
Modèles et interfaces d'hyperlivres pour l'apprentissage collaboratif

Gilles Falquet, Jacques Guyot, Jean-Pierre Hurni

Centre universitaire d'informatique, Université de Genève
24, rue Général-Dufour
CH-1211 Genève 4 – Suisse
(Gilles.Falquet|Jean-Pierre.Hurni|Jacques.Guyot)@cui.unige.ch

RÉSUMÉ. Après avoir mené des expériences d'écriture collaborative d'hypertextes avec les technologies ordinaires du Web, nous avons constaté que celles-ci sont encore insuffisantes à bien des égards. C'est pourquoi nous avons développé et implémenté un modèle d'hyperlivre qui étend les possibilités du Web. Ce modèle peut être vu comme une réalisation de la notion de document virtuel. Le contenu informationnel de l'hyperlivre est formé d'un ensemble de fragments d'informations et de liens. L'interface de lecture et d'écriture est formée de documents XML ou HTML générés à partir des fragments et des liens par un mécanisme de sélection et d'assemblage. La technique des vues hypertextuelles sur les bases de données permet de spécifier l'interface de manière purement déclarative et par conséquent de l'adapter ou de l'étendre sans recourir à la programmation. Les fragments qui décrivent les concepts du domaine étudié forment le niveau terminologique de l'hyperlivre. Ce niveau terminologique sert non seulement à décrire le domaine mais également à établir des liens sémantiques entre fragments et à générer des éléments d'interface.

ABSTRACT. Collaborative writing experiments with standard Web technologies showed us that they are still unsatisfactory in several respects. Thus, we developed and implemented a hyperbook model that goes beyond current Web functionalities. This model can be seen as a realization of the virtual document idea. The information content of the hyperbook is a set of information fragments and links, while the reading and writing interface is a set of XML or HTML documents generated from fragments and links through a selection and assembly mechanism. The hypertext technique enables a book designer to specify the interface in a purely declarative way, thus no programming is required to adapt and extend the interface. Fragments that describe domain concepts make up the terminological level of the hyperbook. This terminological information does not only describe the domain but is also serves to infer semantic links between fragments and to generate interface components.

MOTS-CLÉS : hyperlivre, hypertexte, XML, travail collaboratif, terminologie, document virtuel, interface.

KEYWORDS: hyperbook, hypertext, XML, collaborative work, terminology, virtual document, interface.

1. Introduction

La diffusion rapide du Web en tant qu'hypertexte mondial a suscité un grand intérêt dans le domaine pédagogique. De même, le Web a donné un second souffle à des recherches dans le domaine des hypertextes, des documents électroniques et des systèmes de travail collaboratif. Mais si le Web s'est rapidement imposé comme un outil efficace de diffusion d'informations et de connaissances, les techniques de production collaborative de contenus restent rudimentaires. Les outils de création de ressources (documents HTML, XML ou autre) et de sites sont encore loin de ce qu'on pourrait attendre d'un environnement interactif et collaboratif de production d'hypertextes. La création collaborative via Internet se base souvent sur l'échange de courriels ou le partage d'un espace commun de stockage de documents (site ftp ou site Web).

Les travaux de recherche que nous menons actuellement visent précisément à créer de véritables environnements d'écriture collaborative sur le Web. Ceci avec deux applications principales en vue : l'apprentissage par l'écriture collaborative de notes de cours et l'écriture collaborative d'hyperlivres scientifiques.

1.1 *Le contexte pédagogique*

Depuis plus de trois ans, nous expérimentons une approche pédagogique basée sur l'écriture collaborative d'hyperlivres scientifiques. L'écriture est doublement collaborative. Tout d'abord les enseignants et les étudiants collaborent à la création du même hyperlivre ; ensuite les étudiants collaborent entre eux. Nous avons expérimenté différents moyens techniques, comme nous le verrons plus loin, mais le principe général reste le même :

1. le ou les enseignants fournissent un contenu de base (notes de cours, exercices, définitions des concepts, etc.).
2. les étudiants ajoutent du contenu (exemples, notes de lecture, exercices, corrigés d'exercices, etc.) qui vient s'ancrer sur le contenu de base.
3. à la fin du cours (un semestre ou une année) l'hyperlivre ainsi produit est figé, distribué aux participants, puis supprimé ou archivé.

Ce dernier point est important et distingue cette démarche des approches cumulatives qui consistent, par exemple, à enrichir un site Web année après année. Ici, l'hyperlivre correspond à la mise en commun des notes de cours des enseignants et étudiants et n'a pas pour objectif de devenir un ouvrage de référence. Il n'a de valeur que pour ceux qui ont participé au cours. Il peut bien sûr être intéressant de consulter les hyperlivres des années précédentes mais là n'est pas l'objectif.

L'un des objectifs pédagogiques poursuivis, en plus de la compréhension des différentes notions abordées, est la mise en relation de ces notions. Notre pratique de

l'enseignement nous a en effet montré que les étudiants ont en général de la peine à replacer ces notions dans le contexte général du cours et à voir les liens qui existent entre elles. Ceci est probablement dû à la grande distance qui peut séparer la présentation de deux notions, que ce soit dans le temps ou dans le texte du cours. L'idée du lien hypertextuel nous a paru extrêmement importante car elle permet précisément de relier explicitement ce qui semble éloigné.

Un objectif secondaire est l'apprentissage de l'écriture collaborative d'hypertextes. En effet, avec le développement du Web, l'hypertexte est devenu un mode de communication extrêmement répandu qui vient s'ajouter à l'écrit traditionnel. Bien qu'il existe désormais de nombreux ouvrages traitant de la conception des sites Web, bien peu d'entre eux traitent de la création de contenus. Ils se concentrent en général sur la présentation de contenus supposés préexistants. En fait, le Web n'est pas encore considéré comme un environnement d'écriture mais plutôt comme un outil de publication.

1.2 *Le contexte technique*

Lors d'une première expérience d'écriture collaborative de notes de cours électroniques [Falquet *et al.*, 1998] nous avons utilisé les technologies de base du Web. Le noyau de l'hypertexte a été fourni par les enseignants. Il s'agissait de notes de cours, mises sous forme HTML ou PDF, et de divers documents de référence (articles, documentations). Chaque groupe d'étudiants (de deux à quatre personnes) était chargé de produire un ensemble de pages HTML correspondant à une vision « transversale » du cours, par exemple : les algorithmes ; les exercices ; un recueil de questions-réponses ; des textes de vulgarisation ; des exemples ; un index ; un glossaire ; une carte conceptuelle de la matière. De plus, chaque groupe était chargé de créer des liens hypertextes vers les documents de base (notes de cours) et les documents des autres groupes.

Aucun outil particulier de création de pages ou de site n'a été fourni, si ce n'est un modèle de page HTML (facultatif) et un modèle d'organisation en répertoires et sous-répertoires.

1.2.1 *Observations*

L'écriture de pages HTML isolées ne pose pas de problèmes particuliers aux étudiants. Certains ont produit de très bons contenus. Par contre l'établissement de liens hypertextuels s'avère beaucoup plus problématique. En effet, s'il est relativement aisé de créer des liens entre ses propres pages, plusieurs obstacles se présentent dès qu'il s'agit de faire référence à des pages créées et gérées par d'autres :

- la localisation des pages est instable : l'auteur d'une page HTML peut à tout moment décider de la déplacer dans la hiérarchie des fichiers, de la renommer, de

l'éclater en plusieurs pages ou de la regrouper avec d'autres. Il s'ensuit que l'URL de la page peut changer au cours du temps et rendre invalide toutes les références vers elle.

– il n'est pas possible de faire référence à un point précis d'une page sans créer à cet endroit une ancre, opération que seul l'auteur a le droit d'effectuer.

– il est impossible d'établir un lien depuis une page dont on n'est pas l'auteur.

L'ensemble de ces obstacles a fait que le livre électronique (le site) produit possédait essentiellement deux types de liens : les liens vers le noyau de base stable fourni par les enseignants (avec le problème de la référence à des points précis d'un long document) et les liens entre les pages d'un même groupe. Par exemple, des liens aussi évidents que ceux reliant un exercice à une entrée de la terminologie ont été très rarement établis.

