HAL
open science

Formalisation des connaissances documentaires et des connaissances conceptuelles à l'aide d'ontologies : application à la description de documents audiovisuels

Raphaël Troncy

▶ **To cite this version:**

Raphaël Troncy. Formalisation des connaissances documentaires et des connaissances conceptuelles à l'aide d'ontologies : application à la description de documents audiovisuels. Interface homme-machine [cs.HC]. Université Joseph-Fourier - Grenoble I, 2004. Français. NNT : . tel-00005263v2

HAL Id: tel-00005263

https://theses.hal.science/tel-00005263v2

Submitted on 9 Feb 2006

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

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

UNIVERSITE JOSEPH FOURIER
SCIENCES. TECHNOLOGIE. MEDECINE. GRENOBLE - ALPES

École Doctorale Mathématiques, Sciences
et Technologies de l'Information, Informatique

UFR IMA

Formalisation des connaissances documentaires et des connaissances conceptuelles à l'aide d'ontologies : application à la description de documents audiovisuels

THÈSE

présentée et soutenue publiquement le 5 Mars 2004

pour l'obtention du

Doctorat de l'université Joseph Fourier – Grenoble 1

(spécialité Informatique : Systèmes et Communications)

par

Raphaël TRONCY

Composition du jury

Président :	Yves CHIARAMELLA
Rapporteurs :	Jean CHARLET Jacques LE MAITRE
Examinateurs :	Asunción GÓMEZ-PÉREZ Yannick PRIÉ
Directeurs de thèse :	Bruno BACHIMONT Jérôme EUZENAT

Institut National de l'Audiovisuel
Institut National de Recherche en Informatique et en Automatique – Unité Rhône-Alpes

institut national de l'audiovisuel

INRIA RHÔNE-ALPES

Mis en page avec la classe thloria.

À Ana Gabriela

Remerciements

Au terme de ce travail de longue haleine, je tiens à remercier ici ceux qui m'ont soutenu et aidé dans mes recherches : Mr Yves Chiaramella, tout d'abord, pour avoir accepté de présider le jury de cette thèse ; Mr Jean Charlet et Mr Jacques Le Maitre, ensuite, pour le travail de rapporteur qu'ils ont effectué sur ce manuscrit, lequel m'a permis d'en améliorer substantiellement la qualité ; Mme Asunción Gómez-Pérez et Mr Yannick Prié, enfin, pour avoir accepté de faire partie de mon jury : leurs critiques m'ont été d'une grande utilité et, je n'en doute pas, le point de départ de futures collaborations fructueuses.

Je tiens à remercier tout particulièrement Mr Jérôme Euzenat et Mr Bruno Bachimont, pour leur encadrement : leurs précieux conseils et leur rigueur m'ont fait chaque jour prendre conscience de la réalité du travail de chercheur, et leur confiance m'a permis d'accomplir ses différentes missions. Je les remercie également pour les nombreuses lectures et corrections qu'ils ont pu apporter à la rédaction de ce manuscrit, ainsi que pour leur patience souvent mise à rude épreuve.

J'adresse par ailleurs mes remerciements à la direction de la recherche de l'INA à travers son directeur, Mr Daniel Teruggi, et son directeur adjoint, Mr Philippe Poncin, pour m'avoir accueilli ; À l'équipe *Métadonnées* puis *Description des Contenus Audiovisuels* pour avoir fourni des problématiques de recherche et un terrain de jeu si stimulant, et en particulier à Vincent, Patrick, Jean, Laurent, Steffen, Fabrice et Michèle qui ont su, chacun dans leur domaine de compétences, apporter des réponses à mes nombreuses interrogations.

J'adresse une mention spéciale à Antoine Isaac et à Véronique Malaisé qui savent qu'ils ont été pour moi bien plus que des collègues, des amis, qui ont infatigablement et avec un sourire constant égayé l'ambiance parfois terne de notre bureau. Je n'oublie pas non plus mes collègues, doctorants ou stagiaires de passage, qui ont su rythmer notre vie banlieusarde : Gwendal, Emmanuel, Karine, Estelle, Bastien, Younes, Jean-Claude, Rémi, Jean-Philippe et Thomas, je vous souhaite une bonne continuation.

Un grand merci enfin à mes deux familles et à tous mes amis pour m'offrir les bouffées d'air frais et d'enthousiasme dont j'ai besoin : Anne-Laure, Antonin, Cathy, Eric, Jérôme, Ludovic, Maryline, Nathalie, Nicolas, Sébastien, Stéphane. Merci à ma femme, Ana, pour être là, tout simplement : je te dédie cette thèse.

Table des matières

Introduction générale **1**

1 Contexte de la recherche 1

1.1 Motivations 1

1.2 Problématique scientifique 2

1.3 Approche et principaux résultats 4

2 Plan du mémoire 5

Chapitre 1
Le document audiovisuel à l'heure du numérique

1.1 Notion de document et de collection 10

1.1.1 Qu'est ce qu'un document ? 10

1.1.2 Le document numérique 11

1.1.3 Qu'est ce qu'une collection ? 13

1.2 Le document audiovisuel numérique 14

1.2.1 Les spécificités du document audiovisuel 15

1.2.2 Le cycle de vie du document audiovisuel 17

1.2.3 L'indexation de documents numériques 21

1.3 De l'annotation à la description des documents audiovisuels 23

1.3.1 Qu'est-ce qu'une annotation ? 23

1.3.2 L'extraction automatique de caractéristiques dans les documents audiovisuels 25

1.4 Synthèse 27

Chapitre 2
Représenter les documents audiovisuels : une approche documentaire

2.1 L'exemple paradigmatique : la famille XML 32

2.1.1 De SGML à XML 32

2.1.2 Validité d'un document XML 33

2.1.3 La norme HyTime 38

2.2 Des standards pour présenter les documents audiovisuels 39

2.3 Des standards pour décrire les documents audiovisuels 40
2.3.1 La description des données bibliographiques 41
2.3.2 MPEG-7 : le nouveau langage documentaire pour représenter le multimédia 46
2.3.3 TV Anytime 51
2.3.4 Synthèse 52
2.4 Des systèmes opérationnels pour annoter de la vidéo 52
2.4.1 Des systèmes « purement » documentaires 52
2.4.2 Les modèles de représentation orientés « connaissances » 57
2.5 Conclusion 64

Chapitre 3
Les langages de représentation des connaissances et les ontologies

3.1 Les ontologies 66
3.2 Quelques formalismes de représentation des connaissances 69
3.2.1 Un peu d'histoire 70
3.2.2 Les graphes conceptuels 70
3.2.3 Les logiques de description 74
3.3 La représentation des connaissances sur le Web 77
3.3.1 Shoe et Ontobroker : des précurseurs 78
3.3.2 Les langages RDF et RDFS 79
3.3.3 Vers un langage pour exprimer des ontologies sur le Web 82
3.4 La construction des ontologies 86
3.4.1 Les méthodologies de construction existantes 86
3.4.2 Un problème non résolu : comment obtenir les taxinomies ? 88
3.4.3 Les outils existants 91
3.4.4 DOE - *Differential Ontology Editor* 93
3.5 L'exploitation des ontologies 94
3.5.1 Le stockage des bases de connaissances 94
3.5.2 Les moteurs d'inférence 95
3.6 Conclusion 95

Chapitre 4
Une architecture pour raisonner dans les descriptions documentaires

4.1 Description du problème par l'exemple 97
4.2 Langage documentaire ou langage de représentation des connaissances ? 100
4.2.1 La solution MPEG-7 + XML Schema 100
4.2.2 La solution OWL + RDF 106

4.2.3 Discussion . . . 108
4.3 Une architecture duale . . . 109
4.3.1 Présentation générale . . . 109
4.3.2 Formaliser la structure dans une ontologie . . . 110
4.3.3 Engendrer les structures génériques des émissions . . . 111
4.3.4 La base des connaissances . . . 118
4.4 Conclusion . . . 118

Chapitre 5
Ontologie de l'audiovisuel et schémas de description

5.1 Construction d'une ontologie de l'audiovisuel . . . 123
5.1.1 Conceptualisation . . . 124
5.1.2 Formalisation . . . 130
5.1.3 Ontologie de l'audiovisuel et Ontologie de MPEG-7 . . . 132
5.2 Schémas de description et description de la structure . . . 132
5.2.1 Les schémas de description . . . 132
5.2.2 La description d'une émission particulière : l'exemple du magazine `Stade2` 134
5.3 Expérimentations . . . 138
5.3.1 La dérivation de types en XML Schema . . . 138
5.3.2 Expérimentations avec le langage MPEG-7 . . . 139
5.4 Conclusion . . . 147

Chapitre 6
Ontologie de domaine : application au domaine du cyclisme

6.1 Construction d'une ontologie du cyclisme . . . 150
6.1.1 Conceptualisation . . . 150
6.1.2 Formalisation et opérationnalisation . . . 153
6.1.3 L'ontologie de haut niveau . . . 154
6.2 Génération d'une base de faits . . . 156
6.3 Expérimentations . . . 160
6.3.1 Expérimentations avec le système *SeBOR* . . . 161
6.3.2 Expérimentations avec le système *Racer* . . . 163
6.3.3 Visionner les séquences audiovisuelles . . . 164
6.4 Conclusion . . . 166

Conclusion générale **167**

Annexe A
Complément sur l'INA

A.1 Les différentes missions confiées à l'INA 171
A.2 Notices documentaires produites par les documentalistes de l'INA 173

Annexe B
Ontologie de l'audiovisuel

B.1 Taxinomies des concepts 177
B.2 Ontologie différentielle : liste alphabétique des concepts 180

Annexe C
Ontologie du cyclisme

C.1 Taxinomies des concepts avec les axes différentiels 193
C.2 Taxinomies des relations avec les axes différentiels 201
C.3 Ontologie formalisée 206

Table des figures **235**

Liste des tableaux **239**

Liste des codes sources **241**

Bibliographie **243**

Introduction générale

1 Contexte de la recherche

La recherche présentée dans ce mémoire s'est déroulée principalement au sein de l'Institut National de l'Audiovisuel (INA [1]), de décembre 2000 à novembre 2003, en partenariat avec l'équipe EXMO [2] de l'INRIA Rhône-Alpes. Elle intervient donc dans un contexte applicatif fort où l'univers de l'audiovisuel est lui-même soumis à de profondes mutations – sans que l'on soit d'ailleurs capable d'en mesurer complètement les effets et les conséquences – engendrées par la numérisation des informations.

1.1 Motivations

La manipulation de contenus audiovisuels est une tâche à la fois complexe et spécifique dans l'absolu, qui nécessite le plus souvent de recourir à des représentations médiatrices. En pratique, cette manipulation s'effectue généralement par le biais de descriptions des documents. Ainsi, dans une chaîne documentaire audiovisuelle, l'exploitation des documents est conditionnée par une phase préalable d'indexation des contenus (et de stockage des supports). Dans un contexte analogique, les seules exploitations possibles, en pratique, se résument à la recherche documentaire ou au visionnage et à l'écoute des documents, les index ne pouvant pas être utilisés par le dispositif de lecture des documents puisque les systèmes techniques chargés de reconstruire le signal analogique et d'utiliser la documentation sont différents.

La fin des années 90 a été marquée par la numérisation progressive des contenus et l'émergence des systèmes techniques permettant de les traiter. L'audiovisuel a été le grand bénéficiaire de cette vague avec l'arrivée du numérique au cinéma et à la maison (caméra DV). Aujourd'hui, l'audiovisuel numérique bénéficie de nouveaux formats de compression [3] et de l'élaboration de nouveaux services (vidéo à la demande, interactivité), se structure de plus en plus (table des matières et chapitrage d'un DVD), s'échange sur les réseaux grâce à l'augmentation des débits (ADSL, câble) et devient un vecteur de diffusion de la télévision (Télévision Numérique Terrestre, TV sur le Web). Les progrès technologiques ainsi accomplis trouvent leur concrétisation avec l'arrivée des nouveaux terminaux numériques, les *Set Top Box*, qui sont équipés d'un disque dur pour enregistrer les programmes et pour contrôler le direct de la même manière que l'on peut voir un DVD (pause, ralenti, retour arrière...). Ils sont aujourd'hui les exemples paradigmatiques de ce que le numérique apporte à l'audiovisuel en terme de nouvelles possibilités.

1. Plus précisément, dans l'équipe *Description des Contenus Audiovisuels* de la Direction de la Recherche de l'INA. Pour plus de précisions concernant les différentes missions de cet institut, voir l'annexe A.1, page 171.

2. *Échanges de Connaissance Structurée Médiatisés par Ordinateur*, `http://www.inrialpes.fr/exmo/`.

3. Par exemple, la norme MPEG-4 [Koenen, 1999] définit des unités de contenu sonore, image ou audiovisuel appelées *objets média*. Elle définit également la manière de multiplexer et de synchroniser les données associées avec les objets, en gérant la qualité de service de la transmission. Elle permet enfin à l'utilisateur final d'interagir avec les scènes qu'il visualise, par exemple en cliquant sur des objets pour les désigner à un service de télé-achat.

Le document audiovisuel numérique présente donc de nombreux avantages. Ceux généralement cités concernent :

- **La conservation.** La numérisation a comme conséquence la pérennisation des formats des documents puisqu'il sera toujours possible de passer d'un format à un autre moyennant une transformation. De plus, elle permet de reproduire un nombre potentiellement infini de fois un document sans le détériorer.

- **La structuration éditoriale.** Le numérique permet au programme de lecture d'utiliser les index pour proposer différentes exploitations du contenu audiovisuel. L'utilisateur va ainsi pouvoir visionner les documents de manière plus interactive grâce aux possibilités de navigation hypertextuelle ou de recomposition dynamique des documents. Au niveau des archives, c'est tout le fonds qui peut être réorganisé pour constituer des corpus thématisés.

- **La diffusion et l'exploitation.** Le numérique permet de stocker les documents audiovisuels dans des bases de données et de les échanger sur des réseaux. Ainsi, si jusqu'à présent les utilisateurs devaient se déplacer dans les centres d'archives pour pouvoir consulter les documents qu'ils souhaitaient, ils pourront d'ici peu y accéder à distance. Mais pour les documents audiovisuels, la nouveauté fondamentale introduite par le numérique, et sans doute la plus importante, est la possibilité d'accéder à une quelconque partie du document directement (à condition toutefois que le support le permette).

L'apparition du numérique et l'« association » qu'il permet entre l'audiovisuel et l'information amènent donc de nombreuses évolutions dans le domaine de l'archivage et de la consultation des documents audiovisuels. Le système technique devenant unique, les problèmes concernent maintenant le format des descriptions et le type des informations qu'elles doivent contenir pour permettre l'instrumentation des contenus audiovisuels. Les documentalistes voient ainsi leur pratiques profondément bouleversées en raison des nouvelles possibilités technologiques.

1.2 Problématique scientifique

Ces nouvelles possibilités sont au cœur de notre travail. En effet, la question de l'accès du grand public au patrimoine audiovisuel public devient de plus en plus d'actualité à présent que les technologies rendent une telle perspective envisageable. Dans cette optique, il est nécessaire d'explorer les solutions techniques qui pourraient rendre un tel accès possible. Cela pourrait se matérialiser par la création d'un site Web mis à jour quotidiennement et reprenant le catalogue documenté d'un fonds d'archives comme celui de l'INA. Ce catalogue structuré serait accessible par mots clés ou préférablement en naviguant dans des thématiques pré-établies.

L'utilisateur pourrait ainsi avoir une meilleure connaissance de ce qui a été radiodiffusé puis conservé pour un événement donné (éventuellement d'en pré-visualiser quelques extraits) dans l'objectif d'acquérir les droits pour pouvoir visualiser la séquence intégrale. L'utilisateur devrait également pouvoir compléter la base des descriptions existantes en ajoutant sa propre description reflétant son point de vue sur un document (ou une partie) tout en suivant un cadre de documentation standard à des fins d'interopérabilité. Il pourrait ainsi se constituer sa propre base de descriptions sur des documents qu'il n'a pas. Enfin, l'utilisateur devrait pouvoir se constituer un espace personnel en indiquant ses préférences afin d'être alerté dès qu'une séquence correspondant à ses critères de choix entre dans le fonds.

Pour que ces promesses deviennent des réalités, de nombreux problèmes restent à résoudre. D'une manière générale, le système technique proposé doit être intégré et faire le lien entre le

contenu et sa description qui devient alors la condition *sine qua non* à son accès. La mise en place de ce service nécessite une représentation de la *structure* et du *contenu* du catalogue documentant le fonds disponible, afin qu'un utilisateur puisse l'interroger ou le parcourir. Le problème que nous nous posons est de construire cette représentation à partir du système documentaire de l'INA, établi par et pour des professionnels de l'audiovisuel et qui doit nécessairement être adapté dès lors qu'il sera utilisé par des utilisateurs grand public. Il fait ressortir le besoin d'un format homogène et exploitable par la machine, à la fois expressif et optimal en terme de manipulations.

La réponse à cette demande renvoie donc clairement à des problématiques qui peuvent se regrouper en thèmes assez bien délimités (voire même des disciplines académiques) :

L'ingénierie documentaire. Dès lors qu'il s'agit de stocker et d'archiver de grands volumes documentaires, d'instrumenter leur structuration, leur accès et leur consultation, il est d'usage d'avoir recours aux outils et concepts développés par l'ingénierie documentaire. Celle-ci trouve son origine dans le contexte des documents numériques textuels. Les techniques utilisées sont issues du monde SGML/XML [Goldfarb, 1990] et reposent sur le fait qu'il est possible d'enrichir un document textuel par des balises véhiculant des informations sur la structure et sur le contenu de ce document. Le problème est que cette approche ne s'applique pas telle quelle pour les documents audiovisuels puisque le flux audiovisuel ne peut pas s'enrichir simplement par l'ajout de balise dont la position indiquerait quelle partie du document est annotée puisqu'il est multidimensionnel. Il faut donc aborder les problèmes de la localisation des contenus dans le document audiovisuel (repérer des positions spatiales dans l'image et temporelles dans le flux) et de la caractérisation de ces contenus (qualifier le contenu par des annotations exploitables dans une perspective donnée).

La recherche d'information. L'accès au système documentaire pour le plus grand nombre amène le problème de la recherche d'information dans une base de descriptions, à savoir, pouvoir trouver les réponses les plus pertinentes à des requêtes en s'aidant des descriptions. Le système documentaire doit donc être adapté pour répondre à tous types de requêtes, ou mieux, fournir les interfaces pour permettre de formuler ces requêtes avec les termes utilisés par les professionnels de l'audiovisuel.

L'ingénierie des connaissances. La description du contenu des documents, pour être efficace, est généralement contrôlée. Elle fait appel à des concepts sur lesquels des modèles de connaissances formels permettent d'entreprendre un calcul symbolique et de mener des inférences. L'ingénierie des connaissances a proposé depuis plusieurs années de formaliser les notions essentielles à un domaine à travers des *ontologies*. Mais si celles-ci permettent bien de mener des raisonnements, force est de constater que les langages de représentation des connaissances se prêtent mal à la description documentaire car il leurs manque un élément essentiel pour correspondre à une description : la localisation et la structuration des contenus.

Le problème peut finalement se reformuler de la manière suivante :

> *Comment représenter et utiliser conjointement les connaissances documentaires et conceptuelles dans la description des documents audiovisuels ?*

1.3 Approche et principaux résultats

L'approche proposée consiste tout d'abord à utiliser l'inférence comme moyen de réduire la distance entre requêtes et descriptions, et donc de mieux répondre aux requêtes. Cela pré-suppose une modélisation et une représentation des connaissances mobilisées lors des raisonnements. Nous proposons d'utiliser les ontologies comme moyen pour structurer les connaissances tant documentaires que conceptuelles. Notre travail a alors porté sur les moyens d'articuler les connaissances documentaires et les connaissances conceptuelles propres à un domaine à travers l'ontologie dans un contexte d'indexation de documents audiovisuels.

L'ontologie est le point de rencontre des notions nécessaires à la description documentaire et par ailleurs nécessaires à la définition des connaissances liées au(x) domaine(s) dont relève(nt) un document. Nous avons largement étudié les différentes méthodologies de construction d'ontologies, les langages dans lesquels elles sont exprimées ainsi que les outils permettant de les manipuler. En particulier, nous nous sommes inscrit dans le cadre théorique fixé par B. Bachimont [Bachimont, 1996; Bachimont, 2000a]. Nous avons alors implémenté cette méthodologie dans l'outil logiciel **DOE**, pour *Differential Ontology Editor*, basé sur le langage JAVA [Troncy and Isaac, 2002; Bachimont *et al.*, 2002]. Cet outil, disponible depuis deux ans sur le Web à `http://opales.ina.fr/public/`, a déjà été téléchargé par près de 300 utilisateurs et a été intégré dans le projet OPALES (*Outils pour des Portails AudiovisueLs Educatif et Scientifique* [4]).

Parallèlement, nous avons d'appréhendé le contexte industriel de cette thèse en analysant le travail de description effectué par les documentalistes de l'INA. Cette étude préalable de leurs pratiques documentaires et de leur ressource principale – la *bible de l'indexation* – a permis de modéliser leurs tâches et a donné lieu à la construction d'une ontologie de l'audiovisuel qui contient les concepts nécessaires à la description documentaire des programmes.

Nous avons finalement étudié les différents modèles de représentation de documents audiovisuels, et en particulier la norme MPEG-7 [MPEG-7, 2001]. Mais, sur la constatation que le modèle proposé n'était pas encore adapté à notre volonté de gérer connaissances documentaires et connaissances conceptuelles de manière formelle, nous avons décidé de l'étendre grâce à l'ontologie de l'audiovisuel. Nous avons alors proposé une architecture générale outillée, faites de plusieurs ontologies, et permettant d'une part de contraindre la structure documentaire des documents audiovisuels et d'autre part d'exprimer formellement les connaissances tant documentaires que conceptuelles mobilisées par leurs descriptions [Troncy, 2003a; Troncy, 2003b]. Cette architecture a pour finalité la constitution d'une base de connaissances dans laquelle il est désormais possible de *raisonner sur les descriptions documentaires*. Un prototype, constitué de résultats de projets de recherche européens pour la partie gestion de la base de connaissances et raisonneur, et d'une interface Web simple, pour la partie interrogation de cette base, permet finalement de valider la démarche entreprise pour un petit corpus de vidéos annotées, mais permet surtout de lancer des expérimentations prometteuses à plus grande échelle.

Au terme de cette thèse, nous pouvons dire que l'apport principal de notre travail a été :

1. la spécification d'une architecture générale à base d'ontologies permettant tout à la fois de contraindre la structure logique des documents audiovisuels et de représenter les connais-

4. OPALES est un projet PRIAMM (2001-2003) qui réunit des acteurs du monde audiovisuel public éducatif (Centre National de Documentation Pédagogique, La Cinquième-BPS), des partenaires confirmés sur l'indexation, la représentation des connaissances et la consultation multimédia (l'INA, le LIRMM de l'Université Montpellier II, équipes IHM et Graphes Conceptuels), des acteurs importants du monde industriel (CS-Systèmes d'Information, GIP Renater) et des utilisateurs directement visés par le projet (Maison des Sciences de l'Homme).

sances liées à cette structure et au contenu afin de permettre le raisonnement dans les descriptions documentaires ;

2. un outil (DOE) dédié à une méthodologie de construction d'ontologies, qui est aujourd'hui source de recherches linguistiques sur les phases en amont de la conceptualisation de ces ontologies et précédant généralement leur formalisation ;

3. la conception d'une ontologie de l'audiovisuel en s'appuyant sur les pratiques existantes des documentalistes professionnels de l'audiovisuel, qui est utilisée dans l'architecture et qui devrait devenir une ressource intéressante pour tous les futurs algorithmes d'analyse automatique guidés par les connaissances ;

4. le développement d'un prototype mettant en œuvre cette architecture, et d'une interface permettant d'interroger la base des descriptions montrant ainsi la pertinence de produire des descriptions formalisées pour la recherche de séquences audiovisuelles mais également ses limites.

2 Plan du mémoire

Le chapitre introductif de ce mémoire *1- Le document audiovisuel à l'heure du numérique* nous permet de définir les notions fondamentales qui seront mobilisées tout au long de nos travaux (telles que « document », « collection » ou « annotation »). Il nous permet de mieux positionner le problème en montrant clairement les limites du modèle d'indexation actuellement suivi par les documentalistes dans l'optique de profiter pleinement des nouvelles possibilités apportées par le numérique. Nous donnons alors une définition du processus de description des documents audiovisuels en indiquant quel type d'opération doit être effectuée à chacune des étapes.

La première partie de ce document est ensuite consacrée à un état de l'art sur les langages documentaires et les langages de représentation des connaissances. Cette partie se compose des chapitres 2 et 3.

Le chapitre *2- Représenter les documents audiovisuels : une approche documentaire* présente les modèles et langages issus du monde documentaire permettant de décrire des documents audiovisuels numériques. En particulier, nous étudions les approches prenant appui sur le paradigme des « documents structurés » en rappelant le principe des langages de marquage. Comme il est habituel de distinguer en ingénierie documentaire le fond de la forme, nous décrivons rapidement quelques standards permettant de présenter des documents multimédias, avant de nous attarder beaucoup plus longuement sur la norme MPEG-7, le nouveau standard pour décrire les documents multimédias, ainsi que sur ses dérivés. Nous étudions finalement quelques systèmes opérationnels permettant d'annoter de la vidéo et qui intègrent parfois de la connaissance dans les modèles de représentation des documents audiovisuels, autorisant ainsi le calcul d'inférences sur les descriptions. Nous concluons alors ce chapitre sur le constat que les approches « purement » documentaires sont pertinentes pour décrire la structure des documents, mais ne sont pas suffisantes pour décrire la sémantique de leur contenu.

Le chapitre *3- Les langages de représentation des connaissances et les ontologies* commence par définir la notion d'« *ontologie* » en Ingénierie des Connaissances. Nous présentons ensuite

quelques formalismes de représentation des connaissances, et leur adaptation au Web pour représenter des ontologies. Nous étudierons également le problème délicat de la construction des ontologies en proposant une revue des méthodologies et des outils existants. Nous verrons comment nous nous sommes tournés vers la méthodologie proposée par B. Bachimont et comment nous avons cherché à l'implémenter dans un outil d'aide à la structuration d'ontologies. Nous compléterons cet état de l'art en présentant quelques outils permettant de stocker des bases de connaissances, et de les utiliser. Ces outils seront utilisés pour effectuer des raisonnements sur les descriptions des documents audiovisuels.

La deuxième partie de ce mémoire présente notre contribution aux problèmes posés par cette thèse, à savoir la spécification et la réalisation d'une architecture générale permettant le raisonnement dans les descriptions documentaires. Cette partie se compose des chapitres 4 à 6.

Le chapitre *4- Une architecture pour raisonner dans les descriptions documentaires* rappelle tout d'abord notre problématique en l'illustrant par un exemple. Comment décrire et contrôler la structure logique de documents audiovisuels tout en représentant la sémantique de cette structure et la sémantique du contenu de manière à ce qu'un système informatique puisse conduire des raisonnements sur l'ensemble des descriptions ? Nous discutons alors du langage adéquat pour exprimer ces descriptions en se basant sur l'état de l'art de l'ingénierie documentaire et de l'ingénierie des connaissances effectué dans les chapitres précédents. Nous proposons finalement une architecture générale qui combine ces deux approches, et nous détaillons les différentes composantes de cette architecture.

Le chapitre *5- Ontologie de l'audiovisuel et schémas de description* présente les réalisations et les expérimentations menées autour de la modélisation d'une ontologie de l'audiovisuel. Nous commençons par détailler le contenu de cette ontologie compte tenu de la méthodologie de construction que nous avons suivi. Nous montrons ainsi comment l'ontologie a été conceptualisée puis formalisée et finalement traduite dans le langage OWL. Nous présentons alors comment les schémas de description sont construits à partir de cette ontologie et nous donnons un exemple d'instance d'un schéma particulier contraignant la structure logique des magazines sportifs composites. Nous menons finalement diverses expérimentations sur les schémas produits pour tester leur validité.

Le chapitre *6- Ontologie de domaine : application au domaine du cyclisme* détaille tout d'abord la construction d'une autre ontologie, mais cette fois pour un domaine particulier : le cyclisme. Cette ontologie ayant été modélisée selon la même méthodologie que précédemment, nous présentons sa conceptualisation et sa formalisation. Nous montrons alors comment cette ontologie sert de ressource à un système d'extraction d'information qui, appliqué sur un corpus textuel, extrait un ensemble d'assertions formalisées prêtes à décrire le contenu de séquences audiovisuelles en rapport avec ces textes. Nous avons alors utilisé ce système pour décrire quelques séquences vidéos. Nous menons finalement deux expérimentations en interrogeant cette base de séquences annotées avec deux systèmes à base de connaissances implémentés. Nous montrons ainsi qu'il est possible de retrouver des séquences audiovisuelles selon leur structure et leur contenu grâce à l'architecture générale que nous avons proposé.

Nous concluons ce mémoire en rappelant le problème posé par cette thèse et la solution que nous avons apporté. Cette solution a commencé à être validée par un certain nombre d'ex-

périmentations concluantes. La conclusion est surtout l'occasion de présenter les perspectives d'application et de poursuite de la recherche entreprise.

Chapitre 1

Le document audiovisuel à l'heure du numérique

Sommaire

1.1 Notion de document et de collection 10

1.1.1 Qu'est ce qu'un document ? 10

1.1.2 Le document numérique 11

1.1.3 Qu'est ce qu'une collection ? 13

1.2 Le document audiovisuel numérique 14

1.2.1 Les spécificités du document audiovisuel 15

1.2.2 Le cycle de vie du document audiovisuel 17

1.2.3 L'indexation de documents numériques 21

1.3 De l'annotation à la description des documents audiovisuels 23

1.3.1 Qu'est-ce qu'une annotation ? 23

1.3.2 L'extraction automatique de caractéristiques dans les documents audiovisuels 25

1.4 Synthèse 27

Ce premier chapitre revient sur les notions essentielles dont nous aurons besoin pour analyser les problèmes de description de documents audiovisuels et les thèmes de recherche qu'ils suscitent.

Premièrement, il faut définir ce que l'on entend par *document* et considérer la mutation qu'entraîne dans la nature des documents leur numérisation. Nous définirons également la notion de *collection de documents* qui est primordiale pour tout centre d'archives. Nous pouvons alors nous intéresser au *document audiovisuel* en abordant ses spécificités, son cycle de vie, et les problèmes d'indexation particuliers qu'il pose. Nous ne prétendons pas épuiser, au cours de ces quelques pages, les sens de ces notions extrêmement polysémiques, mais nous essaierons plutôt de préciser leur sémantique dans le cadre restreint de nos travaux.

Deuxièmement, il faut définir ce que recouvre la notion d'*annotation*. Celle-ci est en effet le point de départ de toute indexation mais elle mérite d'être analysée. Nous verrons alors qu'elle peut être plus ou moins structurée, plus ou moins formalisée, obtenue automatiquement ou manuellement. . . Nous pourrons finalement définir ce que nous entendons par *description de documents audiovisuels*, une notion centrale dans notre travail qui sera continuellement utilisée dans la suite de ce mémoire.

1.1 Notion de document et de collection

1.1.1 Qu'est ce qu'un document ?

Les documents sont omniprésents dans notre vie courante. Quasiment toutes les disciplines (les scientifiques, informaticiens, linguistes, sémiologues, psychologues, sociologues, économistes, historiens, géographes, juristes...) utilisent et appréhendent, parfois sous des vocables différents, des documents. La notion de document, en soi banale, est donc intuitive pour chacun d'entre nous sans que nous ressentions le besoin de la préciser. Cependant, comme le note [Pédauque, 2003], ce flou entretenu pose aujourd'hui problème car le numérique bouscule profondément la notion de document.

Étymologiquement, un document est « *ce qui sert à enseigner* » (document de *docere* = enseigner en latin). Le sens contemporain de la notion s'appuie, lui, sur deux fonctions, la preuve et le renseignement, comme l'atteste la définition donnée par le dictionnaire Larousse : « *objet quelconque servant de preuve ou de témoignage* ». Ainsi, le document est la trace d'une activité. Mais il ne suffit pas à un objet d'être support de traces pour atteindre le statut de document. Encore faut-il qu'il ait été conçu dans « l'objectif d'être interprété (lu, vu, écouté, visionné) par un certain nombre de personnes différentes de la ou les personnes qui l'ont mis en place » [Prié, 1999]. Ainsi, les traces d'un document participent d'un acte délibéré d'enregistrement et d'une mise à disposition vers un public potentiel pour une réinterprétation future.

D'autre part, B. Bachimont souligne l'importance du support sur lequel est inscrit le document en rappelant donc sa matérialité [Bachimont, 1999]. Selon lui, le document est indissociable d'un support matériel (une feuille de papier, une cassette VHS...) qui devient un support d'inscription où le contenu est exprimé. Ce contenu est finalement l'abstraction des différents documents exprimant la même chose : par exemple, les différents exemplaires matériels d'une même œuvre littéraire par exemple.

[Auffret and Bachimont, 1999], étendant une définition de [Furuta, 1997] considère alors qu'un document est

> « *une unité représentant une contribution intellectuelle identifiée et publiée sur un média pour des raisons spécifiques. Un document exhibe, dans une certaine limite, une structure intentionnelle qui définit comment les éléments de son contenu sont organisés selon des axes dans l'objectif d'être interprétés par un lecteur comme témoignage de cet objectif original de publication.* »

Cette structure intentionnelle se manifeste selon [Bachimont, 1999] par deux types de contraintes :

- Une structure *logique* qui impose un ordonnancement d'éléments de contenu. Les structures logiques se formalisent habituellement par des définitions de types de documents tels que l'on peut les rencontrer dans des formalismes comme SGML ou XML (voir page 32).

- Une structure *matérielle* qui impose une présentation de ces éléments conformément à l'ordonnancement défini précédemment. Cette structure matérielle doit alors obéir à des règles de formatage définies habituellement à l'aide de feuilles de style (voir page 39).

En conclusion, B. Bachimont définit le document comme étant un support matériel d'inscription, une forme sémiotique, et des structures logique et matérielle [Bachimont, 1999].

Ces différentes composantes se retrouvent dans le travail collectif de réflexion effectué au sein du réseau thématique pluridisciplinaire (RTP) 33 du département STIC du CNRS et synthétisé

par R.T. Pédauque [Pédauque, 2003]. En effet, ce travail se propose de préciser la notion de document dans son passage au numérique à partir de recherches qui privilégient plutôt la forme (comme un objet matériel ou immatériel), le signe (comme un porteur de sens) ou le médium (comme un vecteur de communication). Ainsi, [Pédauque, 2003] définit la notion de document selon trois dimensions qui ne sont pas orthogonales mais qui reflètent des points de vue différents sur le document :

- **Le document comme forme** : sous cette rubrique, sont rangées les approches qui analysent le document comme un objet dont on a repéré les frontières (les contours) et celles qui en étudient la structure, les règles qui le constituent (par référence au formalisme) pour mieux l'analyser, l'utiliser ou le manipuler.

- **Le document comme signe** : pour ces chercheurs, le document est perçu avant tout comme un objet signifiant et doté d'une intentionnalité. On passe alors de la notion d'information à celle de connaissance qui a l'avantage sur la première d'intégrer le raisonnement.

- **Le document comme médium** : selon cette dimension, le document est considéré comme un élément tangible d'une communication entre des personnes humaines. La question du statut du document dans les relations sociales est alors posée. Le document doit être diffusé (dépasser le cercle de l'intime) pour être légitime et cette légitimité doit s'affranchir de l'éphémère (dépasser le moment de son énonciation) et donc être enregistrée, inscrite.

Finalement, nous adoptons pleinement la définition proposée par [Bringay, 2003] qui résume les contributions de [Pédauque, 2003], [Buckland, 1997], [Buckland, 1998] et [Fernández and Arroyo, 1982] :

> « *Un document est un ensemble de traces, inscrites par un (ou plusieurs) auteur(s) sur un support, et tel que les informations soient structurées suivant des règles imposées par le support d'inscription* (dimension 1 : le document comme une forme), *et suivant des règles qui font sens pour l'auteur et le destinataire. Le document est source d'information pour ces deux acteurs, qui vont interpréter cet ensemble de traces (parfois de manière différente) après l'avoir perçu* (dimension 2 : le document comme un signe). *En général le document prend place dans une relation sociale entre plusieurs personnes, pour légitimer une communication* (dimension 3 : le document comme une relation). »

Nous retenons également l'importance de la structure logique d'un document, qui permet de le décrire indépendamment de sa présentation finale à l'utilisateur (la structure matérielle). Comme nous le verrons dans la suite de ce mémoire, la description de ce niveau logique pour les documents audiovisuels a été l'objet d'une partie de notre travail.

1.1.2 Le document numérique

La fabrication d'un document numérique peut tout d'abord découler de la numérisation d'un document « traditionnel ». C'est par exemple le fichier informatique résultat du passage du scanner sur un document inscrit sur un support papier, ou issu de la numérisation d'archives de télévision à des fins de sauvegarde et de pérennisation. Mais un document peut également être directement pensé, écrit et émis en numérique, par exemple à partir d'un logiciel informatique. Dans tous les cas, la notion de document numérique apparaît quand le support matériel de l'expression devient numérique ou informatique.

B. Bachimont souligne dans [Bachimont, 1999] que le document numérique possède les caractéristiques d'un document papier traditionnel mais qu'il partage surtout les cinq spécificités suivantes en commun avec, entre autre, le document audiovisuel :

- Le *support d'enregistrement du document* : le document numérique est codé par une structure discrète.
- La *forme d'enregistrement* : l'enregistrement numérique se fait selon un format de codage, qui permet d'établir la correspondance entre le document tel qu'il doit être consulté ou lu, et sa structure interne.
- Le *support d'appropriation du document* : l'enregistrement numérique est en mémoire et ne peut pas être accédé directement. Il faut donc qu'un calcul produise à partir de cet enregistrement une représentation lisible et intelligible pour un utilisateur sur un autre support. Celui-ci est nommé support d'appropriation et sera par exemple un écran, des hauts parleurs ou encore le papier.
- La *forme d'appropriation du document* : la représentation affichée sur le support d'appropriation respecte une structure ou une forme telle qu'elle est directement intelligible par l'utilisateur. Par exemple, il a appris à lire, et la forme alphabétique d'affichage du document est une forme d'appropriation intelligible pour lui.
- La *modalité d'appropriation* : la forme d'appropriation s'adresse à une ou des modalités perceptives (le son, l'image...).

Mais comme le souligne M. Buckland dans [Buckland, 1998], s'il est aisé de dire qu'un courrier électronique ou un rapport technique généré par un traitement de texte sont des documents électroniques, le concept de document devient lui beaucoup plus obscur au-delà de ces simples exemples. En particulier, le code source d'un logiciel ou le système d'exploitation d'un ordinateur sont-ils des documents ?

Nous avons insisté dans la section précédente sur le fait qu'un document était une structure logique mise en forme matériellement. B. Bachimont précise alors que [Bachimont, 1999] :

> « *Le document correspond nécessairement à ce qui est consulté dans le cadre d'une forme d'appropriation sur un support d'appropriation. Il ne correspond pas à l'enregistrement interne, contrairement à ce que la locution de « document numérique » pourrait laisser entendre. Cela implique qu'un document numérique n'est un document que lorsqu'il est consulté sur un support d'appropriation à travers une forme d'appropriation.* »

Le document numérique doit donc correspondre à ce qui est perçu par l'utilisateur grâce à un périphérique (écran, enceinte, imprimante). Les fragments d'un fichier stockés sur plusieurs pages d'un disque dur d'ordinateur ne sont pas des documents, mais l'impression ou l'affichage du fichier dans une interface, une fois les opérations de reconstitution des différents fragments effectuées, en est un.

