

# Distributed PolyTeXML

## Une nouvelle plateforme de partage d'items didactiques

Nicolas Delestre<sup>1</sup>, Stéphane Frénot<sup>2</sup>, Stéphane Mottelet<sup>3</sup>, Michel Vayssade<sup>3</sup>

<sup>1</sup>Laboratoire PSI, INSA de Rouen, BP 08 Place Émile Blondel 76131 Mont Saint Aignan Cedex  
Nicolas.Delestre@insa-rouen.fr

<sup>2</sup>Département Télécommunications Services & Usages, INSA de Lyon, 20 Avenue Albert Einstein, Villeurbanne Cedex  
Stephane.Frenot@insa-lyon.fr

<sup>3</sup>Université de Technologie de Compiègne, BP 60319, 60205 Compiègne Cedex  
{Stephane.Mottelet, Michel.Vayssade}@utc.fr

### Résumé

La création de documents pédagogiques est un travail de longue haleine, et il l'est encore plus lorsque le support de présentation est électronique. Or aujourd'hui, à l'heure d'Internet, où le mot d'ordre est l'échange d'informations, le partage de documents pédagogiques n'en est qu'à ses premiers balbutiements. Bien que des systèmes ont été développés, ou sont en cours de développement, bien que des normes de caractérisation d'éléments pédagogiques ont été proposées, ou sont en cours de spécification, leur utilisation reste très confidentielle. Pourquoi ? C'est en voulant répondre à cette question, après une étude de l'existant, que nous avons défini une nouvelle catégorie de système, qui s'est concrétisée par le développement d'un prototype nommé "Distributed PolyTeXML".

### Abstract

Pedagogical document building is something that can not be done overnight, and specially when the destination support is the computer. But, although one main goal of Internet is the share of documents, it is not true with pedagogical documents. Few systems and few norms were, and are, specified, but they are not really used. Why ? In searching to answer to this question that we have designed a new kind of system and developed a prototype called "Distributed PolyTeXML".

### Introduction

La création de documents pédagogiques est un travail de longue haleine, et il l'est encore plus lorsque le support de présentation est électronique (par expérience, nous considérons qu'une heure de cours présenté à l'aide de transparents électroniques vidéo-projetés demande au minimum dix heures de travail). Or aujourd'hui, à l'heure d'Internet, où le mot d'ordre est l'échange d'informations, le partage de documents pédagogiques n'en est qu'à ses premiers balbutiements. Des études ont toutefois été menées, aboutissant à la définition de normes et à la création de systèmes. Toutes ces études ont en commun la création de documents pédagogiques par agencement

de ce que nous appelons grains didactiques, et que l'on retrouve dans la littérature derrière les termes d'items didactiques [Del00], de briques d'information [CR00] ou encore de briques élémentaires comme nous allons le voir par la suite dans l'étude du projet SEMUSDI [ABD99]. Cependant en étudiant les articles scientifiques publiés, en étudiant les plate-formes de téléformation du commerce, qui devrait être les premiers à mettre en pratique les résultats théoriques dans ce domaine, on remarque que la plupart du temps les prototypes restent des prototypes peu utilisés, et que les résultats ne sont pas mis en œuvre dans les logiciels commerciaux. Pourquoi ? C'est en essayant de répondre à cette question que nous avons caractérisé une nouvelle catégorie de système de création de documents pédagogiques.

Cet article va s'articuler de la façon suivante. Après une présentation des principaux systèmes qui ont été développés en France, en Europe et aux États-unis, nous essaierons de déterminer les raisons de ces échecs relatifs. Nous présenterons alors notre système, *Distributed PolyTeXML* que nous sommes en train de concevoir et qui, de par ces caractéristiques, se veut être une réelle alternative.

### État de l'art des systèmes d'enseignement basés sur l'utilisation de grains

Nous allons étudier dans cette partie cinq projets dont le leitmotiv est l'utilisation plus ou moins partagée de grains didactiques. À l'issue de cette étude, nous en tirerons quelques conclusions.