1.2.2 *Concepts et technologies autour de l'écriture collaborative et hypertextuelle*

L'approche que nous proposons pour obtenir un outil d'écriture collaborative efficace repose sur des concepts et techniques bien connus et datant pour la plupart d'avant le Web, en particulier en ce qui concerne les hypertextes et la gestion des documents.

Il existe à l'heure actuelle différents outils pour le travail collaboratif sur le Web, dont certains ont été directement utilisés dans des buts pédagogiques, par exemple BSCW [Appelt *et al.*, 1999], ou adaptés à une utilisation pédagogique, par exemple Learning Space construit sur la base de Lotus Notes. Cependant, ces outils fonctionnent plus comme des lieux de stockage de documents et de coordination entre acteurs que comme des hypertextes.

Dans le domaine des hypertextes, on constate que la plupart des systèmes et modèles développés avant le Web possèdent une richesse structurelle ou fonctionnelle qui ont été abandonnées au profit de la simplicité et de la non centralisation. La recherche sur les hypertextes a abordé plusieurs domaines qui nous intéressent directement. Des systèmes tels qu'Intermedia [Garret *et al.*, 1986] ou Storyspace [Bernstein, 2002] ont été développés essentiellement dans le but de produire de la littérature hypertextuelle, d'autres tels que KMS [Ackscyn *et al.*, 1988] ou MacWeb [Nanard *et al.*, 1993] ont pour but la gestion et le partage de connaissances. Les notions de lien, d'ancrage, de composition de nœuds, etc. ont été étudiées en détail, ce qui a conduit, entre autres, à la définition du modèle de référence Dexter [Halasz *et al.*, 1994].

Différents modèles et systèmes ont également été proposés pour intégrer sous forme d'hyperlivres la notion de livre et de publication électronique. Il peut s'agir de transformer un livre papier en hypertexte [Rada, 1990] ou en livre électronique [Landoni *et al.*, 2000] ou d'écrire directement sous forme électronique (hypertextuelle) [Fröhlich *et al.*, 1997] ou encore d'intégrer des documents électroniques existants [Brusilivsky *et al.*, 2002].

Le problème de la personnalisation des hypertextes a conduit à la définition de modèles et de techniques d'adaptation et d'adaptativité. [De Bra *et al.*, 1997] [Brusilovsky, 1998]. La capacité d'adaptation correspond à la présentation de contenus différents ou différemment organisés, en fonction d'un profil de l'utilisateur. L'adaptativité consiste à mettre à jour automatiquement le profil de l'utilisateur en fonction de son comportement. Un exemple bien connu d'adaptativité est le changement de couleurs des liens menant à des pages Web déjà visitées. Dans [Wu *et al.*, 2001] les auteurs proposent un modèle d'hypertexte adaptatif qui comprend un modèle du domaine, un modèle de l'utilisateur et des règles d'adaptation. Le modèle du domaine est un réseau sémantique formé des concepts du domaine et de relations entre ces concepts. Ce modèle sert essentiellement à définir des règles d'adaptation, en fonction, par exemple, des concepts connus (ou maîtrisés) de l'utilisateur.

Parallèlement, de nombreux travaux ont été menés depuis une vingtaine d'années dans le domaine des documents électroniques. Ils ont conduit en particulier à la distinction entre la structuration logique des documents, exprimée dans un langage tel que SGML ou XML, et la restitution des documents sur écran, sur papier, ou autre, exprimée dans un langage de définition de styles tel que CSS ou dans un langage de transformation comme XSLT. Des applications de XML ont également été développées pour structurer des documents dans des domaines d'application particuliers : MathML (formules mathématiques), SVG (graphisme vectoriel), SMIL (contenus multimédias synchronisés), X3D (scènes de réalité virtuelle 3D), etc. La plupart de ces standards sont créés et gérés par le W3 Consortium (www.w3c.org).

Un courant de recherche se dégage autour de la notion de document virtuel personnalisable [Crampes *et al.*, 1999 ; Ranwez *et al.*, 1999 ; Garlati *et al.*, 2002]. Les documents virtuels personnalisables sont définis comme des ensembles d'éléments (souvent appelés fragments) associé à des mécanismes de filtrage, d'organisation et d'assemblage. En fonction du profil de l'utilisateur ou de ses intentions, ces mécanismes vont produire différents documents adaptés à ses besoins. Le modèle que nous présentons s'inscrit dans cette approche, tout en utilisant la notion de publication hypertextuelle de bases de données.

Si l'on admet que les fragments constituant un document virtuel sont stockés dans une base de données, il faut obligatoirement passer par un système de publication de bases de données sur le Web pour instancier les documents et liens visibles par l'utilisateur. Dès les débuts du Web, différentes techniques ont été mises en œuvre pour produire des documents Web (en général des pages HTML) à partir de contenus de bases de données [Nguyen *et al.*, 1996]. À l'heure actuelle on distingue :

- les techniques procédurales, basées sur un langage de programmation (Java, Visual Basic, C) ou de « script » (PHP, Perl) associé au langage de requête SQL et produisant du code HTML ou autre.

– les techniques déclaratives, dans lesquelles on utilise un langage non procédural de spécification des contenus de pages [Atzeni *et al.*, 1997 ; Fernandez *et al.*, 1998 ; Falquet *et al.*, 1999b].

C'est cette dernière approche que nous utiliserons, car elle correspond bien au principe des documents virtuels. Elle permet en effet de spécifier (plutôt que programmer) la sélection, le filtrage et l'assemblage de fragments et de liens contenus dans une base de données.

2. Un modèle d'hyperlivre orienté documents virtuels

Le modèle d'hyperlivre que nous présentons ci-dessous a pour but de répondre aux besoins mis en évidence lors de nos expériences avec les technologies du Web. Ces besoins concernent différents aspects du modèle :

Documents. Il s'agit d'assurer la stabilité de l'identification des documents ou fragments de documents, indépendamment de leur localisation. Un document doit être un objet qui possède une identité fixe tout au long de son existence. Il est également important de pouvoir classer les documents, dans le but de faciliter la recherche et la réutilisation des documents. La composition du contenu d'un document doit pouvoir se faire sans se préoccuper de la présentation finale à l'écran ou de l'aspect navigationnel (contrairement aux pages HTML qui requièrent l'inclusion explicite d'ancres dans le texte).

Liens. Les liens entre documents doivent exister indépendamment du contenu des documents et de leurs caractéristiques. En particulier, un utilisateur doit pouvoir créer des liens entre documents dont il n'est pas l'auteur. Le lien devient un objet situé « hors » des documents. En ce qui concerne l'écriture proprement dite, il doit également exister des outils d'aide à la création de liens, que ce soit au niveau du choix des documents les plus pertinents à lier (aspect sémantique) ou au niveau de la facilité de création des liens.

Interface de lecture et d'écriture. La lecture efficace d'hyperdocuments nécessite un important travail narratif et rhétorique de la part de l'auteur. Ce travail peut se matérialiser sous la forme de liens narratifs entre les documents, ou sous forme de documents composés par assemblage de noeuds de l'hypertexte ou encore sous forme d'aides à la navigation (index, visites guidées, tables des matières, cartes conceptuelles, etc.). On ne peut pas non plus dissocier entièrement lecture et écriture, la seconde implique toujours la première. Mais l'inverse est aussi vrai, la lecture « active » met en général en jeu des techniques telles que l'annotation, le surlignage, le résumé, etc. Ceci implique que l'interface de lecture/écriture offre différentes vues, adaptées aux activités de l'utilisateur.

Terminologie. D'un point de vue sémantique, nous avons remarqué que l'aspect terminologique (termes, définitions, hiérarchie de concepts) est très important. La constitution d'un « niveau terminologique », en plus de son aspect pédagogique

(recueil des définitions des principaux concepts du cours), peut grandement aider à décrire les documents et à établir des liens pertinents entre eux.

Dans le modèle d'hyperlivre que nous proposons il y a une séparation nette entre la partie structurale (représentation de l'information) et l'interface de lecture et d'écriture de l'hyperlivre.

La partie structurale du modèle est composée de fragments de documents structurés et de liens typés entre ces fragments. Étant donné que nous ne considérons que des hyperlivres couvrant un domaine restreint (par exemple la matière d'un cours), nous appellerons « monographie » cet ensemble qui forme le contenu informationnel de l'hyperlivre.