Le travail de réflexion mené au sein du RTP 33 du département STIC du CNRS a également pour objectif de proposer une interprétation de l'évolution des points de vue dans le passage du document traditionnel au document numérique [Pédauque, 2003]. Les trois dimensions vues précédemment (forme, signe et médium) constituent à ce titre les catégories dominantes abordées

par l'analyse, ou plus exactement les entrées à partir desquelles l'objet de recherche « document » est abordé afin d'en construire progressivement une définition.

- **Le document comme forme** : lors du passage au numérique, la notion de support perd de son importance et l'inscription sur un support relève désormais du codage. Les informaticiens ont alors cherché à isoler les éléments logiques qui constituent cette dimension du document, pour les modéliser, automatiser les opérations et réagencer les différents éléments ainsi perfectionnés. Selon cette dimension, le document numérique est donc un ensemble de données organisées selon une structure stable et auquel on peut associer des règles de mise en forme pour permettre sa lisibilité. C'est pas exemple un document XML qui va exhiber la structure du document et auquel on va associer des feuilles de style pour le présenter.

- **Le document comme signe** : comme précédemment, les informaticiens ont cherché à isoler les éléments logiques pour les modéliser, mais pour s'attaquer cette fois directement au contenu des documents. Le sens est ainsi remplacé par la notion de « connaissances » exploitables par la machine. Cette tendance rejoint la vision du Web Sémantique [Berners-Lee *et al.*, 2001], une évolution du Web actuel où les machines auraient accès au sens des informations ce qui permettrait d'améliorer les échanges et les traitements des données.

- **Le document comme médium** : le passage au numérique accroît finalement considérablement les facilités de contrôle et de manipulation de l'information. Celle-ci devient de plus en plus ré-utilisable et le nombre de documents disponibles se multiplie exponentiellement.

L'analyse que nous venons de présenter permet de dresser les contours des nouvelles possibilités offertes par le document numérique. Outre l'accès direct à une quelconque partie du document, ou sa très grande facilité de transmission qui permet un accès à distance, nous retiendrons la possibilité d'instrumenter sa description. La structure logique des documents représentée dans les descriptions peut désormais être utilisée par le système de lecture et offrir ainsi une navigation personnalisée ou des mécanismes de reconstruction dynamique de documents. De la connaissance formalisée peut être représentée et intégrée dans les descriptions. Le système documentaire peut alors effectuer des raisonnements sur celle-ci et améliorer la recherche d'information.

Comme nous le verrons par la suite, le choix de bons langages pour représenter et intégrer dans les descriptions de documents audiovisuels ces deux niveaux – *structure logique* et *connaissance formalisée* – constitue une partie essentielle de notre travail.

1.1.3 Qu'est ce qu'une collection ?

Aux notions de document et de document numérique, il convient d'ajouter celle de *collection de documents*. En effet, les documents sont généralement assemblés en des ensembles documentaires pour répondre à des objectifs pratiques déterminés [Bachimont, 1999]. Une collection est donc un « *ensemble unifié et cohérent de documents* » [Belaïd *et al.*, 2002]. En général, deux types de critères permettent alors de regrouper les documents :

- ceux relatifs à la *forme* des documents (par exemple, tous les documents sonores archivés par Radio France) ;

- ceux relatifs au *sens* des documents (par exemple, toutes les intrigues policières télévisées et conservées par l'INA).

Les centres d'archives doivent gérer des collections pour décrire leur fonds car celles-ci permettent de fixer un contexte dans le cadre d'un usage pour exploiter les documents. La notion de collection est même parfois encore plus restrictive puisqu'elle peut conditionner la description des émissions rentrant dans la collection. Ainsi, le manuel des bonnes pratiques d'indexation donné aux documentalistes de l'INA définit la collection comme étant un « *ensemble d'émissions ayant en commun un titre, une thématique, au moins un auteur (en général le producteur), une origine, et donc chaque numéro est diffusable indépendamment* » [Pichon, 1996]. La collection est alors vue comme une suite d'émissions, dont le nombre n'est pas fixé *a priori*, et dont chaque épisode est dit clos, c'est-à-dire qu'il forme un tout cohérent et diffusable isolément. Par exemple, les journaux télévisés, les émissions de variétés comme `Taratata` et les magazines comme `Envoyé spécial` ou `La Marche du siècle` sont des collections.

L'idée de produire des descriptions au niveau de la collection présente un sérieux avantage dès lors qu'il faut décrire chacun de ses constituants : elle permet de factoriser et de réunir dans un même lieu l'ensemble des informations communes à tous les éléments de la collection. Ainsi, l'INA produit pour chacune des collections ce qu'elle appelle une *fiche collection*. Celle-ci se compose d'une notice documentaire, d'un enregistrement du son du générique et d'un fichier d'imagettes provenant de l'émission dans le cas de la télévision. La notice documentaire n'est pas le résultat de l'addition des contenus et des genres de toutes les émissions de la suite mais s'attache à la description de ses constantes thématiques et formelles, de la scénographie ou encore du dispositif technique mis en place. Elle peut être éventuellement complétée lors de chaque changement significatif de générique, d'organisation ou de contenu de l'émission. La fiche collection sert ainsi de fiche d'identité et de mode d'emploi à la collection. Le tableau A.1 page 173 montre par exemple un large extrait de la notice documentaire correspondant à la fiche collection de l'émission `Stade2` diffusée à la télévision depuis 1975.

La lecture de cette notice nous apporte donc de précieuses informations qui sont communes à l'ensemble des émissions appartenant à la collection. Ces informations résument le principe de l'émission et son déroulement chronologique, mais renseignent également sur tous les dispositifs techniques mis en place : le nombre et la situation des caméras, l'organisation spatiale des protagonistes, le décor et le mobilier utilisé, l'ambiance générale... De plus, certaines séquences particulières de l'émission (par exemple, le générique de début et de fin ou les duplex) sont finement décrites et indiquent par avance comment l'image pourra être divisée et quelle sera alors sa disposition. Nous pensons que ces informations sont une ressource inestimable pour tout ce qui concerne l'analyse automatique de la vidéo, car elles sont à même de piloter les algorithmes d'analyse. Les travaux de J. Carrive montrent par exemple qu'il est possible d'exploiter une fiche collection pour reconnaître automatiquement les séquences caractéristiques des émissions, tel que les reportages ou les séquences plateau, pour pouvoir segmenter la vidéo [Carrive, 2000].

1.2 Le document audiovisuel numérique

Après avoir évoqué la notion de document et de document numérique, intéressons-nous aux documents numériques audiovisuels puisque ceux-ci sont au centre de nos préoccupations. Nous allons tout d'abord caractériser ce que nous entendons par audiovisuel avant d'aborder ses spécificités et les difficultés que celles-ci entraînent dès lors qu'il faudra exploiter des documents audiovisuels. En effet, leur exploitation nécessite une interprétation de leur contenu, la plus systématique possible, et l'instrumentation de cette interprétation, bref, une *indexation* des documents audiovisuels. Nous verrons ensuite comment cette indexation est réalisée aujourd'hui par les documentalistes de l'INA, à travers le cycle de vie du document audiovisuel. L'analyse ainsi

menée nous conduira à constater que les pratiques actuelles de documentation ne permettent pas d'exploiter pleinement les nouvelles possibilités offertes par le numérique.

1.2.1 Les spécificités du document audiovisuel

Un document audiovisuel est composé d'images animées et de sons. Mais si, techniquement parlant, il peut être seulement perçu comme une superposition de flux, il se réduit pourtant pas qu'à ça. Comme le souligne Y. Prié, le document audiovisuel est surtout une composition d'éléments pour lesquels son auteur a mis en place des contraintes d'interprétation en obéissant à des règles [Prié, 1999].

L'explosion du nombre de documents électroniques disponibles, en particulier à cause du Web, amène souvent une confusion entre documents audiovisuels et documents multimédia. Cependant, audiovisuel et multimédia ne sont pas synonymes. Selon B. Bachimont, est audiovisuel « un document composé d'images animées et/ou de sons se déroulant de manière linéaire selon un rythme temporel particulier le rendant intelligible pour un lecteur », alors qu'est multimédia « un document mêlant d'une part différents médias et d'autre part construit à partir de différents sous-documents unis par un réseau de liens » [Bachimont, 1999]. Le document multimédia ajoute donc la notion de liens entre les éléments, voire d'interactivité pour l'utilisateur s'il existe une instrumentation de ces liens. On retrouve ici la notion d'hypertexte pressentie par V. Bush [Bush, 1945], mais étendue à tous les types de média [1]. Cette notion d'interactivité est également fondamentale pour Y. Prié qui considère qu'un document audiovisuel est un document multimédia appauvri, mais que si on lui ajoute des possibilités d'interaction – par exemple en liant une table des matières au document – celui-ci devient pleinement multimédia [Prié, 1999].

La nature audiovisuelle des documents appartenant à un fonds d'archive comme celui de l'INA engendre un certain nombre de contraintes sur leur gestion documentaire. En effet, les spécificités de ce type de média, c'est-à-dire le fait qu'il soit composé d'images mais également son caractère temporel, entraînent des difficultés quand à sa description.

Un document audiovisuel est composé d'images. Mais comme le souligne B. Bachimont, « l'image est une forme sémiotique particulière dans la mesure où la forme signifiante n'est pas arbitraire par rapport au contenu signifié » [Bachimont, 1999]. Elle fait appel le plus souvent à des signes concrets iconiques renvoyant à des objets du monde réel. C. Metz parle alors du caractère *analogique* des images [Metz, 1968]. L'image fonctionne donc comme une pseudo-réalité puisque le rapport au monde relève du domaine de l'analogie, mais elle ne prescrit pas son interprétation. Bien sûr, un premier niveau passe par la reconnaissance des objets montrés, mais cela ne suffit pas à épuiser toutes ses significations (par exemple, l'enseigne de telle chaîne de restauration rapide pourra symboliser tantôt l'impérialisme américain, tantôt de nouvelles habitudes alimentaires, le *fast food*, voire la « mal bouffe »). Il est alors fréquent d'apposer un commentaire ou une légende aux images, c'est-à-dire une « paraphrase langagière pour gagner l'intelligibilité qui leurs manque » [Bachimont, 1999]. L'image est donc sujette à de très nombreuses interprétations (potentiellement, autant qu'il y a de lecteurs), même s'il ne faut pas non plus faire fi de son contexte de production (voire de diffusion) qui contraint plus ou moins sa signification. Retenons donc qu'une image est nécessairement interprétée par un être humain et que cette interprétation est fortement conditionnée par l'environnement culturel du lecteur.

La deuxième spécificité des documents audiovisuels est d'être des objets temporels. Cette

1. Le terme *hypermédia* est utilisé, mais nous lui préférons le terme générique *multimédia*.

temporalité, par essence, ne se représente pas et ne se stocke pas, ce qui a plusieurs conséquences :

- Une imposition du rythme de lecture du document : il faut une heure pour visualiser une vidéo d'une heure, et si l'information recherchée est à la 10ème minute, alors il faut passer les 9 premières pour la retrouver. Il n'existe donc pas de moyen d'accéder directement à un contenu, sauf à le repérer et à noter explicitement sa localisation pour être capable de s'y reporter directement.

- L'absence d'unité minimale documentaire : comme le souligne B. Bachimont, « la forme audiovisuelle ne mobilise pas d'unités répertoriées composant le contenu (comme les lettres de l'alphabet) » [Bachimont, 2000b]. Le document audiovisuel exhibe certes plusieurs unités signifiantes (l'image, le plan ou la séquence) mais aucunes d'entre elles ne peut prétendre être une unité documentaire définie *a priori*.

L'image est une unité signifiante proposée par le système technique, a l'instar des mots pour la langue, mais réduire l'analyse d'un document audiovisuel à l'analyse des images le constituant, c'est ignorer la temporalité du document. Le plan est défini comme étant « *une portion de film impressionnée par la caméra entre le début et la fin d'une prise ; sur un film fini, le plan est limité par les collures qui le lient aux plans précédent et suivant* » [Vanoye and Goliot-Lété, 2001]. Les ruptures de plan correspondent donc aux blancs de l'imprimerie. Ils sont les unités manipulées lors de l'opération de montage, mais leur interprétation, sortie de ce contexte, peut remettre en cause leur pertinence. L'expérience du réalisateur russe Koulechov [2] montre qu'un plan ne signifie pas grand chose en lui même, et que la majeure partie de sa signification provient de son entour visuel et sonore. La séquence, que nous définissons comme « *un ensemble de plans constituant une unité narrative définie selon l'unité de lieu ou d'action* » [Vanoye and Goliot-Lété, 2001], fait généralement disparaître la perception du plan chez le spectateur. En tant qu'unité narrative, elle est la plus à même de correspondre à une unité documentaire. Mais il arrive parfois que le segment temporel obtenu soit trop large. Ainsi, le *plan-séquence*, qui correspond à la réalisation d'une séquence en un seul plan, doit généralement être découpé en unités de sens inférieur pour être analysé.

En conclusion, nous constatons donc que la temporalité des contenus audiovisuels modifie sensiblement le problème de la description tel qu'il se pose pour les images. Il faut tout d'abord repérer explicitement dans le document des unités signifiantes ou des zones d'intérêt, une opération que nous appellerons par la suite la *localisation*. Il faut ensuite associer à ces zones, également explicitement, une signification, une opération que nous nommerons par la suite la *caractérisation*. Cette seconde opération est tout aussi délicate que la première à cause de la difficulté à sémiotiser [3] l'image. C'est pourquoi, la plupart des centres d'archives audiovisuels recourent à la description linguistique manuelle pour décrire les contenus. Nous allons donc voir dans la section suivante comment l'INA, en particulier, opère pour décrire son fonds.

2. L'effet Koulechov est l'association de deux plans qui donne un sens. Il doit son nom au réalisateur russe qui en 1923 choisit un gros plan statique qui n'exprimait aucun sentiment du visage de l'acteur Mosjoukine. Il assembla ce plan avec des morceaux de films pour trois montages différents et montra le résultat à des spectateurs qui eurent des réactions très différentes selon le plan qui était associé au visage inexpressif de l'acteur.

3. Nous reprenons ici B. Bachimont qui définit la sémiotisation comme étant « le processus interprétatif selon laquelle une certaine unité est définie comme faisant sens ; cette unité est reconnue fonctionner comme un *signe* et comme telle, posséder une valeur *sémiotique*. » [Bachimont, 1999]. Sémiotiser l'image revient alors à déterminer des unités signifiantes faisant sens dans l'image.

1.2.2 Le cycle de vie du document audiovisuel

Comme nous l'avons vu dans la section précédente, il est d'usage d'avoir recours à la paraphrase textuelle pour décrire le contenu audiovisuel. Mais il faut garder à l'esprit que toute indexation ne sera jamais qu'une interprétation du contenu. Or, comme le souligne B. Bachimont dans [Bachimont, 2000b], la vision consistant à assimiler l'indexation à une interprétation du contenu appelle les remarques suivantes :

- « L'indexation est une interprétation qui est soumise, d'une part, à la subjectivité de l'indexeur, dont on tente de surmonter la variabilité et, d'autre part, à la finalité de la consultation (pour qui indexe-t-on ?) que l'on veut optimiser » ;

- « L'indexation ne rend pas compte des propriétés intrinsèques et objectives d'un document mais exprime un point de vue en fonction d'une application visée des documents (retrouver des documents, pour quoi faire ?). Autrement dit, il n'existe pas d'indexation universelle, ou d'indexation qui soit indépendante d'un contexte ».

Le fait qu'il n'y ait pas d'indexation universelle des documents ne signifie pas pour autant que toutes les interprétations doivent être mises sur le même pied d'égalité. Comme le souligne G. Auffret dans [Auffret, 2000], l'INA est un lecteur particulier qui a une légitimité institutionnelle à proposer une interprétation normalisée du contenu des documents. Cette interprétation n'est pas plus objective qu'une autre, mais elle est contrôlée par un ensemble de règles et d'usages, propres aux métiers des documentalistes, qui contraint la description des documents. Elle est également interprétable hors de son contexte de création car ces règles sont explicites. La relativité de l'indexation est donc maîtrisée même s'il en résulte un manque de souplesse pour exprimer les index. Ces règles et usages sont consignés dans un manuel d'indexation, véritable bible qui accompagne le travail des documentalistes, dont la référence est [Pichon, 1996]. Finalement, la phase d'indexation à l'INA aboutit à la rédaction d'une notice documentaire textuelle dont nous présentons un extrait dans les tableaux 1.1 et 1.2.

Avant de détailler les différents éléments composant ces notices, il convient de revenir sur le cycle de vie des documents audiovisuels. En effet, un des premiers problèmes auxquels ont été confronté les documentalistes concerne l'unité de traitement documentaire à adopter, c'est-à-dire la détermination d'une entité sur laquelle va porter le traitement. La figure 1.1 empruntée à G. Auffret [Auffret, 2000] nous permet de visualiser ce cycle. Ainsi, le producteur, le diffuseur puis l'archive ne vont pas aborder les contenus de la même façon, ce qui pourrait donner lieu à autant de traitements documentaires possibles.

Au commencement, est donc le *programme* audiovisuel, une œuvre fabriquée par une société de production. Selon [Miquel *et al.*, 1990], la production est « *une opération qui consiste à réunir les moyens financiers et techniques pour la réalisation d'un téléfilm, d'une émission de télévision ou de radio ainsi qu'en l'organisation du tournage (constitution du tournage, planning), également appelé pré-production, et le tournage lui-même* ». Du point de vue de sa structure de production (ou encore *structure propre*), le programme peut être de structure simple (homogène) ou complexe, c'est-à-dire composé d'un ou plusieurs éléments autonomes. L'unité documentaire à associer pourrait donc être le programme dans son intégralité lorsqu'il est simple ou ses différents éléments lorsqu'il est composite.

Le programme est ensuite acheté par un diffuseur. Il entre alors dans une logique de programmation qui reflète la politique éditoriale de la chaîne de diffusion. Une nouvelle structure est donnée lors de sa diffusion grâce à un habillage ou à des coupures publicitaires. On parle alors

Titre propre	Stade 2 : [émission du 17 Mars 2002]
Titre collection	Stade 2
Chaîne de diffusion	France 2
Date de diffusion	17.03.2002
Jour	dimanche
Statut de diffusion	Première diffusion
Heure de diffusion	18 :18 :07
Heure de fin de diffusion	19 :13 :05
Durée	00 :54 :58
Thématique	Sports
Genre	Magazine
Type de description	Emission composite
Médiamétrie	Sport, magazine, multisports
Générique	RED, Prudhomme Christian ; REA, Godard Fred ; PRE, Luyat Laurent

Séquences

...

11. [Plateau extérieur invités : Laurent Jalabert et Sandy Casar]
à 18 :37 :56 :00 - 00 :02 :43 :00. - France 2
En direct de Nice, Jean-René GODART interviewe Laurent JALABERT et Sandy CASAR au sujet de la dernière étape de la course cycliste Paris-Nice.

12. [Cyclisme. Paris-Nice, les Français qui brillent]
à 18 :40 :39 :00 - 00 :03 :17 :00. Rodolphe Gaudin. - France 2
Reportage consacré à la course cycliste Paris-Nice. Pour les équipes françaises et les coureurs elle représente un enjeu important en prévision du Tour de France. Commentaires sur images de la course en alternance avec les interviews de Jean-Marie LEBLANC, Didier ROUS et Sandy CASAR.

13. [Plateau : 6ème partie]
à 18 :43 :56 :00 - 00 :09 :06 :00. - Eurosport
Plateau composé de la suite de l'interview en direct de Nice de Sandy CASAR par Jean-René GODART au sujet de la course cycliste Paris-Nice et d'une succession de brèves en images commentées par Alexandre BOYON et Laurent PUYAT.

...

Société de programmes	France 2
Nature de production	Coproduction
Producteurs	Producteur, Paris : France 2, 2002
Doc. d'accompagnement	Conducteur ; Fiche d'émission
Type de traitement	Catalogage analytique
Audience globale	5.20
Audience + 15 ans	5.70
Audience femme	4.30
Audience homme	7.40
Part de marché	21.30
Part de marché +15 ans	22.20
Numéro DL	DL T 20020317 FR2 001.003
Titre matériel	[Journée de programme F2 du 17 mars 2002]

TAB. 1.1 – *Extrait d'une notice documentaire rédigée par l'Inathèque et concernant l'émission* `Stade2` *diffusée le 17 mars 2002*

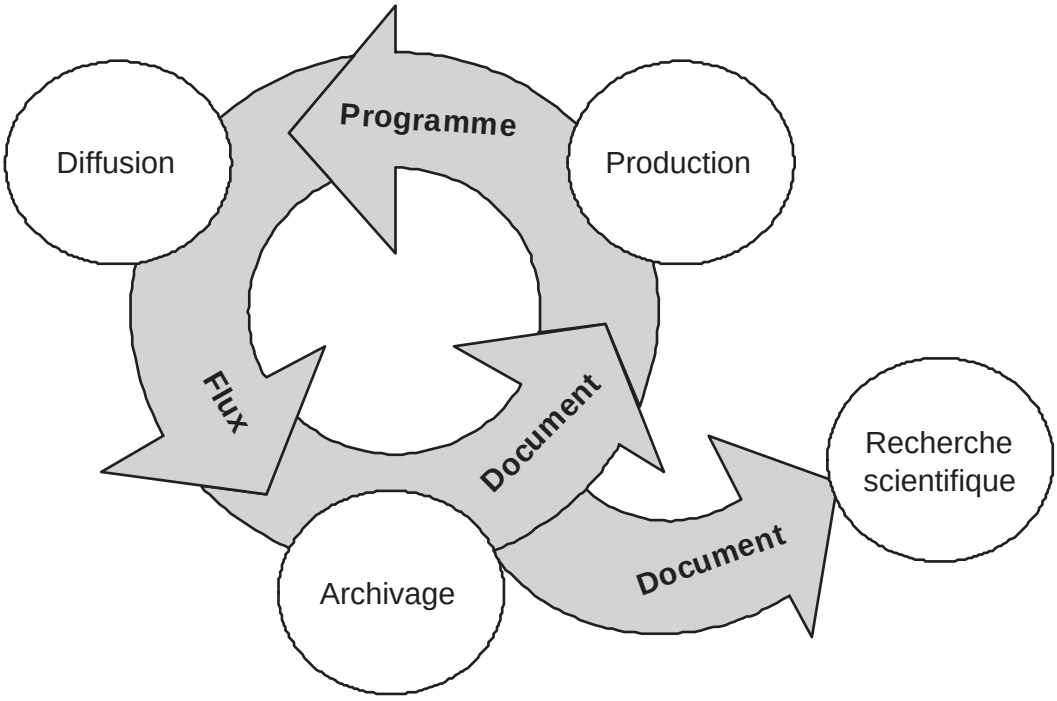


FIG. 1.1 – *Le cycle de vie d'un document audiovisuel d'après [Auffret, 2000]*

d'*émission*. Celle-ci peut appartenir à une tranche horaire caractérisée par une thématique récurrente ou encore à une collection. Ici, le traitement documentaire peut donc s'effectuer au niveau de ces deux « macro-entités » considérées comme des cases dans une grille de programmes.

Finalement, le flux diffusé est capté par l'INA qui s'occupe de le conserver et de l'archiver. On parle alors de *document*. Le document peut alors être consulté à l'Inathèque de France[4] par les chercheurs disposant d'une accréditation, ou ré-injecté dans un nouveau cycle de production : c'est typiquement le cas quand une émission a recours à des images d'archives.

Les documentalistes de l'INA ont tenté de conserver ces différents points de vue (production et diffusion) dans la rédaction des notices documentaires [Troncy, 2001]. Celles-ci sont ainsi hiérarchisée et comprennent deux niveaux de structuration : un document dit de *synthèse* et des *sujets de sommaire*. Le tableau 1.1 présente un extrait d'une notice rédigée par l'Inathèque et concernant l'émission `Stade2` diffusée le 17 mars 2002.

– Le document de synthèse sert à décrire un programme dans sa globalité. Les informations concernant l'unité sont réparties, par catégorie, dans des zones formatées appelées champs. Il comprend entre autre : un titre, une heure de début et une heure de fin de diffusion, le mode de diffusion (*direct, différé, duplex*...), sa périodicité (*quotidien, week-end, hebdomadaire, mensuel, annuel*...), une mention précisant la sélection Dépôt Légal ou non, un type de traitement (qui précise la finesse des descriptions), un type de description qui permet de préciser la structure de l'unité au niveau de la production ou de la diffusion, et des typologies.

4. La consultation des documents audiovisuels conservés au titre du Dépôt Légal s'effectue à la Bibliothèque Nationale de France (`http://www.bnf.fr`), sur le site de Tolbiac à Paris.

– Les sujets de sommaire permettent de décliner les différents éléments constituant le programme. Selon sa structure propre, il pourra s'agir des éléments composant une émission composite ou une tranche horaire. Selon sa structure de diffusion, il s'agira de référencer les interprogrammes[5], les éléments d'habillage de l'émission (séquences de présentation ou de lancement, séquences de conclusion) ou encore les ruptures dues à un incident technique. Depuis peu, les sujets de sommaire peuvent eux-mêmes faire l'objet d'une nouvelle notice documentaire chargée de décrire finement le contenu de la séquence. Le tableau 1.2 présente justement un extrait d'une telle notice mais cette fois rédigée par le Département des Archives. La notice concerne la séquence numérotée 13 dans l'émission `Stade2` diffusée le 17 mars 2002.

Titre propre	[Plateau : 6ème partie]
Titre collection	Stade 2
Générique	PAR, Ravva Vittoria
Descripteurs thématiques	volley ball ; Coupe d'Europe (Ligue des champions) ; basket ball ; handball ; biathlon ; Jeux paralympiques (d'hiver, 2002)
Diffusion	17.03.2002 (D), 18H 43MIN 56SEC, 2 (A2), N
Durée	00H 09MIN 06SEC
Producteurs	PRD, Paris : Pathé Sport (PATSP), 2002 ; PRD, Paris : Eurosport (EUSPO), 2002
Nature de production	ACHAT DE DROITS DE DIFFUSION
Forme	Brève

Résumé

• Plateau composé de la suite de l'interview en direct de Nice de Sandy CASAR par Jean-René GODART au sujet de la course cycliste Paris-Nice et d'une succession de brèves en images commentées par Alexandre BOYON et Laurent PUYAT.
• Volley ball. Ligue des champions féminine. DP Commentaires sur images factuelles du match entre Cannes et Bergame, suivi d'une interview de Vittoria RAVVA, joueuse de Cannes ITW.
• Volley ball. Ligue des champions homme. DP Commentaires sur images factuelles du match entre Tourcoing et Montpellier.
• Basket ball. 21ème journée Pro A. DP Commentaires sur images factuelles du match Châlon-Dijon - Hand ball. Commentaires sur images factuelles du match Istres Chambéry.
• Biathlon. Poursuite 12,5 kms. GP RAL Commentaires sur images factuelles de la victoire du français Raphaël Poiree en Finlande.
• Jeux paralympiques d'hiver de Salt Lake City. DP Commentaires sur images factuelles de la victoire de Denis Barbet à l'épreuve de la descente en ski assis.

Lien notice principale	1973901001

TAB. 1.2 – *Extrait d'une notice documentaire rédigée par les Archives et concernant une séquence de l'émission `Stade2` diffusée le 17 mars 2002*

Il paraît intéressant ici de s'arrêter sur les méthodes de construction des notices, de l'Inathèque d'une part et du Département des Archives d'autre part. Tout d'abord, on peut noter que le

5. Tout ce qui constitue l'habillage d'une chaîne : bandes annonces, espaces publicitaires, jingles...

document de synthèse contient globalement moins d'information dans la notice provenant du Département des Archives, et que l'intitulé de certains champs a même parfois été changé. Mais la différence la plus notable réside dans l'emploi de toute une série d'abréviation dans la description des sujets de sommaire afin d'indiquer la mise en images des séquences. Ainsi, pour chaque séquence, on pourra trouver l'information concernant le type de plan utilisé (GP = gros plan, DP = divers plan), ou concernant le mouvement de caméra (RAL = ralenti). Finalement, les notices issues de ces deux départements sont différentes (bien que décrivant la même émission), mais ceci s'explique par le fait qu'elles ne visent pas le même public. Le Département des Archives vise les professionnels du secteur audiovisuel alors que pour l'Inathèque, les utilisateurs finaux sont des chercheurs qui seront, par exemple, peut être plus intéressés par les indices d'écoute des émissions.

Enfin, comme nous l'avons vu dans la section 1.1.3, la collection fait aussi l'objet d'une notice documentaire appelée *fiche mère* (voir le tableau A.1).

1.2.3 L'indexation de documents numériques

Comme nous l'avons vu plus haut, l'accès au contenu audiovisuel repose en grande partie sur la documentation qui l'accompagne, la notice documentaire pour l'INA. La numérisation des contenus amène de nouvelles possibilités comme nous l'avons entr'aperçu dans l'introduction (voir page 1). Or, comme le fait remarquer B. Bachimont, le numérique, comme système technique, reconfigure les relations entre les documents audiovisuels et leur description. En particulier, il permet la mise en œuvre de quatre principes fondamentaux pour la documentation [Bachimont, 1999] :

- « *l'intégration de la documentation* » : il est possible d'intégrer différents niveaux au sein d'une même description structurée, par exemple lier la description du contenu avec celle des droits attachés au contenu ;
- « *l'alignement de la documentation* » : la description peut être alignée avec le flux audiovisuel, établissant ainsi une correspondance entre tout segment ou structure de la description documentaire, et un segment audiovisuel ;
- « *l'intégration de la chaîne documentaire* » : la documentation peut mieux rendre compte des étapes du cycle de vie du document audiovisuel en intégrant les éléments techniques composant ces moments (documents de production, grille des programmes du diffuseur) ;
- « *l'intégration de la documentation et des contenus* » : la description documentaire devient un document à part entière, consultable pour lui-même, mais qui permet d'accéder directement au document audiovisuel qu'elle décrit.

Ainsi, les notices documentaires, qui avaient anticipé l'irruption du numérique, proposent déjà la localisation temporelle des différents sujets de sommaire composant une émission. Par exemple, on remarquera dans le tableau 1.1 que chaque séquence numérotée contient un repère temporel (on dit encore *timecode*) indiquant le début de la séquence et sa durée. L'instrumentation de ces repères permet alors de visualiser directement la séquence en cliquant sur sa description. Cependant, on peut regretter que la description détaillée de chacune des séquences ne soit pas elle même plus structurée et *timecodée* (voir le tableau 1.2), pour des problèmes de coût de fabrication. En fait, les pratiques documentaires de l'INA établies dans un contexte analogique ont simplement été transposées lors du passage au numérique. L'accès direct aux sujets de sommaire des émissions est incontestablement un progrès, mais il est encore loin de bénéficier de toutes les nouvelles possibilités offertes par le numérique.

Le faible niveau de structuration des notices actuelles (l'émission dans sa globalité et ses sujets de sommaire) est à l'origine d'un certain nombre de problèmes. Pour les illustrer, reprenons l'exemple de la notice documentaire de l'émission `Stade2` donnée dans le tableau 1.1, et en particulier les séquences numérotées 11, 12 et 13.

Imaginons qu'une personne souhaite retrouver *les séquences audiovisuelles concernant la course cycliste Paris-Nice où Sandy Casar est interviewé*. Une recherche en « texte intégral » effectuée sur cette notice documentaire avec les mots clés `Paris-Nice`, `Sandy Casar` et `interview` va bien retourner les trois séquences numérotées 11, 12 et 13, mais des limitations apparaissent :

1. Les réponses données ne correspondent pas exactement à la requête puisque, par exemple, la séquence numérotée 13 contient aussi un certain nombre de brèves qui concernent d'autres sports que le cyclisme. Le bruit qui est engendré peut être particulièrement gênant dès lors que l'on souhaite reconstruire automatiquement un nouveau document multimédia par mise bout-à-bout des séquences retournées. Le problème vient ici du modèle documentaire qui n'autorise pas plus de deux niveaux de structuration et qui est donc incapable d'isoler une `interview` dans une `séquence` particulière pour une `émission` donnée.

2. Les réponses données peuvent paraître incomplètes puisque les séquences 11 et 13 sont en fait la même `interview` mais diffusées en deux parties avec un `reportage` entre les deux. Le problème vient encore du modèle documentaire puisque celui-ci ne permet pas de gérer des éléments discontinus dans le temps.

3. Enfin, la recherche de séquences n'est pas généralisable puisque si la requête avait été de retrouver *toutes les séquences audiovisuelles où un coureur cycliste donne une interview dans le cadre d'une course à étapes*, la recherche en texte intégral n'aurait donné aucun résultat.

Cet exemple fait ressortir l'importance de la modélisation de la structure logique et conceptuelle des documents audiovisuels. Il illustre bien la nécessité d'avoir un bon modèle documentaire pour représenter cette structure logique. Mais cette structure contenant de la connaissance, il est également indispensable de l'expliciter, surtout si l'on souhaite conduire des raisonnements dessus. Nous pouvons donc commencer à dresser les contours de ce que serait un système d'information documentaire audiovisuel non pas idéal, mais exploitant mieux les nouvelles possibilités introduites par le numérique. Ce système devrait être capable :

- d'exprimer des modèles capables de contraindre la structure logique d'une description (par exemple, identifier une `interview` à l'intérieur d'un `reportage` particulier d'un `magazine sportif` donné ;

- de représenter la sémantique de cette structure pour que celle-ci soit accessible à la machine qui pourra conduire des raisonnements (par exemple, déduire qu'un `dessin animé` est une `fiction` sans personnages réels) ;

- de représenter la sémantique du contenu proprement dit de chaque séquence, toujours dans un langage formel pour être exploitée par la machine (par exemple, déduire que dans le domaine du cyclisme, le `prologue` est toujours un `contre la montre individuel`, étape numérotée 0 d'une `course à étapes`.

On peut noter ici que la problématique soulevée se retrouve dans celle posée par le *Web Sémantique* [Berners-Lee *et al.*, 2001] puisque celui-ci contient d'une part de plus en plus de documents

multimédias, et d'autre part parce qu'il ambitionne de rendre accessible à la machine la sémantique des documents publiés sur le Web. Le Web étant constitué de ressources, la technique généralement employée consiste à annoter des portions de ces ressources avec des informations exploitables par des agents logiciels. Nous allons donc définir dans la section suivante ce que recouvre la notion d'annotation.

1.3 De l'annotation à la description des documents audiovisuels

Nous nous sommes attachés jusqu'à présent à étudier ce que l'arrivée du numérique dans l'audiovisuel allait apporter en termes de nouvelles fonctionnalités. En particulier, nous avons souligné l'importance du modèle de représentation des documents audiovisuels et détaillé celui actuellement suivi à l'INA. Nous avons vu que l'*annotation* d'un document audiovisuel, c'est-à-dire le fait d'attacher une description à l'une de ses parties, était le concept de base de cette représentation. Nous allons donc maintenant définir ce que l'on entend par annotation – d'une manière générale puis appliquée à l'audiovisuel –, comment elle peut être obtenue et quelles sont ses caractéristiques. Nous aborderons alors le cas particulier des annotations obtenues automatiquement par extraction des caractéristiques d'un flux audiovisuel, afin de clarifier ce que l'on entend par *contenu* d'un document audiovisuel.

1.3.1 Qu'est-ce qu'une annotation ?

Comme le rappelle [Prié and Garlatti, 2003], « une annotation est à la base une note critique ou explicative accompagnant un texte, et par extension, une quelconque marque de lecture portée sur un document, que celui-ci soit textuel ou image ».

Dans [Marshall, 1998], C. C. Marshall prend comme point de départ l'annotation comme une activité fondamentale d'un lecteur de documents. Elle propose alors une caractérisation des annotations selon plusieurs dimensions avant d'aborder les annotations comme une activité individuelle, puis dans une perspective plus large où celles-ci vont bénéficier à toute une communauté de lecteurs. Ces dimensions décrivent les propriétés de l'annotation au niveau de sa forme, de sa fonction et de son rôle dans la communication avec les autres.

- Une première dimension reflète la forme que peut prendre l'annotation : celle-ci peut être de très formelle et structurée (puisque basée sur une formalisation rigoureuse) à informelle (par exemple, les notes manuscrites apposées dans la marge d'un ouvrage sur support papier) ; mais elle peut également être tacite, c'est-à-dire plutôt destinée à son auteur, ou explicite, c'est-à-dire compréhensible pour la plupart des lecteurs (par exemple, les notes de bas de page).

- Une autre dimension révèle les fonctions de l'annotation pour le lecteur : ces fonctions indiquent dans quelle mesure l'annotation peut devenir elle-même une écriture, si l'annotation permettra une lecture superficielle du document (un ensemble d'hyperliens autorisant une lecture rapide) ou au contraire approfondie (l'annotation est alors vraiment inscrite dans le texte et utilise généralement une typographie spéciale), et la « durée de vie » de l'annotation, c'est-à-dire si celle-ci sera permanente ou éphémère.

- Une dernière dimension concerne le rôle de l'annotation dans la communication avec les autres, autrement dit, le public potentiel de l'annotation : celle-ci peut en effet être privée, ou publique et destinée alors à un groupe restreint, à une institution ou au monde entier.

En partant de l'analyse effectuée par C. C. Marshall, [Desmontils and Jacquin, 2002] abordent eux aussi le cas spécifique des *annotations sémantiques* qui ont comme objectif majeur « de *désambiguïser* le document pour un traitement automatique ». Les auteurs proposent alors de décomposer le processus général d'annotation par des ontologies en trois phases :

- *Repérer* : placer dans le document des références à des concepts de l'ontologie, de manière manuelle ou automatique ;
- *Instancier* : valuer les attributs des concepts à l'aide des informations présentes dans le document, de manière manuelle ou automatique ;
- *Enrichir* : ajouter des informations par l'intermédiaire des attributs de concepts qui n'ont pas pu être valués à la phase précédente, de manière manuelle.

Selon ces mêmes auteurs, les annotations sémantiques doivent donc être directement insérées dans le code source du document. Cette affirmation est largement discutable car, dans le cas des documents textuels publiés sur le Web par exemple, il est désormais facile de référencer une quelconque partie de document avec des expressions de type XPointer [XPointer, 2001] ou XPath [XPath, 1999], et qui plus est si le document est fortement structuré. Dans le cas des documents audiovisuels, il est généralement extrêmement difficile, et souvent non souhaitable, de modifier le document. La solution que nous avons adoptée, comme nous le verrons plus tard, a donc consisté à séparer les descriptions des documents décrits.

Pour [Euzenat, 2002], la connaissance représentée dans les annotations est forcément liée au type d'application utilisant le contenu des documents. Il donne alors une classification alternative des annotations selon leur statut : il y a tout d'abord les données média (encodage, format, date de création...), puis les métadonnées (auteur, date de production...), les descripteurs de contenu (mots clés, catégories...) et finalement la représentation du contenu (résumé). Pour notre part, il s'agit d'une représentation du contenu dans un langage formel que l'on souhaite mettre en œuvre pour décrire les documents audiovisuels.

