spécifique à la plateforme de formation. L'atelier Construction, destiné à l'ingénieur pédagogique et à l'enseignant, permet la modélisation et la construction d'un dispositif de formation sur une plateforme de formation. Ces actions sont effectuées dans l'usage de l'infrastructure réalisée lors de l'atelier Outillage.

Automatisable : Nous utilisons ce terme pour indiquer des données pouvant être effectivement manipulable par un système d'information, (cf. chapitre 4, page 53). Un modèle de scenario est automatisable s'il peut être exprimé dans un formalisme manipulable par un tel système. Nous distinguons les adjectifs "automatisable" et "opérationnalisable", ce dernier étant utilisé pour indiquer la réalisation possible d'un but.

Bricolage : Le chapitre chapitre 8 aborde la notion de bricolage, nous utilisons cette notion pour caractériser l'activité qui consiste pour un enseignant à faire avec les moyens du bord. Nous distinguons cette activité de celle de type ingénieuriste qui consiste à construire de toute pièce un objet adapté à son projet. Le paradoxe de notre démarche consiste à mettre en place une ingénierie pour favoriser ce bricolage enseignant. *"Le bricoleur ne subordonne pas les tâches qu'il mène, à l'obtention de matières premières et d'outils, conçus et procurés à la mesure de son projet : son univers instrumental est clos, et la règle de son jeu est de toujours s'arranger avec les 'moyens du bord'" (Levi Strauss, 1962).* Nous utilisons également la notion de Bricolage pour faire référence aux potentialités d'une activité de type improvisation réglée. Pour mettre en place cette espace de potentialisation permettant le bricolage enseignant nous proposons une approche dispositif (cf. chapitre 5). *"Il est donc indispensable dans ce cas de quitter un idéal de maîtrise ..."* (Audran, 2005) *"La question qui taraude le pédagogue en permanence est donc de laisser à l'autre la place qui lui revient*

pour qu'il construise son propre savoir. Pour cela, le pédagogue en est réduit au bricolage" (Merrieu, 1999).

Catachrèse : Ce terme permet de caractériser les actions de détournements et d'adaptations qu'un objet subit lors de son instrumentation. Ce terme peut être utile pour éclairer la définition d'une plateforme de formation que nous proposons (page 34) Il est également sous-jacent à la notion de "finalisation dans l'usage" abordée au chapitre 8. Enfin il permet de comprendre l'usage des applications de type Web 2.0 en enseignement (cf. 12.3.2).

Chaîne de modélisation : Nous utilisons ce terme en IDM pour décrire, chapitre 6, une suite de modélisations s'effectuant dans un cadre de conformité mettant en œuvre des métamodèles différents relier par des règles de transformations (cf. 6.4). En EIAH, des chaînes de modélisation, permettent la production de documents (Bachimont et al., 2002), ou d'environnement d'apprentissage (Giacomini-Pacurar et al., 2006) (cf 7.1.1).

Collection : Pour construire un dispositif à partir du modèle qui le décrit, nous utilisons la notion de collection, un élément du modèle représente une collection d'objet, chacun des objets de la collection possède un mécanisme lui permettant de construire sur la plateforme les éléments constituant le dispositif décrit (cf. 9.3.4).

Concrétisation : La notion de concrétisation est utilisée à la fin du chapitre 5 pour établir un parallèle entre un processus MDA (cf. 6.4) et la nécessaire concrétisation que des intentions pédagogiques doivent subir pour construire à partir de celles-ci un dispositif permettant de les mettre en œuvre (cf. 5.4.2).

Contextualisation : La phase de contextualisation permet dans une démarche de bricolage pédagogique de prendre en compte la réalité des services disponibles sur une plateforme. Nous utilisons également cette notion pour décrire le dialogue qu'un scénario ou un dispositif doit mettre en œuvre pour prendre en compte des activités de co-construction émergeant de l'usage d'une plateforme de formation (cf. chapitre 11 et chapitre 12) Pour mettre en œuvre un enseignement socio-constructiviste, cette phase est réputée essentielle (cf. page 66).