Le modèle d'interface utilisateur se base sur la notion de vues hypertextuelles. Il se compose d'une spécification des vues qui produit des documents XML et des liens hypertextuels à partir de la monographie. Suivant l'idée des documents virtuels, les documents et liens qui apparaissent dans l'interface de lecture/écriture ne sont donc pas forcément ceux de la monographie, ils sont créés par sélection, filtrage et assemblage des fragments et liens de la monographie. On s'éloigne ici du modèle Web habituel où chaque document stocké sur un serveur est envoyé tel quel à l'utilisateur.

La figure ci-dessous montre l'articulation entre les éléments du modèle.

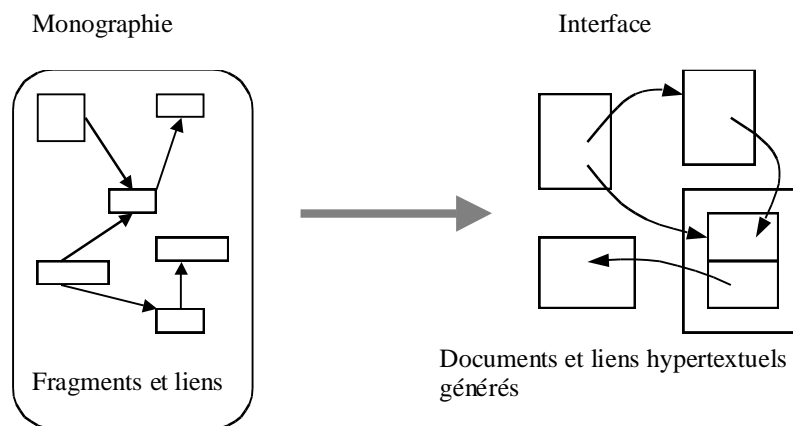


Figure 1. *Éléments du modèle d'hyperlivre*

3. Le modèle de monographie

Tout le contenu informationnel de l'hyperlivre se trouve dans les fragments d'information et les liens de la monographie.

3.1 Structure des fragments

Un fragment possède un identificateur unique, invariable au cours du temps, un titre (qui peut changer), une catégorie et un contenu. Le contenu d'un fragment est un document structuré hiérarchiquement, de type XML. Suivant le modèle « XML Query Data Model » [Fernandez *et al.*, 2002] du W3 Consortium, un document est un ensemble de noeuds de différents types (élément, attribut, valeur, instruction, texte, espace de nom, etc.). Les noeuds éléments sont munis d'une relation enfant qui définit un arbre ; ils peuvent être liés à des noeuds attributs qui eux-mêmes sont liés à des valeurs. Ainsi, le texte XML d'un document n'est en fait qu'une représentation linéaire d'un arbre dont les feuilles sont des textes ou des valeurs d'attributs. La Figure 2. montre un texte XML et son interprétation en tant qu'arbre.

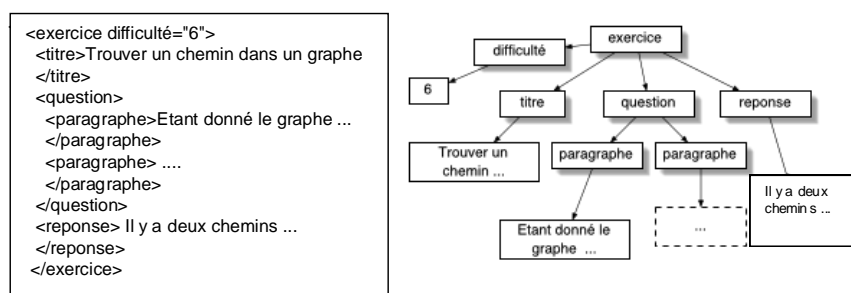


Figure 2. Texte XML et arbre correspondant

Suivant le degré de précision souhaité et les traitements à effectuer sur les fragments, on pourra soit stocker le texte XML en bloc, soit représenter explicitement chaque élément par un objet de la base de données.

Si dans un document réel la nature d'un fragment de texte est souvent déterminée par sa position, il n'en va pas de même dans un document virtuel. C'est pourquoi chaque fragment de la monographie doit être catégorisé explicitement. Par exemple, pour la création collective de notes de cours dans le domaine des structures de données, nous avons identifié les catégories : concept, algorithme, exercice, vulgarisation, carte conceptuelle, index et question fréquente.

Les fragments de la catégorie concept forment le niveau terminologique que nous détaillerons plus loin.

3.2 Liens entre fragments

Un lien établit une relation typée entre un fragment source et un fragment destination. Le type précise la signification de la relation. On pourra définir, par exemple, des relations de structuration des documents, de type chronologie ou causalité [Auffret, 1999] ou des relations sémantiques du type instance-de, spécifique/générique, propriété, partie-de, etc. ou encore des relations argumentatives : question, position, argument, contradiction, etc. Pour indiquer sur quelle partie du fragment porte la relation, celle-ci possède une ancre de départ et une ancre d'arrivée qui sont des éléments du contenu des fragments liés. Une relation peut être définie soit en liant explicitement des fragments, soit par un calcul sur les contenus des fragments ou sur les relations (par exemple en utilisant des techniques de recherche d'information ou d'inférence logique).

3.3 Conception des fragments et liens

L'orientation document virtuel de l'hyperlivre implique que les fragments et liens de la monographie soient écrits dans l'idée qu'ils vont servir à composer les documents qui seront vus par les utilisateurs. Chaque fragment doit donc former un tout cohérent, lisible pour lui-même et réutilisable. Ce sera par exemple, un énoncé d'exercice, un théorème, une hypothèse, un fait historique, une définition. Le genre de contenu du fragment sera indiqué par sa catégorie. Pour que le fragment soit réellement indépendant, la catégorisation doit réellement refléter la nature du contenu et non pas son rôle par rapport à un autre fragment. Ainsi, classer un fragment comme exemple n'est pas satisfaisant car ce même fragment pourrait servir également de contre-exemple par rapport à un autre fragment. Le rôle joué par un fragment sera indiqué par le type du lien qui le relie au fragment concerné. Cette distinction entre catégorie (propriété intrinsèque) et rôle (par rapport à un autre fragment) donne une première piste pour la conception de la monographie. Cependant elle n'est pas suffisante car, comme nous l'avons constaté, les auteurs ont tendance à créer un grand nombre de catégories et de types de liens. On assiste également à une prolifération de liens dont la pertinence est faible. Pour résoudre ces problèmes, nous préconisons de recourir au niveau terminologique.

4. Niveau terminologique

Les ouvrages scientifiques, et en particulier les notes de cours, s'appuient sur un ensemble de concepts du domaine étudié. Il apparaît donc naturel de réserver une place particulière aux fragments de la monographie qui définissent l'environnement conceptuel de celle-ci. Il peut s'agir soit de rappeler au lecteur des définitions de concepts nécessaires à la compréhension de la monographie, soit de définir de nouveaux concepts qui forment le coeur de la matière étudiée.

Un fragment de type *concept* est appelé fragment terminologique, son nom est un terme qui désigne le concept et son contenu est une définition du concept. La définition elle-même peut être soit sous forme d'un texte en langue naturelle, soit exprimée dans un langage formel.

4.1 Description (ontologie) du domaine

Le rôle premier du niveau terminologique est d'offrir au lecteur des définitions précises des termes employés dans le domaine de la monographie. Comme nous l'avons indiqué précédemment, les étudiants ont en général de la peine à voir les relations existant entre les différents concepts présentés dans un cours. C'est pourquoi il est nécessaire de créer des liens sémantiques entre les fragments terminologiques. On distingue en général deux sortes de liens :

– les liens taxonomiques, de type générique/spécifique, placent les concepts dans une « hétéarchie »¹ et permettent de les ordonner (partiellement) des plus généraux aux plus spécifiques.

– les liens non-taxonomiques indiquent plutôt des propriétés des concepts. Parmi ceux-ci, les liens méronymiques (partie de) montrent la composition de certains concepts. Par exemple, *mémoire* –partie de→ *ordinateur*. D'autres liens, appelés « rôles » en logique descriptive, indiquent le rôle qu'un concept joue par rapport à un autre. Par exemple, *automate à états fini* –accepte→ *langage régulier* (un automate à états fini a pour propriété d'accepter un langage régulier).