Nous pouvons remarquer à ce stade que la littérature utilise généralement deux termes pour qualifier le fait d'associer une information à une ressource : *annotation* et *métadonnée*. La communauté anglophone du Web Sémantique semble utiliser indifféremment ces deux termes puisqu'ils concourent à rendre le Web plus utilisable par les machines. Par exemple, [Bechhofer *et al.*, 2002] considère que l'annotation n'est qu'un moyen particulier d'associer une métadonnée à une ressource sur le Web, donnant ainsi une primauté à la fonction.

[Prié and Garlatti, 2003] propose pourtant de distinguer l'annotation et la métadonnée en s'appuyant sur le processus de leur mise en place. Ainsi, il propose le distinguo suivant :

- « une métadonnée sera plutôt attachée à une ressource identifiée en tant que telle sur le Web, aura plutôt une pertinence *a priori* et sera plutôt saisie suivant un schéma » ;
- « une annotation sera plus *située* au sein de cette ressource et *écrite* au cours d'un processus d'annotation/lecture ».

Pour reprendre les exemples donnés précédemment, l'annotation sera donc souvent un commentaire libre et informel alors que la métadonnée sera, elle, formelle, structurée et indépendante (voire ressource en tant que telle), bref *sémantique* pour la machine.

La description des documents audiovisuels, telle qu'elle est réalisée aujourd'hui à l'INA et comme nous l'avons vu dans la section 1.2.2 (tableaux 1.1 et 1.2), se situe sans doute au milieu de ces deux situations extrêmes. En effet, les notices documentaires sont structurées et elles

contiennent un certain nombre de champs formatés qui peuvent être contrôlés par un thésaurus. Mais cette structure est relativement limitée puisque restreinte à seulement deux niveaux (l'émission et ses séquences). De plus, les notices contiennent de larges portions rédigées en texte libre par les documentalistes et correspondant aux résumés des séquences. C'est donc vers une plus grande formalisation des descriptions, et notamment de la partie s'attachant au contenu proprement dit, que nous souhaitons aller, afin que ces descriptions soient utilisables par la machine.

1.3.2 L'extraction automatique de caractéristiques dans les documents audiovisuels

Le terme « contenu » (*content* en anglais) est fréquent dans les titres des articles consacrés aux systèmes de recherche d'informations audiovisuels. Mais chez de nombreux auteurs, pour la plupart ceux de la communauté du traitement du signal, ce terme ne regroupe que les caractéristiques (*features* en anglais) extraites automatiquement ou calculables à partir du signal audiovisuel. Nul doute que ces recherches basées uniquement sur de l'analyse automatique concourent à la description des documents audiovisuels. Elles peuvent concerner la segmentation des documents audiovisuels, la détection et la transcription de la parole (voire la reconnaissance des locuteurs), le calcul de points d'intérêts dans l'image ou encore la détection et la reconnaissance de textes incrustés dans les documents. Nous présentons brièvement ci-après quelques résultats apportés par ces recherches.

Détection de scènes Comme nous l'avons déjà dit, le plan est l'unité de montage vidéo. Les chercheurs ont donc assez naturellement cherché à les détecter. La technique utilisée consiste alors à calculer les ruptures de plan à l'aide de caractéristiques globales telle que le mouvement, la couleur ou les contours. Mais, nous l'avons vu, le plan n'est que très rarement l'unité d'un traitement documentaire, puisque les documentalistes s'attachent plutôt à décrire des scènes. Quand on sait qu'elles passent près de 30% du temps de documentation à *segmenter* le flux, on comprend mieux les raisons d'automatiser cette tâche, appelée par la suite *macro-segmentation* ou segmentation en séquences. La technique généralement utilisée consiste à segmenter le flux audiovisuel en plans, puis à opérer des regroupements de ces plans selon différents critères.

Pour un certain nombre d'émissions (journaux télévisé et magazines par exemple), il est nécessaire de segmenter le flux en séquences *plateau* et *reportage*. Ce type de découpage s'apparente à un cas particulier de macro-segmentation, mais plus simple dès lors qu'il est réalisé sur une collection d'émissions où, pour tous les numéros, le décor des plateaux, l'habillage des reportages et le présentateur seront identiques.

Travaux sur l'image Les recherches dans le domaine du traitement d'image proposent aujourd'hui de calculer des signatures caractérisant le contenu de manière plus compacte que le contenu lui-même. Ce principe général, dénommé *fingerprinting*, apparaît comme une alternative au principe de *watermarking* dans le domaine de la protection des droits auteurs, avec l'avantage de préserver les œuvres puisque celles-ci ne sont plus tatouées. Il est alors possible de calculer la signature d'extraits diffusés à la volée et de les comparer avec une base de signatures caractérisant un catalogue vidéo de référence. Un tel système permettrait par exemple à un ayant-droit de tracer la diffusion des copies d'extraits fournies aux exploitants. Outre la lutte contre le piratage, il offrirait un retour précis sur l'exploitation des extraits et sur leur intérêt commercial.

Il est également possible de caractériser automatiquement le contenu d'extraits de vidéo, c'est-à-dire d'apposer des descripteurs sémantiques à ces extraits. [Veneau, 2002] propose par

exemple d'analyser le mouvement pour caractériser des documents de sport. Une méthode de classification par apprentissage fondée sur les machines à vecteurs de support (SVM) a ainsi été utilisée et permet d'indiquer le sport (football, cyclisme, patinage. . .) concerné par les documents.

Travaux sur le son Les progrès récents effectués dans l'analyse de la bande audio permettent aujourd'hui de la découper en parole/bruit/musique, de transcrire automatiquement la parole (grâce à des modèles de langage) et même de reconnaître les locuteurs. En particulier, l'état de l'art pour la transcription automatique de documents d'information en français correspond à un taux d'erreurs sur les mots de l'ordre de 20% [Allauzen and Gauvain, 2003].

Travaux sur les textes La détection et la reconnaissance des textes incrustés dans les images peut s'avérer d'un grand intérêt par exemple dans le cas de sous-titres, de bandeaux à l'écran ou de résultat sportifs.

Le *fossé sémantique* Tous ces travaux fournissent indéniablement de précieuses informations pour décrire les documents audiovisuels. Mais ils ont également leur limite, qui est le saut qualitatif à franchir pour proposer des annotations sémantiques à partir des descripteurs numériques extraits du flux. La description manuelle produite par les documentalistes est donc foncièrement différente de que ces outils fournissent. [van Slype, 1987] l'explique par le fait que l'indexation manuelle et la classification ou l'analyse automatique reconnaissent des unités significatives différentes (concepts *vs.* descripteurs), et n'analysent pas la même chose dans le document (analyse critique globale *vs.* classification synthétique locale). [Veneau, 2002] en conclut alors que l'indexation automatique n'est pas une automatisation de l'indexation manuelle, mais peut servir plutôt d'assistance à tout processus de description de documents.

Des travaux tentent cependant de franchir ce *fossé sémantique* (*semantic gap*), c'est-à-dire de proposer une interprétation sémantique à partir d'une analyse automatique effectuée sur un document audiovisuel[6]. L'idée en vogue consiste à combiner les descripteurs numériques extraits à partir du signal (par exemple, audio et vidéo) pour produire du « sémantique » [Bimbo, 2000]. Mais si l'utilisation de descripteurs de nature différente permet effectivement d'améliorer les résultats obtenus pour certaines tâches, ces algorithmes n'ont pas encore fait émerger des outils capables de créer et de manipuler des concepts sémantiques [Veneau, 2002].

En témoigne la conférence TREC (*Text REtrival Conference*) qui organise depuis 3 ans une session dédiée à la recherche de séquences audiovisuelles[7]. Les participants doivent retrouver automatiquement, dans un corpus vidéo de référence, un certain nombre de caractéristiques fixées au départ telles qu'un plan contenant un visage, la distinction ville/campagne, la distinction intérieur/extérieur. . . . En 2002, l'évaluation portait ainsi sur 10 *features* à reconnaître et les meilleurs se situaient – seulement – autour de 60% de bonne reconnaissance. En 2003, l'évaluation a concerné 17 *features* à reconnaître dans un corpus composé de 60 heures de journaux télévisés provenant des chaînes américaines CNN et ABC. La encore, les résultats peuvent paraître décevants et montrent toute la difficulté à extraire du « sémantique » à partir d'une combinaison de descripteurs primitifs.

Finalement, les seules méthodes ayant réussi à franchir ce saut qualitatif semblent être celles qui utilisent de l'information *a priori*, dans un cadre particulier, par la mise en œuvre de règles, de modèles ou d'algorithmes d'apprentissage [Carrive, 2000], [Landais *et al.*, 2004]. Ces méthodes

6. L'Action Spécifique STIC du CNRS intitulée *Modélisation Sémantique et Indexation Multimédia* s'intéresse en particulier à ces questions [Bachimont and Vignaux, 2002].

7. TREC Video : `http://www-nlpir.nist.gov/projects/trecvid/`.

apportent satisfaction dès lors qu'elles sont appliquées dans le cadre restreint pour lesquelles elles ont été conçues. Les critiques régulièrement adressées concernent donc d'une part le manque intrinsèque de généralité de ces méthodes, et d'autre part le travail de modélisation souvent coûteux qui est demandé à l'opérateur humain en préalable.

Le terme « contenu », nous l'avons vu, a donc des acceptions assez différentes dépendamment des communautés qui l'emploient. Pour notre part, cette notion renverra toujours à des concepts sémantiques. Ainsi, la description du contenu des documents audiovisuels tentera de répondre aux questions *qui? quand? quoi? où? comment?*

Nous établissons dans la dernière section de ce chapitre une synthèse de toutes les remarques émises jusqu'à présent afin de proposer une définition du processus de description des documents audiovisuels.

1.4 Synthèse

Dans ce chapitre introductif, nous avons étudié et présenté un certain nombre de notions qui seront utilisées tout au long de ce mémoire : documents, documents numériques, documents audiovisuels, collections, indexation. . . . Nous avons eu l'occasion de souligner l'importance de la structure logique qu'exhibe tout document. Nous avons également montré la difficulté à décrire des documents audiovisuels, en raison de la spécificité même de ce média, et rappelé la solution généralement utilisée, à savoir le recours à la paraphrase textuelle.

La numérisation des documents permet, entre autres, d'instrumenter leur description. Dans le cas des documents audiovisuels, une description fine de la structure va ainsi autoriser de multiples parcours de navigation, ou rendra possible la recomposition dynamique de documents. Il est également possible d'intégrer de la connaissance formalisée dans les descriptions, afin de conduire des raisonnements sur celles-ci et améliorer la recherche de séquences audiovisuelles particulières par le contenu.

Nous avons vu enfin que le processus de description suivi actuellement à l'INA ne permettait pas de profiter pleinement de toutes ces nouvelles possibilités apportées par le numérique. Nous proposons donc de reprendre et d'étendre les pratiques actuelles de documentation. Ainsi, nous considérons que toute description de documents audiovisuels est issue d'un processus, résumé par la Figure 1.2, composé de trois étapes :

- *Identification* : il s'agit de donner des informations de catalogage sur le document (titre, auteur, format, droits. . .) ; on utilise généralement pour cela un ensemble de descripteurs normalisés comme les catégories VRA (*Visual Resources Association*[8]), celles du *Dublin Core*[9] ou son extension pour l'audiovisuel telle que l'a proposée Jane Hunter [Hunter, 1999].

- *Localisation* : il s'agit de repérer dans le document des entités spatio-temporelles d'intérêt qui seront annotées ; on utilise pour cela des coordonnées cartésiennes et des dates pour localiser dans l'espace et dans le temps des segments pertinents.

- *Caractérisation* : il s'agit de donner une signification formelle aux éléments préalablement repérés ; on utilise pour cela un ensemble de descripteurs contrôlés auxquels on a donné une sémantique formelle pour la machine.

8. `http://www.vraweb.org/`.
9. `http://www.dublincore.org/`.

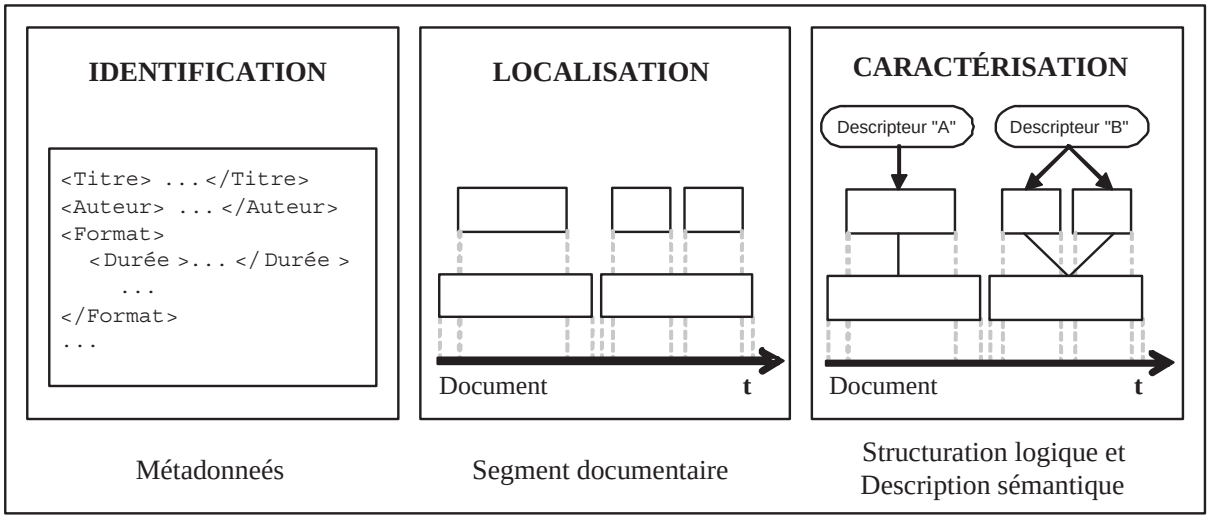


FIG. 1.2 – *Les différentes étapes du processus de description des documents audiovisuels*

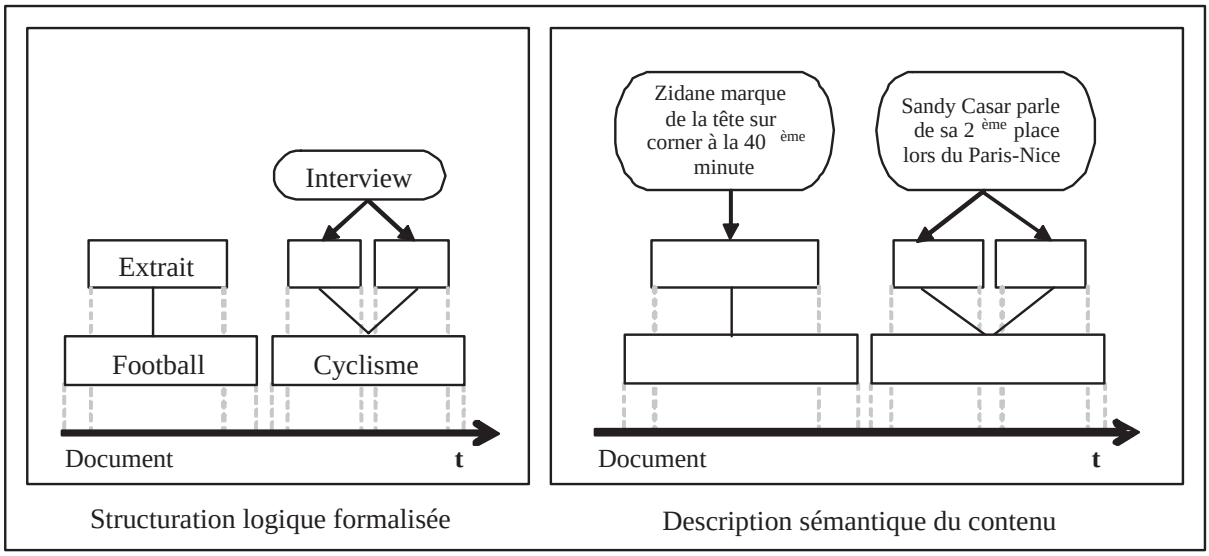


FIG. 1.3 – *Structuration logique et description sémantique du contenu des documents audiovisuels*

La caractérisation des segments documentaires fait appel à deux sortes de descripteurs afin de faire ressortir les deux dimensions que sont la structure et le contenu des documents comme l'illustre la Figure 1.3. Dans un premier temps, chaque segment est typé selon un (ou plusieurs) genre audiovisuel (par exemple, *extrait*, *reportage* ou *fiction*) et une (ou plusieurs) thématique générale (par exemple, *football*, *politique* ou *cinéma*) afin de rendre compte de la structure logique du document. Dans un deuxième temps, une description sémantique du contenu est apposée aux segments (par exemple, la représentation formelle de l'assertion *Zidane marque de la tête sur corner à la 40ème minute*).

Notons enfin que la description de la structure logique du document (et par le même coup le repérage des segments d'intérêt) peut être contraint par avance par des schémas. C'est notamment le cas quand la description concerne une émission particulière rentrant dans le cadre d'une collection (telle que définie dans la sous-section 1.1.3), un cas extrêmement fréquent à l'INA. Le schéma reflète alors la structure générale commune à tous les éléments de la collection, et la description de la structure d'une émission doit être une instance de ce schéma.

Il nous faut désormais nous interroger sur les différentes manières de représenter ces descriptions de documents audiovisuels. Nous allons donc étudier au cours des deux chapitres suivants les langages et les systèmes fournis d'un côté par l'ingénierie documentaire et multimédia, et d'un autre côté par le Web sémantique et la représentation des connaissances.

Chapitre 2

Représenter les documents audiovisuels : une approche documentaire

Sommaire

2.1 L'exemple paradigmatique : la famille XML 32
2.1.1 De SGML à XML 32
2.1.2 Validité d'un document XML 33
2.1.3 La norme HyTime 38
2.2 Des standards pour présenter les documents audiovisuels 39
2.3 Des standards pour décrire les documents audiovisuels 40
2.3.1 La description des données bibliographiques 41
2.3.2 MPEG-7 : le nouveau langage documentaire pour représenter le multimédia 46
2.3.3 TV Anytime 51
2.3.4 Synthèse 52
2.4 Des systèmes opérationnels pour annoter de la vidéo 52
2.4.1 Des systèmes « purement » documentaires 52
2.4.2 Les modèles de représentation orientés « connaissances » 57
2.5 Conclusion 64

L'explosion documentaire, c'est-à-dire la brutale augmentation du nombre de documents dès la fin du 19ème siècle et sans rémission depuis, a conduit à l'élaboration de ce qu'on a appelé les *langages documentaires* (références bibliographiques, index, thésaurus, résumés...) [Fayet-Scribe, 2000]. De longue date, les archivistes ont également collecté des métadonnées sur les documents et leurs producteurs, des données généralement qualifiées de bibliographiques. Comme nous l'avons vu dans le chapitre introductif, les documentalistes utilisent tous ces outils pour décrire les documents audiovisuels.

Dans ce chapitre, nous présentons les modèles et langages issus du monde documentaire qui s'offrent à nous pour décrire des documents audiovisuels numériques. En particulier, nous étudions les approches prenant appui sur le paradigme des « documents structurés » qui trouve sa concrétisation la plus spectaculaire avec le succès manifeste de la norme XML [XML, 2000] (section 2.1). Comme il est habituel de distinguer en ingénierie documentaire le fond de la forme, nous décrivons rapidement quelques standards permettant de présenter des documents multimédias

(section 2.2), avant de nous attarder beaucoup plus longuement sur le langage MPEG-7 [MPEG-7, 2001], le nouveau standard pour décrire les documents multimédias, ainsi que sur ses dérivés (section 2.3). Nous étudions finalement quelques systèmes opérationnels permettant d'annoter de la vidéo et qui intègrent parfois de la connaissance dans les modèles de représentation des documents audiovisuels, autorisant ainsi le calcul d'inférences sur les descriptions (section 2.4). Nous concluons alors ce chapitre sur le constat que les approches « purement » documentaires ne sont pas suffisantes pour décrire la sémantique du contenu des documents.

2.1 L'exemple paradigmatique : la famille XML

Indépendamment de leur représentation informatique, tous les documents possèdent des structures. Par exemple, un roman est souvent structuré en chapitres, un poème en strophes, un document audiovisuel en séquences... On appelle *systèmes de documents structurés* « les systèmes informatiques qui permettent de représenter explicitement et de manipuler les structures des documents » [Brunie, 1999]. Ceux-ci prennent appui sur le paradigme des « documents structurés » qui est apparu dans les années 80, et qui s'est développé avec la création des normes d'encodage numérique du texte telles que *Standard Generalized Markup Language* (SGML) [SGML, 1986].

Dans cette section, nous rappelons brièvement les principes de la norme SGML avant de présenter le langage XML qui est aujourd'hui utilisé dans de très nombreux types d'applications. Nous détaillons ensuite les moyens de contraindre la structure des documents XML avec les DTDs d'une part, et les schémas XML d'autre part qui introduisent la notion de type. Finalement, nous analysons la norme HyTime [HyTime, 1997] qui propose une façon de représenter des documents hypermédias en SGML.

2.1.1 De SGML à XML

Standard Generalized Markup Language [SGML, 1986] fait figure de référence dans le domaine des documents structurés, mais malgré son acronyme, il n'est pas, à proprement parler, un langage mais plutôt un métalangage, c'est-à-dire un formalisme qui permet de définir des langages servant à décrire des documents. Son principe de base repose sur le fait qu'il est possible d'enrichir un document textuel par des balises véhiculant des informations sur la structure et le contenu du document. Chaque langage de balisage, défini en SGML, est ainsi une grammaire de balises que les documents doivent respecter. L'ouvrage de référence de C. Goldfarb [Goldfarb, 1990] commente dans ses moindres détails la norme et est donc incontournable pour celui qui veut utiliser ce langage pleinement.

eXtensible Markup Language (XML) [XML, 2000] est un langage recommandé, lui, par le W3C (*World Wide Web Consortium*). C'est également un langage de balisage qui se présente comme un sous-ensemble de SGML dans la mesure où tout document XML valide est également SGML valide vis-à-vis de la même DTD. Son but est de permettre au SGML générique d'être transmis, reçu et traité sur le Web de la même manière que l'est HTML aujourd'hui. Il est donc délibérément intermédiaire entre HTML et SGML, plus complexe que le premier mais plus simple et moins contraignant que le second [Euzenat, 2000]. Une description complète en français de ce langage se trouve dans [Michard, 1998].

Les documents XML suivent une syntaxe relativement simple qui peut se résumer par :

```
<élément att1="val1" att2="val2" ... attn="valn"/>
```

ou

<élément att$_1$="val$_1$" att$_2$="val$_2$" ... att$_n$="val$_n$"> contenu </élément>

où `élément` est un nom d'élément, `att`$_i$ est un nom d'attribut, `val`$_i$ la valeur de cet attribut, et `contenu` une suite d'expressions XML ou une chaîne de caractères. Un document XML peut donc être vu comme un arbre dont la racine est le document et les fils d'un nœud sont les éléments de son contenu.

Le code 2.1 donne un exemple de document XML [1]. Il représente une émission intitulée `Stade2` et diffusée sur `France2` qui a comme thématique générale `Multisports` et qui est composée de deux séquences. La première séquence a comme thématique le `Football` tandis que la seconde a comme thématique le `Cyclisme` et contient elle-même une autre séquence.

```
<?xml version="1.0" encoding="iso-8859-1" standalone="no" ?>
<!DOCTYPE ina:Emission SYSTEM "emission.dtd">

<ina:Emission xmlns:ina="http://www.ina.fr/schema/"
              nom="Stade2" chaineDiffusion="France2">
  <ina:Typologie theme="Multisports"/>
  <ina:Sequence tcDebut="00H01M32S24F" tcFin="00H19M07S17F">
    <ina:Typologie theme="Football"/>
  </ina:Sequence>
  <ina:Sequence tcDebut="00H19M07S17F" tcFin="00H24M46S4F">
    <ina:Typologie theme="Cyclisme"/>
    <ina:Sequence tcDebut="00H23M40S3F" tcFin="00H23M51S2F"/>
  </ina:Sequence>
</ina:Emission>
```

CODE 2.1 – *Exemple de document XML*

Le modèle de données XML peut être complété par la recommandation concernant les espaces de noms ou *namespaces* [XML NS, 1999]. Les noms des constructeurs peuvent alors être décomposés en deux parties : le nom – par exemple, `Emission` – proprement dit et l'espace de noms auquel il est rattaché. Ce dernier est alors identifié par un URI (*Uniform Resource Identifier*) [Berners-Lee *et al.*, 1998]. Dans notre exemple, le préfixe `ina` est ainsi identifié par `http://www.ina.fr/schema/`. Cela permet de faire coexister des ensembles d'éléments ayant des noms en commun et surtout de développer des vocabulaires sans se soucier des interactions possibles avec d'autres [Euzenat, 2000].

2.1.2 Validité d'un document XML

Déclaration de Type de Document (DTD)

Les documents XML sont donc décrits par un texte balisé. Cependant, si ces balises ne sont pas contraintes par une grammaire, aucune application ne pourra interpréter le document. Par exemple, pour pouvoir présenter un document grâce à une feuille de style, l'organisation

1. Les exemples donnés dans ce mémoire suivent les conventions d'usage relatif au nom des constructeurs, à savoir, chaque mot composant ce nom doit commencer par une majuscule sauf pour le premier mot selon les cas suivants : si le constructeur est un *élément*, le premier mot commence par une majuscule ; si le constructeur est un *attribut*, le premier mot commence par une minuscule.

des balises doit être spécifiée[2]. Tout comme en SGML, les balises autorisées dans le document peuvent obéir à une Définition de Type de Document (DTD).

La DTD est donc la grammaire qui définit la syntaxe légale des balises au sein du document. Elle a pour but de définir chaque élément en précisant :

- son contenu, comme une expression régulière introduisant la séquence (,) et l'alternative (|) fonction d'un nombre variable (1/?/+/*) d'autres éléments,
- et ses attributs, en précisant le type de valeur prise, la présence exigée ou optionnelle et éventuellement la valeur par défaut.

L'exemple de document XML présenté dans le code 2.1 est ainsi défini par la DTD suivante :

```
<?xml version="1.0" encoding="iso-8859-1" ?>

<!ELEMENT ina:Emission (ina:Typologie?, ina:Sequence*)>
<!ATTLIST ina:Emission xmlns:ina CDATA #FIXED "http://www.ina.fr/schema/"
                       nom CDATA #REQUIRED
                       chaineDiffusion CDATA #REQUIRED>

<!ELEMENT ina:Typologie EMPTY>
<!ATTLIST ina:Typologie theme CDATA #REQUIRED>

<!ELEMENT ina:Sequence (ina:Typologie?, ina:Sequence*)>
<!ATTLIST ina:Sequence tcDebut CDATA #REQUIRED
                       tcFin CDATA #REQUIRED>
```

CODE 2.2 – *Exemple de DTD*

Cette DTD déclare les trois types d'éléments visibles dans le document : `Emission`, `Typologie` et `Sequence`. Chacun de ces éléments est défini par son contenu (à l'aide du mot clé `!ELEMENT`) et ses attributs (à l'aide du mot clé `!ATTLIST`). Ici, tous les attributs sont de type chaîne de caractères (`CDATA`) et obligatoires (`#REQUIRED`).

Le document XML présenté dans le code 2.1 est un document dit *valide* au sens XML vis-à-vis de la DTD ci-dessus (code 2.2), car la disposition des balises y est conforme à celle définie par la DTD. Cette notion de validité était déjà présente dans SGML, mais la norme XML ajoute une nouvelle notion – moins forte – qui est celle de document *bien formé*. Un document est dit bien formé si les balises qui le composent forment un et un seul arbre, ce qui est bien entendu une condition nécessaire à sa validité. Ainsi, les documents XML peuvent être manipulés indépendamment de leur DTD. Cette particularité est même une des motivations qui a présidé à son élaboration, le faisant passer d'un langage documentaire à un langage d'échange de données structurées. Une conséquence directe de cette notion est qu'elle a favorisé l'apparition d'autres langages pour exprimer la structure des documents, et des données. Nous évoquons en particulier le langage XML Schema [XML Schema, 2001] dans le paragraphe suivant.

XML Schema

L'approche proposée par les DTDs de XML impose un certain nombre de limitations quant à la manière de spécifier un modèle de document :

- les types de données sont peu nombreux puisqu'ils se limitent au type « chaîne de carac-

2. Nous revenons sur les différentes manières de présenter des documents structurés dans la section 2.2.

tères » ;

- les constructeurs et indicateurs d'occurrence ne permettent pas d'exprimer des contraintes sur leur nombre (sont seulement disponibles : ?, + et *) ;
- la notion d'entité n'offre pas de mécanismes de haut niveau d'abstraction pour la définition de composants de documents réutilisables ;
- il n'y a pas de notion d'espace de noms dans une DTD, d'où un risque de collisions entre symboles importés et symboles définis localement.

Par ailleurs, la syntaxe utilisée pour encoder des DTDs diffère de celle utilisée pour représenter les instances de documents XML.

Ainsi, dès la création d'XML, d'autres langages ont été développés afin de répondre aux limites des DTDs [Euzenat *et al.*, 2003b]. Un groupe de travail du W3C reprenant ces diverses propositions a finalement élaboré une nouvelle recommandation nommée XML Schema [XML Schema, 2001] qui se compose de deux parties normatives : la partie *Structure* spécifie le langage de définition des schémas qui permettront de contraindre la structure des documents XML, tandis que la partie *Datatypes* permet de définir ses propres types de données à inclure dans les schémas.

Un schéma définit une classe de documents XML et permet donc de valider des documents XML instances. Synthétiquement, XML Schema permet :

- de déclarer les éléments (et leurs attributs) susceptibles d'apparaître dans un document XML en précisant leur ordre et leur arrangement ;
- de différencier les types simples des types complexes[3] (en précisant leurs usages), et de définir ces derniers ;
- de dériver des types existants (par restriction ou par extension) en contrôlant ces dérivations ;
- d'annoter les schémas ;
- de réutiliser des définitions de type ou des déclarations d'éléments grâce au mécanisme des espaces de noms ;

Le code 2.3 montre un exemple de schéma[4] qui définit les types complexes `TypologieType`, `ContentType` et `SequenceType` qui est un dérivé du précédent, ainsi que l'élément `Emission` qui est de type complexe, également dérivé de `ContentType`. Le document XML donné dans le code 2.1 est alors dit schéma-valide[5] vis-à-vis du schéma donné dans le code 2.3, moyennant l'ajout du couple attribut-valeur :

```
xsi:schemaLocation="http://www.ina.fr/schema/ emission.xsd"
```

dans l'élément racine `Emission`. Il est à noter que cet attribut possède deux valeurs séparées par un espace : la première est l'URI identifiant l'espace de nom dans lequel les éléments de schémas utilisés sont définis, tandis que la deuxième valeur est l'emplacement du schéma lui-même.

3. Un type est dit *simple* si et seulement si il définit un modèle de contenu qui ne contient ni des éléments, ni des attributs. Sinon, le type est dit *complexe*.

4. Ici encore, nous suivons les conventions d'usage spécifiques au nommage des types en XML Schema, à savoir, chaque mot composant le nom d'un type doit commencer par une majuscule sauf pour le premier mot selon les cas suivants : s'il s'agit d'un *type complexe*, le premier mot commence par une majuscule ; s'il s'agit d'un *type*

```
<?xml version="1.0" encoding="iso-8859-1"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema"
        xmlns:xml="http://www.w3.org/XML/1998/namespace"
        xmlns:ina="http://www.ina.fr/schema/"
        targetNamespace="http://www.ina.fr/schema/"
        elementFormDefault="qualified"
        attributeFormDefault="unqualified">

  <import namespace="http://www.w3.org/XML/1998/namespace"
          schemaLocation="http://www.w3.org/2001/xml.xsd"/>

  <annotation>
    <documentation xml:lang="fr">
      Ce schéma fournit une structure simple pour décrire des émissions
    </documentation>
  </annotation>

  <element name="Emission">
    <complexType>
      <complexContent>
        <extension base="ina:ContentType">
          <attribute name="nom" type="string" use="required"/>
          <attribute name="chaineDiffusion" type="string" use="required"/>
        </extension>
      </complexContent>
    </complexType>
  </element>

  <complexType name="ContentType">
    <sequence>
      <element name="Typologie" type="ina:TypologieType"
               minOccurs="0" maxOccurs="1"/>
      <element name="Sequence" type="ina:SequenceType"
               minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
  </complexType>

  <complexType name="TypologieType">
    <attribute name="theme" type="string" use="required"/>
  </complexType>

  <complexType name="SequenceType">
    <complexContent>
      <extension base="ina:ContentType">
        <attribute name="tcDebut" type="string" use="required"/>
        <attribute name="tcFin" type="string" use="required"/>
      </extension>
    </complexContent>
  </complexType>
</schema>
```

Code 2.3 – *Exemple de Schéma XML*

XML Schema est donc une recommandation qui concerne essentiellement le typage des documents XML. Les documents XML instances doivent être non seulement bien formés et bien structurés mais également bien typés. Une des possibilités les plus intéressantes offerte par le langage consiste à dériver des types existants. Deux types de dérivations sont alors possibles :

1. **La dérivation *par restriction*** : les valeurs possibles du nouveau type défini sont alors un sous-ensemble des valeurs possibles du type de base :

   – dans le cas des types simples, elle consiste à restreindre le domaine des valeurs que peut prendre un élément ou un attribut de ce type ; la dérivation produit alors un type simple ;

   – dans le cas des types complexes, elle reprend directement le modèle de contenu définit dans le type de base, en jouant par exemple sur les contraintes d'occurrence des sous-éléments ;

   L'invariant obtenu est qu'une instance d'un type défini par restriction *est* toujours une instance du type de base.

2. **La dérivation *par extension*** : elle consiste, dans tous les cas, à ajouter des attributs et/ou des éléments au type de base, mais dans le cas d'un type de base simple, le dérivé devient complexe. L'invariant est alors qu'une instance d'un type défini par extension *contient* toujours une instance du type de base comme préfixe.

Cette dérivation de type n'est pas sans rappeler le mécanisme d'héritage des langages de programmation à objet. Pourtant, comme le font remarquer [Euzenat *et al.*, 2003b], il manque encore beaucoup à XML Schema pour approcher un langage à objet :

– L'héritage est ici de la réutilisation dans les types dans la mesure où il s'agit de reproduire la définition d'un modèle de contenu pour le modifier. Il y a bien une spécialisation logique entre les éléments mais pas de réelle spécialisation au sens des objets.

– Les deux invariants donnés ci-dessus ne sont valables qu'entre deux types qui ont un lien de parenté direct (le dérivé et son type de base). Ainsi, si un type `ExtType` est obtenu par extension d'un type `RestType` qui est lui même une restriction d'un type `BaseType`, alors il est impossible d'affirmer quoi que ce soit entre `ExtType` et `BaseType`.

– Les mécanismes de restriction et d'extension mènent parfois à des situations alambiquées. Par exemple, il n'est pas possible de restreindre directement un type en contraignant le type d'un de ses éléments par un de ses dérivés qui aurait été obtenu par extension, bien que ce dernier soit plus spécifique, au sens logique du terme [6].

Dans le cas d'une description d'un document audiovisuel, il est possible de contraindre une structure temporelle d'une émission en utilisant un modèle exprimé en XML Schema. Implicitement, une relation univoque sera alors établie entre l'ordre temporel des séquences composant

simple, le premier mot commence par une minuscule ; dans les deux cas le nom se termine par le suffixe `Type`.

5. Tous les exemples présentés dans ce mémoire ont été validé à l'aide du *parser* Xerces Java version 2.5.0 (`http://xml.apache.org/xerces2-j/index.html`), reconnu comme étant un des plus conformes à la norme XML Schema.

6. Un exemple de cette situation est présenté dans la section 5.3.1 page 138. Une solution possible consiste alors à créer un type intermédiaire, mais cette astuce syntaxique ne fait que mieux mettre en exergue le manque de sémantique donnée à la dérivation des types par les concepteurs du langage.

l'émission et la suite (c'est-à-dire la séquence) des éléments XML. Toutefois, l'utilisation de XML comme méta-langage documentaire pour décrire de l'audiovisuel peut poser quelques problèmes. En effet, comme nous l'avons vu dans la section 1.2.1, une des particularités du document audiovisuel est d'être multidimensionnel. Pour le décrire, il faut donc tenir compte de ses deux dimensions temporelles (celle du son et celle de la vidéo), mais également de la dimension spatiale de l'image. Or, on peut parler du caractère mono-axial de XML puisque l'on ne peut contraindre la structure de ses éléments que selon un seul axe : celui de l'apparition des caractères. Dans ce contexte, il apparaît difficile, pour ne pas dire impossible, de contraindre l'intégration de descripteurs localisés spatialement et temporellement en même temps. Ce problème de localisation de descripteurs selon plusieurs axes est mieux géré dans le langage HyTime que nous décrivons dans la section suivante.

2.1.3 La norme HyTime

La norme HyTime [HyTime, 1997] propose une façon de représenter des documents hypermédias en SGML. Le langage se présente plus exactement comme une *forme architecturale SGML*. Cette technique permet de définir une sémantique standard réutilisable à volonté lors de l'écriture de DTD, plutôt que de définir une DTD standard. Ainsi, la forme architecturale peut être considérée comme une « méta-DTD » définissant des éléments SGML, leurs relations, et la sémantique qui leur est associée. Au lieu d'être des balises standards, les éléments de la pseudo-DTD fonctionnent en fait comme des *types* de balises SGML [Brunie, 1999].

Outre le module de base, qui contient les outils permettant de définir ses propres formes architecturales ou nécessaires au repérage des objets au sein d'une structure, HyTime se compose de quatre modules principaux que nous présentons rapidement :

- Le module d'adressage qui permet d'identifier des objets, soit dépourvus d'identifiants SGML, soit externes aux documents, via trois types de relations : le nommage des éléments, le repérage dans un espace de coordonnées ou le repérage sémantique.

- Le module d'hyperliens qui permet d'établir des relations très riches entre des objets appartenant soit à un seul document, soit à un hyperdocument.

- Le module de définition de l'espace de coordonnées finies (FCS) qui se veut un espace unique à n dimensions dans lequel tout objet manipulé est localisé à partir de coordonnées. HyTime propose des mesures standard (telles que le mètre ou la seconde) mais permet surtout de définir ses propres dimensions pour peu qu'elles soient munies d'une relation d'ordre partiel.

- Le module de présentation qui permet de contrôler l'apparence future du document, soit en indiquant l'ordre et la manière dont les objets peuvent être modifiés lors de leur présentation, soit en précisant une zone d'un FCS susceptible d'être projetée sur un autre FCS.

HyTime apparaît donc comme une norme complexe et peu utilisée. Toutefois, d'intéressantes idées ou intuitions en sont sorties et sont aujourd'hui reprises. Ainsi, les travaux concernant la description des liens XML (*XML Linking Language* ou XLink) et la localisation de l'information (*XML Pointer Language* ou XPointer) s'inspirent fortement de HyTime. De même, comme le souligne [Auffret, 2000], le FCS est de façon évidente un modèle pertinent de localisation au sein du document, à la fois efficace et générique. Par exemple, pour décrire des documents audiovisuels, il est possible de définir un FCS à quatre axes représentant les dimensions temporelles de la vidéo et du son et l'espace plan permettant la localisation spatiale des descripteurs. Une

description HyTime est alors une description SGML dont certains attributs renvoient vers des segments documentaires définis dans le FCS. Comme toute instance SGML, la description peut être contrôlée par une DTD. Cependant, cette grammaire portera sur l'agencement de balises et non pas sur les axes du document, c'est-à-dire non pas en fonction des contraintes émises sur les éléments et définies dans le FCS. L'affiliation du langage HyTime à SGML et le caractère mono-axial de ce dernier empêche donc de profiter pleinement du mécanisme du FCS. On retrouve les mêmes limitations déjà évoquées avec les DTDs et les schémas XML concernant la description de documents audiovisuels avec ces langages.