### Le projet ARIADNE

ARIADNE<sup>1</sup>, pour *Alliance of Remote Instructional Authoring and Distribution Networks for Europe*, est un projet européen, regroupant principalement la Belgique, l'Espagne, la Finlande, la France, la Grande-

<sup>1</sup><http://ariadne.unil.ch/>

Bretagne et la Suisse, et dont l'objectif est de définir des méthodologies et de produire des outils d'échange de données pédagogiques numériques pour les différentes catégories d'enseignement (classique, continu, à distance, etc.).

Les outils créés par ce consortium se veulent transversaux :

- Il y a tout d'abord ceux dont l'objectif est la création d'items didactiques, que l'on nomme les *Authoring tools*.
- Il y a ensuite des outils de caractérisation de ces items didactiques.
- Il y a enfin des outils d'administration et de création de cours à partir de ces items didactiques.

Bien que peu spectaculaire à priori, la caractérisation des items didactiques est l'un des résultats les plus importants des travaux effectués au sein de ce consortium. Il en a résulté une norme, nommée *Learning Object Metadata*, qui permet d'indexer un élément pédagogique à l'aide de huit catégories d'attributs :

1. Catégorie Générale, qui permet de spécifier les caractéristiques générales du document, tel que le nom, l'identifiant, la langue, etc.
2. Catégorie Cycle de vie, qui permet de spécifier les caractéristiques du cycle de vie, tel que le numéro de version, la date de création, la date d'expiration, etc.
3. Catégorie Méta-données, qui détermine la signification de tous ces attributs.
4. Catégorie Technique, qui regroupe l'ensemble des caractéristiques physiques du document, tel que le format, la taille, les *plug-ins* indispensables, etc.
5. Catégorie Pédagogique, qui détermine les spécificités pédagogiques du document tel que son type, son approche, ou encore sa granularité.
6. Catégorie Gestion des droits, qui détermine toutes les particularités juridiques du document.
7. Catégorie Relation, qui permet d'associer des documents entre eux.
8. Catégorie Annotation, qui permet d'ajouter des commentaires.

Ainsi chaque élément pédagogique (du plus petit, ce que l'on appelle les grains jusqu'au plus grand, comme par exemple un cours) peut être précisément référencé, c'est-à-dire aussi bien au niveau de ses caractéristiques physiques, de ses caractéristiques pédagogiques, de son état courant (gestion du cycle de vie) qu'au niveau de sa diffusion (gestion des problèmes juridiques).

Outre la caractérisation des items, c'est aussi grâce à son architecture répartie que ce projet est innovant.

En effet, de part le caractère européen de ce projet, les ressources ne sont pas centralisées, mais au contraire réparties sur plusieurs sites, nommés "centre de ressources". Ces centres sont organisés hiérarchiquement : il y a un centre de ressources principal, qui est relié à des centres de ressources régionaux, eux-mêmes reliés à des centres de ressources locaux.

## Le projet CDE

Le projet CDE, pour *Course Designer Environment*, de l'école Polytechnique Fédérale de Lausanne, s'inscrit dans le grand projet de l'EPFL nommé *ClassRoom2000*<sup>2</sup> dont le but est de développer des environnements auteurs pour la production de cours interactifs avec un haut niveau de partage et de réutilisation d'items didactiques. CDE est, comme ARIADNE, un projet transversal puisque outre la problématique de partage d'items didactiques, il propose aussi un outil nommé *CoDes Authoring Tool* pour *Course Designer Authoring Tool* permettant aux enseignants de créer, structurer et visualiser des cours interactifs [RV99].

Cet outil se distingue du système précédent par le fait qu'il ne stocke pas ces items, il les référence seulement. Cette caractérisation s'effectue à l'aide du langage XML. La DTD choisie permet de caractériser un item suivant quatre axes :

1. Les attributs physiques, tels que le type mime, la taille, etc.
2. Les attributs logiques, tels que le copyright, la version, l'auteur, etc.
3. Les attributs structurels, tels que le contexte, ou les liens.
4. Les attributs sémantiques, tels que la description, les mots-clefs, le titre, etc.

L'entité référencée avec les informations la caractérisant est nommé un "composant de connaissance" (*Knowledge Components* KC dans les textes). Ces KC sont regroupés au sein d'une base de données qui dans le cadre de ce projet est appelée *Shared Knowledge Space* ou SKS. Ces SKS peuvent être utilisés depuis :

- un navigateur Web. Un moteur de recherche a été développé pour retrouver des KC.
- CoDes pour construire de nouveaux cours.
- toutes applications qui utilisent l'API<sup>3</sup> proposée.

