

# Réécriture de requêtes multimédias : approche basée sur l'usage d'une ontologie

Samira Hammiche, Salima Benbernou, Mohand-Saïd Hacid  
Université Claude Bernard Lyon 1 – Bâtiment Nautibus  
43, boulevard du 11 novembre 1918  
69622 Villeurbanne - France  
{shammich, sbenbern, mshacid}@liris.univ-lyon1.fr

**Résumé.** Nous proposons dans cet article une stratégie de réécriture de requêtes sur des données multimédias décrites moyennant le standard MPEG-7. Ce standard se base sur XML schéma qui permet de décrire la structure des données. Cependant, aucune sémantique n'est assignée à cette structure. Nous proposons d'étendre ce standard d'une ontologie permettant d'exprimer les connaissances du domaine. Ainsi, l'ontologie sera utilisée durant l'indexation des données multimédias et la réécriture de requêtes. Le but de la réécriture de requêtes est de transformer une requête initiale en une ou plusieurs requêtes équivalentes ou sémantiquement proches compte tenu des connaissances représentées dans l'ontologie.

## 1 Introduction

L'avènement du Web a contribué à la prolifération massive d'informations multimédias. Cette augmentation rend la recherche de ce type de données de plus en plus difficile. Afin de pallier ce problème, le standard MPEG-7 (Martinez 2003) a été proposé. La principale ambition de MPEG-7 est de rendre les informations multimédias aussi facile à trouver sur le Web que l'est le texte aujourd'hui.

Le standard MPEG-7 fournit différents outils pour décrire la plupart des informations du contenu multimédia. Il est basé sur XML schéma, ce qui permet de structurer et de contrôler les descriptions des contenus. Cependant, aucune sémantique n'est assignée aux descriptions. La formalisation de ces descriptions est une piste pour faciliter la recherche dans les documents MPEG-7. Nous proposons d'augmenter le standard MPEG-7 d'une ontologie du domaine, qui peut être utilisée comme source de descripteurs permettant d'indexer les données multimédias. Afin d'éviter les réponses vides, nous proposons de réécrire les requêtes, avant leur évaluation, en s'appuyant sur les mécanismes de raisonnement offert par le formalisme de spécification de l'ontologie. La requête réécrite est, par la suite, évaluée. A cet effet, nous nous basons sur l'appariement d'arbres comme stratégie de recherche vu la représentation arborescente des descriptions MPEG-7 et des requêtes.

Cet article est organisé comme suit. Dans la Section 2, nous présentons le standard MPEG-7, la représentation et l'interrogation des descriptions MPEG-7. Dans la Section 3, nous décrivons de façon générale le cadre de recherche. La Section 4 présente les ontologies et leur représentation en logique de description. La Section 5 décrit le processus de réécriture de requêtes sur des données multimédias à travers un exemple.

La Section 6 est un état de l’art sur la réécriture des requêtes en utilisant les ontologies, suivi d’une conclusion.

## 2 Le standard MPEG-7

### 2.1 Présentation et limites du standard

Publiée en septembre 2001, la norme MPEG-7 propose une syntaxe basée sur XML pour la description des caractéristiques du contenu multimédia et définit la notion d’outils de description multimédia (Martinez 2003).

MPEG-7 est basé sur XML schéma. Ce dernier permet d’ajouter de la structure aux descriptions MPEG-7. Cependant, il fournit très peu de moyens pour exprimer de la connaissance, autrement dit la *sémantique* associée à cette structure. Il faut alors ajouter de la connaissance aux descriptions MPEG-7 pour pouvoir effectuer des raisonnements et inférer des informations implicites.

### 2.2 Représentation des descriptions MPEG-7

Nous avons représenté dans (Hammiche et al. 2004) une description MPEG-7 sous forme d’un arbre XML étiqueté et ordonné  $T = (N, E, root(T))$  composé de trois types de nœuds :  $N_e$  pour les éléments internes,  $N_t$  pour les feuilles et  $N_a$  pour les attributs.  $root(T)$  représente le type de la donnée décrite en MPEG-7. Donc, les éléments d’un arbre correspondant à une description MPEG-7 sont :

- $N = N_e \cup N_t \cup N_a$  l’ensemble des nœuds de l’arbre.
- $E$  : les arcs de l’arbre capturent l’imbrication des éléments de la description MPEG-7.