Conforme : Relation qui lie un modèle à son métamodèle, la relation de conformité peut être vérifiée de différentes façons, dans le cadre de cette thèse la relation de conformité est mis en œuvre par des spécifications exécutable au sein de notre plateforme de formation ModX (cf. 6.3.2).

Dispositif : Nous définissons la notion de dispositif au chapitre 5. Dans notre thèse un dispositif est une entité triple, c'est une offre de signification (cf. 5.3), c'est un objet frontière (cf. 8.2.2), c'est la mise en œuvre sur une plateforme de formation de moyens techniques (cf. 10.8.1).

DSL : Domain Specific Language, en IDM les langages spécifiques de domaines proposent une approche pragmatique de la modélisation, ils peuvent être mis en œuvre au sein d'une plateforme de modélisation, en définissant un métamodèle de domaine conforme par exemple au MOF. Un tel métamodèle permet alors de définir un modèleur spécifique au sein duquel des modèles peuvent facilement être élaborés.

Etude de cas : Nous présentons dans notre thèse trois études de cas, ces études permettent de valider une partie de nos choix instrumentaux (cf. chapitre 11). Bien que le lecteur puisse y trouver quelques réflexions ethnométhodologiques, ces études consistent essentiellement en des études de faisabilité.

Factorise : Terme que nous introduisons pour décrire la relation qui lie un élément du modèle à la partie du système qu'il décrit (cf. 9.3.4).

GenDep : Notre générateur de constructeur spécifique (cf. 9.3.3 et 10.5).

Générateur de code : La notion de générateur de code est abordée section 6.6, un générateur de code est en génie logiciel une solution banale pour automatiser la production de code à partir de spécification. Dans cette thèse nous suivons la démarche préconisée par (Tolvanen, 2004) (cf. page 95).

IDM : L'ingénierie dirigée par les modèles (IDM), en anglais Model Driven Engineering (MDE), est une approche spécifique du génie logiciel qui a pour objectif de définir un cadre théorique pour générer du code en utilisant des transformations successives de modèle. L'IDM a également pour objectif la manipulation, par des métamodèles, de modèles **productifs** par opposition à des modèles descriptifs, contemplatifs. Cette productivité se décline de deux manières : un modèle doit être manipulable informatiquement, il doit aussi être opérationnalisable. (cf chapitre 6)

Infrastructure : L'objet de l'atelier Outillage que nous envisageons est de produire pour chaque plateforme de formation une infrastructure de modélisation adaptée (cf. 8.6). *"Une infrastructure peut être définie comme un substrat social et technique qui stabilise et permet rapidement des activités instrumentées intentionnelles dans un domaine donné"* (Derycke et al., 2006).

Instrument : Nous considérons dans cette thèse les "plateformes de formation" en tant qu'instruments choisis, définis (dans leur usage) et utilisés par les enseignants. Pour (Béguin & Rabardel, 1999) *"l'artefact désigne un objet nu, indépendamment de toute relation avec un usager et l'instrument est une entité mixte composée de l'artefact et des schèmes d'utilisation construits par l'usager"*.

Mash-up : Un Mash-up désigne est une agrégation précaire de fonctionnalité Web, ce terme est utilisé dans cette thèse au chapitre 12 pour désigner des mécanismes permettant aux usagers de composer leur propre environnement Web.

Méta métamodèle : Un méta métamodèle, définit les notions de base permettant l'expression des métamodèles, et des modèles, il permet d'exprimer les règles de conformités qui lient les modèles aux métamodèles. Un méta métamodèle peut être réflexif c'est-à-dire que dans ce cas il se définit lui-même. Le MOF (Meta Object Facility) standardisé par l'OMG, eCore défini dans le cadre du "Eclipse Modeling Framework" (EMF) (Budinsky et al., 2003), différents formalismes comme des profils UML (Unified Modeling Language), des graphes conceptuels, sNets, CDIF, OWL (Dinh et al., 2006) sont des exemples de méta métamodèles (cf chapitre 6).