Enfin très peu d'information est précisée concernant la mise en œuvre éventuelle de plusieurs centres de ressources et a fortiori d'une possibilité d'échange de KC entre ces centres.

<sup>2</sup><http://www.eduswiss.ch/data/afe/old/cr2000-cd/start.htm>

<sup>3</sup>*Application Programming Interface*

## Le système OLA

Le système OLA, pour *Online Learning Application*, était, un système développé par la société Oracle, qui a été stoppé il y deux ans.

Toutefois, il est important de ne pas oublier ce système car il a permis de définir la norme RCOS, pour *Reusable Content Object Strategy*, qui a été adoptée par plusieurs distributeurs de logiciel auteur, tels que la société Macromédia, pour caractériser les documents produits.

La norme RCOS définit quatre niveaux hiérarchiques de document [HK97]. Le niveau le plus bas, nommé niveau (ou catégorie) composant, est constitué de documents monomédia, dont les caractéristiques didactiques ne sont pas spécifiées. Ensuite, il y a le niveau des items didactiques simples, nommés RCO, qui sont formés d'une succession d'éléments du niveau composant. Le troisième niveau est constitué de documents pédagogiques simples (agrégation de RCOS) tels que des cours. Et pour finir le dernier niveau est constitué de documents pédagogiques complexes (agrégation de documents pédagogiques simples) pour former par exemple des modules.

Tous les éléments de la norme RCOS possèdent des attributs communs, dont les objectifs sont l'aide à la recherche (tel que le nom, la liste de mots clefs, etc.) et l'aide à la gestion (la version, la date de création et d'expiration, etc.). Ensuite chaque catégorie de documents possède ses propres attributs. Par exemple, comme le montre le tableau 1, une hiérarchie pour la catégorie "composant" a été élaborée permettant ainsi de caractériser pleinement le document; par exemple les vidéos posséderont un attribut durée qui n'a pas lieu d'être pour les textes. De même les éléments de la catégorie RCO sont caractérisés par trois catégories d'attributs (les attributs de stratégie de sélection, les attributs de lien, et les attributs d'utilisation).

Les RCO sont renseignés et stockés dans une base de données centralisée Oracle. L'objectif du projet OLA était surtout la formation continue au sein des entreprises, la possibilité de communications inter-établissements n'est donc pas proposée.

## Le projet SEMUSDI

Ce projet de l'INSA de Rouen [ABD99, DR98] a débuté en 1996 et se termine cette année. Le serveur Semusdi<sup>4</sup> est conçu pour collecter, gérer, stocker, rechercher, consulter, diffuser et organiser (pour former des ouvrages consultables ou téléchargeables) des items didactiques ciblés, pour des utilisateurs répartis sur plusieurs sites géographiques distants. Pour insister sur l'aspect réutilisable et réorganisable de ces items, les au-

<sup>4</sup><http://semusdi.insa-rouen.fr>

<sup>5</sup><http://www.adlnet.org/>

teurs de SEMUSDI utilisent plutôt le terme de brique élémentaire. Ces briques élémentaires peuvent être de différentes formes :

- texte et hypertexte (HTML),
- images fixes ou animées,
- son,
- logiciel d'animation et de simulation,
- exercices d'application courts,
- exemples de cas concrets.

L'ensemble de ces briques est géré au sein d'une base de données accessible via le Web. En plus des fonctions utilisateurs spécifiées précédemment, des fonctions d'administration du système sont également disponibles, mais elles ne sont accessibles que par un nombre très restreint de personnes identifiées.

La fiche d'identification d'une brique élémentaire dans SEMUSDI est beaucoup moins complète que les différentes normalisations que l'on vient de voir. Tout d'abord de par son aspect francophone, il n'y pas d'information au sujet de la langue. Ensuite, la gestion des problèmes juridiques a été écartée, car on considère que les personnes qui ont accès au site peuvent faire ce qu'ils veulent des documents une fois récupérés (ils sont libres de droit). Enfin, deux versions d'une même brique correspondent à deux briques distinctes, ce qui résout les problèmes de cycle de vie.