L'ensemble des concepts et des liens qui les relient forme une conceptualisation du domaine de la monographie qui est commune à tous les utilisateurs, c'est-à-dire une ontologie du domaine. En plus de son rôle purement descriptif, l'ontologie va aider à la création de liens pertinents entre fragments et à l'automatisation de certaines tâches.

4.2 Liens entre fragments terminologiques et autres fragments

Les fragments terminologiques jouent à la fois le rôle d'information et de méta-information, puisqu'on peut les consulter pour eux-mêmes ou s'en servir pour décrire le contenu d'un autre fragment. Le typage des relations entre fragments terminologiques et autres fragments permet de qualifier précisément ces derniers. Les types de liens que l'on rencontre fréquemment sont, par exemple :

- *instance de, exemple, illustration* : le fragment décrit une instance du concept ;
- *propriété* : le fragment décrit une propriété du concept ;

¹ Dans une hétéarchie, contrairement à une hiérarchie, un élément peut posséder plusieurs éléments qui lui sont directement supérieurs. Une hétéarchie doit former un graphe orienté sans circuit.

– *note, observation, commentaire, précision* : le fragment décrit plus en détail un aspect du concept ;

– *référence, utilisation* : le fragment fait référence à ce concept (il faut avoir compris le concept pour pouvoir lire le fragment).

La méthode que nous proposons pour résoudre le problème de la prolifération de liens dont la pertinence est parfois douteuse, consiste à se concentrer sur l'établissement de liens entre les fragments terminologiques et les autres. En général, l'auteur d'un fragment n'a pas de peine à créer des liens correctement typés vers les concepts dont relève son fragment. Une fois ces liens établis, on peut inférer des liens entre fragments non terminologiques par un parcours « aller-retour » vers le niveau terminologique. La figure 3 montre un lien obtenu par un parcours direct (1) (pour trouver un autre exemple du même concept) et un parcours plus long (2) (pour trouver un autre exemple en passant par un concept plus spécifique).

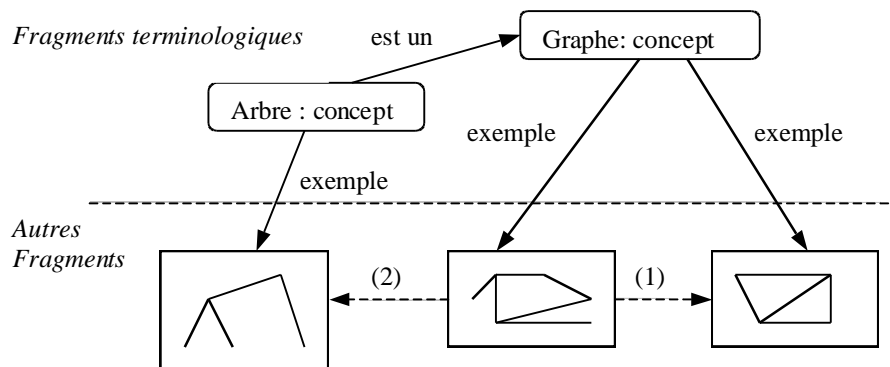


Figure 3. Inférence de liens en passant par le niveau terminologique.

On peut exprimer de manière générale ce type de règle d'inférence par des chemins dans la monographie. Une expression de chemin prendra la forme suivante :

*fragment:catégorie lien:type fragment:catégorie lien:type ...
fragment:catégorie.*

Ainsi, le lien (1) de la figure 3 peut être obtenu par la règle d'inférence

$f1 \leftarrow l1:exemple - c:concept - l2:exemple \rightarrow f2$

(Partir d'un fragment $f1$, parcourir en sens inverse un lien $l1$ de type *exemple* pour arriver à un fragment concept c , puis « redescendre » par un lien *exemple* à un fragment $f2$.)

De la même manière, en fonction des types de liens admis dans la monographie, on pourra définir des inférences de liens ayant un sens précis et jugé utile pour le lecteur.

Exemples. Les exemples qui suivent montrent des inférences de liens et leurs expressions sous forme de chemins.

a. Si $f1$ est un exemple du concept c , créer un lien vers un fragment $f2$ qui est un exemple de c ou d'un concept d plus spécifique que c .

$$f1 \leftarrow l1:exemple - c:concept \leftarrow est-un^* - d:concept - l2:exemple \rightarrow f2$$

La notation $\leftarrow est-un^*$ représente le parcours de zéro, un ou plusieurs liens taxonomiques *est-un*, dans le sens du générique au spécifique.

b. Lier un fragment $f1$, exemple de c , à tout concept d lié à c par un lien quelconque.

$$f1 \leftarrow l1:exemple - c:concept - l2 \rightarrow d:concept$$

c. Si le fragment $f1$ utilise le concept c , créer un lien vers les propriétés de c et des concepts d plus généraux que c .

$$f1 - l1:utilise \rightarrow c:concept - est-un^* \rightarrow d:concept - l2:propriété \rightarrow f2$$

Dans le cas où $f1$ est un exercice on obtiendra, par exemple, un lien vers toutes les propriétés des concepts mis en jeu dans l'exercice.

Une propriété intéressante de cette méthode est sa robustesse par rapport à l'évolution de la monographie. Les documents et liens d'interface étant générés dynamiquement, les liens inférés que l'utilisateur voit dans l'interface sont forcément à jour. De plus, le niveau terminologique est en général stable (on ne reconceptualise pas un domaine tous les jours), l'auteur d'un lien depuis ou vers un fragment terminologique peut être pratiquement certain que celui-ci va perdurer tant que le fragment non terminologique existera.

Nous verrons à la section suivante que ce type d'inférence de liens est facile à spécifier avec le langage de génération d'interfaces. Ainsi, le lecteur d'un fragment pourra voir non seulement le contenu d'un fragment mais également les liens inférés par la ou les méthodes choisies.

Pour que cette méthode soit applicable il faut que les utilisateurs se servent des types de liens pour lesquels des règles d'inférence existent. C'est pourquoi, dans les dernières versions des hyperlivres que nous avons implémentés, il existe un choix limité de types de liens : *est-un*, *partie-de*, *instance-de*, *rôle* (accompagné d'un nom de rôle), *exemple-de*, *référence*, *remarque*, *réponse*, *preuve* et *contradiction*.

4.3 Formalisation des définitions dans les domaines complexes

Lorsque la monographie traite d'un domaine complexe où le nombre de concepts est élevé, certaines tâches deviennent ardues. Par exemple :

- la comparaison de concepts est rendue difficile car il peut exister de subtiles différences qui n'apparaissent pas clairement dans la définition textuelle ni dans les liens entre concepts ;
- la recherche du concept à partir d'éléments de sa définition nécessite une exploration quasi exhaustive de la terminologie (par exemple pour répondre à la question « Comment s'appelle le meuble muni de tablettes et de crochets qui sert à suspendre des ustensiles à anse ? ») ;
- l'inférence de propriétés (par exemple « Quelles sont toutes les pièces qui entrent dans la construction d'une automobile ? ») ;

Dans ces cas il devient nécessaire de formaliser plus précisément les définitions. C'est pourquoi nous avons étendu la notion de lien pour pouvoir représenter des énoncés de logique descriptive. En particulier, chaque lien peut être muni d'un nombre minimum et maximum d'occurrences et d'un quantificateur. On peut également construire des concepts par conjonction et disjonction de concepts (la définition détaillée du modèle est donnée dans [Falquet *et al.*, 2000]).

L'exemple ci-dessous montre les liens à créer pour définir le concept d'armoire comme « meuble de rangement de grande dimension, composé d'une porte (ou plus), de deux ou plusieurs étagères, d'un corps et dont l'usage principal est uniquement de ranger du linge ou des habits ».

```

armoire –est un→ meuble
armoire –dimension→ grand
armoire –partie→ porte
armoire –partie (2, *)→ étagère
armoire –partie→ corps
armoire – all usage principal→ usage armoire
usage armoire –verbe→ ranger
usage armoire –objet→ ( linge OU habits )

```

Le quantificateur **all** indique que si un objet appartenant au concept *armoire* possède un *usage principal*, ce dernier doit appartenir au concept *usage armoire*. En d'autres termes, il n'y a pas d'autre usage principal pour une armoire que celui indiqué.