Cette constatation semble donc remettre en question l'utilisation de XML (ou l'un de ses dérivés) comme méta-langage documentaire pour décrire de l'audiovisuel. Cependant, en pratique, on observe souvent une prédominance de la dimension temporelle sur la dimension spatiale de l'image. Ainsi, l'usage privilégie souvent le découpage dans le temps des différentes séquences d'un document audiovisuel plutôt que la description de l'agencement des différents objets composants l'image. Cette primauté de la dimension temporelle est également notre hypothèse de travail. Il reste ensuite à choisir « une » dimension temporelle puisque nous avons vu que celle du son et celle de la vidéo étaient le plus souvent asynchrones dans le média audiovisuel. XML peut alors être utilisé comme méta-langage pouvant contraindre la structure temporelle des documents audiovisuels à décrire. Nous présentons dans les sections suivantes des langages (toujours déclaratifs) permettant soit de présenter (section 2.2) soit de décrire (section 2.3) des documents multimédias.

2.2 Des standards pour présenter les documents audiovisuels

Nous avons présenté dans la section précédente des langages de marquage permettant de représenter des documents structurés. Une particularité de ces langages est de rendre la description des documents indépendante de leur présentation. Cette technique d'abstraction consistant à séparer la structure du document de son apparence visuelle est bien connue et a déjà montré tout son intérêt en terme de flexibilité et de ré-utilisabilité. L'apparence visuelle est alors spécifiée au moyen de feuilles de style qui établissent une correspondance entre l'abstraction de la structure logique des documents et celle de la structure de présentation. Des langages fonctionnels permettant de déclarer ces correspondances ont alors été élaboré comme par exemple CSS, pour la décoration, ou XSLT [XSLT, 1999] pour la transformation.

Ces langages se sont développés avant tout pour les documents textuels du Web, et ils ne peuvent pas s'appliquer directement pour la génération de documents multimédias en raison des spécificités de ce type de média [van Ossenbruggen *et al.*, 2001]. [Bes and Roisin, 2002] posent alors la question du langage à utiliser pour contrôler le formatage des documents multimédia. Les expériences les plus probantes dans ce domaine ont été faites à partir de résolveurs de contraintes. Ainsi, le système **Cuypers**, présenté dans [van Ossenbruggen *et al.*, 2001], assimile le processus de génération de présentations multimédia à un système de contraintes fixées par les préférences de l'utilisateur, les capacités techniques du support de restitution ou les conditions du réseau. Dans ce système, un interpréteur Prolog est alors utilisé pour formater les documents. Sur la constatation qu'il manque encore des opérateurs de haut niveau pour contrôler globalement le résultat, [Bes and Roisin, 2002] proposent d'ajouter des expressions de priorité et de repli en cas d'échec du formatage et en complément des langages déclaratifs existants. Finalement, ces différents systèmes traduisent la présentation finale des documents multimédia dans le langage SMIL, que nous présentons rapidement dans le paragraphe suivant.

SMIL (*Synchronized Multimedia Integration Language*) est un langage déclaratif (basé sur XML) pour décrire le format et le comportement temporel de présentations multimédia interatives sur le Web, et la façon dont les objets multimédia exploitent les lien hypertextes ([SMIL, 1998]). Les concepts de base d'une présentation SMIL sont les régions, qu'il faut définir et disposer sur un « canevas », et la ligne d'échelle de temps, sur laquelle un scénario temporel mettant en jeux les différents composants de la présentation sera construit. Depuis le mois d'août 2001, une deuxième version du langage est recommandée par le W3C [SMIL2, 2001]. SMIL 2.0 apporte ainsi de nombreuses nouveautés comme par exemple la possibilité d'ajouter des effets de transition entre les éléments multimédia, de les animer dans le temps, ou encore de spécifier des alternatives permettant à l'outil en charge du formatage multimédia une adaptation – bien que limitée – de la présentation. Enfin, un sous-ensemble de SMIL 2 peut s'intégrer dans des pages HTML, et donc être interprété par des navigateurs Web. Nous verrons dans la section 6.3.3 comment nous avons implémenté cette possibilité pour permettre la visualisation des séquences retournées par notre système de recherche documentaire.

De la même façon que SMIL est utilisé pour décrire des combinaisons de composants multimédias, il peut aussi l'être pour « déconstruire » un document composite (par exemple, entre la vidéo et le son, pour un document audiovisuel) ou pour décrire la structure temporelle des composants. Ainsi, [Hunter and Iannella, 1998] exploitent cette idée pour annoter des vidéos : SMIL est alors utilisé pour décrire la structure temporelle tandis que des annotations RDF[7] permettent de décrire la sémantique du contenu. Mais cette solution ne permet pas de décrire la structure logique des documents puisque les segments ne sont pas typés. De plus, il n'est pas possible de contraindre et de valider cette structure à l'aide du seul langage SMIL qui, de toute façon, sert à générer des présentations – et non des descriptions – de documents multimédia.

Nous avons vu, dans le chapitre précédent, que la description fine de la structure et du contenu des documents audiovisuels étaient indispensable pour exploiter au mieux les nouvelles possibilités apportées par le numérique. Cela permettra de rechercher avec plus de précision des séquences audiovisuelles dans le système documentaire, mais également d'adapter les contenus pour des besoins de publications vers différents supports ou publics [Euzenat *et al.*, 2003a]. Nous présentons donc dans la section suivante des standards permettant de décrire les documents multimédia.

2.3 Des standards pour décrire les documents audiovisuels

La croissance explosive du Web a rendu disponibles de nombreuses ressources nouvelles, souvent difficiles à localiser parce que mal indexées. La disparité des modèles et standards pour la description de ces ressources renforce cette difficulté à pouvoir les retrouver. Il s'avère donc utile de définir un formalisme simple et universel permettant de décrire des ressources, quelles qu'elles soient, dans un format compatible avec les standards existants.

Ce problème s'applique d'autant plus aux documents multimédias, qui sont de plus en plus nombreux sur le Web. Ainsi, dès 1996, lors d'une réunion en Finlande, le comité MPEG a souligné la nécessité d'une solution puissante pour identifier et décrire les données multimédias. En effet, si de nets progrès ont été effectués ces dernières années pour accéder à ce type de documents, l'extraction d'informations utiles pour des systèmes techniques tels que des moteurs de recherche multimédia reste un problème. L'obstacle majeur mis en lumière par le comité était *le manque d'une représentation standard, compréhensible et flexible pour le multimédia.*

7. Voir la section 3.3.2 page 79.

Dans cette partie, nous allons présenter différents standards permettant de décrire les documents multimédias. Nous avons tenté de structurer le domaine par niveau de description. Ainsi, nous débuterons cette section avec les standards qui permettent une description bibliographique des documents dans leur ensemble (section 2.3.1). Nous présentons ensuite la réponse apportée par le comité MPEG qui a élaboré la norme MPEG-7 [MPEG-7, 2001], un standard pour décrire des documents multimédias (section 2.3.2). Nous verrons finalement une autre initiative, *TV Anytime*, qui semble être mieux adaptée aux besoins des professionnels de l'audiovisuel (section 2.3.3).

2.3.1 La description des données bibliographiques

Le monde des archives et des bibliothèques a l'habitude d'utiliser des normes de description bibliographiques telles que *Marc* (ou *MODS*, sa version XML) par exemple. Les ressources électroniques étant toujours plus nombreuses à décrire, un consensus international s'est porté sur l'utilisation du standard *Dublin Core*, dont l'objectif n'est pas de supplanter des outils de catalogage comme le Marc ou des outils d'indexation automatique, mais d'en être le complément. Nous commençons donc par présenter ce standard, avant d'étudier quelques une de ces extensions conçues pour mieux prendre en compte les spécificités des ressources multimédias.

Le *Dublin Core*

Le *Dublin Core*[8] est une initiative née en 1995 à Dublin dans l'Ohio qui a pour objectif de développer et de promouvoir des standards pour exprimer des métadonnées sur le Web, et dans le but de couvrir un large spectre d'applications. Elle rassemble des groupes de travail qui ont mis en place une liste minimale de champs de métadonnées bibliographiques connue sous le nom du *Dublin Core Metadata Element Set* (DCMES). Chaque définition du Dublin Core concerne une *ressource* à décrire. D'après [Berners-Lee *et al.*, 1998], une ressource est définie comme « *tout ce qui a une identité* ». Dans le cas des métadonnées du Dublin Core, une ressource sera typiquement une source d'information ou un service, mais la définition peut être appliquée plus largement.

8. *Dublin Core Metadata Initiative*, `http://dublincore.org/`.

Élément	**Description et liste des raffinements**
Title	Le nom donné à la ressource. Pour indiquer un autre type, on peut utiliser le raffinement suivant : • *alternative* : alternative pour le titre, par exemple une abréviation ou une traduction
Creator	L'entité principalement responsable de la création du contenu de la ressource (par exemple, une personne, une organisation ou un service).
Subject	Le sujet du contenu de la ressource. Il sera typiquement décrit par un ensemble de mots-clefs ou par un code de classification. L'utilisation de vocabulaires contrôlés et de schémas formels de classification est encouragée.
Description	Une description du contenu de la ressource à l'aide d'un résumé, d'une table des matières ou d'un texte libre sur le contenu. Le type de description peut être précisé à l'aide des raffinements suivants : • *tableOfContents* : table des matières ; • *abstract* : résumé.
Publisher	L'entité responsable de la diffusion de la ressource, dans sa forme actuelle, tels, un département universitaire, une entreprise.
Contributor	Une entité qui a contribué à la création du contenu de la ressource.
Date	Une date associée avec un événement dans le cycle de vie de la ressource. Il est fortement recommandé d'encoder la valeur de la date en utilisant le format défini par l'ISO 8601 sous la forme *AAAA-MM-JJ*. Pour préciser de quelle date il s'agit, on utilise les raffinements suivants : • *created* : date de création ; • *valid* : date ou période de validité ; • *available* : date ou période de mise à disposition ; • *issued* : date de publication ; • *modified* : date de modification ; • *dateAccepted* : date d'acceptation (par exemple, acceptation d'une thèse par une université, d'un article par un journal...) ; • *dateCopyrighted* : date du copyright ; • *dateSubmitted* : date où le document a été soumis (par exemple, soumis à un comité de lecture s'il s'agit d'un article).
Type	La nature ou le genre du contenu de la ressource. Il est recommandé de choisir la valeur du type dans une liste de vocabulaire contrôlé (par exemple, le vocabulaire *DCMI Type* définit par le Dublin Core).
Format	La matérialisation physique ou digitale de la ressource. On peut spécifier le matériel et le logiciel nécessaires pour afficher ou faire fonctionner la ressource. Les raffinements suivants sont disponibles : • *extent* : taille ou durée ; • *medium* : support physique.
Identifier	Une référence non ambiguë à la ressource dans un contexte donné. Il est recommander d'utiliser un système formel d'identification (URI, ISBN...). La référence peut être précisée avec le raffinement suivant : • *bibliographicCitation* : une référence bibliographique pour la ressource.
Source	Une référence à une ressource à partir de laquelle la ressource actuelle a été dérivée.
Language	La langue du contenu intellectuel de la ressource.
Relation	Une référence à une autre ressource qui a un rapport avec cette ressource. On précise le type de lien avec des raffinements : • *isVersionOf* : on a affaire à une nouvelle version, une modification ou une adaptation du document lié. Les changements concernent le contenu et pas seulement la forme ;

suite sur la page suivante

suite de la page précédente	
Élément	**Description et liste des raffinements**
	• *hasVersion* : réciproque d'*isVersionOf*. Le document lié est une version modifiée du présent document ; • *isReplacedBy* : le présent document a été remplacé par le document lié ; • *replaces* : réciproque de *isReplacedBy*. Le présent document remplace le document lié ; • *isRequiredBy* : on a besoin du présent document pour interpréter correctement le document lié ; • *requires* : réciproque d'*isRequiredBy*. Le présent document a besoin du document lié pour être correctement présenté, transmis, ou pour assurer sa cohérence ; • *isPartOf* : le document est une partie (physique ou logique) d'un autre document ; • *hasPart* : réciproque d'*isPartOf* : le document inclut le document lié, physiquement ou logiquement ; • *isReferencedBy* : le document courant est référencé, cité, ou lié par le document indiqué ; • *references* : réciproque d'*isReferencedBy* : le document courant référence, cite ou pointe vers le document indiqué ; • *isFormatOf* : le présent document a le même fond que le document indiqué, mais présenté sous une forme différente ; • *hasFormat* : réciproque d'*isFormatOf* : le présent document possède une variante sous une forme différente ; • *conformsTo* : référence à un standard établi auquel se conforme le présent document.
Coverage	La portée ou la couverture spatio-temporelle de la ressource. La couverture typiquement inclut une position géographique (le nom d'un lieu ou ses coordonnées), une période de temps (un nom de période, une date, ou un intervalle de temps) ou une juridiction (telle que le nom d'une entité administrative). Le type de couverture peut être précisé : • *spatial* : couverture spatiale ; • *temporal* : couverture temporelle.
Rights	Information sur les droits sur et au sujet de la ressource (par exemple, présence d'un Copyright). Si cet élément est absent, aucune hypothèse ne peut être faite sur l'état de ces droits, ou de tout autre, par rapport à la ressource. Les droits peuvent être précisés à l'aide du raffinement : • *accessRights* : information sur qui peut accéder à la ressource.
Audience	L'audience représente le groupe de personnes à qui le document est destiné. Elle est déterminée par l'auteur, le publicateur, ou un tiers. On peut utiliser les raffinements suivants : • *mediator* : entité qui sert d'intermédiaire pour l'accès au document ; • *educationLevel* : position du niveau de l'audience par rapport à un contexte d'éducation ou de formation.

TAB. 2.1 – *Liste des éléments du* Dublin Core

Les champs de métadonnées du Dublin Core sont appelés *éléments* et la signification de certains d'entre eux peut être précisée à l'aide de *raffinements*. Un raffinement restreint la signification d'un élément, mais sans la changer fondamentalement. Son utilisation est facultative. Une description utilisant le Dublin Core est donc composée de ces éléments auxquels on associe des valeurs. Ces valeurs peuvent être libres (de simples chaînes de caractères) ou contrôlées par des formats de données[9] bien définis. Le tableau 2.1 donne la liste exhaustive des différents éléments

9. Par exemple, le MeSH (*Medical Subject Headings*), un URI (*Uniform Resource Identifier*) selon la RFC 2396 [Berners-Lee *et al.*, 1998], le W3CDTF pour le codage des dates et des heures selon le W3C, le RFC 3066 pour le nouveau codage des langues...

avec leurs raffinements possibles selon la version 1.1 du DCMES[10]. Dans une description, tous ces éléments sont optionnels et répétables.

Les éléments provenant du DCMES peuvent, entre autre, être représentés à l'aide de la syntaxe RDF/XML[11]. Cette solution, promue par le W3C, s'accompagne d'une DTD et d'un schéma XML permettant de valider les descriptions et décrit une méthode pour lier celles-ci à des pages Web. Pour notre part, nous avons utilisé cette syntaxe pour décrire les données bibliographiques d'une ressource particulière : les ontologies.

La généralité du modèle du Dublin Core est à la fois un atout, puisqu'elle le rend applicable à tout type de ressource, mais également une faiblesse dès lors qu'on veut l'appliquer à un domaine précis. Par exemple, dans le domaine de l'audiovisuel, on aimerait préciser d'avantage le genre d'une émission ou le rôle d'un de ses contributeurs (s'il s'agit d'un présentateur, d'un acteur, d'un chanteur, d'un sportif...).

Une extension du *Dublin Core*

Pour pallier ces défauts, [Hunter and Iannella, 1998] puis [Hunter, 1999] proposent d'étendre le modèle du Dublin Core pour tenir compte des spécificités de l'audiovisuel. Ils reprennent ainsi les conclusions d'un workshop consacré à l'applicabilité du Dublin Core pour décrire des images ou des vidéos. Celui-ci assujettissait l'utilisation des éléments du DCMES moyennant, entre autre, quelques ajouts via des raffinements supplémentaires et d'avantage de liste de termes contrôlés pour valuer les éléments.

[Hunter and Iannella, 1998] définissent ainsi une liste hiérarchique composée de genres audiovisuels pour qualifier le type du document. Cette liste contient également des descripteurs permettant de spécifier les sous-parties du document (séquence, scène, plan, image et objet). Les auteurs proposent alors de raffiner l'élément `Description` et de lier les valeurs de ces nouveaux attributs au type de document décrit (figure 2.1).

L'extension proposée contient finalement de nouveaux raffinements pour les éléments `Format` – afin de spécifier par exemple la chaîne de diffusion ou la durée de document – et `Coverage` – afin de permettre la localisation spatio-temporelle des segments à décrire.

Ce nouveau modèle tient mieux compte des spécificités audiovisuelles mais il montre rapidement quelques limites. Il fige, tout d'abord, un découpage hiérarchique prédéterminé qui n'a rien d'universel puisque les notions de « scène » et de « séquence » sont loin d'être communément acceptées. De plus, comme nous l'avons vu, les notions de « plan » et d'« image », si elles ont un fondement objectif, ne représentent pas forcément une unité documentaire pertinente. Enfin, comme l'avouent les auteurs eux-mêmes, vouloir décrire quelque chose d'aussi détaillée que la texture d'un objet d'une image particulière d'un film avec les éléments du Dublin Core semble dépasser largement les objectifs (et les possibilités) de ce standard. Celui-ci est en effet caractérisé par une adoption minimale des éléments à décrire quelque soit le type de ressource, privilégiant la simplicité et la généralité.

Finalement, une utilisation sage du Dublin Core et de cette extension réside dans la description du document audiovisuel dans son ensemble, c'est-à-dire se cantonnant à la description des données bibliographiques du document.

10. Pour une traduction complète en français de la recommandation DCMES, voir [Vercoustre, 2002].
11. Voir la section 3.3.2 page 79.

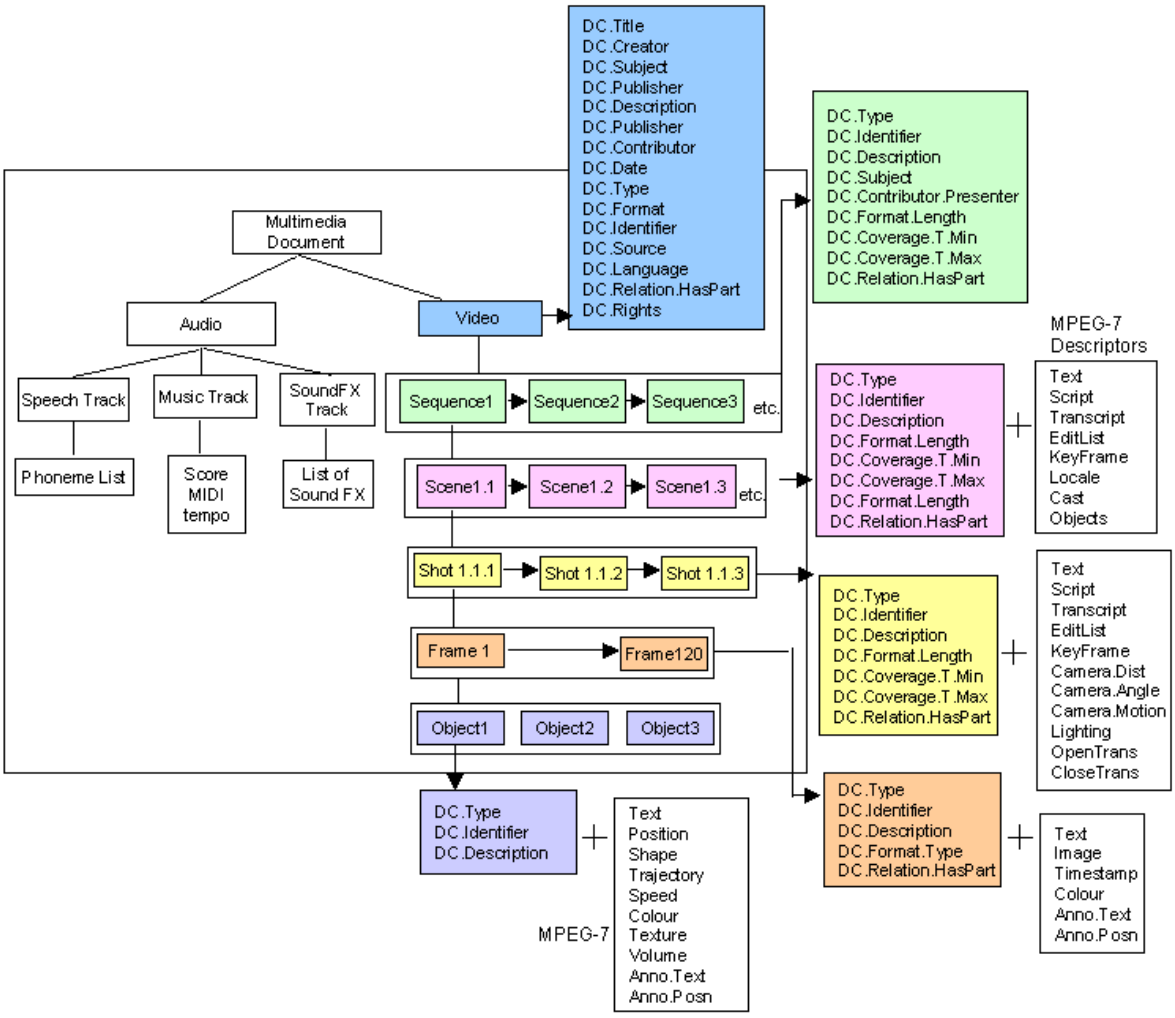


FIG. 2.1 – *Structure hiérarchique avec ses attributs exprimés selon le Dublin Core pour la description de documents vidéo, d'après [Hunter, 1999]*

Les catégories VRA

A la suite des travaux du Dublin Core, un autre comité a proposé un ensemble de catégories permettant de décrire les ressources à vocation culturelles. Ainsi, la *Visual Resources Association* (VRA[12]) définit la notion d'œuvre (*work*) comme une entité physique qui existe, ou a pu exister dans le passé ou encore pourrait exister dans le futur. Cette notion recouvre donc les création artistiques (peinture, sculpture), mais également les bâtiments ou toute autre construction, les œuvres littéraires ou encore les interprétations d'un artiste. De plus, la VRA définit l'*image* comme une représentation visuelle d'une œuvre (par exemple, une photographie ou un document audiovisuel).

La version actuelle (3.0) du standard fournit un ensemble de 17 descripteurs pour décrire ces deux types d'entités. A l'instar du Dublin Core, ces éléments peuvent être précisés à l'aide d'attributs, et il est recommandé de contrôler leur valeur en utilisant des listes d'autorités ou des thésaurus. L'interopérabilité avec les autres standards est assurée au moyen de correspondance entre ces catégories et les éléments du Dublin Core ou provenant d'autres standards. Au final, les descripteurs s'avèrent un peu plus précis pour décrire globalement les documents, mais ils ne sont clairement pas adaptés pour décrire les contenus.

XMLTV

Contrairement aux langages précédents, le projet XML TV[13] est une initiative personnelle relayée depuis par une foule d'anonymes du Web qui apportent leur contribution. Ce projet a défini une DTD permettant de décrire les grilles de programmes des chaînes de radio et de télévision dans un format XML, et continue de développer des petits logiciels chargés d'extraire automatiquement ces grilles de programmes à partir des sites Web des diffuseurs ou des guides de programmes électroniques (cela concerne plus de 15 pays aujourd'hui). Le format de description est clairement orienté vers le téléspectateur puisqu'il regroupe les programmes par genres plutôt que selon leur diffuseur. Il contient des informations de type catalogage bibliographique ainsi qu'un résumé pour chacun des programmes. De nombreuses évolutions du format sont prévues, dont une traduction vers le langage RDF pour laquelle un vocabulaire ontologique spécifique serait modélisé.

2.3.2 MPEG-7 : le nouveau langage documentaire pour représenter le multimédia

Le langage MPEG-7 (*Multimedia Content Description Interface*) [MPEG-7, 2001] est devenu officiellement une norme ISO pour la description de contenu multimédia en décembre 2001[14]. Pour une présentation complète, on peut consulter [Manjunath *et al.*, 2002], alors que [Martínez *et al.*, 2002] et [Martínez, 2002] constitue une excellente introduction au langage.

Notions de base

Le langage MPEG-7 définit la notion d'*outils* de description multimédia. Dans la terminologie de la norme, les outils font référence à un ensemble de descripteurs (Ds) dont les valeurs

12. `http://www.vraweb.org/`.
13. `hhtp://membled.com/work/apps/xmltv/`.
14. Cependant, les 8 parties composants la norme ne sont pas encore toutes publiées, et certaines d'entre elles sont mêmes toujours en cours de vote.

permettent de décrire des caractéristiques physiques audiovisuelles (couleur, texture, mouvement...), à un ensemble de schémas de descriptions (DSs) qui permettent d'organiser les descripteurs dans des modèles pour les objets multimédias, et au langage de définition des descriptions (DDL) qui permet d'encoder le tout. Le langage XML Schema, présenté dans la section 2.1.2, étendu de quelques types primitifs a finalement été retenu comme le DDL. Il est à noter que les descripteurs de bas niveau (couleur dominante, mouvement de caméra, spectre sonore, mélodie...) prédominent largement dans la norme car celle-ci a, pour l'essentiel, été élaborée par la communauté de l'analyse automatique et du traitement du signal.

Si MPEG-7 fournit des outils normatifs pour décrire les différents types d'information composant les documents multimédias en général, la partie la plus importante du langage est consacrée aux informations audiovisuelles (*MPEG-7 Part 5, Multimedia Description Schemes* ou MDS). Cette partie couvre un très large éventail puisqu'elle permet de décrire la structure du document, des aspects sémantiques (objet, événement), des données concernant le média, la création et la production du document et son utilisation compte-tenu de ses droits d'accès, et même la manière permettant aux utilisateurs d'accéder ou de naviguer dans le contenu ou encore leurs préférences et leurs usages lors de leurs interactions avec celui-ci (figure 2.2). Au final, le MDS standardise plus de 600 types.

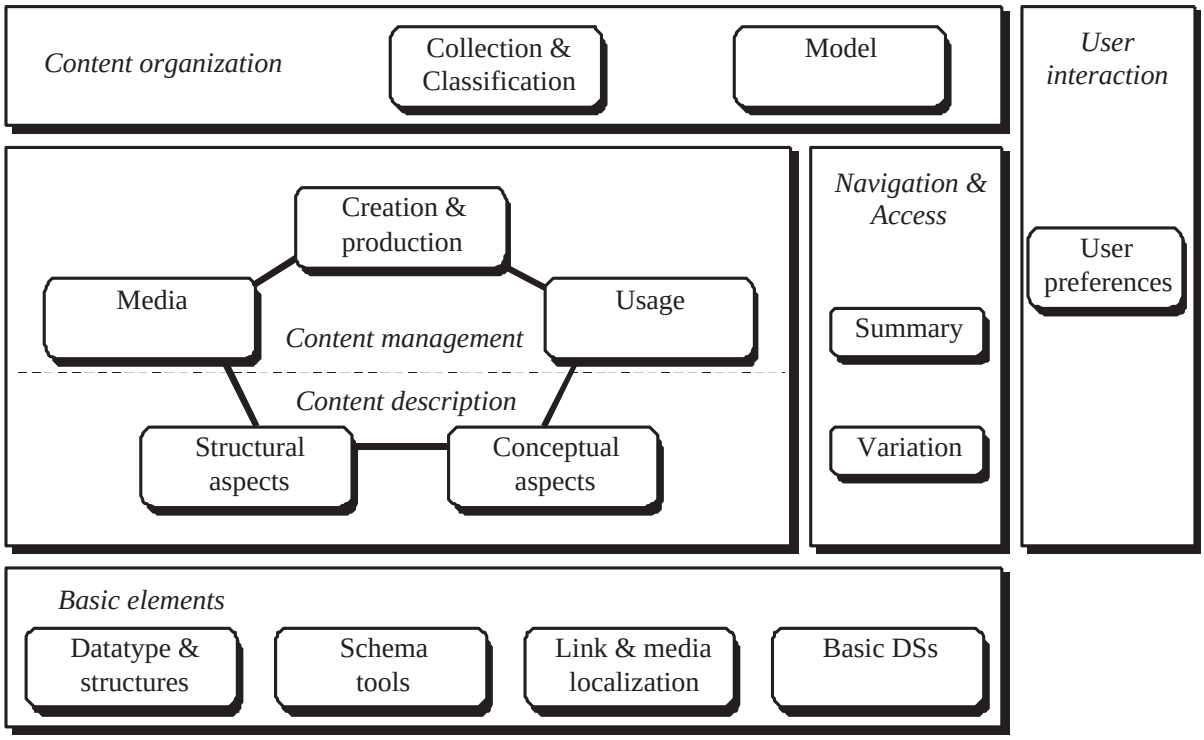


FIG. 2.2 – *Schéma général des outils de description disponibles dans le MDS*

Une description valide MPEG-7 est unitaire ou complète, et toujours encapsulée entre les éléments XML `<Mpeg7>` et `</Mpeg7>`. Une description unitaire contient une instance de n'importe quel descripteur, et sert donc à l'échange, entre applications, d'une partie spécifique d'une description complète. Une description complète implique que sa structure suive la définition d'un des types de haut niveau, qui sont eux répartis en trois groupes (figure 2.3) : la gestion du contenu, qui concerne les informations relatives au média, à sa création ou à son usage ; les entités de contenu,

qui font référence à la structure et à la sémantique des différents médias comme l'image, le son ou la vidéo ; et l'abstraction du contenu, qui permet de décrire un modèle ou un résumé du document.

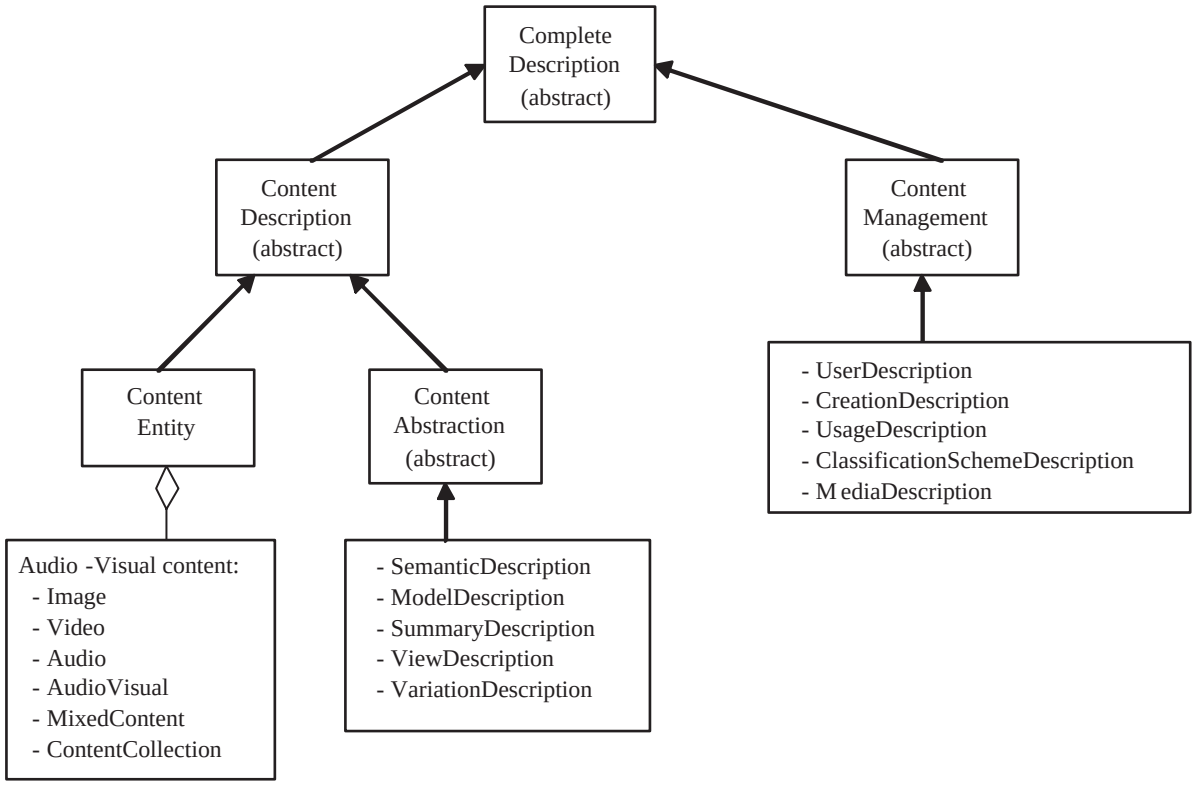


FIG. 2.3 – *Hiérarchie des types de description MPEG-7*

Un exemple d'une description MPEG-7 complète est donné dans la section 4.2.1 page 100.

Structure

L'intégration des caractéristiques structurelles et sémantiques est considérée comme la contribution la plus importante du langage MPEG-7. La description structurelle est basée sur l'idée de *segment* qui est une portion spatiale, temporelle ou spatio-temporelle du contenu audiovisuel. Le *SegmentDS* se spécialise en différents types utilisables selon le média à décrire (audio, image, vidéo, multimédia). La figure 2.4 illustre la hiérarchie des types de segments disponibles. Ces types ajoutent les notions de *temps média*, qui permet d'obtenir un segment temporellement connecté, et de *masque* qui permet de construire des régions et des segments non connectés spatialement ou temporellement.

Chaque segment autorise certaines décompositions (dans le temps, dans l'espace, par média) selon le média auquel ils sont liés, et ils définissent alors les types résultats issus de ces découpages (figure 2.5).

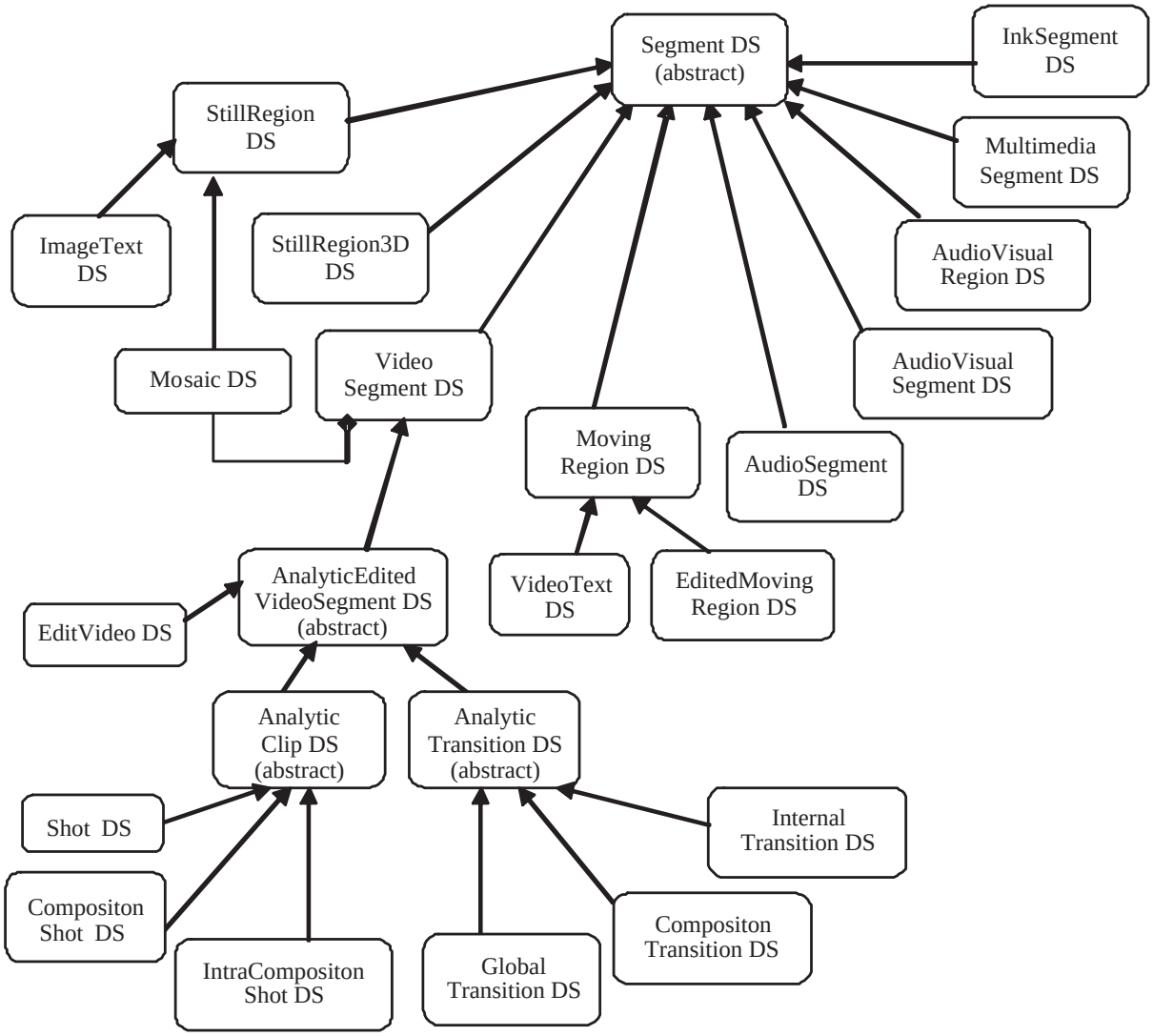


FIG. 2.4 – *Hiérarchie des types de segments disponibles dans le MDS pour décrire le contenu multimédia*

Sémantique

La description sémantique traite, pour sa part, du monde dépeint dans le contenu audiovisuel (*SemanticDS*). L'approche adoptée par MPEG-7 est un modèle centré sur l'événement interprété comme un moment où il se passe quelque chose. Les objets, les personnes et les lieux permettent de décrire cet événement ainsi que le temps où il se produit. De plus, ces entités ont des propriétés qui les relient. Enfin, il y a un monde en toile de fond qui contient d'autres entités et événements qui fournit le contexte d'interprétation des descriptions [Benitez *et al.*, 2002]. Ainsi, trois sortes de composants sont utilisés pour la partie sémantique des descriptions :

- des **entités** appartenant au monde narratif (*ObjectDS*, *EventDS*, *SemanticPlaceDS*, *SemanticDateDS*) ;
- des **attributs** (ou propriétés) permettant de qualifier les entités ou d'indiquer leur niveau d'abstraction (instance ou concept) ;
- des **relations** indiquant comment les entités sont liées au monde narratif (agent, patient, cause), comment elles sont liées entre elles (spécialisation, combinaison) ou encore quels rapports elles entretiennent avec le média (dépeint, symbolise).