### 2.3 Interrogation des descriptions MPEG-7

Nous considérons des requêtes qui peuvent porter sur la structure et sur le contenu des descriptions MPEG-7. Nous utilisons un sous ensemble du standard XQuery comme langage d’interrogation (XQuery 2004). On note ce langage  $XQuery^-$ . La portion que nous considérons est limitée aux requêtes sous forme d’expression de chemin dont la grammaire est la suivante :

$$[“/”|“//”]( < StepExpr > (“/”|“//”))* < PrimaryExpr > [“ < Expr > ”]$$

sachant que “/” et “//” dénotent les relations directes (père-fils) et indirectes (ancêtre-descendant) respectivement. La partie se trouvant entre les crochets dans  $< Expr >$  spécifie les conditions de recherche à vérifier. Pour des raisons de simplicité, les opérations sur les prédicats sont restreintes à l’égalité. Une combinaison de prédicats à travers les opérateurs logiques “and” et “or” est possible. Par exemple, la requête “rechercher les CDs dont le nom de l’auteur est John et contenant la chanson imagine” est écrite en  $XQuery^-$  comme suit :

$$CD[ auteur/nom = “john” and chanson/titre = “imagine” ]$$

### 3 Cadre de recherche

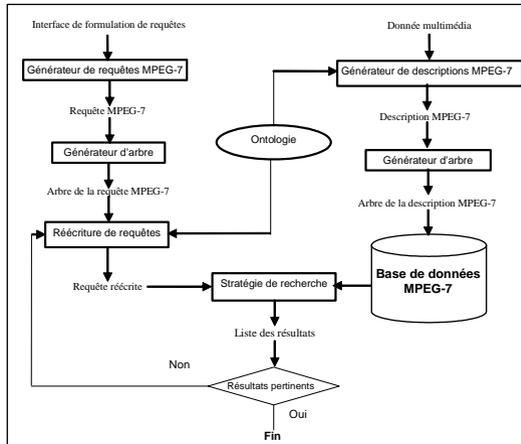


FIG. 1 – Architecture du cadre de recherche

Nous avons spécifié, dans les Sections 2.2 et 2.3, la représentation sous forme d'arbres, adoptée aux descriptions MPEG-7 et aux requêtes. Dans cette section, nous présentons le cadre (cf. figure 1) utilisé lors du processus de recherche. Ce dernier se compose de cinq modules : le générateur de description MPEG-7, le générateur de requêtes MPEG-7, le générateur d'arbre, le module de réécriture de requêtes et le module qui définit la stratégie de recherche. Le but de ce papier est de présenter le module de réécriture de requêtes nouvellement introduit. Son rôle est de réécrire la requête initiale en un ensemble de requêtes équivalentes compte tenu des connaissances véhiculées dans l'ontologie. Dans le cadre de recherche, il est spécifié que si les réponses d'une requête ne sont pas pertinentes, la requête est réécrite afin de satisfaire les besoins l'utilisateur.

### 4 Les ontologies : Définition et Représentation

Une ontologie est une spécification explicite d'une conceptualisation (Gruber 1993). Elle se réfère généralement à un ensemble de termes structurés de façon hiérarchique, conçu afin de décrire un domaine. L'ensemble de ses composants sont (Baader 2003) : les concepts et leurs propriétés, les liens organisant les concepts (à travers la relation de subsomption), les relations entre les concepts, les axiomes pour modéliser les assertions toujours vraies et les individus ou instances de concepts.

Propulsées par les travaux réalisés autour du Web sémantique, les logiques de descriptions (LDs) se retrouvent parmi les langages les plus utilisés pour la représentation

de la connaissance et l’opérationnalisation des ontologies. La définition d’une ontologie en LD nécessite de choisir la LD répondant aux besoins exprimés. Toute LD est définie par une syntaxe, une sémantique formelle et des mécanismes de raisonnement. Le raisonnement dans les LDs comprend, entre autre, deux mécanismes d’inférences, tous deux basés sur des calculs de relations de subsomption (un concept  $A$  subsume un concept  $B$  si  $A$  est plus général que  $B$ ) : la classification de concepts et la reconnaissance d’instances.