En somme, les briques dans SEMUSDI possèdent quatre types d'informations :

- Informations générales, tel que son nom, un texte de présentation et des mots clefs,
- Informations au sujet de l'auteur, tel que son nom et prénom et ses coordonnées,
- Informations pédagogiques, tel que le thème et le type (présentation, application, etc.),
- Informations techniques, tel que le format de la brique, et s'il y a besoin, le plug-in indispensable pour la visualiser.

Les briques dans SEMUSDI sont donc moins caractérisées, mais le travail de validation et d'indexation est précis, car toute brique insérée dans le système est d'abord validée par un expert du domaine et indexée par un spécialiste avant d'être accessible aux autres membres de la communauté.

## Le projet/norme SCORM

Ce dernier système [Dod02] est le plus récent, son développement a commencé il y a deux ans et en est aujourd'hui à la version 1.2. SCORM<sup>5</sup>, pour *Sharable Content Object Reference Model* est à la fois une norme que nous allons étudier tout de suite, mais aussi un prototype écrit en Java mettant en application cette norme.

Média visuel	Texte	Mots, phrases, paragraphes, listes, tableaux
	Images fixes	Haute définition (photo, capture d'écran, etc.) Basse définition (graphique, diagramme, etc.)
	Images animées	Haute définition (vidéo) Basse définition (animation)
Média auditif	Parlé	Langue
	Son	Musique Bruit

TAB. 1 – Caractérisation des composants dans le modèle RCO

La norme SCORM, de part par la composition du comité qui l'a créé, est grandement inspirée du projet ARIADNE, tout particulièrement au niveau de la caractérisation des éléments pédagogiques. On retrouve en effet les huit catégories présentées précédemment auxquelles vient s'ajouter une neuvième que les auteurs qualifient de catégorie de classification permettant de spécifier dans quel contexte d'apprentissage on peut utiliser cet élément pédagogique. Cette norme est implantée concrètement sous la forme d'une DTD, et donc tout document qui s'y réfère est un document XML.

La norme SCORM prévoit trois grands niveaux de granularité :

1. Les *assets* qui représentent les documents de granularité de plus bas niveau, qui peuvent être par exemple une image gif, un document XML, un objet flash, une page Web, etc. Lorsque un élément de ce type est caractérisé (le document plus sa fiche XML) on parle alors de *tagged asset*,
2. Les SCO, pour *Sharable Content Object*, représentent un assemblage d'*assets* encadrés par du code Javascript. Ce code Javascript est utilisé pour référencer l'élément lors de son utilisation dans de l'environnement SCORM. Encore une fois, si un SCO est caractérisé on parle alors de *tagged SCO*.
3. Les hiérarchies de SCO forment des *tagged content aggregation*.

Alors que la caractérisation des items est des plus complètes, aucun mécanisme n'est par contre spécifié quant à leurs partages. Les *assets* sont référencés par des URL (*Uniform Resource Locator*), mais le prototype et les exemples proposés n'utilisent que des éléments locaux.

## Conclusions

Qu'en est-il de l'utilisation réel de ces systèmes ? En fait, à part Semusdi<sup>6</sup>, nous n'en savons pas grand chose, car les publications n'en font pas grand état. Nulle part on

ne trouve de statistique concernant l'évolution du nombre de grains disponibles, et surtout dans combien de cours (publiques ou privés) ils sont utilisés. Comment peut-on donc évaluer ce relatif échec ?

Une première réponse est issue du type de média utilisé : dans la plupart des cas "un grain est égal à un fichier html". Or un fichier html contient les informations à présenter mais aussi la façon de les présenter : mise en forme (couleur, taille, police, style, mise en page (organisation, couleur du fond), etc. De ce fait, agencer des grains pour former un document, revient à fusionner des fichiers html, ce qui rapidement pose de nombreux problèmes techniques (par exemple si les grains possèdent des fonctions javascript qui portent le même nom) et surtout d'uniformisation visuelle, si une charte graphique n'a pas été définie auparavant. Alors que ce type de charte peut être facilement mis en œuvre dans une entreprise, dans une école, il est difficile de l'imposer lorsque la communauté est composée d'utilisateurs très divers et répartis géographiquement.