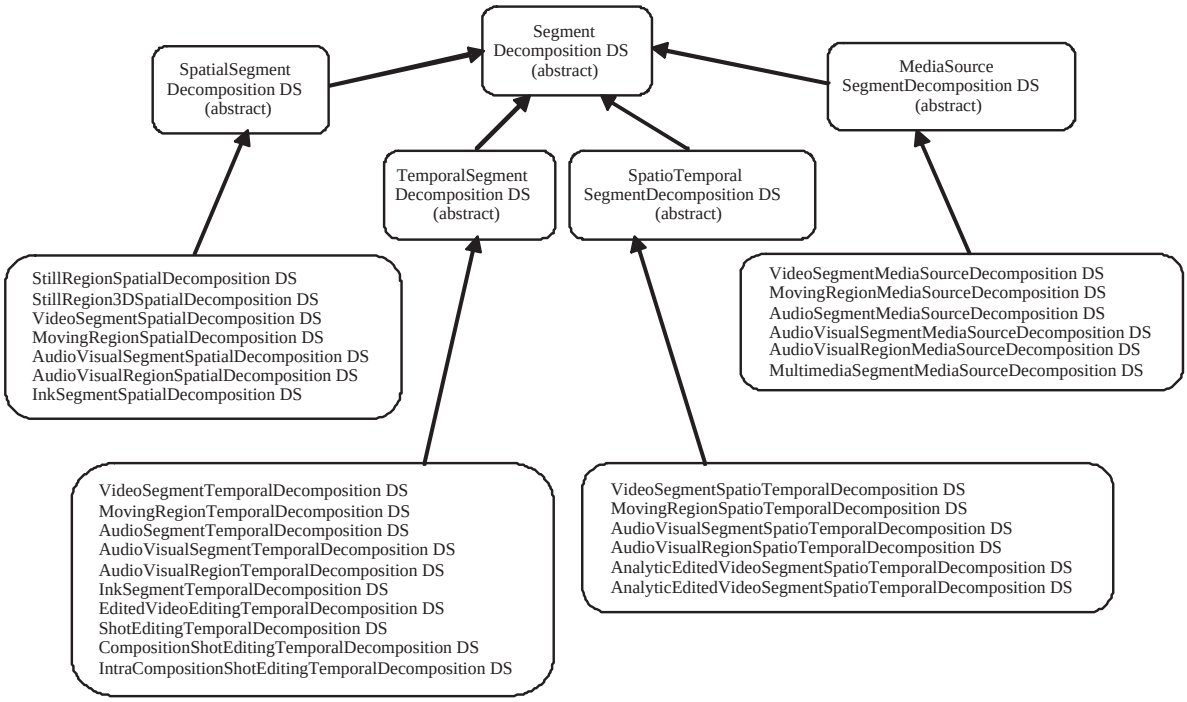


FIG. 2.5 – *Hiérarchie des différentes décompositions structurelles applicables aux segments*

Schémas de classification

MPEG-7 a laissé la porte ouverte à la création de structures de connaissance très simple, de type thésaurus, à travers les schémas de classification (*ClassificationSchemeDS*) (CS). Ceux-ci permettent de définir des termes et de les organiser grâce à cinq relations : `plus spécifique`,

`plus général`, `est lié à`, `utilise` et `est utilisé par`. Cependant, les CS sont vus comme des ressources externes, utilisables lors de la description pour valuer des entités, mais ils ne peuvent pas être utilisés dans un schéma pour contraindre la structure d'une classe de documents.

Implémentations

Une plate-forme logicielle de référence nommée XM (*eXperimentation Model*)[15] vient compléter la définition du standard. Elle doit montrer comment le langage MPEG-7 peut être utilisé simplement pour produire rapidement des applications. Cette plate-forme se compose de modules implémentant divers algorithmes de traitement du signal pour extraire automatiquement des caractéristiques des médias. Elle produit ainsi des descriptions composées pour l'essentiel de descripteurs proches du signal (son, couleur, texture, mouvement...). Il est ensuite possible d'interroger ces descriptions à l'aide d'un langage de requête XML spécialisé.

L'Institut National des Standards et Technologies américain (NIST) fournit, lui, un service de validation de descriptions MPEG-7 en ligne[16]. Cet outil est basé sur un parseur XML Schema et permet de vérifier la validité d'une description avec le schéma MPEG-7. Finalement, divers outils permettent d'annoter des vidéos semi-automatiquement tout en produisant des descriptions conformes au langage MPEG-7 comme nous le verrons dans la section 2.4.1.

L'outillage existant autour de ce langage est donc encore très insuffisant et l'on ne peut pas vraiment dire que l'industrie a décidé de l'adopter : il n'existe pas d'API programmable du langage par exemple. Le schéma MPEG-7 (`xsd`) lui-même pose problème, puisque comme nous le verrons dans la section 5.3.2, celui-ci n'existe pas globalement et n'est pas disponible sur le Web. Ainsi, pour pouvoir valider ses propres descriptions, il est nécessaire de le reconstruire par extraction dans les différentes parties de la norme. Cette opération a d'ailleurs mis en évidence un certain nombre d'erreurs syntaxiques présentes dans la version publiée de la norme.

Au contraire de MPEG-7, le langage **TV Anytime** a été développé selon des besoins professionnels bien identifiés. Nous le présentons dans la section suivante.

2.3.3 TV Anytime

TV Anytime[17] est un consortium international fondé en 1999, rassemblant plus de 150 entreprises, dont l'objectif est de produire des standards pour les décodeurs avec stockage pour la TV numérique. Il produit entre autre un langage pour décrire des méta-données « riches » sur les contenus audiovisuels diffusés afin de fournit des services interactifs sur la télévision numérique (par exemple, le guide de programme électronique). Tout comme pour MPEG-7, le langage de description est basé sur XML et XML Schema. Il reprend des descripteurs de la norme MPEG-7 et l'étend, à l'aide de XML Schema, pour ajouter des descripteurs de plus haut niveau permettant de décrire, par exemple, si le programme est isolé ou s'il appartient à une série ou à une collection, s'il est diffusé en plusieurs parties, le public destiné au programme ou encore les conditions de la diffusion [Pfeiffer and Srinivasan, 2000]. La liste des termes définis n'est pas très structurée du point de vue des connaissances, mais nous verrons comment nous l'avons utilisé pour construire l'ontologie de l'audiovisuel (voir section 5.1, page 123).

15. `http://www.lis.e-technik.tu-muenchen.de/research/bv/topics/mmdb/e_mpeg7.html`.
16. Ce service est disponible à `http://m7itb.nist.gov/M7Validation.html`.
17. `http://www.tv-anytime.org/`

2.3.4 Synthèse

En conclusion, nous remarquons que des langages comme le Dublin Core ou les catégories VRA peuvent être utilisés pour décrire les données de catalogage bibliographique des documents. Ils ont l'avantage de produire des descriptions minimales mais interopérables avec un nombre maximal de systèmes. La norme MPEG-7 était annoncée comme le standard permettant de décrire les documents multimédia, mais finalement, nous constatons que les descripteurs standardisés sont de trop bas niveau pour prendre en compte tous les besoins de description, et notamment ceux que nous avons mis en évidence dans la section 1.2.3. Nous avons ainsi vu qu'il était nécessaire d'exprimer la sémantique de ces descripteurs dans un langage formel et utilisable par la machine pour véritablement permettre la manipulation (voire l'échange) du contenu multimédia.

On assiste aujourd'hui à une sorte de « démodularisation » du langage MPEG-7 dans le sens où un certain nombre d'initiatives (par exemple, TV Anytime) essaient de reprendre des parties du langage pour ensuite l'étendre afin de correspondre à des besoins réels d'application. Telle a été également notre démarche, puisque nous verrons dans la section 4.3.3 comment nous ré-utilisons la partie la plus aboutie du langage – consacrée à la description de la structure des documents – pour décrire les documents audiovisuels. Après avoir décrit les langages de représentation existants, nous présentons dans la section suivantes des systèmes opérationnels permettant d'annoter des contenus audiovisuels.

2.4 Des systèmes opérationnels pour annoter de la vidéo

Nous avons étudié dans la section précédente les langages et modèles permettant de décrire la structure et le contenu des documents audiovisuels. Nous avons insisté en particulier sur la norme MPEG-7, qui a comme ambition de devenir le standard pour ce genre de description, tout en mettant en avant ses faiblesses : la norme n'a pas été développée selon des usages identifiés.

Nous allons présenté dans cette section les systèmes opérationnels permettant d'annoter des documents audiovisuels. Nous commençons par décrire un certain nombre d'outils capables de segmenter un flux audiovisuel, de localiser des régions d'intérêts et de les annoter en utilisant un vocabulaire plus ou moins contrôlé (section 2.4.1). Tous ces outils ont comme particularité de produire des descriptions conformes à la norme MPEG-7. Sur le constat que ces approches ne sont pas suffisantes pour décrire la sémantique des contenus, nous décrivons d'autres systèmes, basés sur des approches orientées « connaissances », qui sont à l'origine de notre proposition (section 2.4.2).

2.4.1 Des systèmes « purement » documentaires

***MPEG-7 Annotation Tool* - IBM** L'outil d'annotation **VideoAnnEx**[18] est développé par le centre de recherche d'IBM alphaWorks (figure 2.6). Il propose tout d'abord une segmentation en plans des vidéos et permet ensuite d'annoter chacun des segments repérés à l'aide de descripteurs contrôlés. Ceux-ci sont rangés dans trois catégories : la description de l'arrière-plan (la scène statique), les objets clés et les événements. Dans chacune de ces catégorie, les descripteurs sont structurés par la relation de spécialisation. L'ensemble des descripteurs forment le *lexicon set* qui est complètement personnalisable puisque l'utilisateur peut construire ces trois catégories au fur et à mesure de l'indexation. Une seconde interface permet de délimiter des régions dans l'image

18. L'outil est disponible gratuitement à `http://www.alphaworks.ibm.com/tech/videoannex`.

pour décrire encore plus finement les vidéos. Finalement, l'outil peut importer et générer des descriptions MPEG-7 des documents audiovisuels annotés.

Cet outil fut un des tous premiers à être capable de lire des descriptions MPEG-7. Il a le mérite d'afficher une bonne compatibilité avec le langage sans toutefois l'implémenter complètement. D'un point de vue documentaire, les possibilités sont faibles : on ne peut pas contraindre la structure des documents, et les seuls segments qu'il est possible d'annoter sont les images ou les plans, sans niveau d'abstraction supplémentaire. Au niveau des connaissances, outre sa faible expressivité, on peut regretter que la représentation du *lexicon set* ne soit pas elle-même en MPEG-7 puisque les schémas de classification sont justement conçus pour cela.

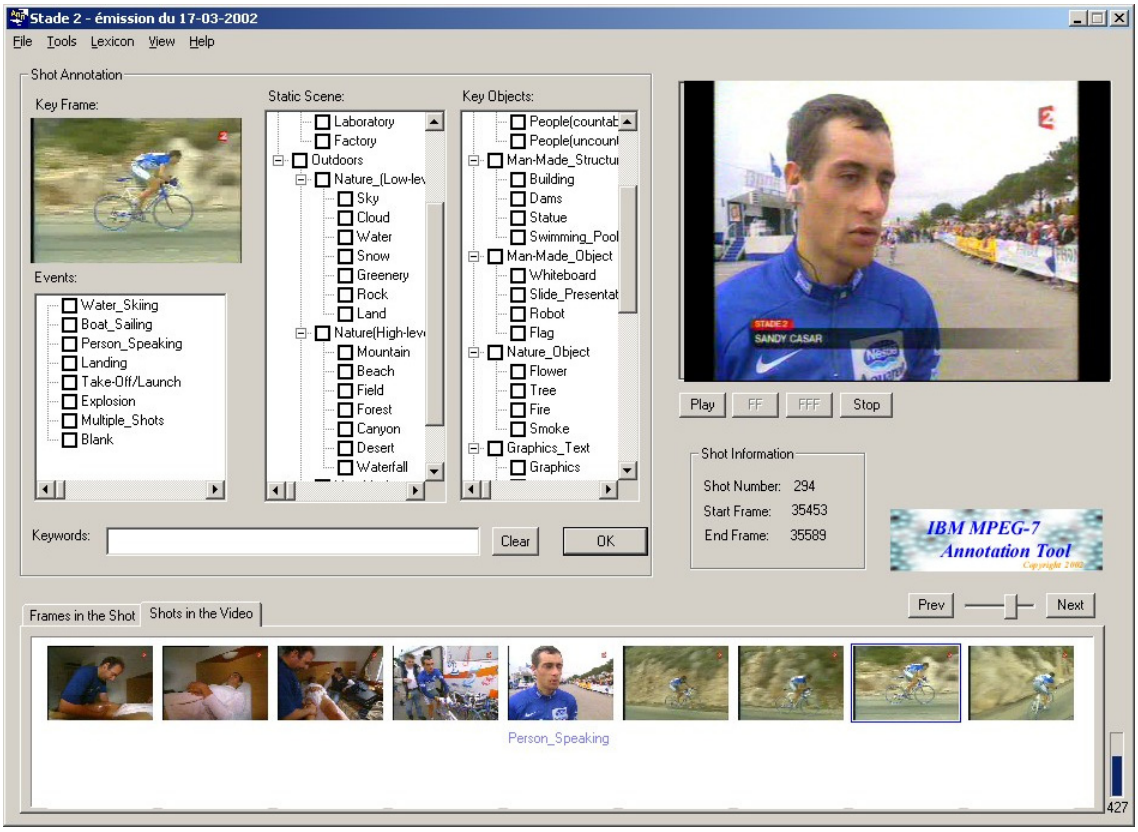


FIG. 2.6 – *Annotation de séquences vidéos dans l'outil* VideoAnnEx. *La vidéo (en haut à droite) est découpée en plans repérés par des images clés (en bas) et annotée grâce au vocabulaire contrôlé appartenant à l'une des trois catégories : arrière-plan, événement et objet (en haut à gauche)*

***COALA - Log Creator* - EPFL** Le projet **COALA** (*Content Oriented Audiovisual Library*), conduit par l'EPFL en Suisse, a débouché sur la réalisation d'une plate-forme prototype d'indexation et d'annotations du téléjournal de la TSR (Télévision Suisse Romande). Contrairement aux autres outils, il se présente comme une application Web[19] spécialisée dans l'annotation d'un genre particulier de document audiovisuel, les journaux télévisés.

Le système se compose d'une interface (**LogCreator**) de segmentation et d'annotation des journaux télévisés [Fatemi and Abou Khaled, 2001; Fatemi, 2003]. La segmentation consiste à

19. `http://coala.epfl.ch/demos/demosFrameset.html`.

découper *a priori* le document selon une structure hiérarchique (figure 2.7). L'annotation consiste à ajouter une description appropriée pour chaque type de segment (figure 2.8). La description de la structure et du contenu sont validés par un schéma de description exprimé par un sous-ensemble de MPEG-7. Le système permet enfin d'interroger les descriptions audiovisuelles grâce à une interface implémentant un langage de requête dédié nommé SVQL (*Semantic Views Query Language*) et tirant partie du schéma de description.

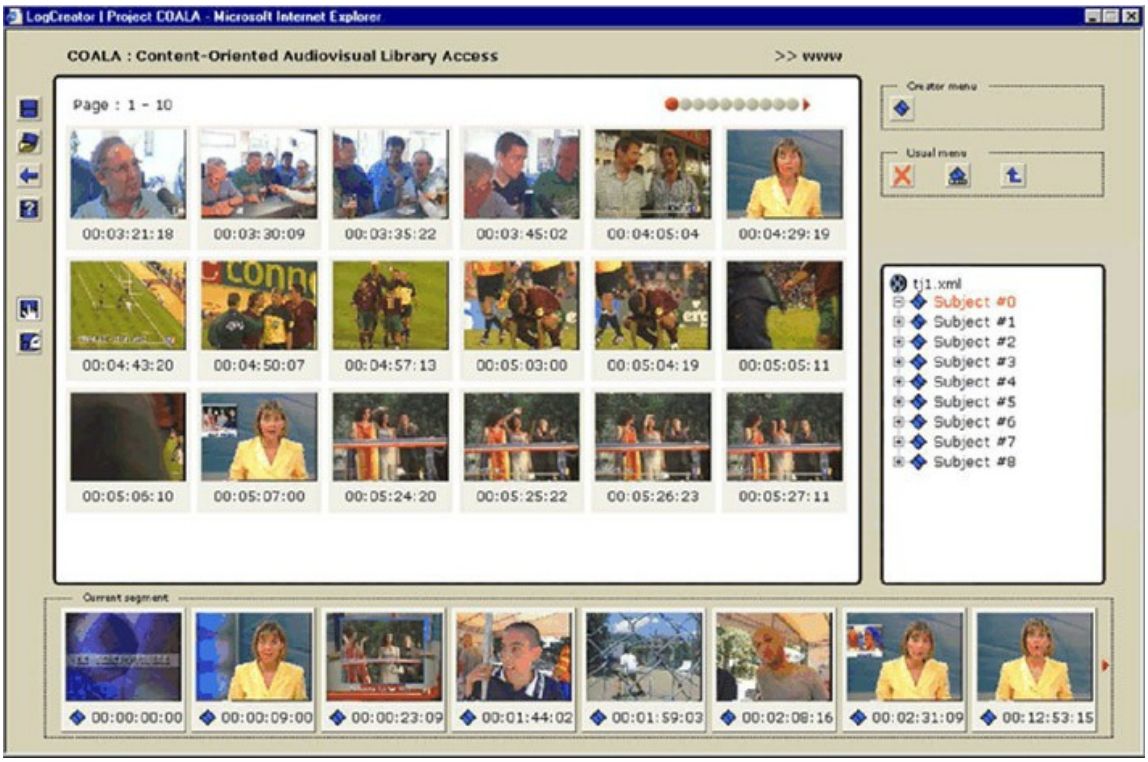


FIG. 2.7 – *Interface de segmentation de* Log Creator

Le système COALA prend bien en considération les enjeux documentaires de la description audiovisuelle : il permet de décrire finement la structure des émissions pour pouvoir naviguer ensuite dedans. Son principal défaut est cependant de se restreindre à un seul type d'émission : le journal télévisé. Le schéma de description est ainsi figé et ne peut pas s'appliquer à d'autres types de programmes. La représentation du contenu est contrôlée par des connaissances représentées en MPEG-7, c'est-à-dire basées sur un thésaurus. Elle gagnerait à utiliser un langage de représentation de connaissances plus complet apportant une meilleure expressivité et donc des raisonnements plus poussés lors de l'interrogation de la base.

***Mdéfi* - INRIA** L'environnement **Mdéfi** (*Multimedia Description and Finegrained Integration*) est un environnement auteur pour l'intégration fine de média, développé par T. Tran Thuong à l'INRIA Rhône-Alpes [Tran Thuong, 2003]. Il étend l'outil **MADEUS** [Layaïda, 1997; Tardif, 2000; Villard, 2002] en proposant de composer des documents multimédias à partir de descriptions détaillées des contenus.

L'environnement permet donc, entre autre, de produire des descriptions de documents multimédias (figure 2.9). L'annotation de documents audiovisuels peut reposer sur une segmentation

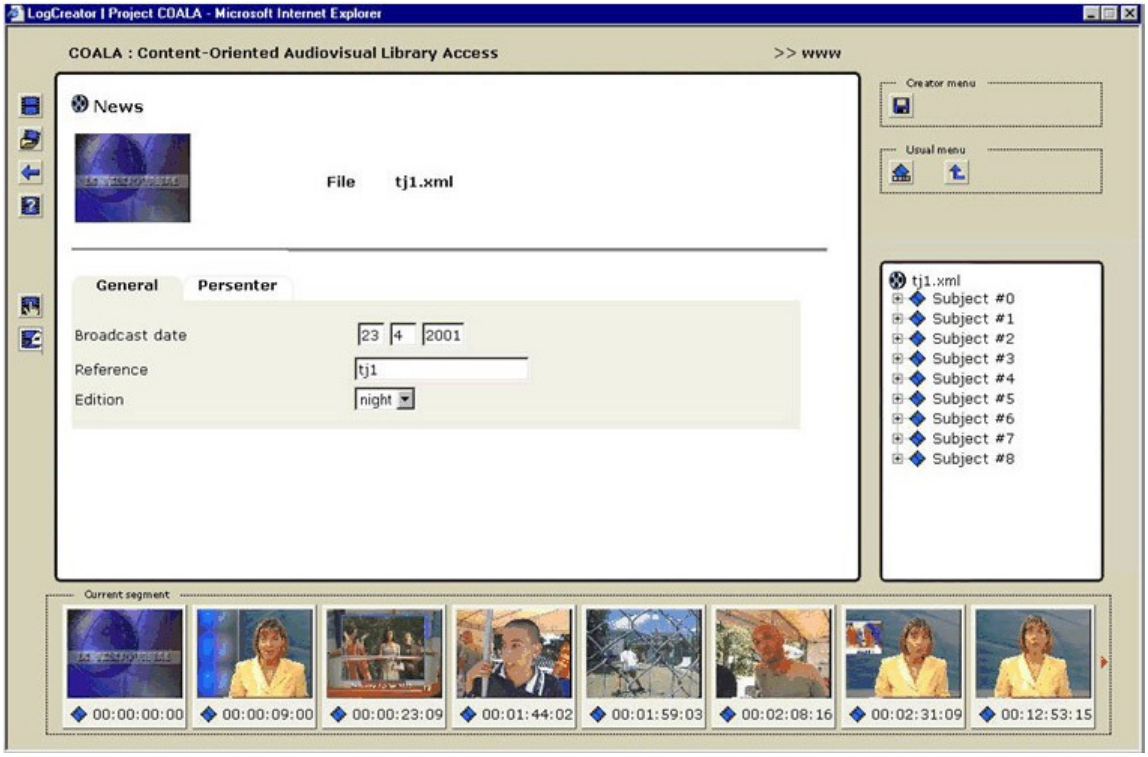


FIG. 2.8 – *Interface d'annotation de* Log Creator

automatique effectuée au préalable. Le modèle de description du contenu sous-jacent fournit lui plusieurs niveaux d'abstractions :

- Le niveau *concret* est la description de la structure du contenu.
- Le niveau *significatif* permet de lier des descriptions de bas niveau (une région dans l'image) avec le monde réel (un personnage particulier). Il se base pour cela sur un *modèle sémantique* qui permet de définir des termes qui vont référencer des descriptions de bas niveau.
- Le niveau *thésaurus* permet de catégoriser les connaissances représentées dans la partie sémantique.

L'implémentation de ce modèle de description utilise finalement une version étendue du langage MPEG-7. L'extension concerne la définition de nouveaux types construits à partir des descripteurs MPEG-7 en utilisant le DDL.

La partie la plus développée de ce travail est sans conteste la description de la structure des différents éléments composant les documents multimédias. Il met à profit une longue tradition d'étude de la temporalité de ce type de document ainsi que la gestion de leur structure. Il montre ainsi qu'une description fine de la structure permet de recomposer de nouveaux documents multimédias plus facilement et dynamiquement. En revanche, la description de la sémantique des contenus est nettement moins abordée. L'auteur suggère d'utiliser des thésaurus existants comme *WordNet* [20] pour contrôler cette partie de la description. Nous verrons dans la suite que nous proposons d'aller beaucoup plus loin dans la description fine des contenus.

20. WordNet est un thésaurus lexical en anglais disponible à `http://www.cogsi.princeton.edu/~wn`.

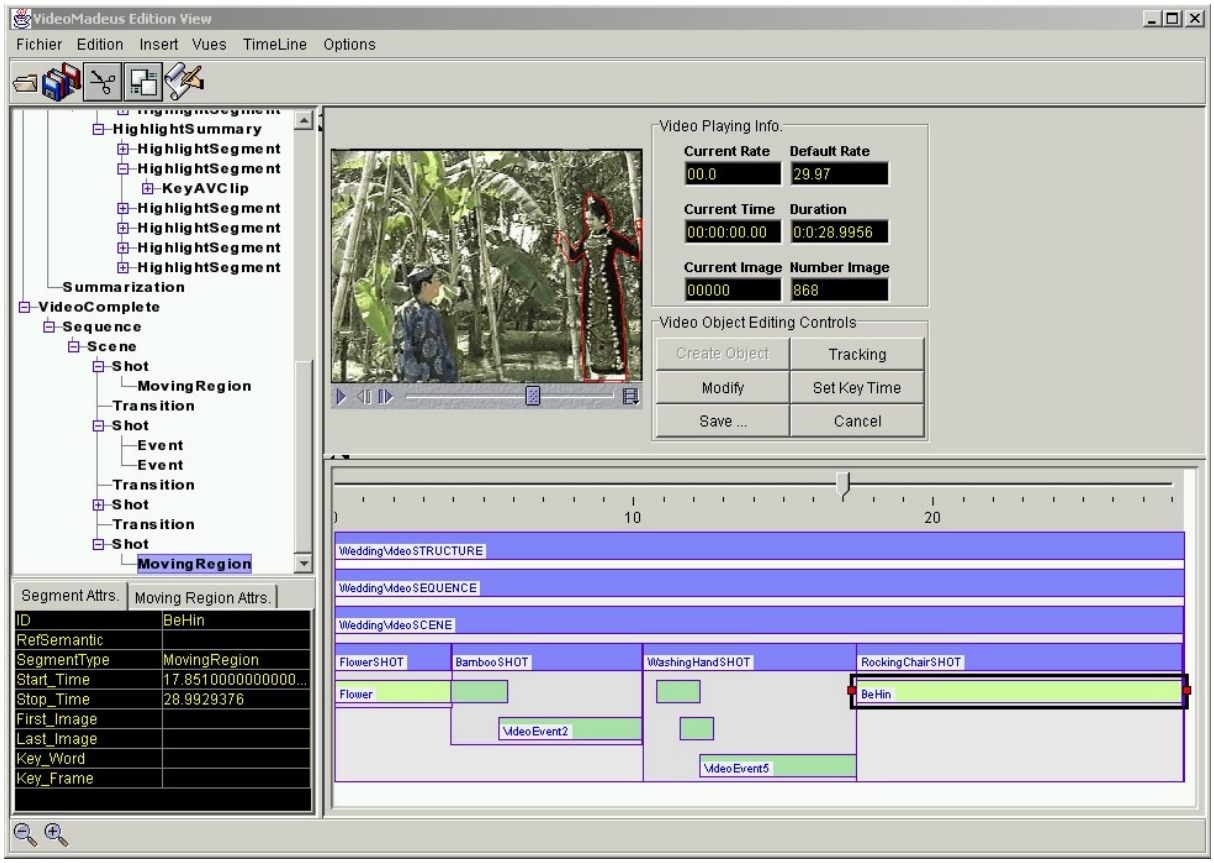


FIG. 2.9 – *Interface de l'environnement auteur* Mdéfi *: vues hiérarchique (à gauche) et temporelle (en bas à droite) du contenu*

***Vizard* - Projet EU-IST** Le projet européen **Vizard** a pour but de permettre l'annotation, la publication et la gestion de collections de vidéos. Le système Vizard se compose de trois modules dont **VAnnotator**, l'outil permettant de décrire des vidéos[21]. Cet outil s'attache particulièrement à traiter le problème de la gestion de la temporalité des documents audiovisuels. Il définit la notion de point de vue (*video-lenses*) pour l'annotation, et de ligne de temps (*timeline*) qui est composée de plusieurs pistes et où chacune des pistes correspond justement à un point de vue [Costa *et al.*, 2002].

Le modèle de la ligne de temps proposé est similaire visuellement à ce que propose maintenant les outils d'édition multimédia tels que *Adobe Premiere* ou *Macromedia Director*. La ligne de temps est un ensemble d'événements qui ont une durée spécifique. Ces événements sont organisés dans des pistes, où ils occupent des segments mutuellement exclusifs, et sont qualifiés par des valeurs. Chaque point de vue a une sémantique dans la mesure où il définit l'interprétation à donner aux événements et le type des valeurs qualifiant ceux-ci. Le modèle permet aussi de définir des relations entre les événements afin d'établir des structures autres que celle linéaire et implicite induite par l'ordre séquentiel des événements que l'on place sur chaque ligne de temps correspondant aux différentes pistes. Finalement, le système utilise le format MPEG-7 pour le stockage et la recherche dans les descriptions des vidéos.

Cet outil pose donc clairement le problème de la description de la structure des documents audiovisuels et apporte une bonne solution avec la gestion des différentes vues composant la ligne de temps. Il permet ainsi de produire des descriptions extrêmement structurées des vidéos et de naviguer ensuite dedans. Cependant, les vues offertes par le système sont pour l'instant restreintes, puisqu'elles ne permettent que d'annoter les segments par des images clés et du texte libre. La possibilité de fabriquer ses propres points de vue, qui pourraient définir des champs contrôlés, est à l'étude et fera certainement de Vizard l'environnement générique qu'il prétend être.

***VIDETO* - ZGDV** L'outil de description vidéo **VIDETO**[22] est développé au laboratoire ZGDV à Rostock en Allemagne (figure 2.10). Il permet aux fournisseurs de contenus de générer des descriptions spécifiques pour un grand nombre d'application. L'interface propose d'abord de détecter les plans qui composent la vidéo, mais le résultat peut ensuite être corrigé à la main. L'annotation des segments est alors complètement paramétrable puisqu'elle se compose de différents champs qui sont totalement prédéfinis dans des modèles (*template*) XML. Ces champs sont renseignés lors de l'annotation et ont comme valeur soit du texte libre, soit des termes provenant d'un vocabulaire contrôlé. Les descriptions peuvent enfin être exportées dans différents formats, dont MPEG-7, *via* des feuilles de style XSLT.

La possibilité de définir des modèles indiquant le type d'information à remplir pour décrire les segments vidéos fait de VIDETO un outil particulièrement intéressant. Cependant, aucune distinction n'est faite selon le type des segments puisque les propriétés permettant de les annoter sont les mêmes pour tous. Cette limitation réduit l'intérêt documentaire puisqu'elle empêche toute description de la structure de la vidéo.

2.4.2 Les modèles de représentation orientés « connaissances »

Ce large tour d'horizon des outils d'annotation vidéo effectué, nous constatons que tous permettent de décrire finement la structure des documents audiovisuels en se basant sur des

21. Une version d'évaluation du système est disponible à `http://www.video-wizard.com/index-e.htm`.
22. Une version d'évaluation est disponible à `http://www.rostock.zgdv.de/ZGDV/Abteilungen/zr1/Produkte/videto/index_html_en`.

FIG. 2.10 – *Interface de l'outil* VIDETO. *La vidéo (en haut à gauche) est découpée en plans (à droite) et est annotée en fonction d'un schéma (en bas à gauche)*

modèles de ligne de temps plus ou moins évolués. À ce titre, nous pouvons les qualifier d'outils « purement documentaires ». En revanche, la description sémantique du contenu est généralement faible puisque, dans le meilleur des cas, elle se résume à l'utilisation d'un thésaurus très général comme *WordNet*. Or, décrire la sémantique des contenus de manière à ce que la machine puisse accéder au sens des documents est une priorité affichée de notre système documentaire. Nous décrivons dans la suite des systèmes qui ont également affiché cette priorité lors de leur élaboration. En particulier, nous présentons le modèle des **Strates-IA**, puis son évolution, le système **E-SIA**. Nous terminons cette section par un autre système auquel nous avons nous-même participé : **OPALES**.

Les Strates Interconnectées par les Annotations (Strates-IA)

Les *Strates Interconnectées par les Annotations* (Strates-IA) sont un modèle de représentation pour l'indexation audiovisuelle mis au point par Y. Prié [Prié, 1999] dans le cadre du projet SESAME[23]. Une description d'un document audiovisuel dans ce modèle correspond à la mise en place de graphes de description éclairés par une base de connaissances.

L'élément de base du modèle des Strates-IA est l'*annotation primitive*, qui consiste à associer une caractéristique interprétée à un morceau du flux audiovisuel. Ainsi, le modèle définit l'*unité audiovisuelle* (UAV) comme une entité abstraite caractérisée par un identifiant et deux instants du flux permettant de situer sans ambiguïté la partie du document audiovisuel qu'elle représente. Une UAV existe dès lors qu'on lui attache un *élément d'annotation* (EA) qui est l'expression d'une caractéristique d'annotation. L'élément d'annotation est alors en *relation d'annotation* (R_a) avec l'UAV. Mais cet élément est en fait une instance d'un concept, et il possède à ce titre des valeurs pour les différents attributs du concept (par exemple, il possédera toujours une valeur pour l'attribut `nom`). Les EA sont donc mis en relation avec des *éléments d'annotation abstraits* (EAA) justifiant ainsi une approche orientée vers les connaissances. Les EAA sont en effet des concepts qui se structurent dans une base de connaissances. Ils sont définis à l'aide d'attributs typés, mais essentiellement par rapport aux autres concepts via la relation de spécialisation (R_s).

Enfin, les différents EA peuvent être reliés par une relation générique dénommée *relation élémentaire* (R_e). La pauvreté sémantique de cette relation est présentée comme un atout du modèle puisqu'elle permet de mettre en relation deux EA quelconques et donc de prendre en compte la structuration des annotations à différents niveaux. Si un usage demande d'avoir des relations plus détaillées, alors l'auteur conseille d'en faire un nouveau concept dans la base de connaissances. La figure 2.11 illustre ces différents éléments avec un graphe Strates-IA correspondant à l'annotation d'une interview diffusée dans le magazine sportif `Stade2` et dont un plan montre le coureur cycliste `Sandy Casar` qui est interviewé.

Le modèle des Strates-IA permet donc de définir – dans une certaine mesure – la sémantique des éléments d'annotation. De plus, il est possible de définir des graphes génériques d'annotation qui prennent la forme de *graphes potentiels*. Ceux-ci peuvent être assimilés à des schémas de descriptions qui seront instanciés lors de l'annotation. Ces graphes représentent donc des visées de description, liées à des tâches particulières, qui jouent le rôle de patron d'indexation.

Nous constatons donc que ce modèle peut convenir dès lors qu'il est utilisé dans le cadre pour lequel il a été conçu, à savoir l'annotation la plus libre possible des documents audiovisuels tout en fixant un contrôle permettant de faire quelques calculs informatiques. En revanche, il nous paraît nettement moins adapté ni dans un cadre documentaire, où les descriptions doivent être

23. SESAME (Système d'Exploration de Séquences Audiovisuelles et Multimédia enrichi par l'Expérience) est un projet qui a été financé par le CNET-CCETT France Télécom dans le cadre d'une CTI (Concertation Thématique Informelle) sur le thème *indexation et recherche par le contenu pour les services multimédias*.

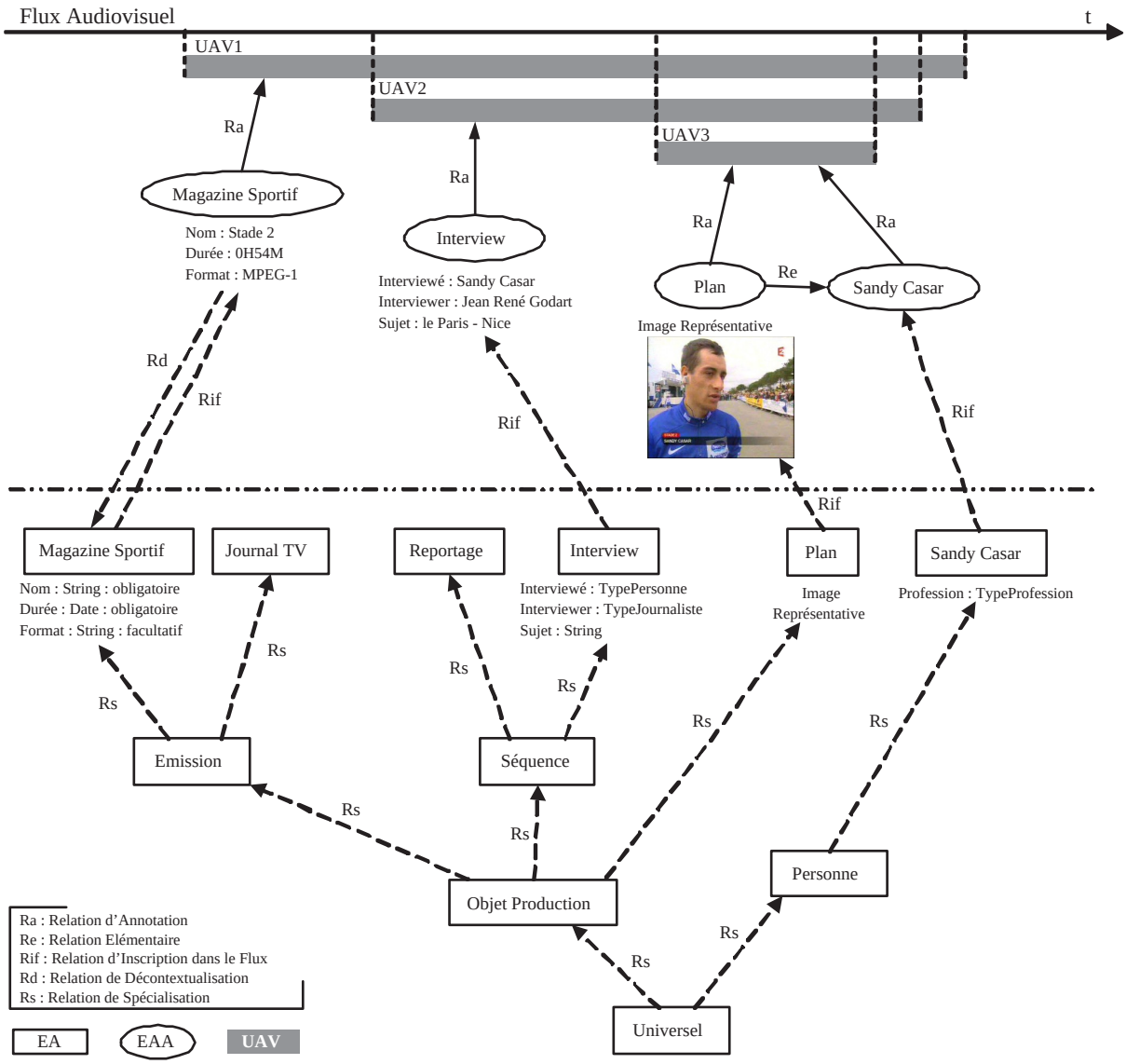


FIG. 2.11 – *Structuration des annotations dans le modèle Strates-IA*

systématiques, ni dans un cadre de manipulation des contenus évolué, où les descriptions doivent être formalisées.

En effet, la seule relation existante entre les EA est la relation élémentaire. Par exemple, il n'y a pas de relation d'ordre entre ces éléments. Ainsi, en dépit des graphes potentiels, il n'est pas possible de contraindre la structure d'un document ce qui est pourtant primordial pour assurer la systématicité des descriptions. Concernant l'aspect connaissance, la relation de spécialisation permettant de hiérarchiser les EAA est singulière puisqu'elle ne permet pas de faire hériter les attributs des concepts. De plus, la distinction entre connaissance ontologique et connaissance assertionnelle est floue puisque chaque individu est présent deux fois dans le modèle : comme instance (EA) lorsqu'il est utilisé pour annoter (les attributs sont valués), mais également comme concept (EAA) pour lui spécifier des attributs (les attributs sont typés). Au final, et de l'aveu même de l'auteur, la base de connaissances produite fait plus penser à un thésaurus enrichi (grâce aux attributs) qu'à un modèle formel sur lequel il sera possible de conduire des raisonnements.

Ce modèle a ensuite fait l'objet de nouvelles recherches que nous décrivons par la suite.

Les Strates-IA Étendues (E-SIA)

Le modèle des Strates-IA Étendues (E-SIA), élaboré par E. Egyed-Zsigmond, est présenté dans [Egyed-Zs., 2003]. Il se veut un système d'annotation et d'exploitation de documents multimédias fonctionnant sur des ressources distribuées sur un réseau. Ainsi, tous les éléments du modèle (documents, annotations, éléments abstraits) sont distribués et plusieurs personnes peuvent consulter le système en laissant des traces d'utilisation qui permettront d'assister les futurs utilisateurs. Par rapport à son prédécesseur, le modèle E-SIA propose donc une opérationalisation du modèle théorique et une représentation XML de celui-ci.