## 5 Réécriture de requêtes à base d’ontologie

### 5.1 Exemple

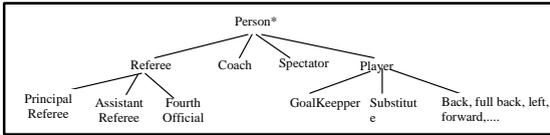


FIG. 2 – Hiérarchie de concepts des personnes

Afin d’illustrer le processus de réécriture de requêtes, nous allons d’abord introduire un exemple d’ontologie et une partie de sa formulation en logique de description. Puis, nous donnerons un exemple de requête qu’il faudrait réécrire avant d’évaluer. La figure 2 décrit une des hiérarchies de concepts de *l’ontologie du Football*. Dans cette hiérarchie, nous distinguons deux types de concept. Les concepts abstraits - *marqués d’une étoile* - qui ne sont jamais référencés dans la base des descriptions MPEG-7 et les concepts concrets - *non marqués* - utilisés lors de l’indexation. Nous avons défini l’ensemble des relations suivantes entre les différents concepts : *Agentof*, *Patientof*, *Memberof* et *Resultof*.

Pour formaliser cette ontologie en LD, nous avons choisi le langage  $\mathcal{ALCN}\mathcal{R}$  assez expressif pour prendre en compte les besoins de l’ontologie. Cependant, pour distinguer les deux types de concepts présents, nous avons introduit un rôle *asterisk* qui caractérise uniquement les concepts abstraits. Une version modifiée de la subsomption classique notée  $\sqsubseteq^*$  est utilisée. Cette subsomption marquée permet de ne pas propager le rôle *asterisk* d’un concept aux concepts subsumés par ce dernier.

La définition d’un extrait de la hiérarchie des personnes en  $\mathcal{ALCN}\mathcal{R}$  est :  $Person \leq \top \sqcap \exists asterisk$ ,  $Referee \sqsubseteq^* Person \sqcap \exists arbitrate$ ,  $Coach \sqsubseteq^* Person \sqcap \exists coach$ ,  $Player \sqsubseteq^* Person \sqcap \exists play$ ,  $Goalkeeper \sqsubseteq^* Player \sqcap \exists locationof GoalArea$ .

Cette définition correspond au niveau terminologique de l’ontologie. L’instantiation des concepts et des relations constitue le niveau assertionnel. L’ensemble  $\mathcal{A}$  représenté ci-après est un sous ensemble des assertions.

$\mathcal{A} = \{ Player(ronaldo) , GoalKeeper(g) , Substitute(s) , left(l) , Referee(r) , Agentof(ronaldo, goal) , Agentof(r, whistle) , Patientof(r, yellowcard) ,$

$Resultof(expulsion, redcard), Resultof(Yellowcard, fault)\}$ .

**Exemple de requête :** Supposons que l'utilisateur formule la requête suivante : "rechercher les extraits vidéo où une personne marque un but". Cette dernière est transformée en une requête  $XQuery^-$  conforme au standard MPEG-7, Les contraintes de recherche (*person*, *goal* et *agentof* (*person*, *goal*)) sont formulées en conjonction de concepts définis dans l'ontologie. Donc, avant son évaluation, on lui appliquera l'algorithme de réécriture.

## 5.2 Algorithme de réécriture

---

### Algorithm 1 Rewrite ( $Q$ )

---

**Require:**  $Q$  – la requête de l'utilisateur

$C = \{c_1, \dots, c_n\}$  – Les concepts de l'ontologie présents dans la requête

$R = \{r_1, \dots, r_m\}$  – Les relations identifiées entre les concepts

$KB$  – La base d'assertions de l'ontologie

**Ensure:**  $Q'$  – l'ensemble des requêtes sémantiquement équivalentes à  $Q$

$Q' = \{\}$

1. **for each**  $C_i \in C$  **do**

2.   **if**  $C_i \sqsubseteq \exists asterisk$  **then**

3.      $D \leftarrow \{\}$

4.     **for each** assertion in  $KB$  stating that  $D_j \sqsubseteq C_i$  **do**