Une seconde réponse est encore liée au fait que dans la plupart des cas "un grain est égal à un fichier html". En effet, qui dit document html, dit navigateur Web et donc un support unique de présentation de l'information. Or il se peut très bien que l'on veuille d'un côté produire un polycopié (document dont le support physique principal est la feuille de papier au format A4). Mais il se peut aussi que l'on veuille proposer sur le Web ce même polycopié (or lire un document au format A4 présuppose l'utilisation d'écrans encore peu répandus !). Enfin, il se peut aussi que l'on veuille utiliser ces grains pour des cours en présentiel (sous forme de transparents en vidéo projection). Chaque support physique de part ses caractéristiques a des avantages et des inconvénients. Par exemple, le papier est plus reposant pour la lecture, alors que l'outil informatique peut proposer des données hypermédias (liens hypertextes et/ou données multimédias). Cela oblige les auteurs à concevoir plusieurs versions d'un même grain en fonction de son utilisation future.

<sup>6</sup>Il y a deux ans, environ deux mille briques étaient référencées, mais ce nombre n'évolue que très lentement aujourd'hui (l'un des auteurs de ce papier a participé au projet Semusdi)

Enfin, tous ces systèmes proposent de centraliser tous les grains sur un seul serveur, ou alors de les stocker sur des serveurs organisés hiérarchiquement. Or notre expérience (dans le cadre du projet Semusdi) montre que ce n'est pas facile à faire admettre. En effet un enseignant, d'une façon générale, aime bien avoir le contrôle sur ce qu'il produit. Il veut bien à la rigueur déposer ses grains sur un serveur de son établissement (bien qu'il préfère les garder sur son ordinateur), mais il n'est souvent pas question qu'il les dépose sur une machine dont lui ou son établissement de tutelle n'ont pas le contrôle. Cette démarche inhibe alors rapidement toutes les bonnes intentions concernant la mise en commun de matériaux pédagogiques.

Pour essayer de résoudre ces trois problèmes, nous avons décidé de suivre une démarche basée sur les règles suivantes :

- Concevoir des grains dont le fond et la forme sont distincts, permettant ainsi de pouvoir les fusionner facilement et de pouvoir les “projeter” sur des supports de communication divers : c'est le rôle des grains PolyTeXML,
- Laisser les grains sur les machines des enseignants, ne pas centraliser les données : c'est le rôle des paquetages PolyTeXML et de l'utilisation de la technologie de communication pair à pair.

### Les grains PolyTeXML

Le produit PolyTeXML est une évolution du produit Polytex<sup>7</sup>. Polytex, projet initié par [BC98], est un ensemble de macros et de classes L<sup>A</sup>T<sub>E</sub>X permettant de construire des photocopiés de cours structurés. Les photocopiés sont structurés en trois niveaux : les chapitres, les sections, les grains. Un grain Polytex est typiquement une partie de texte de 1 à 3 pages, éventuellement accompagné d'exercices, de documents et de transparents destinés à la vidéoprojection. Par exemple les figures 1 et 2 présentent un même document (en l'occurrence la documentation de PolyTex) qui peut être :

1. visualisé sur un écran (figure 1). La police de caractère utilisée est plus grosse, des liens hypertextes sont présents et le texte est découpé en plusieurs pages écrans accessibles via les liens hypertextes “suivant” et “précédent”.
2. imprimé (figure 2). La police de caractères est de taille standard, les liens hypertextes ont été remplacés par des références.

Afin de faciliter le rôle des concepteurs de grain (tout le monde n'est pas censé connaître le langage L<sup>A</sup>T<sub>E</sub>X), et de les empêcher d'insérer des commandes de mise en forme dans un grain (ce qui est tout de même possible avec Polytex), nous avons décidé d'utiliser XML comme

langage de description de grain. Lors de la visualisation, un processeur XSL transforme ce code XML en code PolyTeX, qui peut être compilé pour produire finalement un fichier au format PDF (format adapté à l'impression et à la visualisation sur écran par le caractère vectoriel de ces données et part ses capacités de navigation hypertexte). Durant cette compilation, l'utilisateur spécifie ce qu'il veut visualiser, un document ou des transparents, et dans le cas où c'est un document, quel sera le support utilisé. A la fin, l'utilisateur aura toujours un document PDF dont le contenu sera adapté à l'utilisation qu'il veut en faire.