Métamodèle : Un métamodèle est conforme au méta métamodèle choisi pour l'exprimer. Le MDA repose sur un métamodèle principal UML extensible par exemple par des profils. Les métamodèles que nous produisons avec notre plateforme de modélisation ModX sont conformes au méta métamodèle MOF, ils permettent de définir des modeleurs spécifiques basés sur un ou plusieurs métamodèles décrivant un domaine particulier (cf chapitre 6).

Méthode : La troisième étude de cas, que nous décrivons section 11.4, aborde la notion de méthode. Une méthode est une entité composée de principes, d'une démarche et de moyens de mise en oeuvre. Dans le cadre de la méthode Mepulco Université, la démarche est ensemble coordonné d'étapes permettant de guider les étudiants dans la réalisation de leurs projets et offrant aux tuteurs une assistance à l'encadrement de ces projets. Les principes de la méthode sont les suivants: 1. La répartition des responsabilités. 2. La sollicitation régulière de l'équipe. 3. Le respect d'une méthode qualité de type PDCA (Plan Do Check Act) (Deming, 1982). 4. La concrétisation d'un espace de communication et de gestion. 5. La capitalisation finale. Enfin des moyens complètent et permettent de mettre en oeuvre la démarche initiée par les principes.

Micro contenu : Un micro-contenu est typiquement un article soumis dans un blog, un commentaire, une photo placée sur un site de partage etc...Ce qui caractérise un micro contenu, d'un autre type de contenu, disponible sur le Web, c'est de pouvoir être référencé, et interprété, informatiquement, via la lecture par exemple de fichier RSS. La publication individuelle de micro-contenu sur des applications Web 2.0, tend à dessiner une vue subjective du monde réel (Mosel, 2005) (cf chapitre 12).

Modèle : Un des objectifs de l'IDM est de produire des modèles qui puissent être **validés formellement**, cette validation favorisant leur manipulation par des outils informatiques et

leur opérationnalisation (Blay-Fornarino & Franchi, 2005). C'est cette validation qui distingue l'IDM des méthodes de modélisations antérieures. Un modèle est valide s'il est conforme à son métamodèle, les règles de conformités étant définies au niveau du méta méta modèle. (cf chapitre 6). L'IDM a aussi pour objectif la manipulation par des métamodèles de modèle **productifs**, dans le cadre de cette thèse, nous explicitons comment il est possible pragmatiquement de rendre productif un modèle de dispositif pédagogique (cf. 9.3).

ModX : La plateforme de modélisation que nous utilisons pour générer des modeleurs spécifiques (cf. 10.2.2).

Objet frontière: C'est un objet perçu et abordé par tous mais compris différemment (Star, 1989). Un objet frontière permet de coordonner les perspectives des différentes composantes d'une communauté. Il crée un lieu d'échange et permet de mettre en place un processus de conception participative (cf. 8.2.2)

Opérationnaliser : Nous introduisons dans cette thèse la notion d'opérationnalisation, un scénario pédagogique est **opérationnalisable**, si concrètement il est possible, par des moyens informatiques, de construire, à partir de son expression, la séquence pédagogique qu'il scénarise (cf 4.49.3.2). Nous proposons dans cette thèse de rendre opérationnalisable des dispositifs pédagogiques. Cette opérationnalisation nécessite la réalisation pour chaque plateforme de formation d'un constructeur spécifique. L'outil que nous avons développé GenDep permet d'automatiser la production d'un tel constructeur. (cf 9.3.2).

Pervasif : L'apprentissage Pervasif fait référence à un apprentissage utilisant des réseaux pervasifs. Un réseau pervasif est un réseau dans lequel il est possible d'être connecté partout et à tout moment par l'intermédiaire d'objets communicants classiques ainsi que grâce à des objets équipés d'une capacité de mémoire et d'intelligence. Apprendre, dans un tel contexte, c'est utiliser l'intelligence ambiante du réseau pour favoriser l'apprentissage (cf chapitre 12).