Ce genre de définition nécessite évidemment un effort de conception pour mettre en évidence un ensemble si possible minimal de types de liens pertinents pour décrire les concepts du domaine de la monographie. Par exemple, dans le domaine des meubles nous avons abouti à un ensemble de six caractères de premier niveau : usage-principal, utilisateur, partie, forme, dimension, usage-secondaire, particularité.

5. Modèle d'interface hypertextuelle

Quelle que soit la sophistication du modèle d'hyperlivre virtuel et ses capacités de représentation fine de la connaissance, c'est l'interface humain-machine qui va rendre le système utilisable (et utilisé). On peut même dire que plus le modèle sous-jacent est riche, plus l'interface doit être soignée pour que l'utilisateur soit capable d'exploiter cette richesse. Il faut également reconnaître qu'une vision unique des objets de l'hyperlivre ne peut convenir à tous les utilisateurs et à tous les usages. Selon un principe bien connu de la conception des interfaces, il est nécessaire de fournir diverses visions de l'hyperlivre.

5.1 Vues hypertextuelles

Dans notre cas, l'accès au contenu informationnel se fait au travers de vues qui sont des (hyper)documents générés à partir de la monographie. Pour spécifier ces vues nous utiliserons une extension du langage Lazy. Ce langage a été conçu pour spécifier et implémenter des vues hypertextuelles pour les bases de données relationnelles ou orientées-objet [Falquet *et al.*, 1999b]. Dans [Falquet *et al.*, 1999a] nous avons montré que Lazy permettait de créer des documents hypertextes ou des documents réels (hiérarchiques) pour « lire » le contenu de bases de données. Contrairement à d'autres langages de création de documents dérivés (tels PHP, ASP, JSP), celui-ci est entièrement déclaratif, il ne comprend pas d'énoncés impératifs (instructions) mais des déclarations qui spécifient le contenu et les liens de l'hypertexte à construire. On travaille par conséquent à un niveau plus élevé où la programmation est remplacée par la spécification.

L'utilisation de cette approche pour créer l'interface de lecture/écriture de l'hyperlivre implique que la monographie (fragments et liens) soit stockée dans une base de données. Pour simplifier l'exposé nous prendrons un schéma de base de données relationnelle ne comprenant que deux tables :

Fragment (id, titre, categorie, contenu, auteur, data_creation, ...)

Lien (src, dest, type, auteur, date_creation, ...)

– chaque fragment est stocké sous forme d'un n-uplet de la table Fragment. Son identificateur (id) est une chaîne de caractères, de même que son titre et sa catégorie, son contenu est un texte HTML ou XML.

– pour chaque lien on enregistre sa source (src) et sa destination (dest), qui sont des identificateurs de fragments, ainsi que son type, auteur, date de création, etc.

Dans la pratique, la base de données peut être beaucoup plus riche et contenir des données expérimentales, des images, des documents de référence, etc.

La spécification en Lazy d'une vue hypertextuelle est composée de schémas de noeuds. Un schéma de noeud définit le contenu et les liens d'un document d'interface (XML ou HTML) en fonction du contenu de la base de données et de valeurs de

paramètres. Chaque schéma étant paramétrable, il pourra générer différents documents, appelés instances ou noeuds, en fonction des valeurs des paramètres. La vue hypertextuelle sera donc formée d'un ensemble d'instances des différents schémas de noeuds. La figure 4 illustre ce principe.

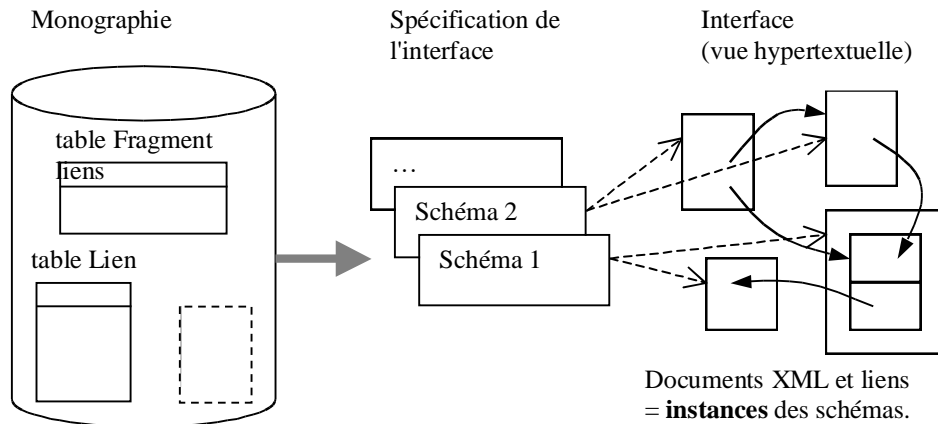


Figure 4. L'interface de l'hyperlivre en tant que vue hypertextuelle de la monographie

Un schéma de noeud est un énoncé de la forme :

```

node nom [paramètres]
      contenu
from collection(s) de données
selected by expression de sélection
ordered by expression

```

La spécification de contenu est une séquence d'éléments qui peuvent être :

- atomiques : chaîne de caractères, nombre, attribut de n-uplet sélectionné, paramètre, fonction, etc. ;
- composites : $\langle \text{type-élément} \rangle \text{élément}_1, \dots, \text{élément}_k \langle / \text{type-élément} \rangle$;
- répétés : $\{ \text{élément}_1, \dots, \text{élément}_k \}$

L'instantiation (ou génération) d'un noeud à partir d'un schéma et de valeurs de paramètres s'effectue de la manière suivante :

1. l'expression de sélection du schéma est évaluée et retourne une liste de n-uplets comme résultat ;
2. les éléments du contenu sont générés, les éléments répétés sont générés pour chaque n-uplet sélectionné.

Exemple. Le schéma de nœud *simple_fragment* sert à afficher l'identité, le titre et le contenu du fragment identifié par le paramètre p.

```

node simple_fragment [ p ]
{ <en-tete> "ID = ", F.id </en-tete> ,
  <titre> F.titre </titre> ,
  <corps> F.contenu </corps>
}
from Fragment F
selected by F.id = p

```

La présentation finale de la page est obtenue à l'aide de feuilles de style XSLT et CSS qui indiquent comment présenter les éléments <en-tete>, <titre> et <corps> qui forment le contenu du nœud. Il est aussi possible de spécifier directement la présentation dans le schéma de nœud si l'on utilise des balises HTML ou XHTML. La figure 5 montre une instance *simple_fragment* ["mdt"] de ce schéma².

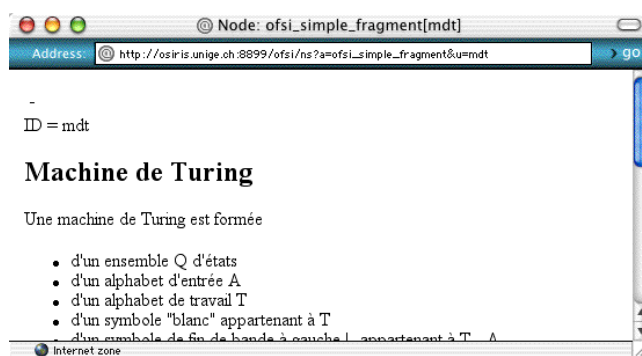


Figure 5. Une instance du schéma *simple_fragment*

Le schéma *fragments_de_categorie* sélectionne tous les fragments de la catégorie *c* donnée en paramètre. La spécification de contenu placée entre { et } est évaluée pour chaque fragment sélectionné. L'énoncé *href* crée, pour chaque fragment, un lien hypertexte vers un nœud de type *simple_fragment* (défini ci-dessus) qui présente le fragment en détail.