Dans PolyTeXML nous avons deux catégories de grain. Tout d'abord les grains minimaux qui correspondent à la présentation d'une notion, d'un exercice, de quelques transparents. Puis les méta-grains qui rassemblent un ensemble de grains en les structurant, en chapitre et/ou section et/ou sous-section etc. Les grains peuvent être inclus directement dans les méta-grains, le tout forme alors un seul fichier. Mais ils peuvent aussi être externes aux méta-grains, afin par exemple d'être réutilisés dans différents cours.

### Les paquetages, où la caractérisation des liens

Pour qu'un grain puisse être partagé par plusieurs enseignants, il faut qu'il puisse être référencé par le système, permettant ainsi aux utilisateurs de pouvoir le retrouver aisément : ceci est le rôle du paquetage.

### Caractéristiques d'un paquetage

Un paquetage est un composant constitué :

- d'un fichier de caractérisations,
- de l'ensemble des fichiers qui constituent l'élément pédagogique (grain ou méta-grain PolyTeXML) proposé.

Le fichier de caractérisation est un fichier XML, dont la DTD oblige l'auteur à préciser les informations suivantes :

- les informations concernant les auteurs, tels que :
  - les nom, prénom et coordonnées du ou des auteurs,
  - les droits de reproduction (libre de droit, licence GPL, licence LGPL, réutilisable sans modification, interdite, etc.),
  - une signature permettant de savoir si un paquetage est conforme à l'origine,
- les informations concernant l'indexation, tels que le titre, un résumé, une liste de mots clefs,
- les informations pédagogiques du paquetage, pour dire par exemple que c'est la présentation d'une notion, que c'est déjà un morceau de cours (regroupement de plusieurs notions) ou que c'est déjà un

<sup>7</sup><http://www.dma.utc.fr/polytex/>

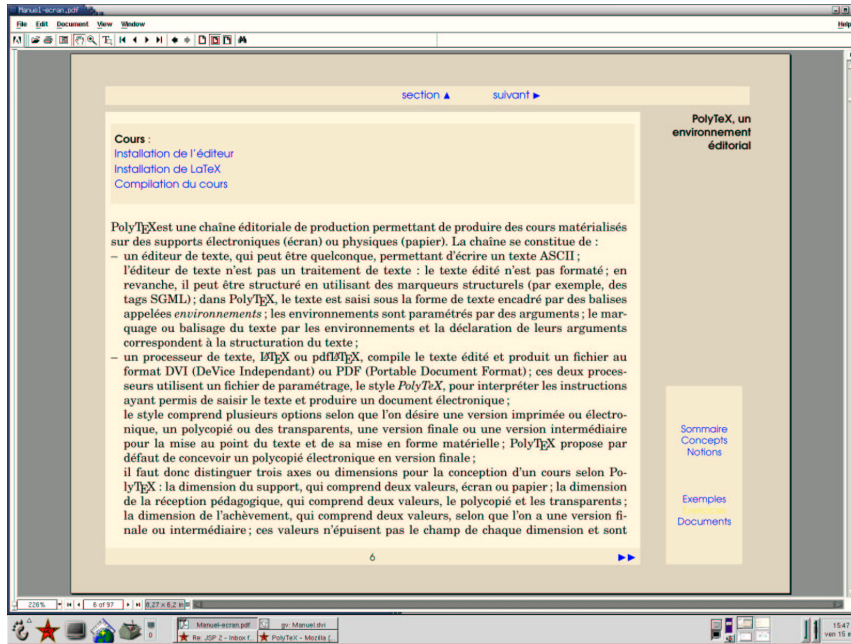


FIG. 1 – Un document PolyTex qui peut être visualisé à l'écran



FIG. 2 – Un document PolyTex qui peut être imprimé

cours.

- les informations concernant le “passé” du paquetage pour indiquer que ce paquetage est une évolution d’un autre paquetage (changement de version, copie intégrale d’un autre paquetage ou produit à partir d’un extrait d’un autre paquetage),
- le point d’entrée dans le paquetage : le document maître lorsqu’il y a plusieurs fichiers PolyTeXML au sein d’un même paquetage.