De plus, il apporte quelques nouveautés au modèle. Les éléments d'annotation abstraits (EAA) peuvent désormais se regrouper en *dimension d'analyse* (DA), *via* la relation d'appartenance (R_{app}), offrant une sorte de point de vue pour l'annotation. Lors de l'annotation, l'utilisateur choisi les dimensions d'analyse à appliquer restreignant du même le vocabulaire possible pour annoter. La notion de *schéma de description* (SD) est également approfondie puisqu'elle permet de pré-sélectionner des dimensions d'analyse (avec la relation de contenance, R_{cont}) et donc de guider véritablement l'annotation. Les SD servent également à exprimer des contraintes temporelles entre les DA (et du même coup les EAA). L'auteur propose d'utiliser les 13 relations de Allen [Allen, 1983] pour les exprimer, tout en précisant qu'elles ne seront pas mises explicitement lors de l'annotation, mais que les contraintes induites seront vérifiées lors du repérage temporel des fragments de documents (UAV).

Enfin, le statut des EAA évolue fortement puisque leur structuration hiérarchique est abandonnée. En effet, désormais tous les EAA sont mis au même niveau et n'ont plus vocation à être des concepts. Ils sont simplement des termes qui forment le vocabulaire utilisable pour l'annotation, ce vocabulaire pouvant contenir des synonymes, des variations langagières...

Nous constatons donc que cette évolution du modèle apporte des avancées sur le plan documentaire puisqu'elle permet de mieux structurer les descriptions. Cependant, les éléments ajoutés ont tendance à complexifier le modèle et il nous paraît difficile à utiliser : les schémas de descriptions sont nécessairement liés à des points de vue d'annotation (DA) qui restreignent le vocabulaire utilisable (EAA) qui sera instancié (EA), tout cela étant autant d'éléments du modèle à définir. Plus grave, l'évolution du statut des EAA constitue selon nous un véritable retour en arrière puisqu'elle annihile toute possibilité d'inférences. En effet, le thésaurus formé par les EAA dans le modèle Srates-IA, et qui était déjà une représentation des connaissances minimale, disparaît dans le modèle E-SIA sous le prétexte qu'une hiérarchie unique conduirait à construire

une ontologie universelle. Mais à cet argument, on peut en opposer un autre : les dimensions d'analyse pourraient être l'occasion de proposer des structurations différentes du domaine afin de constituer des ontologies régionales pour l'indexation.

OPALES

Nous terminons cette section par la présentation d'un système dans lequel nous avons nous-même participé. OPALES (*Outils pour des Portails AudiovisueLs Éducatifs et Scientifiques*[24]) est un système technique permettant à une communauté éducative ou scientifique de se fédérer autour d'une base de contenus et de pratiques. Le résultat du projet est une application logicielle permettant d'indexer des documents audiovisuels, de naviguer et de rechercher des séquences dans une base de contenus à l'aide de requêtes structurées et d'un moteur d'inférences (figure 2.12) [Nanard *et al.*, 2003].

Dans le système OPALES, l'annotation de séquences vidéos peut être libre (du simple texte), structurée (à l'aide de formulaires) ou préférablement formalisée. L'annotation formelle est représentée par un *graphe conceptuel*[25] dont les éléments sont des concepts et des relations structurés dans une ontologie. Un acteur d'une communauté éducative ou scientifique peut, dans le système OPALES, accéder à un contenu, y apporter les siens, annoter et indexer un contenu en fonction de ses propres utilisations, et construire son propre espace de travail *via* la notion de point de vue. Cette notion s'apparente aux *dimensions d'analyse* (DA) du modèle des Strates-IA.

Notre participation dans ce projet a été de développer l'éditeur d'ontologies **DOE**, que nous présenterons dans le prochain chapitre (section 3.4.4, page 93). Cet outil permet de modéliser un domaine de connaissance et de générer le support sur lequel les graphes conceptuels seront construits. L'outil **CoGITANT**[26], intégré dans le système, joue le rôle d'un système à base de connaissances : il permet de dessiner des graphes, de les stocker et de mener des raisonnements sur l'ensemble des connaissances grâce à une implémentation efficace de l'opérateur de projection de graphes [Genest, 2001].

Finalement, le système OPALES est – à notre connaissance – le meilleur outil documentaire pour décrire des documents audiovisuels de manière à ce que la machine puisse les manipuler selon la sémantique de leur contenu. Il nous a donc clairement influencé sur les choix de notre système de représentation. Mais OPALES souffre également de certaines limitations. La première est liée au système CoGITANT qui oblige à déclarer tous les individus utiles aux descriptions dans le support : il n'est ainsi pas possible de créer des individus dynamiquement au moment où l'on décrit les documents. Les inférences réalisées sont locales dans la mesure où elles n'utilisent que le support et un graphe particulier sans tenir compte des autres connaissances assertionnelles (c'est-à-dire des autres graphes) de la base. Ce problème a été discuté dans [Euzenat, 2002] où il est suggéré d'introduire de la connaissance de contexte qui correspondrait à un ensemble d'individus supposés connus par tout le monde et qui pourraient être utilisés pour définir le modèle des connaissances. La seconde limitation du système OPALES provient selon nous des langages utilisés et de leur éloignement par rapport aux standards actuels. Ainsi, la description de la structure des documents ne met pas à profit les résultats obtenus lors de l'élaboration du

24. OPALES est un projet PRIAMM (2001-2003) qui réunit des acteurs du monde audiovisuel public éducatif (Centre National de Documentation Pédagogique, La Cinquième-BPS), des partenaires confirmés sur l'indexation, la représentation des connaissances et la consultation multimédia (l'INA, le LIRMM de l'Université Montpellier II, équipes IHM et Graphes Conceptuels), des acteurs importants du monde industriel (CS-Systèmes d'Information, GIP Renater) et des utilisateurs directement visés par le projet (Maison des Sciences de l'Homme).

25. Le formalisme des graphes conceptuels sera abordé dans le chapitre suivant à la section 3.2.2, page 70.

26. CoGITANT est actuellement maintenu par D. Genest. Il est disponible gratuitement à `http://cogitant.sourceforge.net/`.

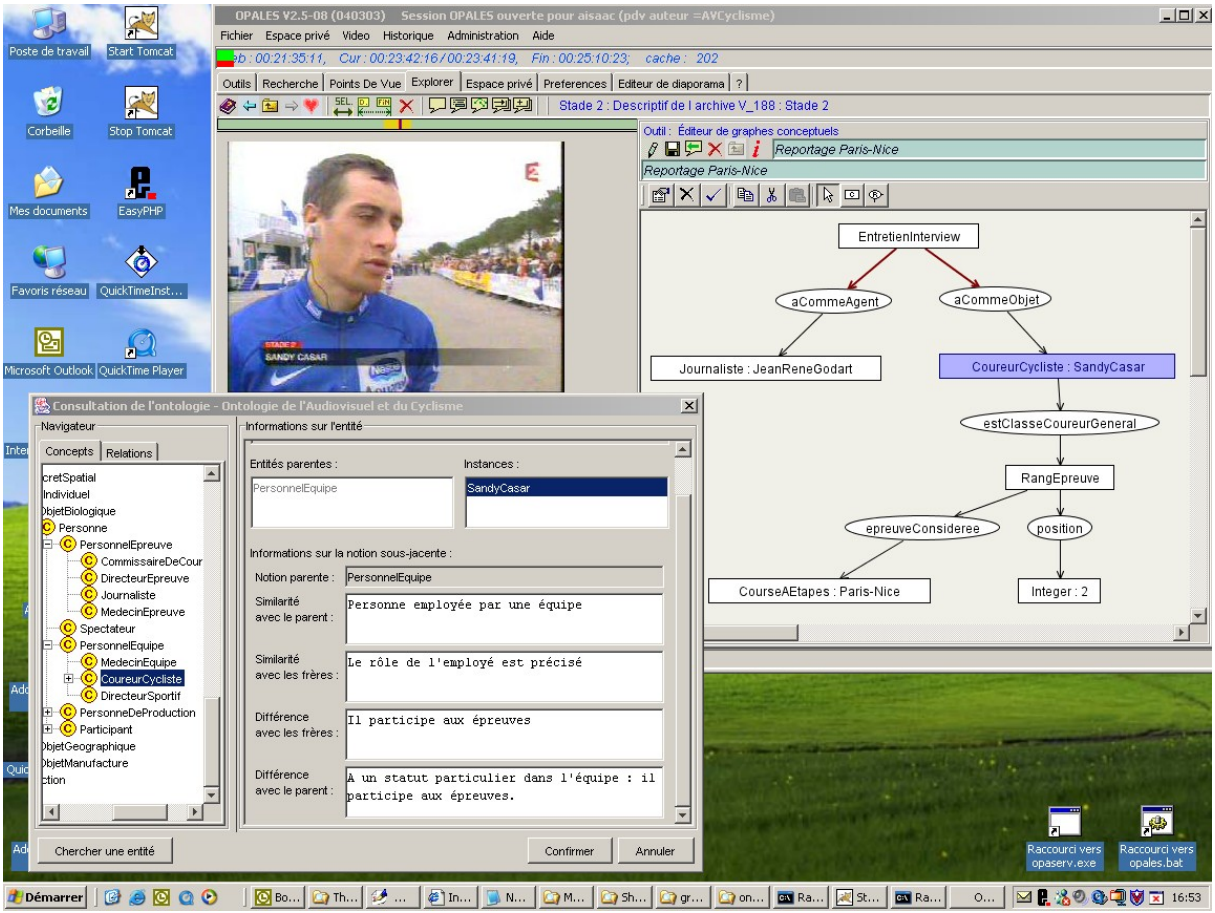


FIG. 2.12 – *Exemple d'annotation d'une vidéo dans le système OPALES : l'annotation est un graphe conceptuel (à droite) dont le support est généré par l'éditeur d'ontologie* DOE *(en bas à gauche)*

standard MPEG-7. De même, le modèle de représentation des connaissances fait fi des langages en cours de standardisation autour du Web sémantique alors qu'ils sont tout-à-fait appropriés. Nous verrons que, au contraire, notre proposition s'intègre complètement dans les efforts de standardisation actuels.

2.5 Conclusion

Dans ce chapitre, nous avons présenté des modèles, des langages et des outils issus de l'ingénierie documentaire pour décrire des documents audiovisuels. Ces modèles s'appuient sur le paradigme des documents structurés qui se développe autour des langages de marquage et notamment autour de XML. Nous avons décrit ce langage ainsi que les différentes manières de contrôler les structures qu'il peux exprimer. Après avoir survolé les langages permettant de présenter des documents multimédia, nous avons étudié les langages permettant de les décrire, que ce soit globalement (données bibliographiques) ou de manière détaillée. Nous avons ainsi longuement présenté la norme MPEG-7 (et ses outils) qui se veut être le langage standard de description de contenu multimédia. Nous avons alors démontré que ce langage est satisfaisant pour décrire finement la structure temporelle des documents audiovisuels.

Nous avons constaté une évolution des systèmes d'indexation audiovisuel avec le développement de modèles permettant de représenter de plus en plus de connaissance. Cette évolution répond également à notre besoin, puisque nous avons vu qu'il était nécessaire d'exprimer la sémantique des descripteurs dans un langage formel et utilisable par la machine pour véritablement permettre la manipulation des contenus audiovisuels. Mais ces systèmes ne vont pas assez loin dans la représentation formelle des informations ou ne tirent pas partie des standards en cours de développement. Nous avons également démontré que le langage MPEG-7 ne remplit pas cet objectif dans la mesure ou les descripteurs fournis, bien qu'ils aient un libellé standardisé, n'ont pas de sémantique exploitable pour la machine. L'évolution naturelle consiste donc à utiliser de véritables langages de représentation des connaissances pour que les descripteurs utilisées dans les descriptions aient un sens pour la machine. Nous présentons donc un état de l'art de ces langages et de la construction d'ontologies dans le chapitre suivant.

Chapitre 3

Les langages de représentation des connaissances et les ontologies

Sommaire

3.1 Les ontologies **66**
3.2 Quelques formalismes de représentation des connaissances **69**
3.2.1 Un peu d'histoire 70
3.2.2 Les graphes conceptuels 70
3.2.3 Les logiques de description 74
3.3 La représentation des connaissances sur le Web **77**
3.3.1 Shoe et Ontobroker : des précurseurs 78
3.3.2 Les langages RDF et RDFS 79
3.3.3 Vers un langage pour exprimer des ontologies sur le Web 82
3.4 La construction des ontologies **86**
3.4.1 Les méthodologies de construction existantes 86
3.4.2 Un problème non résolu : comment obtenir les taxinomies? 88
3.4.3 Les outils existants 91
3.4.4 DOE - *Differential Ontology Editor* 93
3.5 L'exploitation des ontologies **94**
3.5.1 Le stockage des bases de connaissances 94
3.5.2 Les moteurs d'inférence 95
3.6 Conclusion **95**

D'après le dictionnaire *Larousse*, « l'**Ontologie** est la partie de la métaphysique qui s'intéresse à l'Être en tant qu'Être ». Mais en informatique, ce terme est désormais généralement utilisé pour qualifier toute sorte de classification. Depuis ses tout débuts, l'Intelligence Artificielle essaie de concevoir des systèmes qui ont le pouvoir de simuler la connaissance à travers l'utilisation de mécanismes de raisonnement automatique. Ces mécanismes font généralement appel à des bases de connaissances (BC), qui sont des ensembles de termes et d'axiomes permettant de contraindre l'interprétation et d'inférer de nouvelles informations à partir de faits. L'étude rigoureuse du développement et des propriétés de ces bases de connaissances a donné naissance à tout un domaine de recherche, nommé l'Ingénierie des Connaissances.

Dans ce chapitre, nous commençons par définir la notion d'« *ontologie* » en Ingénierie des Connaissances (section 3.1). Nous présentons ensuite quelques formalismes de représentation des

connaissances qui sont à l'origine des langages permettant d'exprimer des ontologies (section 3.2). Nous verrons alors comment ces formalismes ont évolué tout récemment, sous l'influence du langage XML, pour être adaptés à un Web qu'on nous promet sémantique (section 3.3). En particulier, les langages OWL et RDF, permettant respectivement de représenter des ontologies et des assertions à l'aide de ces ontologies, seront abordés. Nous étudierons également le problème délicat de la construction des ontologies en proposant une revue des méthodologies et des outils existants. Nous verrons pourquoi nous nous sommes tournés vers la méthodologie proposée par B. Bachimont et comment nous avons cherché à l'implémenter dans un outil d'aide à la structuration d'ontologies (section 3.4). Nous compléterons cet état de l'art en présentant quelques outils permettant de stocker des bases de connaissances, et de les utiliser (section 3.5).

3.1 Les ontologies

Les ontologies en Ingénierie des Connaissances

Depuis le début des années 90, l'ingénierie des connaissances a sans conteste contribué à populariser le terme « ontologie ». En 1993, T.R. Gruber est crédité pour avoir proposé une première définition de ce terme : *une ontologie est une spécification partagée d'une conceptualisation* [Gruber, 1993]. Mais cette définition laisse déjà la porte ouverte à de nombreuses interprétations puisque, comme l'a montré notamment [Guarino, 1998], elle renvoie à une clarification de la notion de « conceptualisation ». Les travaux menés ensuite par N. Guarino et C. Welty ont permis, dans une large mesure, de définir un cadre conceptuel censé apporter cette clarification [Guarino and Welty, 2000a; Guarino and Welty, 2000b]. Ce cadre consiste : d'une part, à préciser la notion de domaine (d'« application » ou de « discours ») en distinguant parmi les entités peuplant un domaine, les individus et les propriétés[1] ; d'autre part, à préciser les modes de définition des propriétés en consignant dans l'ontologie les (méta)-propriétés *essentielles* (identité, unicité...) les caractérisant.

Contrairement aux définitions précédentes, B. Bachimont propose de s'appuyer directement sur la formalisation et la représentation des connaissances. L'ontologie permet alors d'associer une sémantique aux constructions syntaxiques du langage formel [Bachimont, 2000a] :

> « *Définir une ontologie pour la représentation des connaissances, c'est définir, pour un domaine et un problème donnés, la signature fonctionnelle et relationnelle d'un langage formel de représentation et la sémantique associée.* »

Autrement dit, l'ontologie permet de donner une sémantique aux objets primitifs d'un domaine. Le problème est qu'il faut pour cela déterminer quelles sont les notions élémentaires à partir desquelles toutes les connaissances du domaine sont construites [Bachimont, 2000a]. Nous verrons dans la section 3.4.2 quelle approche B. Bachimont propose pour traiter ce problème.

La démarche consistant à acquérir puis à modéliser des connaissances est courante dans l'élaboration des systèmes à base de connaissances (SBC). En effet, la construction de ces systèmes nécessite de spécifier : d'une part, les connaissances du domaine modélisé et d'autre part, les connaissances de raisonnement décrivant les règles heuristiques d'utilisation de ces connaissances du domaine [Charlet, 2003]. L'ontologie se retrouve donc au centre des SBC. Dans un contexte où le but final est de construire un artefact informatique opérationnel, l'ontologie devient un

1. Pour ces auteurs, les propriétés sont des *universels* correspondant soit à des *concepts*, si la propriété est unaire, soit à des *relations*, si la propriété est d'arité supérieure à un. Des propriétés peuvent également être attribuées à des propriétés, et dans ce cas on parle de méta-propriétés. Leur définition conduit à considérer des ontologies de propriétés [Guarino and Welty, 2000a].

modèle des objets existants qui fait référence à ces objets à travers un niveau d'abstraction supplémentaire, les concepts du domaine. On retrouve ici la notion de « conceptualisation » introduite par T.R. Gruber, dans le sens où il s'agit bien de définir dans une ontologie des concepts. Reprenant les spécifications données dans [Gruber, 1993] et [Uschold and Gruninger, 1996], J. Charlet propose finalement la définition suivante de ce qu'est une ontologie [Charlet, 2003; Charlet *et al.*, 2003] :

> « *Une ontologie implique ou comprend une certaine vue du monde par rapport à un domaine donné. Cette vue est souvent conçue comme un ensemble de concepts - e.g. entités, attributs, processus -, leurs définitions et leurs interrelations. On appelle cela une conceptualisation.*
>
> « *[...]*
>
> « *Une ontologie peut prendre différentes formes mais elle inclura nécessairement un vocabulaire de termes et une spécification de leur signification.*
>
> « *[...]*
>
> « *Une ontologie est une spécification rendant partiellement compte d'une conceptualisation.* »

Cette définition précise les précédentes et laisse déjà augurer de quoi seront constituées les ontologies (voir *supra*). Mais avant de conclure ces quelques définitions, il nous faut nous interroger sur l'utilisation – pour certains abusive – du terme « ontologie ».

Force est de constater que ce terme est de plus en plus utilisé, tout en renvoyant à des objets assez différents dans leur nature et dans leur but. En témoignent les travaux de C. Welty, M. Gruninger et M. Uschold rapportés dans [Smith and Welty, 2001] et illustrés par la figure 3.1 qui, dès 1999, ont tenté d'établir le spectre couvert par les artefacts qui ont pu, à un moment donné, être classifiés comme des ontologies. Un système d'information aussi simple qu'un catalogue peut être considéré comme une ontologie des produits vendus par une société. Légèrement plus complexe, le système peut fournir un ensemble de textes en langue naturelle sur lesquels on pourra établir des correspondances de chaînes de caractères. Le glossaire, en classant les termes référencés, impose une structure au système d'information. Les thésauri permettent d'organiser les termes décrits en hiérarchie *via* la relation de spécialisation. La représentation des connaissances et l'ingénierie logiciel orienté objet utilisent ces taxinomies pour faire hériter des propriétés dans les hiérarchies de classes. Les systèmes de *frames* ajoutent des relations entre les objets et des restrictions sur quoi et comment classes et objets peuvent être liés. Finalement, le système d'information « ontologique » le plus complexe utilise des axiomes de la logique du premier ordre, d'ordre supérieur ou de la logique modale.

Derrière cet apparent chaos, il y a quand même des points communs qui se dessinent : la nécessité d'avoir une classification partagée des entités manipulées par exemple. Si, aujourd'hui, le terme « ontologie » est utilisé pour qualifier certaines ressources terminologiques[2], il renverra pour notre part à un objet formalisé. L'ontologie devant être par la suite un artefact informatique, elle doit être basée sur une théorie logique qui permet de la manipuler formellement car elle sera utilisée au sein d'un module logiciel, type SBC, nécessitant d'effectuer des inférences.

2. Pour mieux clarifier quelles ressources terminologiques peuvent être qualifiées d'ontologiques, certains parlent de *ressources termino-ontologiques* ou RTO [Bourigault *et al.*, 2003].

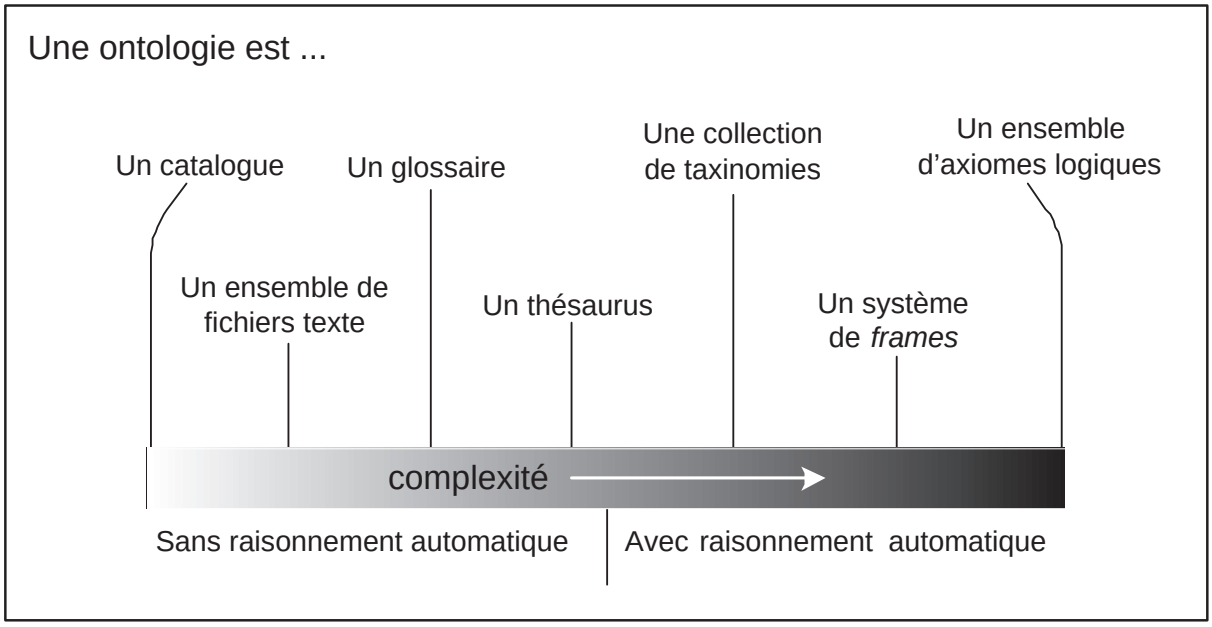


FIG. 3.1 – *Recensement des objets qui ont pu, à un moment donné, être classifié comme des ontologies, d'après [Smith and Welty, 2001]*

Les briques de base des ontologies

Comme nous l'avons dit plus haut, les ontologies fournissent le vocabulaire propre à un domaine et fixent – avec un degré de formalisation variable – le sens des concepts et des relations entre ceux-ci. Ces concepts (et ces relations) sont organisés *via* la relation de subsomption pour former une taxinomie. L'usage montre que cette taxinomie – elle n'est jamais absente, et quelques fois la définition des ontologies se résume même à sa seule construction – constitue la porte d'accès à l'ontologie. Nous détaillons ci-après l'ensemble de ses composantes :

- Les **concepts** (ou classes). Les concepts doivent être compris dans un sens très large : il s'agit de n'importe quoi sur lequel quelque chose est dit, par exemple, la description d'un ensemble d'objets, d'une tâche, d'une fonction, d'une stratégie, d'un processus de raisonnement...

- Les **propriétés** (ou attributs). Les propriétés sont les caractéristiques attachées aux concepts ; elles sont valuées.

- Les **liens organisant les concepts**. La relation de subsomption *is-a*, qui définit un lien de généralisation, est la plus utilisée pour structurer les ontologies. Mais ce n'est pas la seule et, surtout, pas la plus utile dans certain cas [Charlet, 2003]. Dans le domaine de l'audiovisuel, par exemple, il est indispensable de décrire les séquences composant une émission ou les émissions composant une grille des programmes. On a alors besoin de relations de partie-tout (ou *méronymie*) [Winston *et al.*, 1987].

- Les **relations**. Les relations unissent les concepts ensemble pour construire des représentations conceptuelles complexes. La relation *is-a* qui structure l'ontologie est une relation au même type que les autres, si ce n'est qu'elle a été choisie comme une relation de structuration de l'arbre ontologique et qu'elle est par conséquent implicite.

- Les **axiomes**. Les axiomes sont utilisés pour modéliser des assertions toujours vraies. Ils peuvent intervenir dans la définition des concepts et des relations, ou alors sous forme de règles.

- Les **individus** (ou instances). Les individus représentent des éléments singuliers.

Cette énonciation nous permet de clarifier encore un peu plus l'objet informatique « ontologie », mais elle appelle quelques remarques. La formalisation ontologique repose sur le fait que le monde ressemble à la façon dont il est décrit dans la théorie des modèles : on suppose implicitement qu'il existe des individus pour les énumérer explicitement. La difficulté est donc centrée sur l'élaboration du modèle. Des choix de modélisation sont ainsi effectués durant les différentes étapes de l'élaboration de l'ontologie. Ils doivent être assumés par le concepteur d'ontologies et idéalement dictés par la méthodologie de construction. On peut remarquer ainsi que les concepts et les relations sont duals l'un par rapport à l'autre lorsqu'il s'agit de les définir sous forme d'axiomes. Un autre choix de conception est de décider si une connaissance doit être modélisée dans une propriété ou à l'aide d'une relation pointant sur un autre concept. Un critère souvent retenu consiste à dire qu'il s'agit d'une propriété dès lors que les valeurs possibles sont d'un type dit primitif (entier, chaîne de caractères), alors que les valeurs possibles d'une relation sont d'un type dit complexe (c'est-à-dire un autre concept de l'ontologie). Mais cette frontière peut évidemment être remise en question. Enfin, la présence ou non des individus dans l'ontologie est sujette à débat. En effet, la définition des concepts du domaine peut faire appel à des objets ou des assertions individuelles, ce que [Euzenat, 2002] nomme de la *connaissance de contexte*. Cette connaissance est souvent partagée et nécessaire pour la compréhension du domaine, donc incluse dans l'ontologie. Mais là encore, la frontière entre ce type de connaissances et les assertions émises à l'aide de l'ontologie est une question de choix, dictés par l'application à l'origine de l'ontologie construite.

Nous reviendrons sur ces choix de modélisation dans la section 3.4 qui présente quelques méthodologies et outils de construction d'ontologies. Mais avant cela, nous allons étudier dans la prochaine section deux formalismes de représentation des connaissances qui sont à l'origine des langages permettant d'exprimer des ontologies.

3.2 Quelques formalismes de représentation des connaissances

En toute généralité, représenter des connaissances propres à un domaine particulier consiste à décrire et à coder les entités de ce domaine de manière à ce qu'une machine puisse les manipuler afin de raisonner ou de résoudre des problèmes [Kayser, 1997; Euzenat *et al.*, 2000]. Cette définition met en évidence deux composantes complémentaires de la représentation des connaissances (RC), à savoir l'expression et la manipulation des connaissances. D'une part, les connaissances sont exprimées à l'aide d'un langage formel, dit de description des connaissances. Le langage est doté d'une *syntaxe*, précisant l'ensemble des expressions admissibles du langage et d'une *sémantique* qui permet de fournir un sens aux formules justifiant ainsi la validité des opérations effectuées. D'autre part, le but est de mécaniser un certain nombre de manipulations sur les connaissances exprimées. Ainsi, il sera nécessaire de modifier, compléter, inférer de nouvelles connaissances. Ces manipulations possibles sont spécifiées sous forme de mécanismes respectant la sémantique et opérant sur les éléments de la représentation.

Les travaux en RC ont donné naissance à de nombreux formalismes. Nous présentons tout d'abord un rapide historique de ceux-ci (section 3.2.1) avant de décrire un peu plus profondément

les deux plus étudiés : les graphes conceptuels (section 3.2.2) et les logiques de description (section 3.2.3).

3.2.1 Un peu d'histoire

Comme alternative à la logique classique, l'intelligence artificielle a proposé divers formalismes de représentation des connaissances avec une notation diagrammatique, afin de permettre une représentation plus « intuitive » des connaissances [Baader *et al.*, 2003].

Les *réseaux sémantiques* [Quillian, 1968] sont ainsi fondés sur un modèle graphique permettant de combiner la représentation des concepts (sous forme de nœuds) et des relations entre concepts (sous forme d'arcs). Un mécanisme de raisonnement fondé sur le parcours de la structure graphique permet d'établir des liens entre des concepts du même réseau. De plus, les réseaux sémantiques ont introduit la notion d'héritage, matérialisée par un arc particulier (*sorte-de*) entre les nœuds. Les principales critiques émises contre ce formalisme sont son manque de sémantique précise, notamment en ce qui concerne l'héritage par défaut des propriétés. Diverses évolutions ont alors vu le jour pour l'améliorer, dont un nouveau formalisme graphique – les réseaux d'héritage structurés – implémentés dans le système KL-ONE [Brachman and Schmolze, 1985].

Le langage des *frames* ou *schémas* a été introduit par M. Minsky [Minsky, 1975] en 1975. Un des buts de ce formalisme est de représenter toutes les connaissances relatives à une situation (ou un objet) dans un schéma plutôt que distribuer la connaissance à travers de nombreux axiomes. Une *frame* contient des attributs (*slots*) pour décrire les propriétés de l'objet représenté. Le formalisme permet alors de classifier les *frames* (indiquer si une *frame* est plus spécifique ou plus générale qu'une autre) ou de rechercher l'ensemble des valeurs des propriétés pour un objet particulier.

Au milieu des années 80, deux formalismes inspirés à la fois des réseaux sémantiques et des *frames* ont été développés. Tous deux sont dotés d'une sémantique formelle (à la fois logique, et en théorie des modèles) et avaient comme préoccupation de ne pas seulement représenter les connaissances mais également de permettre le raisonnement sur celles-ci. Nous présentons donc le formalisme des *graphes conceptuels* et celui des *logiques de descriptions* dans les deux sections suivantes. Chacun de ces formalismes définit un ensemble de mécanismes concrets pour consulter des connaissances disponibles ou bien pour inférer de nouvelles connaissances qui ne sont pas déjà explicitement représentées. Ils seront donc examinés du point de vue de leurs capacités descriptives (syntaxe, sémantique) mais aussi inférentielles. Une présentation plus détaillée peut être trouvée dans [Euzenat, 1999].

3.2.2 Les graphes conceptuels

Les *graphes conceptuels* ont été introduits par J. Sowa en 1984 [Sowa, 1984]. L'exposé présenté ici est inspiré des notes de cours de [Euzenat, 1999], lui-même fondé sur le traitement formel des graphes conceptuels présenté dans [Chein and Mugnier, 1992] et [Mugnier and Chein, 1996].

Syntaxe d'un langage de graphes conceptuels

Un graphe conceptuel est un multi-graphe connexe biparti composé de deux types de nœuds : des *concepts* et des *relations*. Les nœuds relations possèdent un ou plusieurs arcs qui les lient aux nœuds concepts. Les concepts désignent des entités, des propriétés, des états ou des événements. Ils consistent en une étiquette, c'est-à-dire un couple (*type de concept*, *référent*). Le référent peut être soit individuel et désigner un individu concret, soit générique (noté par $*$) et il désigne alors n'importe quel individu. Le type est un type d'individus, tiré du *treillis de types*. Chaque type

définit un ensemble de contraintes sur la structure des individus qui sont des instances de ce type. Une relation lie un certain nombre de nœuds conceptuels et est d'arité fixe.

Le vocabulaire (ensemble de types, référents et relations) est défini moyennant la notion de *support* permettant de contraindre la construction des graphes. Ainsi, l'ensemble de types est organisé par la relation dite de sous-typage en un treillis de types. Ce treillis est muni d'éléments minimal et maximal. Un ordre est également spécifié entre les référents individuels qui sont deux à deux incompatibles mais plus spécifiques que le référent générique ($*$). Finalement, un ensemble de graphes-étoiles permet de fixer les signatures des relations utilisées. Un graphe étoile est un graphe composé d'un seul nœud relationnel, mais lié à tous les autres nœuds conceptuels. La signature définit les types maximaux (pour le sous-typage) pour les nœuds conceptuels qui peuvent être connectés à chaque arc. L'exemple de la Figure 3.2 illustre notre propos en donnant un exemple avec le treillis de types et son graphe étoile.

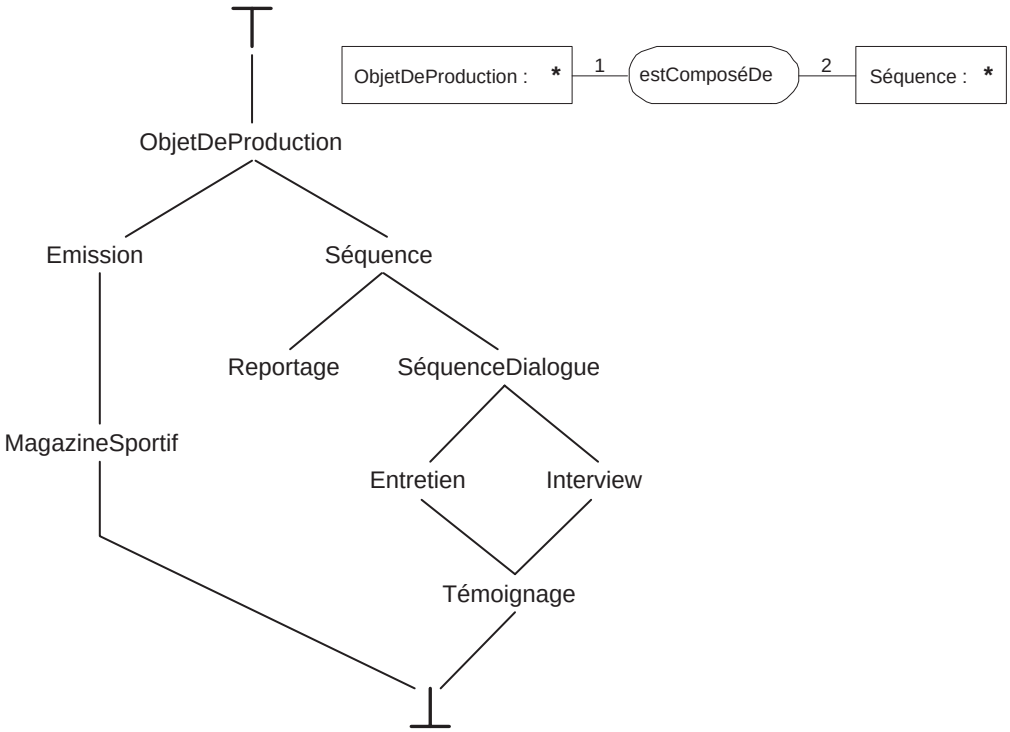


FIG. 3.2 – *Exemple de support avec le treillis de types et un graphe étoile*

Formellement, un support $\mathcal{S}$ est un quintuplet $\langle T_C, T_R, M, :\!\!: B\rangle$ tel que :

- T_C est l'ensemble des types de concepts ;
- T_R est l'ensemble des types de relations ;
- M est un ensemble énumérable de marqueurs contenant $*$ (le marqueur générique) et $\emptyset$ (le marqueur absurde) ;
- $:\!\!:$ est un prédicat de conformité sur $M \times T_C$;
- B est un ensemble de graphes étoiles sur T_C, T_R.

et :

- $\langle T_C, \leq \rangle$ est un ordre partiel fini avec $\top$ et $\bot$ comme supremum et infimum ;
- $T_C \cap T_R = \emptyset$;
- il existe une bijection $\beta : T_R \longrightarrow B$ telle que $\forall r \in T_R, \beta(r)$ soit un graphe étoile pour r ;
- $\langle M, \sqsubseteq \rangle$ soit un treillis construit de telle sorte que $\forall m \in M - \{*, \emptyset\}$, $\emptyset \sqsubseteq m \sqsubseteq *$ et que tous les éléments de $M - \{*, \emptyset\}$ soient incomparables ;
- $\forall m \in M, \forall t, t' \in T_C$,

$$\begin{array}{cc} m :: \top, & (t' \leq t) \wedge (m :: t') \Rightarrow m :: t \\ m :: \bot, & (m :: t') \wedge (m :: t) \Rightarrow m :: t' \wedge t \\ \emptyset :: t, & t \neq \top \equiv * :: t. \end{array}$$

Un graphe conceptuel sur le support $\mathcal{S}$ est donc un S-graphe $G = \langle R, C, U, l \rangle$ tel que :

- $\langle R, C, U \rangle$ soit un graphe biparti fini avec $C \neq \emptyset$;
- l soit une fonction

$$\begin{array}{rcl} l & : & R \longrightarrow T_R \\ & & r \longmapsto type(r) \\ & & C \longrightarrow T_C * (M - \{\emptyset\}). \\ & & c \longmapsto \langle type(c), ref(c) \rangle. \end{array}$$

- $\forall c \in C, ref(c) :: type(c)$;
- $\forall r \in R$,
  1. les arcs sortants de r sont totalement ordonnés de 1 à $degré(r)$ (et leurs extrémités sont nommées $G_i(r)$) ;
  2. $degré(r) = degré(\beta(type(r)))$;
  3. $\forall i \in 1, ...degré(r), type(G_i(r)) \leq type(\beta_i(type(r)))$ (où $\beta_i(r)$ est le nœud concept associé à l'arc numéroté par i dans le graphe étoile associé à $type(r)$ par β.

Des types complexes peuvent ainsi être définis à l'aide des graphes conceptuels et on associe un type à un graphe. La Figure 3.3 nous donne deux exemples de graphe sur le support de la Figure 3.2.

Sémantique

La sémantique d'un graphe conceptuel est donnée par une traduction vers le calcul des prédicats. À chacun des éléments du support, on associe donc son équivalent dans un langage logique d'ordre un. Ainsi, les référents génériques sont traduits par des variables et les référents individuels par des constantes. Chaque type, conceptuel ou relationnel, est traduit par un prédicat. Ceux correspondant aux concepts sont monadiques, alors que ceux correspondant aux relations ont une arité égale au nombre d'arcs de la relation.