² Les exemples montrés ici proviennent d'un hyperlivre que nous avons réalisé avec Lazy et le système de bases de données Oracle.


```

node fragments_de_categorie [ c ]
  <titre>"Fragments [" , c , "]" </titre> ,
  <enumeration>
  { <parag> <evidence> F.titre </evidence> , F.auteur,
    href simple_fragment[F.id] ("ouvrir")
  }
  </enumeration>
from Fragment F
selected by F.category = C ordered by F.title

```

Sur la figure 6 on peut voir une instance *fragments_de_categorie["exercice"]* où apparaissent les liens vers des instances de *fragment_simple*. Dans le système Lazy, les instances de schémas sont générées dynamiquement par le serveur de noeud. Ainsi, un clic de l'utilisateur sur une ancre *[ouvrir]* de ce noeud déclenchera la génération de l'instance de *fragment_simple* correspondante.

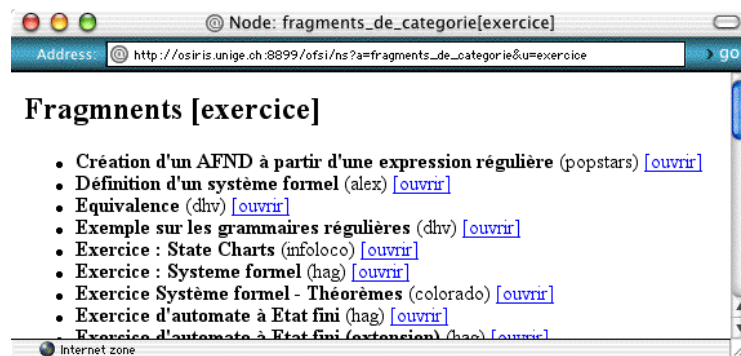


Figure 6. Une instance du schéma *fragments_de_categorie*.

Les deux exemples ci-dessus montrent des documents qui sont encore très proches, par leur structure, de la monographie telle qu'elle est stockée (on reprend tels quels les fragments). Tout l'intérêt de l'approche document virtuelle réalisée par les vues hypertextuelles réside dans la possibilité de créer des documents et liens qui ne correspondent pas directement à la structure des fragments. Le langage Lazy offre pour cela deux mécanismes qui étendent largement le modèle habituel du Web.

Les *liens d'inclusion* permettent d'inclure le contenu d'un noeud dans un autre. On peut ainsi composer des documents hiérarchiques complexes et aux contenus hétérogènes. Un tel document sera présenté dans le paragraphe suivant. Ces liens correspondent à la notion de *transclusion* de Nelson [Nelson, 1995].

Les *liens d'expansion* permettent une inclusion différée. Lorsque l'utilisateur clique sur l'élément de départ du lien, le système ne "saute" pas au noeud désigné mais inclut celui-ci à la place de l'élément de départ. L'utilisateur peut ainsi composer son document en "ouvrant" ou non ces liens (et ceci récursivement).

Ces mécanismes sont également intéressants du point de vue de la conception des interfaces car ils rendent les schémas de noeuds réutilisables. Ils permettent donc de construire et d'étendre l'interface en créant de nouveaux schémas de noeuds par combinaison de noeuds existants.

5.2 Ontologies et interfaces

L'un des problèmes soulevés par Landow [Landow, 1998] à propos de conception des hypertextes est de savoir comment faire pour que l'utilisateur puisse entrer facilement dans l'hypertexte et s'y sentir rapidement à l'aise. L'ensemble des concepts traités par l'hyperlivre peut jouer un tel rôle de point d'accès, en particulier si l'on dispose d'une présentation claire et synthétique de l'ontologie. En effet, le nombre de concepts de l'ontologie est en général beaucoup plus petit que le nombre de fragments. L'accès par les concepts correspond également bien à l'idée de non-séquentialité de l'hyperlivre ; à partir d'un concept, on peut plonger directement sur les fragments s'y rapportant.

Exemple. Cet exemple illustre l'idée du document virtuel. À partir de fragments séparés, on fabrique un document qui possède une unité sémantique et qui est lisible séquentiellement. Les schémas de noeuds de la figure 7 permettent de créer un document contenant :

- la définition d'un concept,
- les contenus des fragments non terminologiques liés à ce concept (commentaires, exemples, exercices, etc.),
- des liens d'expansion vers les concepts liés à ce concept.

On remarquera les expressions de chemins de la forme [F --L-- G] qui indiquent que les fragments F et G doivent être liés par un lien L (de F vers G ou de G vers F). Ce sont des abréviations de l'expression relationnelle (F.id = L.src and L.dest = G.id) or (F.id = L.dest and L.src = G.id).

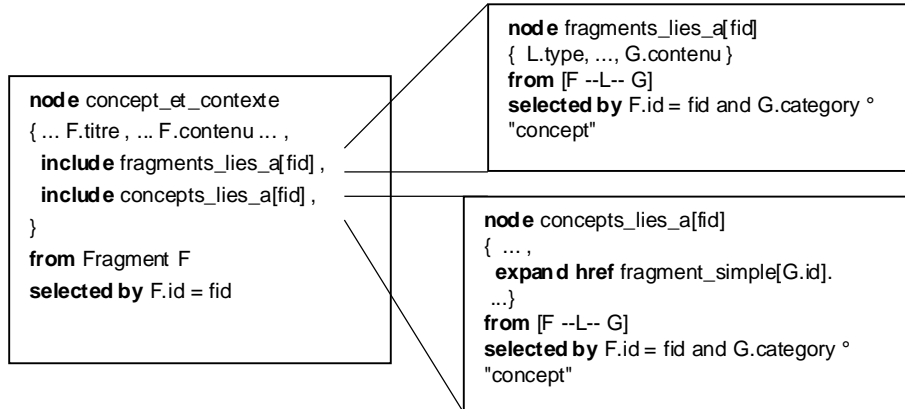


Figure 7. Définition d'un schéma de noeud composite par l'inclusion de schémas

Node: concept_et_contexte[automate etats fini...]

Address: http://osiris.unige.ch:8899/ofs/ns?&concept_et_contexte&u=automate%20etats%20fini&u=0&u=110wzw003jkd4c

Automate à états fini

Un AF est composé de

- un alphabet de symboles reconnus $A = \{s_1, s_2, \dots, s_n\}$
- un ensemble d'états $Q = \{q_1, q_2, \dots, q_m\}$ (les ronds)
- un état initial q_1 de Q
- un ensemble F d'états finals inclus dans Q
- une fonction de transition d de $Q \times A$ dans Q (les flèches)

Commentaire

Un automate fini est déterministe, intuitivement s'il ne peut se comporter que d'une façon pour un mot donné : étant donné un état de départ et un caractère lu, il y a au plus 1 état d'arrivée

Dans le cas d'un **automate déterministe**, si un mot est accepté par le langage, alors il existe une seule dérivation possible pour ce mot. En revanche, si un automate n'est pas déterministe, il peut y avoir des configurations pour lesquelles plusieurs transitions différentes sont possibles.

Exercice

Exercice : On veut accepter le langage forme de a, de b et de c, dans cet ordre.
(p. ex. : aaabcccc, abbc, aaaaabc, ...)

Concepts liés

- [Acceptation d'une chaîne par un AEF \(algorithme\)](#)
- [Acceptation d'une chaîne par un AEF \(math\)](#)
- [Automate à états fini non-déterministe](#)
- [Chaîne de symboles](#)
- [Langage régulier](#)
- [Machine de Turing](#)
- [State Chart](#)

Internet zone

Figure 8. Une instance du schéma concept_et_contexte

Dans le cas présent, la composition du document est élémentaire et n'obéit à aucun critère pédagogique ou rhétorique particulier, si ce n'est le fait d'être fortement liée à un concept. On remarquera aussi sur cet exemple que les liens hypertextuels ne sont pas placés dans le texte mais à côté. Ceci, dans le but de rendre

la lecture beaucoup plus facile que celle des pages Web habituellement parsemées de portions de texte en bleu et soulignées.

On peut, bien entendu, définir des schémas de noeuds pour créer des documents selon des critères plus sophistiqués. Comme nous l'avons mentionné dans la section précédente, l'ontologie peut servir de « pont » entre les fragments non terminologiques. Dans ce cas, le critère d'assemblage est l'existence d'un chemin qui relie deux fragments en passant par un ou plusieurs fragments de l'ontologie. Le langage de spécification de vues est suffisamment puissant pour exprimer toutes les inférences correspondant à des parcours fixes dans l'ontologie, mais également des inférences récursives nécessitant un nombre indéterminé de parcours de liens. On peut, par exemple, retrouver tous les concepts génériques d'un concept donné (en parcourant les liens de type *est-un* tant que c'est possible). L'ensemble des schémas qui entrent (indirectement) dans la composition d'un document (noeud) d'interface forme bien une spécification déclarative de la méthode d'assemblage du document.