Plateforme d'exécution L'IDM aborde de façon générale la notion de plateforme d'exécution : une plateforme n'est pas exclusivement un environnement de programmation dans lequel est généré du code à partir d'un modèle (Czarnecki & Helsen, 2003). L'IDM définit une plateforme d'exécution comme un interpréteur pour l'exécution de systèmes (Marvie et al., 2005). Une plateforme d'exécution établit le modèle des structures de données qu'elle manipule et traite, et pour lesquelles elle définit des règles de cohérence. Dans le domaine des EIAHs, il est possible de considérer une plateforme de formation comme une plateforme d'exécution permettant l'interprétation de dispositifs pédagogiques. Construire un dispositif sur une telle plateforme consiste alors à créer sur celle-ci des éléments en respectant la cohérence du système qu'ils forment (cf section 9.2.1).

Plateforme de formation : Dans notre thèse nous définissons, l'objet plateforme de formation par rapport aux fonctionnalités induites par l'usage que les enseignants font de cet objet. Une plateforme de formation est un EIAH un peu particulier, dans la mesure où sa finalité première n'est pas l'apprentissage mais la mise en œuvre et le suivi de l'apprentissage. C'est un environnement informatique collectif permettant à une communauté d'enseignants de produire les conditions qui favorisent l'apprentissage et le suivi des apprenants (cf 3.2)

Plateforme de modélisation : Une plateforme de modélisation implémentant un méta méta modèle permet d'établir des règles de conformité bien fondées entre métamodèles et modèles. (Blay-Fornarino & Franchi, 2005). De telles plateformes permettent alors la définition efficace de différents modeleurs. GMF (GMF), GME (GME), MerlinGenerator (Merlin), GEMS (GEMS), MetaEdit (MetaEdit), ModX (ModX) sont des exemples de plateforme de modélisation utilisant les concepts de l'IDM. (cf chapitre 6 et section 10.2.2)

Processus : Nous décrivons dans cette thèse un processus de concrétisation qui permet de relier langage, modélisation, transformation et construction au sein d'une infrastructure pouvant être adaptée à chaque plateforme de formation. Ce processus de concrétisation est mis en œuvre dans le cadre de l'IDM en nous référant au processus MDA. Le Model Driven

Architecture (MDA) (OMG, 2003) est un exemple particulier d'ingénierie dirigée par les modèles. C'est une sorte de guide structuré qui préconise une approche de modélisation basée sur un processus de concrétisation proche de notre recherche en EIAH (cf chapitre 6).

Scénario : Nous présentons dans cette thèse trois exemples de mises en oeuvre de la notion de scénario (Koper, 2001), (Schneider, 2003), (Paquette et al., 2002) (cf. 4.3). Pour analyser ces diverses mises en oeuvre nous utilisons la grille d'analyse définie par (Lejeune & Pernin, 2004). Nous montrons la difficulté pour un scénario automatisable et opérationnalisable de décrire un enseignement socio-constructiviste. La nécessaire contextualisation, inhérente à ce type d'enseignement, introduit un indéterminisme antithétique avec la notion de planification caractéristiques des scénarios pédagogiques. Dans le chapitre 5, nous opposons l'espace planifié, décrit par un scénario, à l'espace de potentialité décrit par un dispositif pédagogique.

Services web : Notre ingénierie repose sur l'utilisation d'un greffon de service Web encapsulant l'API des applications ciblées. Ce greffon est utilisé pour établir un protocole de communication entre notre constructeur spécifique et une plateforme de formation. Des initiatives similaires existent, ainsi l'initiative SWEET propose une approche dans le domaine du e-learning, d'applications Web via les Services Web. C'est un kit de développement permettant de mettre en oeuvre la spécification IMS Enterprise Services dans le cadre d'échange de données (étudiants, administration, classes, groupes) entre applications. Le site ELF (The E-Learning Framework) propose quant à lui de référencer les initiatives permettant de spécifier les services utilisables pour adresser les plateformes de formation. Aucune de ces initiatives ne reposent, comme notre proposition, sur l'utilisation de l'IDM pour générer efficacement des outils adaptés aux services définis.