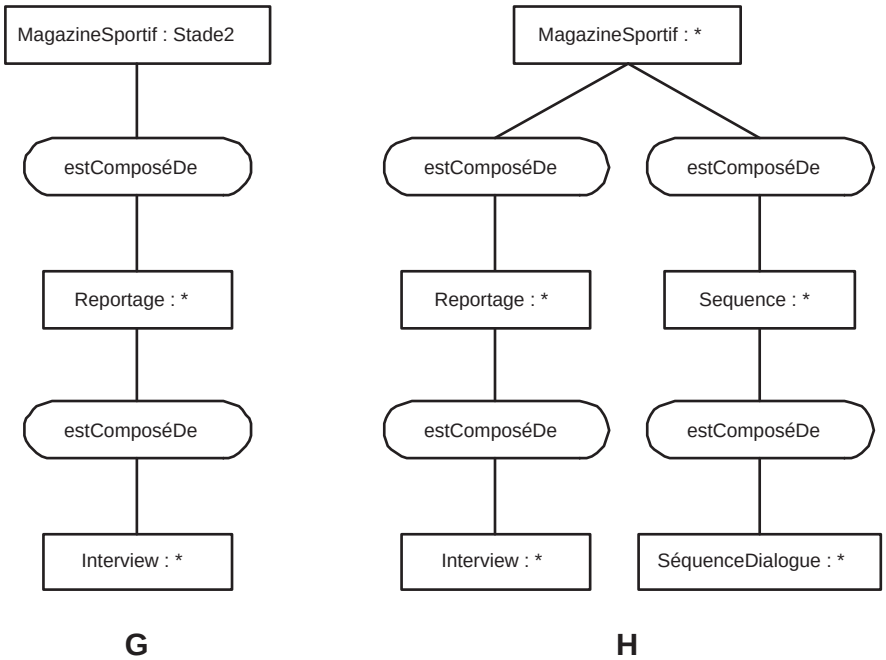


FIG. 3.3 – *Exemples de graphe sur le support de la Figure 3.2*

Soit un support $\mathcal{S} = \langle T_C, T_R, M_{,,}{:}B \rangle$, on considère un calcul des prédicats basé sur un ensemble de constantes $K \supseteq M - \{*, \emptyset\}$, un ensemble de symboles de prédicats $P \supseteq T_C \cup T_R$ et un ensemble de variables V. Les prédicats associés à T_C sont d'arité 1, alors que les prédicats associés à T_R ont une arité égale au nombre d'arcs de la relation sous-jacente. Soit donc un S-graphe $\langle R, C, U, l \rangle$, on définit :

$$
\begin{array}{rcl}
\iota & : & C \longrightarrow V \cup K \text{ injective sur } V \\
 & & c \longmapsto \begin{cases} x \in V \text{ si } ref(c) = * \\ ref(c) \in K \text{ sinon} \end{cases} \\
\phi & : & C \cup R \cup S\text{-graphe} \longrightarrow CP1. \\
 & & c \longmapsto type(c)(\iota(c)). \\
 & & r \longmapsto type(r)(\iota(G_1(r)), ..., \iota(G_{degré(r)}(r))) \\
 & & G \longmapsto \exists_{c \in C; ref(c)=*} \iota(c) \bigwedge_{c \in C} \phi(c) \wedge \bigwedge_{r \in R} \phi(r)
\end{array}
$$

Le graphe G de la Figure 3.3 se traduit en logique du premier ordre par :

$$\exists x, y; MagazineSportif(Stade2) \wedge Reportage(x) \wedge Interview(y) \wedge estComposéDe(Stade2, x) \wedge estComposéDe(x, y)$$

et le graphe H par :

$$\exists x, y, z, t, w; MagazineSportif(x) \wedge Reportage(y) \wedge Interivew(z) \wedge Séquence(t) \wedge SéquenceDialogue(w) \wedge estComposéDe(x, y) \wedge estComposéDe(y, z) \wedge estComposéDe(x, t) \wedge estComposéDe(t, w)$$

Raisonnement

Une relation de généralisation/spécialisation (notée $\preceq_S$) est définie entre graphes conceptuels. Ainsi, un graphe conceptuel G est plus spécifique qu'un autre H si, intuitivement, il contient toutes les informations véhiculées par H et en ajoute d'autres. Cette généralisation/spécialisation s'exprime sur les graphes par une projection qui plaque un graphe sur un autre tout en vérifiant la spécialisation des étiquettes sur les concepts et les relations et la préservation de l'ordre des arcs. Ainsi, $G \preceq_S H$ si et seulement si il existe une projection de H vers G. Formellement, une S-projection entre deux S-graphes $\langle R, C, U, l\rangle$ et $\langle R', C', U', l'\rangle$ est une paire d'applications

$$\langle f : \ R \longrightarrow R',$$

$$g : \ C \longrightarrow C'\rangle$$

telle que :

- $\forall r \in R, \forall i \in 1, ...degré(r), g(G_i(r)) = G'_i(f(r))$;
- $\forall r \in R, l'(f(r)) = l(r)$;
- $\forall c \in C, type'(g(c)) \leq type(c)$ et $ref'(g(c)) \sqsubseteq ref(c)$.

Ainsi, on peut obtenir G en appliquant une suite d'opérations élémentaires de *spécialisation* à H. Les opérations élémentaires de *généralisation* sont obtenues en inversant ces opérations.

Les graphes conceptuels introduisent donc une séparation nette entre les connaissances de nature différente. La description des types conceptuels (et de leurs relations de sous-typage) et la caractérisation des relations sont organisées dans le support, alors que les faits sont organisés au sein d'un ensemble de graphes. Mais il n'y a pas de distinction syntaxique entre classe et instance. Les graphes conceptuels disposent d'un formalisme relationnel et surtout, ne sont pas contraints à un sujet. En revanche, il n'y a pas (ou très peu) de définitions de types puisque le support se limite souvent à une simple hiérarchie.

3.2.3 Les logiques de description

Les *logiques de descriptions* (ou logiques terminologiques, ou encore langages basés sur les termes) constituent l'aboutissement d'une longue période de recherche sur la formalisation dans les langages de représentation de connaissances. La brève introduction présentée ici est principalement inspirée des travaux de B. Nebel [Nebel, 1990] et plus récemment de A. Napoli [Napoli, 1997].

Syntaxe d'une logique terminologique

Les entités manipulées par un langage de description sont les *individus*, les *concepts* et les *rôles*. Les individus correspondent à des entités concrètes de l'univers et sont décrites dans un langage assertionnel (A-box). Les concepts se référent plutôt à des entités génériques. Les rôles sont utilisés pour décrire des relations binaires entre individus. Les concepts et les rôles, eux, sont décrits dans un langage terminologique (T-box). Une base de connaissance comporte donc une terminologie et un ensemble d'assertions. Nous allons voir les caractéristiques de chacun de ces langages.

Langage terminologique (ou T-box)

Au sein d'un système, le langage terminologique est destiné à la composition de termes structurés décrivant les concepts et les rôles. Les termes sont construits sur un ensemble $\mathcal{C}$ de concepts atomiques (dont le concept universel $\top$ et le concept absurde $\bot$) et un ensemble $\mathcal{R}$ de rôles atomiques. À partir de ces deux ensembles, des termes structurés sont construits à l'aide de connecteurs qui permettent de composer des descriptions de concepts ou de rôles, mais aussi de définir des restrictions sur les rôles. Divers langages offrent des ensembles de connecteurs variables. Une terminologie consiste en un ensemble d'introductions de concepts. La description d'un concept précise les caractéristiques communes des individus qui sont ses instances. Cette description peut être une définition ($\dot{=}$) et dans ce cas on parle d'un concept *défini*. Dans le cas contraire, le concept est décrit ($\dot{\leq}$) et on parle de concept *primitif*. La différence entre les deux sortes de concepts est qu'une définition fournit des conditions nécessaires et suffisantes pour l'appartenance de l'individu au concept, alors qu'une description ne fournit que des conditions nécessaires.

Nous donnons ci-dessous un exemple de terminologie avec le langage $\mathcal{FLN}$ qui comprend les connecteurs `and` (conjonction de concepts), `all` (restriction de valeurs), `atleast` et `atmost` (restriction de cardinalité), les symboles $\dot{=}$ (introduction de définition), $\dot{\leq}$ (introduction de description) et $\dot{\oplus}$ (introduction d'exclusion) ainsi que `AnyThing` (concept maximal) et `anyrelation` (relation maximale).

```
Émission              ≤̇ Anything
ÉmissionSimple        ≤̇ Émission
Séquence              ≤̇ Anything
SéquencePlateau       ≤̇ Séquence
Émission              ⊕̇ Séquence
Humain                ≤̇ Anything
Star                  ≤̇ Humain
estComposéDe          ≤̇ anyrelation
aCommeInvité          ≤̇ anyrelation
MagazinePlateau       ≐ (and ÉmissionSimple (all estComposéDe SéquencePlateau)
                        (atleast 3 estComposéDe))
MagazinePlateauInvité ≐ (and MagazinePlateau (all aCommeInvité Star))
                        (atmost 10 aCommeInvité)
```

Cet exemple de terminologie met en jeux des concepts primitifs, et des concepts définis (`MagazinePlateau` et `MagazinePlateauInvité`). Il est possible dans d'autres langages ($\mathcal{ALR}$) de donner une description à un rôle à l'aide de connecteurs appropriés (comme `androle`). Les rôles peuvent alors être primitifs ou définis à leur tour.

Langage assertionnel (ou A-box)

Le langage assertionnel décrit la connaissance individuelle et factuelle. Il permet de formuler des assertions faisant apparaître les termes du langage terminologique. Ainsi, un individu est décrit par ses appartenances à un ou plusieurs concepts de la terminologie et par les rôles qui le lient à d'autres individus. Une base assertionnelle est constituée par des descriptions d'objets et des descriptions de rôles. Celles-ci sont construites à partir d'un ensemble de noms d'objets et en évoquant les termes du langage terminologique sous-jacent. Un exemple de base assertionnelle conforme avec la terminologie du paragraphe précédent est donnée ci-dessous :

```
(MagazinePlateauInvité QuiVeutGagnerDesMillions)
```

```
(Star                  Zidane)
(Star                  Depardieu)
(aCommeInvité          QuiVeutGagnerDesMillions Zidane)
(aCommeInvité          QuiVeutGagnerDesMillions Depardieu)
(aCommeInvité          QuiVeutGagnerDesMillions (atmost 2))
```

Cet exemple comporte six assertions impliquant les objets `Zidane`, `Depardieu` et `QuiVeutGagner-DesMillions`.

Il existe en outre deux contraintes supplémentaires [Euzenat, 1999] :

- l'hypothèse de nom unique, qui signifie que deux noms différents désignent toujours deux entités différentes ;
- l'hypothèse du monde ouvert, qui signifie que tout fait n'étant pas explicitement nié est potentiellement vrai et donc, que l'on ne suppose pas que le monde soit complètement décrit ; Ainsi, il est possible que d'autres `Star` existent.

Sémantique

La sémantique d'un langage terminologique est donnée moyennant une interprétation $\mathcal{E} = \langle \mathcal{D}, \mathcal{I} \rangle$ où $\mathcal{I}$ est une *fonction d'interprétation* vers un ensemble $\mathcal{D}$ dit *domaine d'interprétation*. La fonction $\mathcal{I}$ associe à chaque concept atomique un sous-ensemble de $\mathcal{D}$ et à chaque rôle atomique une relation binaire sur $\mathcal{D} \times \mathcal{D}$. La fonction d'interprétation est aussi étendue sur les expressions du langage de manière inductive comme nous le précise le Tableau 3.1 pour le langage $\mathcal{FLN}$.

$$
\begin{array}{rcl}
\mathcal{I}(\text{Anything}) & = & \mathcal{D} \\
\mathcal{I}(\text{anyrelation}) & = & \mathcal{D} \times \mathcal{D} \\
\mathcal{I}((\text{and } c_1 \ \ldots \ c_n)) & = & \bigcap_{i \in 1 \ldots n} \mathcal{I}(c_i) \\
\mathcal{I}((\text{all } r \ c)) & = & \{x \in \mathcal{D} |\ \forall y, \langle x, y \rangle \in \mathcal{I}(r) \Rightarrow y \in \mathcal{I}(c)\} \\
\mathcal{I}((\text{atleast } n \ r)) & = & \{x \in \mathcal{D} |\ \|\{y \in \mathcal{D} | \langle x, y \rangle \in \mathcal{I}(r)\}\| \ \geq n\} \\
\mathcal{I}((\text{atmost } n \ r)) & = & \{x \in \mathcal{D} |\ \|\{y \in \mathcal{D} | \langle x, y \rangle \in \mathcal{I}(r)\}\| \ \leq n\}
\end{array}
$$

TAB. 3.1 – *Interprétation des termes structurés du langage $\mathcal{FLN}$*

Pour une expression de la terminologie, sa satisfaction par rapport à $\mathcal{E}$, notée $\models_{\mathcal{E}}$ est établie comme suit :

$$
\begin{array}{lcl}
\models_{\mathcal{E}} A \dot{\leq} C & \Leftrightarrow & \mathcal{I}(A) \subseteq \mathcal{I}(C) \\
\models_{\mathcal{E}} A \dot{=} C & \Leftrightarrow & \mathcal{I}(A) = \mathcal{I}(C)
\end{array}
$$

Raisonnement terminologico-assertionnel

La relation de généralité est définie entre couples de termes t et t' vis à vis d'une terminologie $\mathcal{T}$. Ainsi, on dira que t' subsume t (notée $t \preceq_{\mathcal{T}} t'$) si et seulement si dans tout modèle $\mathcal{E}$ de $\mathcal{T}$, $\mathcal{E}(t) \subseteq \mathcal{E}(t')$. Le test de subsomption est à la base d'un ensemble d'autres opérations nécessaires pour le raisonnement terminologique. En particulier, la subsomption permet de vérifier l'incohérence d'un concept ($t \preceq_{\mathcal{T}}$ `Nothing`), l'équivalence ($t \approx_{\mathcal{T}} t'$) ou l'exclusion ($t \oplus_{\mathcal{T}} t'$ ou (`and` t t') $\preceq_{\mathcal{T}}$ `Nothing`) entre deux concepts.

L'intérêt des logiques de descriptions réside donc dans la possibilité de définir des termes structurés (concepts et rôles) et de calculer, de manière efficace, la subsomption entre ces termes. Mais cette obligation de définir tous les termes impliqués dans le test de subsomption est parfois contraignante dès lors qu'il s'agit de modéliser un domaine réaliste. En effet, il est bien difficile de trouver systématiquement des conditions nécessaires et suffisantes d'appartenance à un concept ou un rôle pour une entité. De plus, les principaux travaux portent sur la partie terminologique et peu sur la partie assertionnelle compte tenue de la complexité qu'elle y introduit.

Synthèse

Les deux formalismes de représentation des connaissances que nous venons de présenter ont donc de nombreux points communs : tous deux effectuent une séparation claire entre les connaissances ontologiques (support ou T-box) et les connaissances assertionnelles (graphe ou A-box). Ils sont équipés d'une sémantique formelle (en logique du premier ordre et en théorie des modèles) qui permet d'effectuer des raisonnements sur les bases de connaissances produites.

Mais ces deux formalismes présentent également quelques différences : les logiques de descriptions se sont spécialisées dans la description des connaissances ontologiques alors que les graphes conceptuels ne fournissent qu'une expressivité réduite pour décrire le support mais privilégient plutôt la description des connaissances assertionnelles. On peut noter également que les logiques de descriptions s'adaptent bien au Web, c'est-à-dire à un monde ouvert, puisque tous les faits non présents ne sont pas forcément faux. Nous allons voir dans la section suivante comment l'évolution de ces formalismes a conduit à prendre le meilleur de chacun d'eux pour pouvoir décrire de la connaissance distribuée sur le Web.

3.3 La représentation des connaissances sur le Web

L'avenir du Web, tel que le conçoit le W3C, passe par une structuration des données qu'il contient. XML fournit en cela un excellent support puisqu'il permet d'obtenir un arbre représentant la structure des documents (voir section 2.1). Cependant, ce standard ne permet pas, à lui tout seul, de représenter la sémantique des documents de manière à être compréhensible et manipulable par des programmes informatiques.

Un peu avant l'arrivée de XML, différents systèmes ont essayé de décrire la sémantique des informations contenues dans les documents afin d'améliorer les réponses aux requêtes formulées. Ils s'inspirent des travaux menés autour de la représentation des connaissances tels que nous les avons présentés dans la section précédente. Nous allons commencé par décrire deux de ces langages (SHOE et Ontobroker) souvent jugés comme des précurseurs (section 3.3.1).

Ces deux langages marquent une première étape dans la volonté de rendre les informations du Web accessibles aux machines et contribuent donc à la vision d'un *Web Sémantique* telle que l'a proposée T. Berners-Lee [Berners-Lee *et al.*, 2001] (voir figure 3.6). L'étape suivante est la définition de langages qui permettent d'annoter des ressources du Web (RDF) et de définir le vocabulaire d'annotation (RDFS). Nous les décrivons dans la section 3.3.2. Nous présentons finalement dans la section 3.3.3 le langage OWL, beaucoup plus expressif que RDFS (sa complexité est bien plus grande que celle de RDFS), qui a pour objectif de représenter des ontologies sur le Web.

3.3.1 Shoe et Ontobroker : des précurseurs

SHOE (*Simple HTML Ontology Extensions*)

SHOE est un projet développé à l'université de Maryland par l'équipe de J. Hendler qui a vu le jour en 1996 [Luke *et al.*, 1997]. Il se veut un langage de représentation des connaissances compatible avec HTML et XML, qui permet de construire des ontologies et de les utiliser.

L'ontologie va définir les règles indiquant quels types d'assertions et d'inférences on peut faire. Elle se compose d'une taxinomie de catégories (ou classes) dont le type le plus général est `SHOEEntity`. Elle contient également des relations n-aires établies entre instances de catégories, ou entre instances et types primitifs (*string*, *number*, *date*, *boolean*). Enfin, des règles d'inférences de type clauses de Horn viennent compléter la définition d'une ontologie. Une instance est d'abord un objet au sens des objets dans une base de données. C'est aussi un ensemble de faits, basés sur les règles définissant l'ontologie, qui viennent alimenter une base des connaissances. Ce sont ces faits qui apporteront les réponses à des requêtes structurées.

Autour du langage SHOE, l'équipe a développé un agent nommé *Exposé* qui est chargé de parcourir les pages Web à la recherche d'informations sémantiques. Dès qu'il remarque une page annotée à l'aide du langage SHOE, il charge la ou les ontologies utiles et récolte les données pour compléter la base des connaissances. Un autre outil permet d'annoter les pages Web, masquant ainsi le formalisme du langage SHOE. Enfin, un dernier outil permet à des utilisateurs de formuler des requêtes via une interface graphique.

Pour démontrer les capacités de représentation et d'inférences de SHOE, plusieurs applications ont été construites. En particulier, une ontologie modélisant le domaine des départements d'informatique dans les universités a été créée. Celle-ci est aujourd'hui régulièrement reprise mais sa trop grande simplicité empêche de tirer des conclusions définitives sur la méthode. Des applications portant sur des domaines plus complexes ont ensuite mis en lumière d'autres problèmes : la difficulté de construire des ontologies efficaces lorsque les domaines sont complexes ou mal définis et le compromis à trouver entre le niveau de granularité de l'annotation et le coût de celle-ci. Nous verrons dans la section 3.4 quel type de réponses ont été apportées à ces questions.

Ontobroker

Le projet **Ontobroker** a été développé à l'université de Karlsruhe en 1997 sous l'impulsion de R. Studer [Fensel *et al.*, 1998a; Fensel *et al.*, 1998b]. Il fournit différents langages pour décrire des ontologies, formuler des requêtes, ou encore annoter des documents. L'architecture générale d'Ontobroker est décrite dans la Figure 3.4. Elle consiste en trois éléments principaux (une interface permettant d'effectuer des requêtes, un moteur d'inférence, et un *webcrawler*), chacun de ces éléments étant accompagné d'un langage de formalisation.

Une ontologie va être décrite par une taxinomie de classes et un ensemble de règles (utilisées par le moteur d'inférence). Ontobroker utilise *F-Logic* pour modéliser des ontologies. Ce langage est basé sur le calcul des prédicats restreint aux clauses de Horn (type *Prolog*) [Fensel *et al.*, 1998a]. Il permet, entre autre, de représenter et de hiérarchiser des concepts (ou classes), des instances et des attributs avec leur valeur. Un sous-ensemble du formalisme permet finalement de formuler des requêtes. L'annotation des documents s'effectue directement dans les pages Web grâce à une simple extension du langage HTML. Pour cela, l'éditeur de pages dispose de trois primitives contenues à l'intérieur de la balise ancre `<a ...> ... </a>` et de l'attribut `onto` (la balise contient déjà les attributs `name` et `href`).

Parmi les outils dont disposent Ontobroker, une interface graphique permet de composer des requêtes. Elle affiche l'ontologie sous la forme d'un graphe hyperbolique contenant les classes

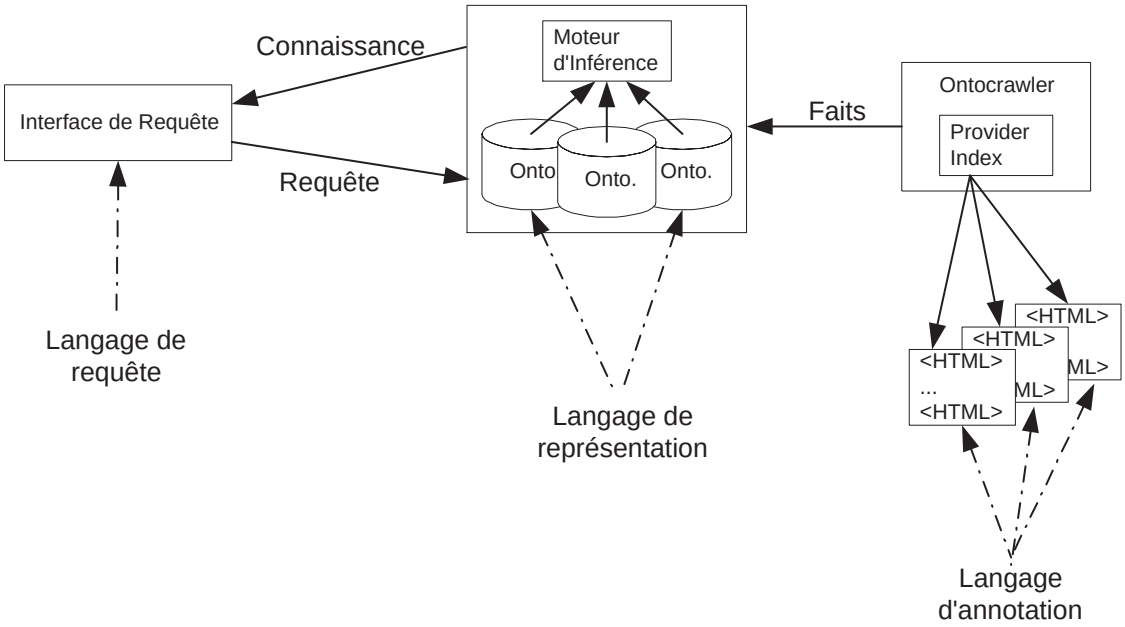


FIG. 3.4 – *Architecture générale d'Ontobroker*

(et leurs attributs) et les liens entre elles. Le moteur d'inférence est alors chargé de dériver les réponses aux requêtes formulées. Il utilise pour cela les faits (contenus dans les pages Web) et la définition de l'ontologie. Une première phase consiste à traduire en prédicats, l'ontologie déclarée en *F-Logic*. Le moteur d'inférence combine alors les faits et les règles obtenus et résout le problème donné en logique des prédicats par substitution et unification. Une dernière transformation est effectuée pour retourner la réponse à la requête en *F-Logic*.

Ontobroker avait pour but de permettre un meilleur accès à l'information dans une gigantesque base de connaissances que serait le Web. Deux ontologies assez importantes ont été produites par le projet. L'une, provenant de la communauté de l'acquisition de connaissances $(KA)^2$ (*Knowledge Annotation Initiative of the Knowledge Acquisition Community*), décrit des organisations (entreprise, université, institut de recherche), des personnes (employé, étudiant, administration), des projets et des événements. L'autre provient du *CIA World Fact Book* et concerne l'organisation géographique, démographique, économique et politique de tous les pays du monde. En trois ans de fonctionnement, l'équipe d'Ontobroker a constaté un certain nombre de limites ou problématiques de recherche (la construction des ontologies, l'expressivité du langage pour les représenter, les performances du moteur d'inférence basé sur *F-Logic*...). Ces problèmes ont commencé à être résolus dans les différentes évolutions d'Ontobroker dont le dernier avatar est la suite **KAON** (voir section 3.5).

3.3.2 Les langages RDF et RDFS

Dans la vision du Web Sémantique proposée par T. Berners-Lee, le langage XML fournit la couche syntaxique permettant de structurer une partie du Web (figure 3.6). L'étape suivante est de permettre l'annotation structurée de ses ressources, ce qui nécessite un modèle de données et un langage de définition du vocabulaire d'annotation. Nous décrivons dans la suite les langages RDF et RDFS répondant à ces deux objectifs.

RDF (*Ressource Description Framework*)

RDF (*Ressource Description Framework*) [RDF, 2004] est un document de travail du W3C qui définit un modèle pour décrire la sémantique des données, de manière compréhensible et utilisable par les machines. C'est une application de XML dans le sens où une de ses syntaxes est définie en XML. Le modèle de données RDF est constitué de trois types d'objet :

- les **ressources** : elles sont identifiées par une URI [Berners-Lee *et al.*, 1998] et ne sont pas obligatoirement accessibles via le Web ;
- les **propriétés** : il s'agit de caractéristiques, d'attributs ou de relations utilisées pour décrire des ressources particulières ;
- les **assertions** : il s'agit de triplets liant une ressource, une propriété et la valeur de cette propriété pour cette ressource ; les trois éléments composant une assertion sont nommées respectivement le *sujet*, le *prédicat* et l'*objet*.

RDF est donc un langage permettant de décrire les informations d'une ressource en lui attachant des propriétés. Il définit le triplet

$$ressource \rightarrow propriété \rightarrow valeur$$

comme primitive de modélisation et introduit une syntaxe standard pour la représenter. La valeur d'une propriété peut être d'un type primitif (par exemple, chaîne de caractères) ou une autre ressource. Enfin, les assertions sont elles-mêmes considérées comme des ressources. Cette réification permet d'attacher des propriétés à des triplets.

Le code 3.1 indique ainsi qu'il existe une ressource identifiée par l'URI `http://www.lance-armstrong.com/` qui a comme nom *Lance Armstrong* qui a terminé *1er* au classement général du *Tour de France 2003* identifié par l'URI `http://www.letour.fr/2003/`.