Comme le mentionne De Bra [De Bra, 2002], l'ontologie peut servir à personnaliser le parcours d'un lecteur dans l'hyperlivre. Si l'on connaît à un instant donné les concepts maîtrisés et non maîtrisés par un lecteur, on peut lui conseiller ou déconseiller la lecture de certains fragments en fonction des concepts utilisés. Pour réaliser ce type de personnalisation il suffit d'ajouter à la base de données les tables nécessaires au stockage des informations personnelles (nous n'aborderons pas ici le problème complexe de l'évaluation de la maîtrise ou non d'un concept par un lecteur).

5.3 Gestion de l'interaction et de l'écriture

Grâce à la notion de lien hypertextuel actif, il est possible d'agir sur la base de données. Lorsqu'un tel lien est suivi, il déclenche une opération d'insertion, de suppression ou de modification de la base de données. L'exemple ci-dessous montre un schéma de noeuds permettant la création d'un lien sortant d'un fragment *f*. La partie de droite de la Figure 9 montre une instance de ce schéma obtenue par expansion d'un lien (*expand href*) dans un autre noeud montrant le contenu du fragment.

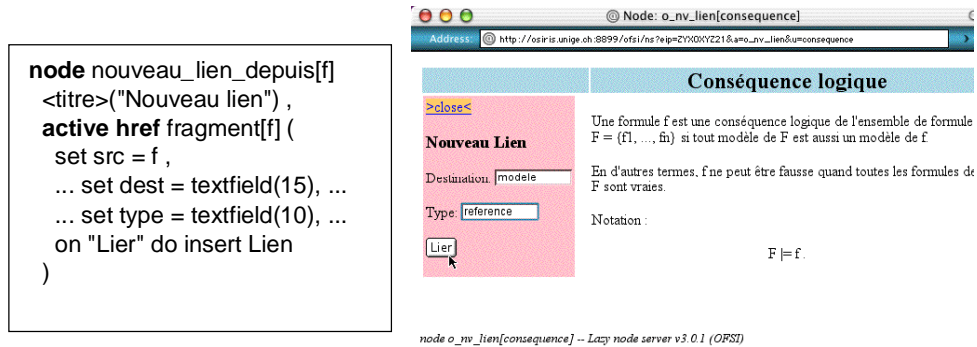


Figure 9. Schéma et instance de noeud avec lien actif

L'effet du bouton "Lier" sera de créer un nouveau n-uplet [$src = f$, $dest = valeur\ entrée$, $type = valeur\ entrée$] dans la relation lien, puis d'afficher le noeud $fragment[f]$.

On emploie de manière analogue les liens actifs dans des schémas de noeuds destinés à la création de nouveaux fragments, à la modification ou la suppression de fragments ou de liens.

L'intérêt des liens actifs est cependant plus vaste car ceux-ci permettent de réaliser des interactions plus sophistiquées, comme, par exemple, la création d'un lien comme résultat d'une navigation : plutôt que de donner l'identité du noeud de destination, on navigue vers lui et une fois arrivé on clique un bouton de confirmation.

Comme nous l'avons fait remarquer en introduction lors de la présentation du modèle, les tâches de lecture et d'écriture ne sont pas forcément aussi distinctes qu'on pourrait le penser au premier abord. La lecture active de textes à caractère scientifique ou technique nécessite en effet bien souvent la prise de notes, l'exécution de calculs, la résolution d'exercices, etc. La technique des liens pourra être utilisée pour offrir au lecteur de telles possibilités d'écriture. Par exemple, un lien "Note" ouvrira une zone pour la saisie d'une note personnelle. Cette note sera stockée dans la monographie sous forme d'un fragment et d'un lien vers le fragment annoté. Dans un hyperlivre récent, nous avons même utilisé ce type de procédé pour créer un forum de discussion entre lecteurs. Dans ce cas, un forum n'est rien d'autre qu'un ensemble de fragments de type *sujet* liés à des fragments de type *réponse*. Ces fragments peuvent bien entendu être liés aux autres fragments de la monographie.

6. Conclusions et perspectives

Le modèle d'hyperlivre que nous avons présenté ici résulte des observations que nous avons pu faire lors de diverses expériences d'écriture collaborative de notes de cours. Du point de vue technique il se base sur nos travaux sur les documents virtuels et sur les vues hypertextuelles des bases de données.

Ce modèle se situe dans la lignée des modèles qui reconnaissent l'importance d'un niveau terminologique ou sémantique au-dessus du niveau informationnel : [Rada, 1990 ; Nanard et al., 1993 ; Brusilovsky *et al.*, 1996 ; Wu et al., 2001], ainsi que la plupart des travaux sur les documents virtuels personnalisables. Dans notre cas, cependant, le niveau terminologique fait partie intégrante de la monographie, les fragments terminologiques peuvent être lus et écrits de la même manière que les autres fragments, il ne s'agit pas d'un glossaire, d'un réseau sémantique ou de toute autre structure définie indépendamment du livre électronique. La modélisation du domaine est donc considérée comme l'une des tâches d'écriture de l'hyperlivre.

L'originalité de ce modèle réside surtout dans le modèle d'interface qui offre la possibilité de spécifier (et non pas de programmer) différentes vues hypertextuelles sur la même monographie. Le langage de spécification permet d'inférer des liens hypertextuels sur la base de chemins simples ou sophistiqués passant à travers les fragments terminologiques. Il permet également de construire des documents complexes qui regroupent plusieurs fragments de la monographie. Les fragments terminologiques possédant la même structure que les autres il en résulte une grande simplicité dans la spécification, il n'y a pas de formalisme particulier pour accéder à l'ontologie.

Plusieurs hyperlivres ont été réalisés en suivant ce modèle et en utilisant comme moyens technologiques un système de gestion de base de données relationnelle standard et le système de vues hypertextuelles Lazy.

Nous avons réalisé un tel livre dans le cadre d'un cours "outils formels pour les systèmes d'information". L'enseignant s'est chargé d'écrire et de lier les fragments terminologiques (essentiellement des définitions des domaines de la théorie des automates, des grammaires formelles et de la logique) alors que les étudiants avaient pour mission d'écrire des exemples, des illustrations, des exercices, des notes, des références historiques, etc. Cependant, chacun était libre de créer des fragments au niveau qu'il souhaitait. Ainsi, plusieurs étudiants ont eux aussi créé des fragments terminologiques. L'hyperlivre obtenu comprend une quarantaine de fragments terminologiques (concepts) et environ 250 fragments d'information (principalement écrits par les étudiants). L'interface, composée de 29 schémas de noeuds, est essentiellement orientée vers l'écriture. Cependant, grâce au mécanisme d'expansion des liens, le lecteur peut composer dynamiquement un document en ouvrant ou non ces liens, ceci permet entre autres de comparer deux définitions en les juxtaposant, de voir à la fois un fragment et les définitions des concepts concernés.

Un nouvel hyperlivre est déjà en cours d'utilisation par des chercheurs, qui veut être une synthèse d'un domaine scientifique. Dans ce cas nous avons aussi intégré un aspect multipoint de vue. En effet, il est rare qu'un concept ne possède qu'une seule définition. Par exemple, une formule mathématique peut avoir un intérêt purement mathématique (elle correspond à une solution d'une classe de problèmes), un intérêt physique (elle synthétise la connaissance qu'a un physicien d'un phénomène), un intérêt historique (c'est la première formule qui utilise un certain formalisme). Ces différentes visions d'un domaine conduisent également à avoir des fragments et liens qui ne sont intéressants que d'un point de vue donné. L'interface doit alors tenir compte du point de vue du lecteur pour lui présenter l'information de la manière la plus pertinente pour lui.

L'un de nos objectifs est d'intégrer l'idée du multipoint de vue dans un hyperlivre pédagogique, de manière à mieux situer la matière du cours dans son contexte. Par exemple, un cours d'outils formels en informatique peut être lu selon différents points de vue : informatique théorique, modélisation des systèmes, programmation, ingénierie des interfaces...