5.       **if**  $D_j \sqsubseteq \exists asterisk$  **then**

6.          $D \leftarrow D$

7.     **else**

8.       **for each**  $R_i \in R$  **do**

9.         **for each** assertion in the  $KB$  stating that  $D_j$  is the component of  $R_i$  **do**

10.          $D \leftarrow D \cup \{D_j\}$

11.       **endif**

12.     **for each**  $D_j \in D$  **do**

13.        $Q' \leftarrow Q' \cup Q(C_i/D_j)$

14.     **endif**

15. **return** ( $Q'$ )

---

Le but de la réécriture est d'éliminer les concepts abstraits (car ils ne sont jamais référencés dans la base MPEG-7) et d'avoir une ou plusieurs requêtes sémantiquement équivalentes à la requête de départ, en se basant sur les mécanismes d'inférences de la LD.

Nous décrivons dans ce qui suit l'algorithme que nous proposons pour la réécriture des requêtes. Nous utiliserons la notation  $Q(C/D)$  pour exprimer le fait de remplacer le concept  $C$  dans la requête  $Q$  par le concept  $D$ . La base de connaissances  $KB$  contient les assertions du domaine (concepts, relations et individus).

L'algorithme présenté nécessite en entrée les concepts présents dans l'ontologie ainsi que les relations entre ces concepts. Il nécessite aussi la base d'assertion. En sortie, il produit un ensemble de requêtes sémantiquement équivalentes. Pour tout concept

marqué (ligne 2), l'algorithme recherche ses subsumés dans la base d'assertions afin de le réécrire. Pour tout concept subsumé trouvé, si ce dernier est marqué, il ne sera pas utilisé. Dans le cas contraire (ligne 7), l'algorithme vérifie si ce concept satisfait les relations entre concepts de la requête puis l'ajoute à l'ensemble  $D$ . Au final, la requête  $Q$  est réécrite en l'*union* des requêtes obtenues en remplaçant les concepts de la requête initiale par les différents concepts retrouvés dans l'ensemble  $D$ .

### 5.3 Application à l'exemple

L'application de cet algorithme à l'exemple présenté précédemment, réécrit la requête avec les concepts *ronaldo*, *goal* et *agentof (ronaldo, goal)* et ces concepts sont utilisés dans la base des descriptions MPEG-7. Ce qui permet d'avoir des réponses. Si la requête était gardée sous sa forme initiale, aucune réponse n'aurait été trouvée.

## 6 État de l'art et conclusion

Notre travail est relatif à l'interrogation des données MPEG-7 (ou plus généralement les données XML) en s'appuyant sur des connaissances du domaine représentées par les ontologies. Dans (Ederman et Decker 2000), une ontologie est utilisée pour formaliser les balises d'un document XML et une approche de recherche basée sur un moteur de recherche XML et un moteur d'inférence a été proposée. Dans (Kiryakov et Simov 1999), une ontologie est utilisée pour représenter les inférences nécessaires utilisées lors du processus de recherche.

Dans cet article, nous avons présenté une stratégie de réécriture de requêtes en se basant sur une ontologie du domaine d'application. L'ontologie est utilisée plus particulièrement lors de l'indexation des données et la réécriture de requêtes. Nous avons distingué les concepts de l'ontologie et introduit un rôle pour les différencier. Nous avons introduit également une nouvelle forme de subsumption permettant de répondre à nos besoins. Nous travaillons actuellement sur la complexité de l'algorithme de réécriture.

## Références

- Baader F. (2003), Description logic terminology. In Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi et Peter F. Patel-Schneider editors, The Description Logic Handbook : Theory, Implementation and Applications, pp 485-495. Cambridge University Press, 2003.
- Ederman M. et Decker S. (2000), Ontology-aware XML-queries. Submission for WebDB 2000.
- Gruber T.R. (1993), A translation approach to portable ontology specifications. Knowl. Acquis. 5(2), pp 199-220, 1993.
- Hamliche S., Benbernou S., Hacid M.S et Vakali A. (2004), Semantic Retrieval of Multimedia Data. In Proceedings of the Second ACM international workshop on Multimedia databases. ACM Press, November 2004.