Annexe

Annexe A : Tableau récapitulatif des liens plateforme modèles d'apprentissages et situation initiale de notre recherche

Théories \ fonctionnalités	Béhaviourisme	Cognitivisme	Constructivisme	Socioconstructivisme
Communiquer	Avec l'enseignant (chat forum, messagerie)	Communiquer avec soit même Support pour la métacognition, (blog)	Communication associée à l'activité, (forum, blog, dépôt de document, messagerie)	Communication avec l'enseignant avec le groupe (Chat forum blog, dépôt de documents, boîte à crier, rating wiki...)
Augmenter la productivité	Positionnement dans son apprentissage	Calendrier d'apprentissage	Gestion de projet	Calendrier partagé, syndication de documents
Mener une évaluation	Test individuel	Pré test Post test	Rendre un devoir, outils de soumission	Zone de dépôts, outils de rating, wiki peer reviewing
Administrer	Positionner l'élève dans son cursus	Positionner l'élève par rapport aux connaissances acquises	Créer pour l'élève les accès aux différents outils de production	Gérer les droits, outils, privilèges associés à l'élève dans son groupe et au groupe dans la formation
Dispenser des cours	Importation de séquence, outils statistique de positionnement, Importer des tests	Accès aux viviers de connaissances, importation de séquence, Importer Test	Permettre des interactions avec des documents riches, accepter des outils de simulation, micro monde etc.	Structurer des espaces, affecter des outils de communication et d'édition collective à ces espaces
Respecter de la charte de la plateforme	Respect de séquençement, interfaçage de test, SCORM IMS-LD, IMS QTI	Métras donnés du LOM, SCORM IMSLD	Protocole de syndication simple de ses propres travaux, support pour e portfolio cf. norme dtd forum	<i>Norme permettant la construction de dispositifs?</i>

Annexe B : Modèles pédagogiques

Le tableau suivant issu du cours de Chantal D'Halluin permet décrire plus facilement les modèles pédagogiques auxquels nous nous référons : (socioconstructivisme, médiationnel et motivation humaine). L'apprentissage collaboratif pratiqué sur la plateforme Accel est décrit dans (Derycke & D'Halluin, 1995) : « *l'acquisition par les individus de connaissances, compétences, ou attitudes à travers des interactions dans des groupes où les membres se partagent le travail et/ou développent une représentation partagée de la tâche à réaliser* »

courant	constructivisme	béhaviorisme / associationnisme	apprentissage social	psychologie sociale génétique /socio constructivisme	courants médiationnels	théorie de la motivation humaine	cognitivisme
initiateur	Piaget	Skinner Watson Pavlov	Bandura	Doise Mugny Perret- Clermont	Vygotski Bruner	Nuttin	Newell et Simon Richard, Anderson
origine	Suisse	USA CEI	USA	Suisse	CEI, USA	Belgique	USA, France
époque d'origine	1920	1938	années 70	1974	1919 et +	1980	1972, 1975
principes	On apprend par l'action Autonomie de l'apprenant	On apprend par les conséquences de nos actes Influence du milieu Hétéronomie de l'apprenant	On apprend en imitant des modèles Importance de l'identification Déni de l'action	On apprend en se confrontant aux pairs	L'apprentissage tire sa source de l'interaction sociale Importance du langage	L'apprentissage est avant tout gouverné par un projet	On acquiert des connaissances en traitant de l'information
notions	équilibre assimilation accommodation schèmes, opérations	Contingences de renforcement Stimulus - réponses Boite noire (ignorée)	Apprentissage par observation Attention sélective Renforcement vicariant	Conflit sociocognitif Marquage social	Zone de proche développement Médiation Etayage	Formation de but Formation de projet Adéquation moyens/buts	Représentations Résolution de problèmes
applications	Educabilité cognitive	Enseignement Programmé Pédagogie par objectifs	Etude des modèles agressifs à la TV	Apprentissage coopératif	Remédiation cognitive Apprentissage coopératif	Formation individualisée	Intelligence artificielle

Annexe C : Texte de description de l'EAPC

Texte non publié de Gilles Leclercq, reproduit avec l'aimable autorisation de l'auteur.