Afin d’identifier un paquetage, son nom est unique au monde. Il est constitué d’un nom générique présentant succinctement le contenu du paquetage, suivi du nom de l’auteur, du nom de la machine sur laquelle le paquetage a été créé, et du numéro de version.

Enfin, un paquetage est un élément réutilisable : il peut être référencé par un autre paquetage.

### Classification des paquetages

Afin d’aider la recherche de paquetage, les paquetages sont classés suivant une taxinomie mi-contrainte. Le haut de la classification est le même pour tous les utilisateurs du système. Cette première classification est normalisée puisque nous reprenons la classification proposée dans la norme Dewey<sup>8</sup>. Malheureusement, cette classification est trop générique. Par exemple, dans le domaine des applications client-serveur et de l’informatique répartie<sup>9</sup>, ce classement nous propose la hiérarchie présentée dans la figure 3.

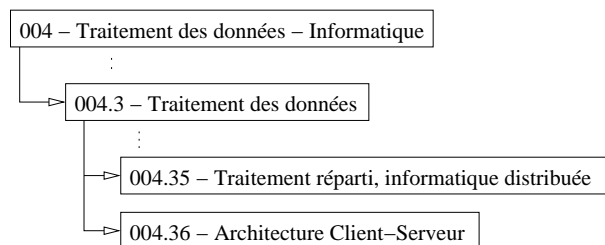


FIG. 3 – Un extrait de la taxinomie Dewey

Cette dernière n’étant pas assez détaillée, nous permettons donc à chaque utilisateur de l’étendre comme bon lui semble. Nous conseillons toutefois pour chaque thématique de former des cercles de réflexion afin d’obtenir une taxinomie consensuelle comme nous l’avons fait pour le domaine de l’informatique répartie. Ainsi un utilisateur pourrait étendre la taxinomie présentée par la figure 3 en une taxinomie présentée par la figure 4.

<sup>8</sup><http://www.oclc.org/dewey/>

<sup>9</sup>Trois des quatre auteurs de cet article enseignent l’informatique répartie.

<sup>10</sup><http://www.jxta.org/>

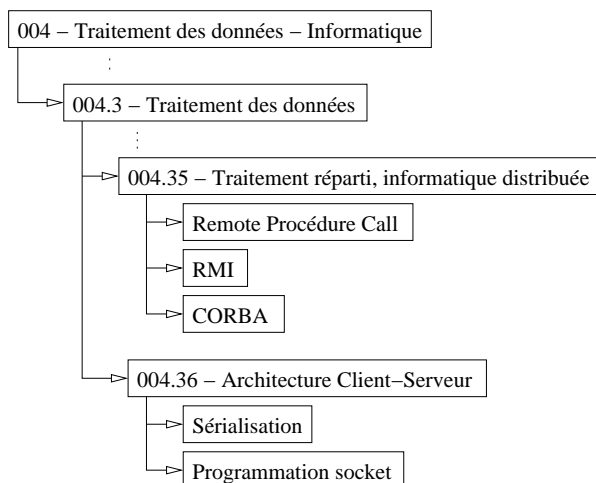


FIG. 4 – Un exemple d’extension de la taxinomie de Dewey

### Répartition et échange des paquetages

À la différence des autres systèmes que nous avons vu précédemment, nous avons opté pour un stockage et un référencement totalement décentralisé des paquetages. Pour ce faire nous nous inspirons des logiciels d’échange de données pair à pair comme FreeNet [CSWH00], en utilisant l’API de référence JXTA<sup>10</sup> définie par Sun[On].

Chaque utilisateur/auteur référence dans son logiciel :

- un “cercle d’amis” (appelé communauté) auprès de qui il peut effectuer des requêtes pour trouver de nouveaux paquetages,
- sa taxinomie et ses paquetages qu’il propose à la communauté.

Lorsque l’auteur recherche un nouveau paquetage pour concevoir un document, il envoie une requête à la communauté en précisant des mots clefs et/ou une branche taxinomique. Les paquetages sélectionnés devront alors :

- posséder une partie ou tous les mots clefs,
- et/ou
- appartenir à la branche, ou à une sous-branche de la branche taxinomique donnée.