Un autre sujet à explorer est la sémantique de l'inférence des liens entre fragments à partir des liens avec les fragments terminologiques. Etant donné un ensemble de types de liens fixés, il s'agit d'établir un catalogue des inférences que permettent ces liens et d'évaluer leur intérêt pour la lecture et la compréhension des hyperlivres.

L'interface étant extrêmement simple à modifier et à étendre, nous comptons réaliser d'autres hyperlivres ou d'autres interfaces pour des monographies existantes, le but étant de créer des interfaces de plus en plus efficaces pour répondre aux besoins des lecteurs et auteurs. Cet objectif s'inscrit dans le cadre plus large de la recherche de nouvelles technologies de lecture, d'écriture et de gestion des connaissances.

Les auteurs tiennent à remercier les relecteurs pour leurs remarques pertinentes qui ont permis l'amélioration de ce texte tant sur la forme que sur le fond.

7. Bibliographie

- Ackscyn R., McCracken D., Yoder E., "KMS: a distributed hypermedia system for managing knowledge in organizations", *Communications of the ACM* vol. 31, July 1988, p. 820-835.
- Appelt W., Mambrey P., "Experiences with the BSCW Shared Workspace System as the Backbone of a Virtual Learning Environment for Students". *Proceedings of the World*

Conference on Educational Multimedia, Hypermedia and Telecommunications ED-MEDIA 99, Seattle, June 1999.

Atzeni P., Mecca G., Merialdo P., "To Weave the Web". *International Conf. on Very Large Databa Bases (VLDB'97)*, Athens, 1997.

Auffret G., "Indexation documentaire et documents virtuels. Vers un nouveau model de lecture des documents audiovisuels ?" Atelier "Documents Virtuels Personnalisables" de la *11ème Conférence Francophone sur l'Interaction Homme-Machine*, Montpellier, 1999.

Bernstein M., "Storyspace 1", Proceedings of the thirteenth conference on Hypertext and hypermedia, 2002, College Park, Maryland, 2002, p. 172-181.

Brusilovsky P., Schwarz E., Weber G., "ELM-ART: An intelligent tutoring system on world wide web". In C. Frasson, G. Gauthier, and A. Lesgold, editors, *Intelligent Tutoring Systems (Lecture Notes in Computer Science, Vol. 1086)*, Berlin, 1996, p. 261-269.

Brusilovsky, P., "Methods and techniques of Adaptive Hypermedia. User Modeling and User-Adapted Interaction" in *Adaptive Hypertext and Hypermedia*, Kluwer, 1998, pp. 1-43.

Brusilovsky P., Rizzo R., "Map-Based Horizontal Navigation in Educational Hypertext *ACM Hypertext 2002 Conf.*, College Park, Maryland, 2002.

Crampes M., Vercoustre A. M. , Nanard M. , Ranwez S. (Eds). Acte de l'atelier Documents Virtuels Personnalisables : De la Définition à l'Utilisation, de la *11ème Conférence Francophone sur l'Interaction Homme-Machine*, Montpellier, novembre 1999. URI: <http://www.lgi2p.ema.fr/~multimedia/ihm99/> (site visité en juillet 2002).

De Bra P., "Adaptive educational hypermedia on the web". *Communications of the ACM*, Vol. 45 , No. 5, May 2002.

De Bra P., Calvi L., "Creating Adaptive Hyperdocuments for and on the Web", *Proceedings of the AACE WebNet Conference*, Toronto, 1997, 149-155,

Falquet G., Guyot J., Nerima L. (Eds), Création collective de notes de cours électroniques en bases de données et structures de données, (CD-ROM), CUI Université de Genève, 1998.

Falquet G., Hurni J.-P., Nerima L., Guyot J., Monographies scientifiques évolutives multi-point de vue. In proc. *Int. Conf. NimesTIC2000*, Nîmes, 2000.

Falquet G., Nerima L., Guyot J., Vanoirbeek C., Rekik Y. A. . "Des documents virtuels pour lire les bases de données". Atelier "Documents Virtuels Personnalisables" de la *11ème Conférence Francophone sur l'Interaction Homme-Machine*, Montpellier, 1999.

Falquet G., Nerima L., Guyot J., "Languages and Tools to Specify Hypertext Views on Databases". In *The World Wide Web and Databases*, P. Atzeni, A. Mendelzon, G. Mecca (Eds.), (LNCS Vol. 1590), Springer, 1999.

Fernandez M., Florescu D., Kang J., Levy A., Suciu D., "Catching the Boat with Strudel: Experiences with a Web-Site Management System". In *Proc. ACM SIGMOD Conf.*, Seattle, pages 414-425, 1998.

Fernandez M., Marsh J., M. Nagi (eds), "XQuery 1.0 and XPath 2.0 Data Model". W3C Working Draft, April 2002. Retrieved from <http://www.w3.org/TR/query-datamodel/> on July 10, 2002.

- Fröhlich P., Nejdil W., "A Database-Oriented Approach to the Design of Educational Hyperbooks", *Proceedings of the workshop "Intelligent Educational Systems on the World Wide Web", 8th World Conference of the AIED Society*, Kobe, Japan, 1997, p. 18-22.
- Garlati S., Crampes M., *Actes du congrès Documents virtuels personnalisables (DVP 2002)*, Brest, 2002.
- Garret L. N., Smith K. E., Meyrowitz N., "Intermedia: Issues Strategies and Tactics in the Design of a Hypermedia Document System." *CSCW '86 Proceedings*, Austin, Texas, Dec 1986.
- Halasz F., Schwartz M., "The Dexter hypertext reference model". *Communications of the ACM*, vol. 37, n° 2, 1994, p. 30-39.
- Landoni M., Crestani F., Melucci M., "The Visual Book and the Hyper-Textbook: Two Electronic Books One Lesson?", *RIAO Conference Proceedings*, 2000, p. 247-265.
- Landow G. P., *Hypertext 2.0 The Convergence of Contemporary Critical Theory and Technology*, Johns Hopkins University Press, 1998.
- Nanard J., Nanard M., "Should Anchors be Typed Too? An Experiment with MacWeb", *In Proc. ACM Hypertext '93 Conf.*, 1993, p. 51-62.
- Nelson T., "The heart of connection: hypermedia unified by transclusion ". *Communications of the ACM*, Vol. 38, No. 8, 1995, p. 31-33.
- Nguyen T., Srinivasan V., "Accessing Relational Databases from the World Wide Web". *In Proc. ACM SIGMOD Conf.*, 1996, p. 529-540.
- Rada R., "Hypertext writing and document reuse: the role of a semantic net", *Electronic Publishing*, Vol. 3, No. 3, 1990, p. 125-140.
- Ranwez S., Crampes M., "Conceptual Documents and Hypertext Documents are two Different Forms of Virtual Document", *Workshop on "Virtual Documents Hypertext Functionality and the Web" of the 8th Intl World-Wide Web Conf.*, Toronto, 1999.
- Sherman S. J., "Cooperative Learning and Science". *In Handbook of Cooperating Learning Methods*, S. Sharan (Ed.), Greenwood Press, 1994, p. 226-244.
- Wu H., de Kort E., De Bra P., "Design Issues for General-Purpose Adaptive Hypermedia Systems", *ACM Hypertext 2001 Conf.*, Aarhus, 2001, p. 141-150.

Gilles Falquet est maître d'enseignement et de recherche à l'Université de Genève. Il enseigne l'algorithmique et les structures de données, les approches formelles en systèmes d'information et la conception des interfaces des systèmes d'information. Ses travaux de recherche portent sur les interfaces des systèmes d'information, les interfaces des bases de connaissances, les systèmes hypermedia et les documents virtuels.

Jacques Guyot est concepteur de systèmes d'information chez Montres Rolex S.A. , il est également chargé d'enseignement à l'Université de Genève. Ses domaines de recherche sont les interfaces des systèmes d'information, la gestion des documents électroniques et les techniques d'indexation et de classification des documents.

***Jean-Pierre Hurni** est docteur en physique, il travaille depuis de nombreuses années à l'ISRI (Independent Scientific Research Institute) sur des sujets aussi variés que le désarmement nucléaire, la physique fondamentale, l'histoire des sciences et les aspects informatiques de la vulgarisation de ces disciplines.*