Il faudrait d'ailleurs plutôt parler d'un Explorateur d'Action Personnelle et Collective (EAPC). Je ne sais pas si le terme est bon, mais prenons le comme raccourci.

Cela se passe comme ceci : dans l'espace potentiel chacun peut créer un espace personnel. Cet espace perso est singularisé (automatiquement) par l'identité de la personne (nom, prénom, année). Dans l'utilisation de l'EAPC cet espace perso est obligatoirement structuré par 5 balises :

- prendre place dans une organisation, construire son tutorat ;
- formaliser sa mission et la questionner, raconter comment elle évolue ;
- mener des investigations de terrain ;
- mener des investigations conceptuelles ;
- capitaliser les références bibliographiques.

Pour bien saisir l'enjeu de cet EAPC, il faut comprendre que nous souhaitons mémoriser un processus et que l'espace est pré-structuré.

A partir de ces "pages balises", l'étudiant pourra créer toutes les pages qu'il souhaite.

Ce qui s'écrit dans chaque espace personnel a un destinataire (le référent universitaire avec qui s'engage un dialogue dans les texte qu'écrit l'étudiant). On obtient ainsi des textes qui rendent progressivement intelligible l'action professionnelle pour l'étudiant et pour le référent universitaire.

On peut comparer l'EAPC à une collection d'espace personnel ayant une structure de base commune (les balises) et s'enrichissant d'années en années (le nombre d'espace collectionné s'accroît). Cette collection forme aussi une mémoire collective que chacun peut visiter. Les espaces sont personnels, ils ne sont pas privés.

C'est la première fonction de l'EAPC. Il permet de constituer une collection d'espace personnel qui forment ensemble un espace collectif où une mémoire collective. Cette mémoire sera utile dans un deuxième temps à chaque étudiant, quand il s'agira de l'exploiter (d'y prélever des ressources) pour écrire son mémoire professionnel.

La première grande famille d'activité consiste donc à composer une mémoire personnelle qui soit en même temps une mémoire collective. Cela se fait en créant des pages dans son espace perso. Le mieux serait de pouvoir les écrire sur un mode what you see, what you get, mais à défaut, l'éditeur d'un wiki pourrait convenir. Il importe néanmoins qu'il soit équipé d'un barre d'outil. En résumé, on crée des pages dans son espace perso avec des mots wiki...

La deuxième grande famille d'activité consiste à exploiter la mémoire collective. Elle se décompose en sous-activités :

Inviter quelqu'un à consulter l'EAPC. L'invitation peut être simple : « je viens d'écrire cela, j'aimerais que vous y jetiez un œil, allez y en cliquant ici... ». Mais elle peut être beaucoup plus construite : allez voir ceci, puis ceci, puis ceci... ». Cette possibilité devient progressivement utile pour le référent universitaire qui peut ainsi prévoir, au fur et à mesure que la mémoire collective s'enrichie, des parcours en son sein. Ce qui se trouve dans l'EAPC peut en effet servir de complément, voir d'alternative aux enseignements... Pour que cet usage soit possible, il faut pouvoir se rendre dans des pages, mais aussi pouvoir déposer ici et là des signets qui permettent un repérage plus précis. Un parcours expérimenté et mémorisé peut ainsi être conseillé à tel étudiant... Dans un Wiki, inviter quelqu'un revient à composer une page wiki qu'on peut analyser comme un métatexte, un texte qui exploite la mémoire collective.

2ème activité d'exploitation : rechercher quelque chose dans l'EAPC. Cela doit pouvoir se faire en croisant : nom, année, date, Balise, signet, mot clef, expression... Cela permet par exemple à un référent universitaire de prendre connaissance de ce qu'a écrit récemment telle personne. La fonction recherche dans un wiki....

3ème activité d'exploitation : faire son marché. .Cette fonction permet de déplacer, d'exporter ou d'importer facilement du texte. L'importation est utile pour déposer un texte écrit par ailleurs. L'exportation sera très utile dans la seconde phase de la formation, quand il s'agira de récupérer des textes dans l'EAPC pour les déposer dans son traitement de texte perso, autrement dit, pour rédiger le mémoire proprement dit. Un presse papier un peu évoluer serait bien venu. On peut le comparer à un ensemble de tiroirs qu'il est possible de remplir et de vider à volonté.