Le programme de chaque adhérent recherche alors les paquetages valables et les retourne aux demandeurs. Mais en plus, il répercute cette même requête auprès des autres adhérents répertoriés dans son “carnet d’adresses”. Bien évidemment, afin d’éviter les effets de bouclages, chaque requête possède un identifiant unique, et elle n’est prise en compte qu’à partir du moment où elle n’a jamais été reçue. De même, afin de limiter la durée de vie de

sa requête, l'émetteur a la possibilité de lui donner une espérance de vie fixe (de quelques secondes à quelques minutes) et/ou un nombre de "rebonds" limité.

L'émetteur voit donc arriver des propositions. Après visualisation, si un paquetage l'intéresse, il a la possibilité :

- de le référencer, un lien est donc créé depuis sa taxinomie vers le paquetage choisi,
- de le copier, un nouveau paquetage est créé au sein de sa taxinomie,
- de copier un élément du paquetage, ce qui comme nous l'avons vu, crée un nouveau paquetage au sein de sa taxinomie avec le fichier désiré ainsi que ses dépendances.

Cette architecture répartie présente plusieurs avantages dont :

- l'absence de maintenance centralisée, chaque enseignant garde ses paquetages sur son ordinateur,
- la dynamique induit par les utilisateurs (un enseignant digne de ce nom se doit de régulièrement mettre à jour son cours et donc d'en faire profiter les membres de la communauté),
- un échange entre enseignants est proposé au niveau des paquetages mais aussi au niveau de leur taxinomie.

### Conclusion

Après avoir étudié cinq projets de recherche sur la création de documents pédagogiques par agencement d'items didactiques (étude qui a porté aussi bien sur la caractérisation des éléments à partager que sur la manière de les partager), nous avons proposé une nouvelle façon de concevoir et partager des documents pédagogiques. Cette nouvelle approche repose principalement sur deux concepts :

- la distinction du fond et de la forme (grâce à l'utilisation de l'outil PolyTeXML),
- l'utilisation des technologies de communication pair à pair en remplacement de la technologie standard client-serveur.

Actuellement, le développement de PolyTeXML est pratiquement fini. Son utilisation est en cours validation via la création commune (UTC et INSA de Rouen) d'un cours sur les EJB<sup>11</sup>. L'outil de partage de paquetages PolyTeXML quant à lui est en phase de développement à l'INSA de Lyon.

### Références

- [ABD99] W. Aziz, S. Barthélémy, and N. Delestre. How to share multimedia data about engineering sciences? In *CAEE'99*, pages 72–78, 1999.
- [BC98] B. Bachimont and J. Charlet. Polytex : un environnement pour l'édition structurée de polycopiés électroniques multisupports. In *EuroTeX'98*, pages 1–16, Saint Malo, France, 1998.
- [CR00] S. Chabert-Ranwez. *Composition automatique de Documents Hypermédia Adaptatifs à partir d'Ontologies et de Requêtes Intentionnelles de l'Utilisateur*. Thèse, Université Montpellier II, 2000.
- [CSWH00] Ian Clarke, Oskar Sandberg, Brandon Wiley, and Theodore W. Hong. Freenet : A distributed anonymous information storage and retrieval system. In *Workshop on Design Issues in Anonymity and Unobservability*, pages 46–66, 2000.
- [Del00] N. Delestre. *METADYNE, un Hypermédia Adaptatif DYnamique pour l'Enseignement*. Thèse, Université de Rouen, Rouen, France, Janvier 2000.
- [Dod02] P. Dodds. Scorm, sharable content object reference model. Technical report, Advanced Distributed Learning, 2002.
- [DR98] N. Delestre and B. Rumpler. Architecture d'un serveur multimédia pour les sciences de l'ingénieur. In *NTICF'98*, pages 383–391, 1998.
- [HK97] J. Hathaway and S. Kirshbaum. The reusable content object strategy. Technical report, Oracle, 1997.
- [On] Ra Ti On. Project jxta : An open, innovative collaboration.
- [RV99] Y.A. Rekik and C. Vanoirbeek. Course designer environment : Authoring environment for interactive courses with a shared course component library. In *CAEE'99*, pages 296–303, 1999.

---

<sup>11</sup>Enterprise Java Beans