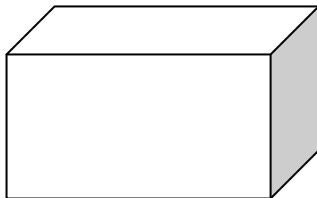
Il faut bien comprendre :

- que l'EAPC est un outil pour accompagner la première phase d'une action professionnelle, celle où l'on écrit pour comprendre.

- Que nous faisons l'hypothèse qu'il est important de mémoriser le processus pour faciliter la vie d'une communauté d'expérience et pour faciliter la relation entre communauté d'expérience et communauté d'apprentissage

- qu'une deuxième phase de travail consistera à exploiter l'EAPC en exportant les ressources qui s'y trouvent dans un traitement de texte, afin d'élaborer un mémoire.

J'ai dépassé les 10 lignes. Pour le dessin à main levé, j'ai envi de faire comme Saint Exupéry... Quand le petit prince lui demande de dessiner un mouton... il n'y parvient pas. Alors il dessine une boîte... L'EAPC, il est dans la boîte.



Je m'excuse pour ce texte un peu long, mais c'était pour moi la manière la plus rapide de procéder.

Annexe D : Implémentation des méthodes "deploy" et "fetch"

La première méthode : "deploy"

Pour qu'un objet puisse se construire via des appels de service il faut implémenter pour la classe non abstraite dont il est une instance, une méthode de construction ("**deploy**").

L'exemple présenté (cf. Figure 120) est extrait de l'étude de cas réalisée sur la plateforme Ganesha

Pour implémenter la méthode "**deploy**", il faut non seulement identifier le service à appeler mais aussi lui communiquer les valeurs des attributs requis. Ceux-ci sont de deux types :

- Les premiers concernent des attributs d'identification rendus disponibles par la classe "**PlatformSpecific**", cet attribut correspond dans l'exemple (cf. Figure 120) au code d'identification renvoyé par la plateforme après la première communication (cf. Figure 82). Ce code est stocké par l'objet "platform" (de type "**PlatformSpecific**").
- Les seconds concerne l'objet (par exemple de type "**User**") proprement dit. Dans l'exemple (Figure 120) il s'agit des valeurs des attributs "user", "md5Password", "name", "firstname" et "email".

```
public String deploy(PlatformObject test) throws GenDepException {
    Ganesha platform = (Ganesha) test;
    GenericUserIn _Learner;
    try {
        _Learner = new GenericUserIn(platform.getAuthCode(), getUser(),
            MD5Password.getEncodedPassword(getMd5Password()),
            getName(), getFirstname(), getEmail());
        GenDepGanesha genDep = new GenDepGaneshaLocator();
        GenDepGaneshaBindingStub stub = new GenDepGaneshaBindingStub(
            new URL(platform.getAdr()), genDep);
        GenericOut lo = stub.createLearner(_Learner);
    } catch (Exception e) {
        throw new GenDepException("Create learner: FAILED, Exception:"
            + e.getMessage(), e);
    }
    return "implantation" + _Learner.getName();
}
```

Figure 120 Ecriture de la méthode "deploy" pour un objet de type "User"

A l'aide de ces attributs, il faut compléter les attributs d'un objet de type "**GenericUserIn**", cet objet est nécessaire pour invoquer le service Web "**createLearner**". "**GenericUserIn**" est la classe souche correspondant via la tâche "**wsdl2java**" au type complexe "**GenericUserIn**" défini dans le greffon de services Web.

Le service "**createLearner**" retourne un objet de type "**GenericOut**", qui dans cet exemple n'est pas utilisé mais qui nous le verrons dans la sous-section suivante permet de gérer des dépendances de construction.

La deuxième méthode "fetch"

Cette méthode permet pour un objet d'interroger la plateforme et de lui demander les instances déjà construites via les services de listage.

L'exemple, Figure 121, est issu de l'étude menée sur la plateforme Moodle, la méthode "**fetch**" de cet exemple retourne une collection d'objet de type "**UserGenDep**", cette collection d'objet est construite à partir de la collection d'objet de type "**UsersOut**" communiquée par le service Web "**getAllUsers**".

```

public Vector fetch ( PlatformObject iPlatform ) throws GenDepException
{
    Moodle _Platform = (Moodle)iPlatform;
    Vector<userGenDep> aRetourner = new Vector<userGenDep>();
    try{
        ServeurRitaine genDep = new ServeurRitaineLocator();
        ServeurRitaineBindingStub stub = new ServeurRitaineBindingStub(
            new URL(_Platform.getAdr()), genDep);
        UsersOut[] _Out = stub.getAllUsers();
        userGenDep itemp;
        for (int i=0;i<_Out.length;i++){
            itemp= new userGenDep();
            itemp.setLastname(_Out[i].getLastname());
            itemp.setCity("");
            itemp.setCountry("");
            itemp.setPassword("");
            itemp.setIdMoodle(Integer.parseInt(_Out[i].getId()));
            itemp.setEmail("");
            itemp.setUsername("");
            itemp.setFirstname(_Out[i].getFirstname());
            aRetourner.add(itemp);
        }
    }catch (Exception e) {
        throw new GenDepException("Exception:"
            + e.getMessage(), e);
    }
    return aRetourner;
}

```

Figure 121 Détail de l'implémentation de la méthode "fetch" pour la classe "userGenDep"

Cette méthode permet des communications avec la plateforme lors de la phase de contextualisation des dispositifs.

Résumé

Notre problématique en EIAH (environnements informatiques pour l'apprentissage humain) concerne la place de l'informatique quant à l'aide qu'elle peut apporter à des enseignants pour accommoder leur enseignement sur des plate-formes de formation. Cette problématique, très générale, nous a amené à nous interroger sur l'organisation sociale des enseignants que nous souhaitons adresser, sur les pratiques de création enseignante, sur les objets pédagogiques que ces pratiques nécessitent, ainsi que sur les médias accueillant ces pratiques. Nous décrivons les contours d'une ingénierie des EIAHs adaptée à des équipes d'enseignants pratiquant le bricolage pédagogique dans une perspective artisanale. Cette ingénierie repose sur une infrastructure exploitant les ressources de l'ingénierie dirigée par les modèles adaptées au domaine des EIAHs. Cette infrastructure est composée d'une plate-forme de modélisation et d'un générateur de constructeur spécifique. Elle permet, de modéliser, de contextualiser et de construire des dispositifs pédagogiques sur des plate-formes de formation ou sur de simples applications Web. Trois études de cas menées pendant cette thèse illustrent les différents aspects de notre travail, et montrent que la proposition d'ingénierie que nous présentons permet d'adresser toute la diversité des environnements informatiques pour l'apprentissage humain. Nous concluons en montrant que les concepts et solutions présentées dans cette thèse permettent de relever les enjeux posés par les récentes évolutions en EIAH, tant au niveau des approches pédagogiques, que des médias utilisés

Abstract

The pedagogical uses of e-learning platforms challenge the type of community of practice teachers build, the type of creativity they employ, the type of learning they implement, and the learning object they use. The aim of our thesis is also to offer the tools and objects that may assist teachers in their preparatory work with e-learning platforms. For the specific case of a small team of teachers using pedagogical "*bricolage*", we will show that it is possible to define an infrastructure allowing "*dispositives*" building on e-learning platforms. Our approach consist in a three step process: modeling, defining the context and building the "*dispositive*". To put this approach into practice, we propose to develop an infrastructure. This infrastructure is made of a modelling environment, a generator of specific constructor and for each e-learning platform a web service plug-in. This plug-in builds the "*dispositive*", in the context of the application, via its services. Our proposal is to easily build this infrastructure using an MDE approach. The concepts and the answers we present in our thesis are adapted to the recent evolutions of the e-learning.