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:ina="http://www.ina.fr/cyclisme-schema/">
  <rdf:Description rdf:about="http://www.lancearmstrong.com/"
                   ina:aCommeNom="Lance Armstrong">
    <ina:estClasséGénéral>
      <rdf:Description ina:position="1er">
        <ina:épreuveConsidérée rdf:resource="http://www.letour.fr/2003/" />
      </rdf:Description>
    </ina:estClasséGénéral>
  </rdf:Description>
</rdf:RDF>
```

CODE 3.1 – *Exemple de description dans la syntaxe RDF/XML*

Cet exemple est écrit dans la syntaxe RDF/XML, mais il se traduit également par le graphe de la figure 3.5 dans lequel les nœuds ovales représentent des ressources identifiées par des URIs, les arcs représentent des propriétés elles-mêmes identifiées par des URIs, et les nœuds rectangulaires représentent les valeurs, de type primitif, des propriétés.

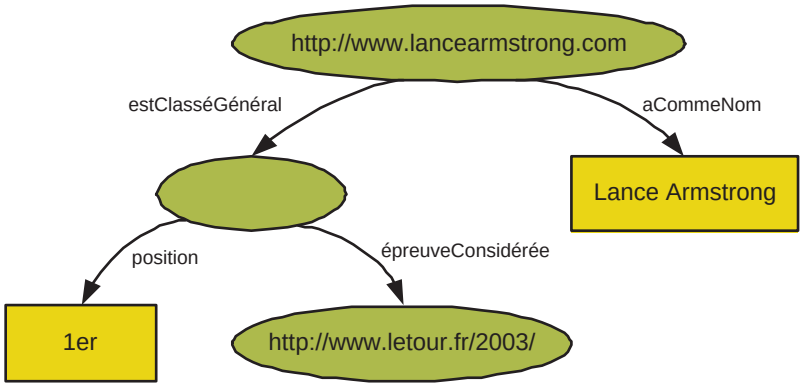


FIG. 3.5 – *Exemple de graphe RDF représentant l'assertion du tableau 3.1*

RDF Schema (*RDF Vocabulary Description Language*)

Pour que les informations présentes dans les graphes RDF soient utilisables par une machine, il faut auparavant les définir. Ceci est le rôle du langage RDF Schema (RDFS) [RDFS, 2004], qui est également un document de travail du W3C. Ce langage permet de déclarer les propriétés des ressources (par exemple `aCommeNom`, `estClasséGénéral`) et le type des ressources (par exemple `CoureurCycliste`, `CourseAEtapes`) décrites.

Contrairement au modèle orienté-objet où la modélisation des classes englobe la modélisation des propriétés (nommées attributs), le langage RDFS permet de définir les classes et les propriétés séparément. Les classes (*resp.* propriétés) sont organisées en taxinomie *via* la relation `subClassOf` (*resp.* `subPropertyOf`). Le langage permet également de contraindre la signature des propriétés en indiquant ses domaine et co-domaine. Du point de vue syntaxique, le vocabulaire défini en RDFS est représenté sous forme de triplets RDF. RDFS peut ainsi être vu comme un langage minimal pour représenter des ontologies qui peuvent ensuite être traduites en un ensemble de triplets RDF.

Sémantique

Comme l'ont remarqué un certain nombre d'auteurs ([Baget, 2003; Corby *et al.*, 2000]), RDF/RDFS entretiennent beaucoup de similarités avec les langages de graphes conceptuels. Au niveau de la représentation tout d'abord, puisque la connaissance est représentée sous forme de graphes. Mais également au niveau de la sémantique, puisque depuis peu, RDF est doté d'une sémantique en théorie des modèles [RDF-MT, 2004]. Ainsi, la sémantique de RDF/RDFS se rapproche de celle des graphes conceptuels simples que avons étudiés dans la section 3.2.2.

Cette sémantique spécifie des règles d'inférence permettant de compléter des graphes RDF, notamment la règle d'implication (*entailment*) qui établit une relation sémantique entre deux expressions de telle sorte que si la première est vraie logiquement alors la seconde est nécessairement vraie logiquement. Le tableau 3.2 donne quelques exemples de ces règles où *xxx*, *yyy*, *zzz* sont des URIs ou des littéraux et *aaa* est un URI. Ces règles indiquent que tout graphe contenant une expression telle que représentée sous forme de triplets dans le prémisse peut être complétée automatiquement par une autre expression représentée sous forme de triplets dans la conclusion.

	Prémisse	**Conclusion**
[rdf 1]	xxx aaa yyy	aaa `rdf:type` `rdf:Property`
[rdfs 2a]	xxx `rdfs:domain` yyy yyy `rdfs:subClassOf` zzz	xxx `rdfs:domain` zzz
[rdfs 7b] (réflexivité)	xxx `rdf:type` `rdfs:Class`	xxx `rdfs:subClassOf` xxx
[rdfs 8] (transitivité)	xxx `rdfs:subClassOf` yyy yyy `rdfs:subClassOf` zzz	xxx `rdfs:subClassOf` zzz

TAB. 3.2 – *Exemples de règles d'inférence induites par la sémantique de RDF/RDFS (d'après [RDF-MT, 2004], section 4)*

3.3.3 Vers un langage pour exprimer des ontologies sur le Web

RDFS fournit les primitives de modélisation permettant de construire des ontologies simples. Beaucoup lui ont donc reproché son manque d'expressivité par comparaison avec ce que les langages de représentation des connaissances pouvaient faire. Des efforts parallèles, puis conjoints, entre l'Europe et les États-Unis ont alors produit un langage de représentation d'ontologies sur le Web qui sera bientôt recommandé par le W3C. Nous présentons rapidement les différentes propositions qui ont conduit à la définition de ce langage.

OIL, DAML+OIL

L'initiative européenne pour produire un langage expressif de représentation d'ontologies a été portée par le projet *On-to-Knowledge* [3] qui a pour but de développer des méthodes et des outils permettant l'accès et la gestion de connaissances issues du Web. Le projet a entre autres proposé le langage **OIL** (*Ontology Inference Layer*), qui combine les primitives de modélisation utilisées dans les langages de *frames* et le raisonnement formel des logiques de description pour exprimer des ontologies sur le Web [Fensel *et al.*, 2000].

Par rapport au langage RDFS, OIL reprend la distinction faite en logique de descriptions entre concept *primitif* (décrit par des conditions nécessaires d'appartenance) et concept *défini* (décrit par des conditions nécessaires et suffisantes d'appartenance). Les propriétés sont des rôles, qui s'organisent en taxinomie, mais qui sont dissociés des concepts et pour lesquels on peut donc spécifier un domaine et un co-domaine. Les concepts sont alors définis au moyen d'une expression booléenne portant sur d'autres concepts, ou au moyen d'une contrainte sur un rôle. Une contrainte sur un rôle peut porter sur les valeurs (ou le type des valeurs) permises, mais également sur la cardinalité. Enfin, les rôles sont équipés de propriétés algébriques telles que *inverse*, *symétrique* ou *transitif*.

Ces primitives de modélisation héritent, en fait, essentiellement du langage XOL [Karp *et al.*, 1999] en l'étendant sur de nombreux points [Fensel *et al.*, 2000]. Sur d'autres aspects, le langage OIL est moins expressif puisque, par exemple, il ne supporte pas les types concrets (nombre, contraintes numériques...). Cette restriction se justifie par la volonté de pouvoir conserver un test de subsomption décidable [Fensel *et al.*, 2001]. OIL dispose d'une syntaxe XML qui le lie étroitement avec RDF puisque toutes les primitives de modélisation sont définies en RDF Schema.

Parallèlement à cette initiative, la DARPA aux États-Unis a lancé le programme de recherche

3. `http://www.ontoknowledge.org/`.

DAML[4] (*DARPA Agent Markup Language*) pour développer des langages et des outils permettant de rendre le contenu de documents accessible et exploitable par des machines. Ce programme de recherche a rapidement proposé deux langages : *DAML-ONT* pour exprimer des ontologies et *DAML-LOGIC* qui permet d'ajouter des règles d'inférence [Hendler and McGuinness, 2000].

Sous la houlette de la DARPA, un comité joint EU/US a alors décidé de développer le langage **DAML+OIL**[5], fusion des langages *DAML-ONT* et *OIL*. Ce nouveau langage supporte désormais les types de données primitifs (tels qu'on les trouve dans la norme XML Schema), et la définition d'un certain nombre d'axiomes comme l'équivalence de classes ou de propriétés. Il est équivalent à une logique de descriptions très expressive, la logique $SHOIQ$, moyennant l'ajout des types de données primitifs et du constructeur indiquant l'existence d'une classe définie (`oneOf`). Pour cette logique, il existe des procédures de décision correctes et complètes, et un certain nombre d'outils l'implémentant sont disponibles (voir section 3.5.2).

OWL

Le 18 décembre 2001, un groupe de travail sur les ontologies intégré à l'activité Web Sémantique du W3C a été créé pour prendre le langage DAML+OIL comme point de départ à la définition d'un nouveau langage permettant de représenter des ontologies sur le Web. Ce langage, baptisé **OWL** (*Ontology Web Language*), est destiné à être utilisé lorsque les informations contenues dans les documents doivent être traitées par des applications logicielles, par opposition aux situations où le contenu est simplement présenté aux humains, cas traditionnel de l'utilisation du Web aujourd'hui [OWL, 2004].

Une ontologie OWL est composée d'un en-tête (métadonnées), d'axiomes et de faits. Les axiomes concernent la définition complète ou partielle de concepts et de relations (ou propriétés), la spécification de propriétés sur les relations (propriétés algébriques), et la définition d'axiomes sur les classes et les relations (équivalence, expression booléenne). Parmi les relations, on distingue celles dont le domaine de valeur sera de type primitif (*attribut*) de celles dont le domaine de valeur sera un autre concept (*relation*). Les faits concernent des individus pour lesquels on donne des valeurs aux propriétés des classes dont ils sont instances.

OWL fournit en fait trois sous-langages, d'expressivité croissante, nommés *OWL Lite*, *OWL DL* et *OWL Full*. Le langage OWL Lite peut être vu comme une extension du langage RDFS, mais auquel on aurait enlevé certaines fonctionnalités (par exemple, la réification[6]). Le tableau 3.3 donne la liste des constructeurs disponibles dans ce langage dont la principale motivation est de permettre la modélisation d'ontologies simples et d'une complexité formelle peu élevée, de sorte qu'il soit facile d'implémenter des raisonneurs corrects et complets aisément. [Horrocks and Patel-Schneider, 2003] montrent ainsi que ce langage est dans la même classe de complexité que la logique de descriptions $SHIF(D)$ – pour laquelle les procédures de décision sont en *ExpTime* – et que le système *Racer* [Haarslev and Möller, 2001], par exemple, l'implémente complètement (voir section 3.5.2).

Le langage OWL DL contient des constructeurs supplémentaires (tableau 3.4) mais il ne peut être utilisé que sous certaines restrictions : par exemple, une classe ne peut pas être une instance d'une autre classe. Il en résulte un langage plus expressif mais toujours *décidable* (les conséquences sont toujours calculables et elles le sont dans un temps fini), qui a finalement une correspondance avec la logique de descriptions $SHOIQ(D)$. Cette logique est dans une classe de

4. `http://www.daml.org`.
5. `http://www.daml.org/2001/03/daml+oil-index.html`.
6. L'opération de réification consiste à représenter quelque chose (par exemple, une assertion) sous la forme d'un objet informatique, celui-ci pouvant alors être lui-même utilisé dans d'autres représentations.

RDF Schema	(In)égalité	Propriétés algébriques
• *Class* • rdf:Property • rdfs:subClassOf • rdfs:subPropertyOf • rdfs:domain • rdfs:Range • *Individual*	• *equivalentClass* • *equivalentProperty* • *sameAs* • *differentFrom* • *allDifferent*	• *inverseOf* • *TransitiveProperty* • *SymmetricProperty* • *FunctionalProperty* • *InverseFunctionalProperty*
Intersection de classes • *intersectionOf* **Types de données** • *DatatypeProperty*	**Version** • *versionInfo* • *priorVersion* • *backwardCompatibleWith* • *inCompatibleWidth* • *DeprecatedClass* • *DeprecatedProperty*	**Annotation** • rdfs:label • rdfs:comment • rdfs:seeAlso • rdfs:isDefinedBy
Restriction sur les rôles • *allValuesFrom* • *someValuesFrom*	**Cardinalité (restreinte)** • *minCardinality* (0-1) • *maxCardinality* (0-1) • *cardinality* (0-1)	**En-tête** • *ontology* • *imports*

TAB. 3.3 – *Liste des constructeurs disponibles dans le langage* OWL Lite

complexité dont les procédures de décision sont en *NExpTime* [Horrocks and Patel-Schneider, 2003] mais les algorithmes « raisonnables » permettant de résoudre ce type de problème sont encore du domaine de recherche.

Axiome de classes	Combinaison booléenne de classes
• *oneOf*, *dataRange* • *disjointWidth* • *equivalentClass* (pour des expressions de classe) • rdfs:subClassOf (pour des expressions de classe)	• *unionOf* • *intersectionOf* • *complementOf*
Cardinalité (arbitraire)	**Restriction sur les rôles**
• *minCardinality* • *maxCardinality* • *cardinality*	• *hasValue*

TAB. 3.4 – *Liste des constructeurs supplémentaires disponibles dans les langages* OWL DL *et* OWL Full

Le langage OWL Full dispose des mêmes constructeurs que OWL DL mais il les interprète de manière plus large. Ainsi, une classe peut cette fois être vue comme un ensemble d'individus (définition extensionnelle), ou comme un individu à lui tout seul (définition intensionnelle) qui pourra, par exemple, valuer une propriété. À ce titre, OWL Full devient clairement un sur-ensemble de RDF (contrairement aux deux autres langages OWL, puisque un document RDF n'est pas forcément un document légal en OWL Lite et OWL DL). Cette expressivité accrue est gagnée au détriment de la complexité puisque le langage OWL Full n'est plus décidable.

La sémantique de OWL est basée sur une hypothèse de monde ouvert qui est particulièrement

adaptée pour le Web. En effet, le système est capable d'effectuer des raisonnements même s'il n'a pas une connaissance complète du monde, dans la mesure où tout fait non présent dans la base n'est pas considéré comme faux *a priori*. A l'instar de DAML+OIL, le langage dispose d'une sémantique en théorie des modèles (compatible avec celle de RDFS pour OWL Full) et d'une axiomatique (une traduction en logique du premier ordre est possible).

La vision d'un Web Sémantique, dans lequel l'information sera accessible et manipulable par la machine automatiquement, peut se résumer finalement par une pile de langages (figure 3.6) jouant chacun un rôle particulier :

- **XML** fournit une manière de représenter des documents structurés, mais il n'impose aucune contrainte sémantique sur les documents produits ;
- **XML Schema** permet de contraindre la structure des documents XML ;
- **RDF** est un modèle de données simple, basé sur des ressources et des relations entre ces ressources, équipé d'une sémantique et qui peut se représenter en XML ;
- **RDF Schema** permet de définir le vocabulaire pour décrire des classes et des propriétés, hiérarchisées en taxinomies.
- **OWL** fournit d'avantage de primitives de modélisation pour décrire des ontologies plus riches sur le Web.

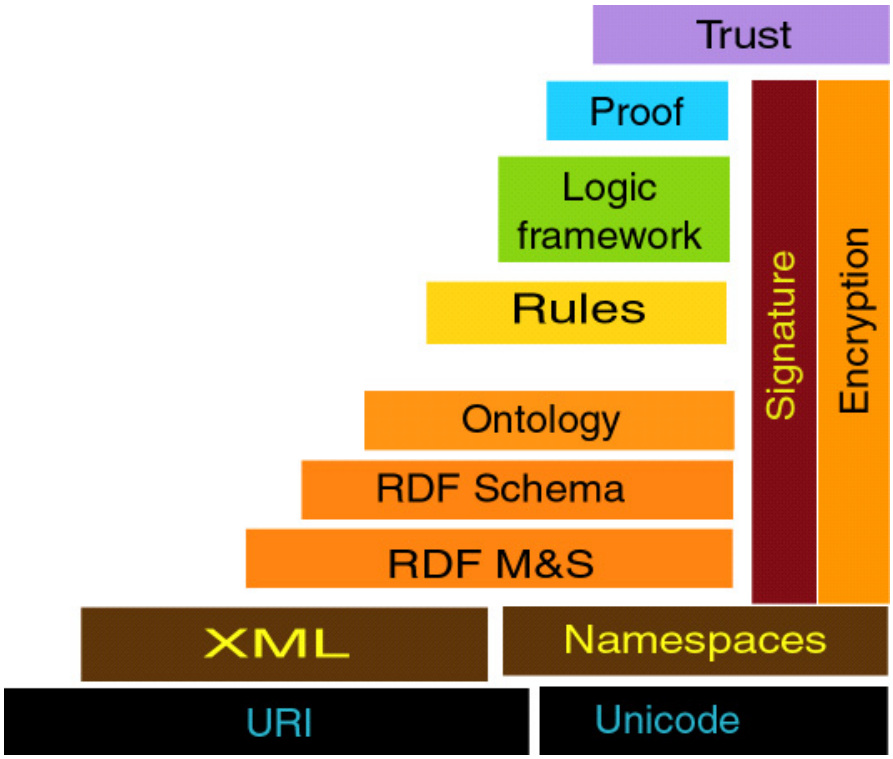


FIG. 3.6 – The Layer Cake - *d'après Tim Berners Lee [28/01/2003]*

Nous venons de voir dans cette section les différents langages qui ont conduit à la définition d'un standard permettant de représenter des ontologies sur le Web. Mais les ontologies sont des objets complexes comme nous l'avons vu dans la section 3.1. Nous étudions dans la section suivante les différentes méthodologies permettant de les construire.

3.4 La construction des ontologies

Nous avons défini dans le chapitre 3.1 ce que l'on entendait par *ontologie* : un objet informatique, résultat d'une démarche d'acquisition et de modélisation des connaissances pour un problème et un domaine donnés. Mais comme le souligne [Gómez-Pérez, 2000], l'élaboration des ontologies relève plus du savoir-faire que de l'ingénierie. Pourtant, l'absence de directives et de méthodes consensuelles entrave le développement d'ontologies communes et partagées. Dans cette section, nous proposons donc une revue des différentes méthodologies de construction existantes (section 3.4.1). Nous montrons alors pourquoi nous nous sommes orientés vers la méthodologie proposée par B. Bachimont que nous détaillons dans la section 3.4.2. Nous décrivons ensuite quelques outils concrets permettant de construire des ontologies (section 3.4.3). Nous terminons cette section en présentant l'outil que nous avons développé : **DOE** pour *Differential Ontology Editor*. DOE est une implémentation de la méthodologie proposée par B. Bachimont et vient en complément des environnements de construction d'ontologies existants (section 3.4.4).

3.4.1 Les méthodologies de construction existantes

La conceptualisation d'une ontologie commence par l'organisation des concepts et des relations d'un domaine en taxinomies. Mais malgré ce consensus, les ontologies produites sont très diverses comme en témoigne une des premières études comparatives consacrées aux méthodologies de construction [Noy and Hafner, 1997]. Le problème est qu'il n'est pas aisé pour l'ontologiste de déterminer ce qui doit apparaître dans les taxinomies, ni d'ailleurs de s'assurer que le résultat produit est bien conforme à la conceptualisation souhaitée du domaine. Nous présentons ci-après quelques éléments méthodologiques pouvant apporter des réponses à ces questions. Nous devons préciser que parmi les – nombreux – travaux en la matière, nous nous sommes surtout intéressés à ceux qui cherchaient à prendre en charge la construction *ex nihilo* des ontologies. Pour un répertoire relativement complet et récent des méthodologies existantes, nous renvoyons le lecteur au travail du OntoWeb Technical RoadMap [Gómez-Pérez *et al.*, 2002].

Uschold et Grüninger La méthodologie présentée dans [Uschold and Gruninger, 1996] se fonde sur l'expérience de la construction de plusieurs ontologies pour la modélisation d'entreprises. Elle propose les étapes suivantes :

1. identifier clairement le domaine concerné, le but et la portée de l'ontologie ;
2. construire l'ontologie et coder les connaissances en les intégrant à d'autres ontologies ;
3. évaluer le résultat ;
4. documenter l'ontologie en établissant des directives pour chacune des étapes.

La deuxième étape est donc la phase de construction proprement dite. Elle aboutit à la spécification d'une ontologie en langage formel. Le modèle n'est pas construit directement et [Uschold and Gruninger, 1996] suggèrent de passer par une étape intermédiaire qui consiste à identifier

un ensemble de *questions de compétences*. Celles-ci constituent l'élément clé qui permet de catégoriser les connaissances que doit inclure une ontologie. Un mécanisme de composition de ces questions de compétences et de leurs réponses permet ensuite de répondre à des questions plus complexes et facilite l'intégration avec d'autres ontologies.

Methontology La méthodologie Methontology, proposée par l'équipe du LAI de l'Université Polytechnique de Madrid, insère l'activité de construction d'ontologies dans un processus complet de gestion de projet (planification, assurance qualité), de développement (spécification, conceptualisation, formalisation, implémentation, maintenance) et de support (intégration, évaluation, documentation) [Fernández-López *et al.*, 1997; Blázquez *et al.*, 1998]. Elle permet de spécifier l'ontologie au niveau des connaissances dans la mesure où, contrairement à la méthodologie précédente, il est suggéré de passer par des *représentations intermédiaires* lors la conceptualisation de l'ontologie. La formalisation et l'implémentation ne sont alors, dans l'idéal, que des phases de traduction quasi-automatique du modèle précédent.

Ce cadre méthodologique est soutenu par l'outil **WebODE**, que nous présenterons dans la section 3.4.3, et a permis de mettre au point plusieurs ontologies. Il rejoint également pour partie la méthodologie décrite par F. Gandon et développée au sein de l'équipe ACACIA de l'INRIA [Gandon, 2002].

Terminae L'acquisition des connaissances d'un domaine est un préalable à sa conceptualisation. Cette tâche étant difficile en général, [Aussenac-Gilles *et al.*, 2003] proposent de l'assister par des outils de traitement du langage naturel opérant sur des textes. Pour ce faire, une « démarche de corpus » et des outils terminologiques sont utilisés afin de commencer à modéliser le domaine. Ces outils, reposent pour la plupart sur la recherche de formes syntaxiques particulières manifestant les notions recherchées comme des syntagmes nominaux pour des candidats termes, des relations syntaxiques marqueurs de relations sémantiques, ou des proximités d'usage - par exemple, contextes partagés - pour des regroupements de notions [Charlet, 2003]. Le système **Terminae** développé par [Aussenac-Gilles *et al.*, 2003] implémente cette méthodologie en intégrant des outils de repérage de candidats termes (**SYNTEX**), de regroupement de contextes (**UPERY**) et de repérage de relations (**YAKWA**).

Guarino et Welty Les derniers travaux que nous présentons ici sont menés par [Guarino and Welty, 2000a; Guarino and Welty, 2000b] et concernent le « nettoyage » de taxinomies, souvent construites de manière anarchique en utilisant abusivement de la relation de subsomption. Il s'agit plus d'une étape de correction à intégrer dans le processus de construction d'ontologies qu'un réel guide méthodologique complet. L'idée est de donner une valeur à des méta-propriétés – ou *propriétés formelles* – pour chacune des propriétés présentes dans l'ontologie. Les auteurs définissent ainsi les notions de *rigidité*, *identité*, *unité* et *dépendance* et donnent les combinaisons valables qui permettront à l'ontologiste de vérifier les règles de subsomption dans la taxinomie. Le module **ODEClean** de l'environnement WebODE propose une implémentation de cette méthodologie, mais celle-ci reste tout de même réservée à des spécialistes compte tenu de sa complexité.

Au final, nous constatons que les méthodologies de construction proposées s'intéressent à l'ensemble du processus de conception d'ontologie en se concentrant sur des questions comme celles du cycle de vie des ontologies construites ou de l'ordonnancement des étapes auquel l'ontologiste

devra se soumettre pour valider son travail. Répondre à ces questions est bien évidemment important si l'on veut voir un jour émerger un véritable « génie ontologique », mais nous pensons que la phase de conceptualisation proprement dite, celle où les concepts de l'ontologie sont dégagés, définis par un certains nombres de propriétés et organisés entre eux, gagnerait à être guidée de manière plus précise qu'elle ne l'est dans ces réflexions. Par exemple, la méthode METHONTOLOGY, alors qu'elle propose un grand nombre de *représentations intermédiaires* [Blázquez *et al.*, 1998] afin de mieux conduire la construction des ontologies au niveau des connaissances, n'insiste pas sur la manière de structurer ses *arbres de classification de concepts*, l'une de ces représentations.

3.4.2 Un problème non résolu : comment obtenir les taxinomies ?

Parmi les méthodologies présentées, peu proposent des aides concrètes pour guider les utilisateurs à organiser les concepts entre eux : tout repose sur leur bonne intuition du domaine. Ce constat est somme toute normal puisqu'aucune des méthodologies exposées – à l'exception du système Terminae – ne prend réellement en charge l'explicitation des concepts sous sa forme la plus naturelle : le langage. En effet, si à peu près toutes préconisent l'utilisation de celui-ci pour tenter de préciser le sens des concepts manipulés, que ce soit dans des commentaires présents dans l'ontologie elle-même ou dans des documents produits lors du processus de modélisation, elles sont peu à énoncer des consignes précises pour la rédaction de ces compléments [Isaac, 2001]. En fait, malgré les efforts de commentaires, les termes utilisés comme primitives de connaissances peuvent toujours être sujets à des interprétations multiples, ce qui nuit fortement à la compréhension de l'ontologie et à son utilisation.

Le problème réside dans le fait qu'aucune méthodologie ne force l'ontologiste à expliciter clairement le sens qu'il attribue aux concepts : les commentaires restent très informels. B. Bachimont propose, dans une méthodologie introduite dans le cadre du projet MENELAS [Menelas, 1994], de contraindre l'utilisateur à un *engagement sémantique* [Bachimont, 1996; Bachimont, 2000a] en introduisant une *normalisation sémantique* des termes manipulés dans l'ontologie. Cette méthodologie propose trois étapes (figure 3.7) dont nous allons à présent rappeler les principes. Cette méthodologie n'ayant pas fait l'objet d'une implémentation satisfaisante, nous l'avons mise en œuvre dans l'éditeur d'ontologies *DOE* (pour *Differential Ontology Editor*) que nous présenterons dans la section 3.4.4.

Étape 1 : la normalisation sémantique

L'étape de normalisation sémantique a pour but d'expliciter le sens des libellés linguistiques pour qu'ils fonctionnent comme des primitives. Ces libellés doivent avoir leurs significations normées pour pouvoir être utilisés. L'usage de la langue naturelle est l'accès privilégié aux connaissances d'un domaine. Il paraît donc logique de commencer par utiliser des procédés d'extraction terminologique afin de trouver des « candidats-termes » (ces candidats seront par exemple les résultats d'un outil comme **LEXTER** [Bourigault, 1994] ou son évolution **SYNTEX**).

Après avoir isolé un certain nombre de libellés, il faut veiller à leurs donner une signification précise et donc utiliser une théorie sémantique appropriée. B. Bachimont propose dans [Bachimont, 2000a] d'utiliser la sémantique différentielle présentée par F. Rastier [Rastier *et al.*, 1994]. Dans ce paradigme, le sens est intra-linguistique : il se construit par des relations d'opposition entre les unités du système linguistique. En fait, il s'agit d'attribuer un sens aux termes grâce à des traits sémantiques ou *sèmes*. Ces sèmes que l'on va associer à une unité sont regroupables en

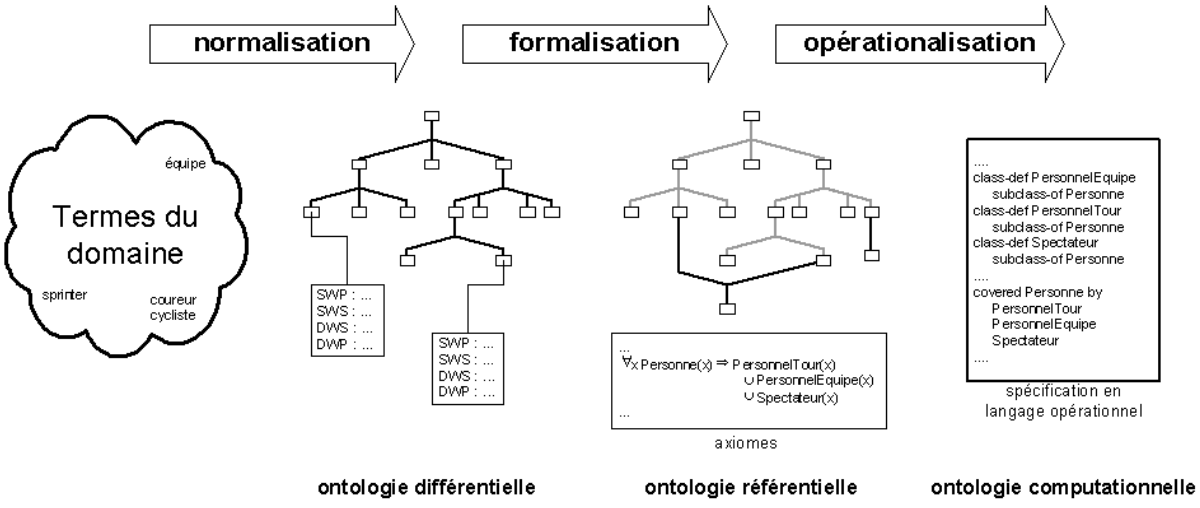


FIG. 3.7 – *Les 3 étapes de la méthode de construction d'ontologies selon B. Bachimont*

deux catégories :

- les *sèmes génériques* qui permettent de regrouper les unités entre elles ;
- les *sèmes spécifiques* qui permettent à une unité de se distinguer de celles avec qui on l'a regroupée.

Il faut cependant signaler que ce sens différentiel est fortement dépendant du contexte, qui encadre l'*interprétation*, le mécanisme d'attribution des sèmes aux termes. Or, pour obtenir une primitive réellement exploitable, il faut contrôler cette interprétation : il s'agit bien d'une *normalisation sémantique*.

Comment concrètement parvenir à cette normalisation du sens des unités manipulées ? Le réseau d'identités et de différences évoqué précédemment reste à structurer, si on veut réellement le rendre opérationnel, pour aboutir à la construction de ce que l'on va appeler une *ontologie différentielle*. Tout d'abord, on observe qu'une unité (dorénavant, on parlera de *notion*) se définit en premier lieu par une relation de subsomption avec une autre unité : celle dont elle hérite tous ses sèmes génériques. En pratique, le créateur d'ontologies doit donc pouvoir exprimer les identités et les différences de chaque notion dans son voisinage proche : la notion-parente et les notions-sœurs. Il obtiendra finalement une taxinomie[7] de notions où la signification d'une unité s'obtiendra en collectant les identités et les différences permettant de caractériser les notions rencontrées sur le chemin qui mène de la notion racine (la plus générique) à cette unité.

B. Bachimont propose quatre principes permettant d'expliciter ces informations :

- le principe de *communauté avec le père* (*similarity with parent* ou **SWP**) : on explicite pourquoi le fils hérite des propriétés de la notion qui le subsume ;
- le principe de *différence avec le père* (*difference with parent* ou **DWP**) : on explicite la différence qui permet de distinguer le fils du père ;

7. Il s'agit d'un arbre : l'héritage multiple est interdit dans la mesure où l'on associe aux notions d'une fratrie des sèmes spécifiques qui sont en opposition.

- le principe de *différence avec la fratrie* (*difference with siblings* ou **DWS**) : on indique ici la propriété qui permet de distinguer la notion considérée de ses notions-sœurs ;
- le principe de *communauté avec la fratrie* (*similarity with siblings* ou **SWS**) : on donne enfin la propriété – admettant plusieurs valeurs exclusives – qui a permis de caractériser les notions d'une même fratrie, justifiant ainsi le principe précédent.

Par le processus de normalisation sémantique exposé ci-dessus, on passe d'un candidat-terme à une notion dont la signification, ancrée dans le domaine et l'application envisagée, est invariable et peut donc fonctionner comme une primitive exprimant une connaissance.

Étape 2 : la formalisation des connaissances

L'arbre ontologique construit dans la première étape a permis de fixer un certain nombre de notions en normalisant leurs sens. Cependant, le travail n'est toujours pas terminé : on ne peut pas vraiment utiliser cet ensemble de notions dans un véritable SBC. Il faut en effet introduire des concepts obéissant à une sémantique formelle et extensionnelle [8] pour que ceux-ci servent en tant que primitives dans un langage formel de représentation des connaissances.

Les notions dont on dispose dans l'ontologie différentielle sont de nature linguistique, toujours soumises à des processus d'interprétation (même si à présent cette interprétation est adaptée à la pratique ciblée lors de la construction de l'ontologie). Le passage à une sémantique extensionnelle va permettre de lier ces notions à un ensemble de référents dans le monde, donnant lieu à ce que l'on va appeler dans la suite une *ontologie référentielle*, composée de *concepts* qui vont agir comme des primitives formelles.

Ces primitives ne sont plus définies par les principes différentiels. Elles n'ont donc pas intrinsèquement de sens interprétatif ; elles n'en acquièrent que parce qu'elles sont reliées – par leur libellé – aux notions différentielles. Cependant, ce sont des primitives et elles peuvent, grâce aux mécanismes de composition de sens d'une sémantique extensionnelle, servir à définir de nouveaux concepts formels. Chaque concept étant lié par référence à un ensemble d'objets du domaine (son *extension*), on peut avoir recours à des opérations de composition de sens qui utilisent les opérations qui existent pour les ensembles (réunion, intersection ou complémentaire).

La comparaison des extensions permet de définir une relation d'héritage extensionnelle entre les concepts : un concept sera subsumé par un autre si et seulement si son extension est incluse dans celle de son parent. On peut alors se poser la question de la structure de la hiérarchie de subsomption obtenue. S'il paraît naturel d'affirmer que les relations d'héritage définies dans l'ontologie différentielle tiendront toujours, on peut se demander dans quelle mesure l'ajout de nouveaux nœuds va modifier la structure arborescente. En effet, les « transcriptions » des notions différentielles en concepts formels peuvent admettre des extensions qui ont un sous-ensemble commun. L'héritage multiple devient donc possible, et la structure hiérarchique obtenue est celle d'un treillis [9].

On a vu que la sémantique référentielle nous permet de définir de nouveaux concepts en fonction d'autres concepts. Il faut signaler que l'on peut également exprimer de telles définitions pour les concepts formels directement issus de la transposition de l'ontologie différentielle. C'est notamment à cette étape que l'on va définir les relations par la donnée de leur arité et de leur domaine, ce qui les associera de fait à des produits cartésiens de références de concepts.

On peut aussi introduire des axiomes logiques, lois qui régissent les comportements des individus qui constituent l'extension des concepts formels. Par exemple, on peut spécifier que les

8. Encore que d'autres types de sémantiques formelles soient envisageables.
9. Il est en effet possible de clore l'ensemble des concepts par le concept « vide ».

sous-concepts d'un concept forment une partition disjointe de celui-ci. On peut enfin exprimer des lois concernant les relations, comme leur attacher des propriétés algébriques : réflexivité, transitivité... Il est clair que l'ontologie référentielle doit expliciter toutes les lois logiques induites par la conceptualisation que l'on veut spécifier.

Étape 3 : l'opérationnalisation des connaissances

Dans cette troisième et dernière étape, on va chercher à munir les concepts présents dans l'ontologie référentielle d'une signification en termes d'opérations informatiques. On spécifie ainsi le comportement des objets informatiques présents dans le système qui utilise l'ontologie, pour aboutir à ce que l'on va appeler une *ontologie computationnelle*.

Les concepts computationnels se définissent par les inférences, les calculs que pourra effectuer un système à partir de la donnée des individus qu'ils couvrent dans le monde. Le système utilise alors un langage opérationnel de représentation des connaissances [10] qui fait appel à des capacités d'inférence précises répondant aux besoins exprimés lors de la spécification du système (voir section 3.2). Pour un langage de représentation reposant sur les graphes conceptuels, il s'agira d'opérations de manipulation de graphes (jointure, projection...). Pour un langage fondé sur le paradigme des logiques de description, il s'agira plutôt des tests de subsomption permettant la classification des individus introduits dans la base de connaissances. L'expression des primitives de l'ontologie référentielle dans un de ces langages assigne donc une véritable signification computationnelle aux concepts manipulés par le système.

3.4.3 Les outils existants

Après avoir étudié les différentes méthodologies de construction d'ontologies et détaillée celle que nous avons finalement adopté, nous présentons maintenant les outils informatiques permettant de les créer afin de voir si l'un d'entre eux peut être utilisé ou adapté pour suivre les principes fournis par B. Bachimont. Ces outils peuvent se regrouper grossièrement en deux catégories. Dans la première, on trouve les plus anciens historiquement, qui permettent de spécifier les ontologies au niveau symbolique (voir, par exemple, le serveur **Ontolingua** [11] [Farquhar *et al.*, 1995]) : une grande partie des définitions des objets se fait directement dans un langage de représentation des connaissances donné (pour Ontolingua, il s'agit de KIF), auquel le créateur et l'utilisateur de l'ontologie doivent se plier. Dans la seconde catégorie, les outils prennent mieux en compte l'importance du niveau des connaissances : ils proposent à l'utilisateur de créer l'ontologie de manière relativement indépendante de tout langage implémenté et prennent ensuite automatiquement en charge l'opérationnalisation de l'ontologie, en la transposant dans divers langages. Cette évolution tend à rapprocher les ontologies de leur but original : il semble en effet naturel de chercher à s'abstraire – dans un premier temps – du niveau symbolique si on veut obtenir une ontologie permettant un réel partage d'une compréhension. Cette dernière catégorie regroupe les outils principalement utilisés aujourd'hui. Nous les détaillons ci-dessous [12].

10. Pour un comparatif récent des langages opérationnels, on peut se reporter à [Barry *et al.*, 2001]. L'action Escrire (`http://www.inrialpes.fr/exmo/cooperation/escrire/`) compare elle, les trois grandes classes de formalismes de représentation des connaissances (logiques de descriptions, graphes conceptuels et paradigme objet) dans le contexte de la représentation du contenu de documents.

11. `http://www-ksl-svc.stanford.edu:5915/`

12. Pour un recensement quasi exhaustif, nous renvoyons le lecteur au déliverable 1.3 du projet OntoWeb [Fensel and Gómez-Pérez, 2002] et à l'étude effectuée par M. Denny disponible en ligne à `http://www.xml.com/pub/a/2002/11/06/ontologies.html`.)

Protégé2000[13] [Noy *et al.*, 2000] est un environnement graphique de développement d'ontologies développé par le SMI de Stanford. Dans le modèle des connaissances de *Protégé*, les ontologies consistent en une hiérarchie de *classes* qui ont des attributs (*slots*), qui peuvent eux-mêmes avoir certaines propriétés (*facets*). L'édition des listes de ces trois types d'objets se fait par l'intermédiaire de l'interface graphique, sans avoir besoin d'exprimer ce que l'on a à spécifier dans un langage formel : il suffit juste de remplir les différents *formulaires* correspondant à ce que l'on veut spécifier. Ce modèle autorise d'ailleurs une liberté de conception assez importante, puisque le contenu des formulaires à remplir peut être modifié suivant les besoins *via* un système de *méta-classes*, qui constituent des sortes de « patrons » de connaissance. L'interface, très bien conçue, et l'architecture logicielle permettant l'insertion de *pluggins* pouvant apporter de nouvelles fonctionnalités (par exemple, la possibilité d'importer et d'exporter les ontologies construites dans divers langages opérationnels de représentation tels que OWL ou encore la spécification d'axiomes) ont participé au succès de *Protégé2000*, qui regroupe une communauté d'utilisateurs très importante et constitue une référence pour beaucoup d'autres outils.

OILed[14] [Bechhofer *et al.*, 2001], développé sous la responsabilité de l'université de Manchester, a été conçu pour éditer des ontologies dans le langage de représentation OIL, un des précurseurs du langage OWL (voir section 3.3.3). Officiellement, il n'a pas d'autre ambition que de construire des exemples montrant les vertus du langage pour lequel il a été créé. À ce titre, *OilEd* est souvent considéré comme une simple interface de la logique de description $\mathcal{SHIQ}$. Néanmoins, il offre la plus grande partie de ce que l'on peut attendre d'un éditeur d'ontologies. On peut créer des hiérarchies de classes et spécialiser les rôles, et utiliser avec l'interface les types d'axiomes les plus courants. Cet éditeur offre également les services d'un raisonneur, *FaCT* [Horrocks, 1998], qui permet de tester la satisfaisabilité des définitions de classes et de découvrir des subsomptions restées implicites dans l'ontologie.

Contrairement aux deux outils précédents, **OntoEdit** [Sure *et al.*, 2002] n'est pas disponible gratuitement dans sa version complète[15]. Il présente les fonctionnalités essentielles communes aux autres éditeurs (hiérarchie de concepts, expression d'axiomes, export de l'ontologie dans des langages divers) et a le mérite de s'appuyer sur une réflexion méthodologique significative. La modélisation des axiomes a fait l'objet de soins particuliers pour pouvoir être effectuée – en tout cas pour les types les plus répandus – indépendamment d'un formalisme privilégié, et cela pour faciliter la traduction d'un langage de représentation à un autre. Il propose également une gestion originale des *questionnaires de compétences*. Des questions pour les réponses desquelles l'ontologie doit fournir le matériel conceptuel, on peut extraire les termes appelés à intégrer l'ontologie. Un petit outil fait une comparaison lexicale des termes extraits des différentes questions pour en déduire automatiquement d'éventuelles subsomptions. Le procédé semble cependant loin d'être fiable, puisqu'il repose sur l'hypothèse que le nom d'un concept se retrouve parfois dans le nom de ses spécialisations.

WebODE[16] [Arpirez *et al.*, 2001], développé par le LAI de Madrid, est une plate-forme de conception d'ontologies fonctionnant en ligne. D'un point de vue méthodologique, l'outil fait suite

13. `http://protege.stanford.edu/index.shtml`
14. `http://oiled.man.ac.uk/`
15. Une version de démonstration est disponible sur le site d'Ontoprise, la société qui le développe en collaboration avec l'AIFB de Karlsruhe.
16. `http://delicias.dia.fi.upm.es/webODE/`

à ODE, un éditeur qui assurait fidèlement la support de la méthodologie maison METHONTOLOGY. Il illustre bien l'évolution des outils de construction d'ontologies, puisque les nombreuses tables de son prédécesseur ont été remplacées par une interface très travaillée, réalisant un pas supplémentaire vers une conception au niveau des connaissances. On peut cependant regretter que cette évolution se soit faite au détriment de l'application des contraintes méthodologiques : les *représentations intermédiaires* utilisées dans le processus de conception sont désormais moins mises en avant, à tel point que le guide de l'utilisateur ne les signale que pour « *assurer la compatibilité conceptuelle avec* ODE ». L'accent a plus été mis sur la possibilité d'un travail collaboratif ou sur la mise à disposition d'outils complémentaires, comme un moteur d'inférences ou le module ODEClean.

3.4.4 DOE - *Differential Ontology Editor*

Si tous ces outils peuvent être considérés – en tout cas dans une première approche – comme satisfaisants en matière d'expressivité ou d'interface, on peut affirmer qu'il existe toujours un certain vide méthodologique. Le problème que nous évoquions, à savoir la structuration des taxinomies produites, est en particulier très peu pris en charge : on ne peut pas dire que les outils existants guident réellement l'utilisateur. La piste d'un traitement plus complet des informations véhiculées par le langage ne semble pas non plus pouvoir être suivie grâce à ces environnements, puisque les commentaires demeurent toujours accessoires [Troncy and Isaac, 2002].

Nous avons donc décidé de développer notre propre éditeur d'ontologies, **DOE** (pour *Differential Ontology Editor* [17]) [Troncy and Isaac, 2002; Bachimont *et al.*, 2002]. Cet outil n'a pas pour ambition de concurrencer les grands environnements existants, mais plutôt de fournir une implémentation à la méthodologie de structuration différentielle que nous avons présenté dans la section 3.4.2. À l'instar des autres éditeurs, il offre une représentation graphique des arbres de concepts et de relations de l'ontologie et permet d'interagir avec les hiérarchies. L'outil assiste également la saisie des principes différentiels issus de la méthodologie en automatisant partiellement cette tâche. Le modèle de représentation de l'ontologie est finalement proche de celui du langage RDFS, à ceci près qu'il autorise la modélisation de relations n-aires. Au niveau formel, l'éditeur est capable de faire quelques inférences en vérifiant la consistance de l'ontologie (propagation de l'arité le long de la hiérarchie des relations et héritage des domaines par exemple). Il permet également d'ajouter des individus à l'ontologie. Finalement, le passage à une ontologie computationnelle s'effectue par un export de l'ontologie formelle dans un certain nombre de langages opérationnels. Cette traduction s'effectue grâce à des feuilles de style XSLT appliquées au format de sauvegarde XML de l'éditeur. De la même façon, DOE peut importer des ontologies modélisées dans d'autres outils grâce à des feuilles XSLT dédiées. Pour être tout à fait complet, il faut noter que l'éditeur permet aussi de sauvegarder un certain nombre de métadonnées concernant l'ontologie elle-même. Il suit ici les propositions du *Dublin Core* [18].

L'interopérabilité de DOE avec les autres outils est fondamentale dans la mesure où il ne met pas en œuvre les fonctionnalités d'expressivité formelle prises en charge par les autres environnements. En fait, cet outil, plutôt que de constituer un environnement complet, se veut complémentaire des éditeurs existants en mettant l'accent sur la structuration des taxinomies. L'export des ontologies a donc été soigné et, aujourd'hui, DOE peut traduire les ontologies modélisées dans les langages RDFS, OWL, OIL, DAML+OIL et CGXML. Il a été intégré dans le système OPALES où il est capable de générer des supports de graphes conceptuels pour la

17. L'outil est disponible gratuitement à `http://opales.ina.fr/public/`.
18. http://www.dublincore.org

plate-forme **CoGITANT** (voir section 2.4.2, page 63). En 18 mois, il a été téléchargé par près de 300 utilisateurs.

3.5 L'exploitation des ontologies

Après avoir détaillé notre méthodologie de construction d'ontologies et son outil, nous allons voir maintenant comment nous allons utiliser les ontologies. Celles-ci sont un élément central des systèmes à bases de connaissances puisqu'elles en fournissent la terminologie. Elles doivent pouvoir être stockées et surtout interrogées. Nous décrivons donc les outils permettant de stocker des bases de connaissances avec les langages qui permettent de les interroger (section 3.5.1). Les ontologies sont également encodées dans un langage formel doté d'une sémantique, ce qui permet de conduire des inférences sur les connaissances représentées. Nous présentons donc quelques moteurs d'inférences opérationnels implémentant la sémantique des langages du Web Sémantique, RDF(S) et OWL (section 3.5.2. Pour une énumération exhaustive des outils permettant d'utiliser les ontologies d'une manière générale, nous renvoyons le lecteur au Déliverable OntoWeb 1.3 [Fensel and Gómez-Pérez, 2002].

3.5.1 Le stockage des bases de connaissances

Dans le cadre de deux projets européens, l'institut ICS-Forth a développé **RDFSuite**[19], qui est une suite d'outils pour l'interrogation et la gestion de données RDF [Alexaki *et al.*, 2001]. La connaissance terminologique (c'est-à-dire l'ontologie) doit être exprimée dans le langage RDFS et elle est ensuite stockée dans une base de données relationnel-objet (par exemple, *PostgresSQL*). Contrairement à un système de gestion RDF orienté « triplets », où tout est stocké sous formes de triplets RDF, le schéma de la base est le schéma ontologique. Il est également clairement séparé des données (des instances RDF). L'institut a également développé le langage de requêtes **RQL** pour interroger le système. Il s'agit d'un langage déclaratif qui suit une approche fonctionnelle à la OQL et qui permet d'interroger à la fois l'ontologie et les instances, de manipuler les types XML Schema, de composer des chemins complexes dans l'ontologie, et de généraliser les requêtes en autorisant des variables pour les classes et les propriétés.

Au contraire, **Sesame** est un système de gestion de données RDF orienté « triplets » développé par Aidministrator dans le cadre du projet OntoWeb [Broekstra, 2002]. L'architecture est composée d'un module pour le stockage des données (un SGBD comme *mySQL*), d'un module d'interrogation et d'un module d'administration pour ajouter, détruire et modifier des schémas d'ontologies et des données (instances RDF). Le système implémente totalement la sémantique du langage RDF Schema et il infère donc tous les triplets possibles au moment où les ontologies sont entrées dans la base à la manière de Datalog [Ullman, 1996]. L'interaction avec le système se fait au travers de pages Web générées dynamiquement. Concernant l'interrogation, le langage **RQL** est implémenté, mais il est également possible de formuler des requêtes dans le langage maison **SeRQL**, et dans le langage **RDQL**[20], l'évolution du langage **SquishQL** utilisée dans le modèle Jena [Miller *et al.*, 2002].

L'architecture Sesame n'implémente pour l'instant que la sémantique de RDF Schema comme langage d'ontologies, mais il devrait supporter très prochainement le langage OWL Lite. De plus, des raisonnements plus poussés sont possibles grâce au raisonneur **BOR** [Simov and Jordanov, 2002] qui a été intégré à Sesame dans le cadre du projet On-To-Knowledge. En fait, BOR im-

19. `http://athena.ics.forth.gr:9090/RDF/index.html`.
20. Un tutoriel de ce langage est disponible à `http://www.hpl.hp.com/semweb/doc/tutorial/RDQL/`.

plémente la sémantique du langage DAML+OIL, et il est donc proche de ce qui pourra être obtenu avec les futurs moteurs d'inférence OWL (Lite ou DL). Nous avons choisi cette dernière architecture pour mener à bien une de nos expérimentations en raison de sa très grande facilité d'utilisation (voir section 6.3.1).

3.5.2 Les moteurs d'inférence

Parmi les moteurs d'inférences, nous distinguons les langages et API programmables qui permettent assez rapidement de construire ses propres applications, des raisonneurs « prêts à l'emploi » généralement développés autour des logiques de descriptions.

Dans la première catégorie, **TRIPLE** est un langage modulaire de règles basé sur la logique de Horn et sur *F-Logic* qui permet à la fois d'interroger et d'effectuer des inférences sur des données RDF [Sintek and Decker, 2002]. Contrairement aux autres langages de requêtes, il n'implémente pas directement la sémantique des langages autres que RDF, mais il permet de la définir avec des clauses de Horn. Lorsque cela n'est pas possible, par exemple pour les logiques de descriptions DAML+OIL ou OWL, il fournit un accès à des classifieurs externes comme **FaCT**.

KAON est la suite d'outils développés à Karlsruhe pour faire suite au système **Ontobroker** [Studer and Volz, 2003]. Elle contient, entre autre, une API JAVA qui permet d'accéder aux ontologies en s'affranchissant de la manière dont celles-ci sont stockées. Les langages RDF Schema et DAML+OIL sont implémentés, mais l'API n'autorise que des requêtes très simples (par exemple, obtenir toutes les instances d'une classe) puisque aucun mécanisme de filtre n'est prévu.

Jena2[21] est également une API JAVA, qui fournit des raisonneurs pré-définis implémentant les langages RDFS et OWL Lite. Cette API permet aussi de construire un raisonneur à base de règles pour faire du chaînage avant, du chaînage arrière ou pour lancer une exécution hybride.

Dans la seconde catégorie, on peut trouver **FaCT**, un classifieur de logique de descriptions développé par I. Horrocks [Horrocks, 1998]. Celui-ci est aujourd'hui relié à différents éditeurs d'ontologies (**OilEd**, **Protégé2000**) afin de tester leur consistance tout au long du processus de construction. C'est avant tout un classifieur terminologique ($T - Box$), et les raisonnements conduits sur la partie assertionnelle ne sont pas implémentés efficacement. Une évolution de ce classifieur prenant en compte la gestion des types primitifs (entier, chaîne de caractères...) est actuellement développée par l'entreprise NetworkInference avec le produit **Cerebra**.

Racer est également un classifieur de logiques de descriptions qui implémente la sémantique du langage OWL Lite [Haarslev and Möller, 2001]. Il est beaucoup plus performant que **FaCT** en ce qui concerne les raisonnements assertionnels ($A - Box$). En outre, il dépasse les capacités d'expressivité de OWL puisqu'il permet de faire des inférences sur les types primitifs, en gérant les équations linéaires et non linéaires sur les nombres entiers et réels, et les (in)égalités de chaînes de caractères. Nous avons également mené une expérimentation avec cet outil comme nous le verrons dans la section 6.3.1.

3.6 Conclusion

Dans ce chapitre, nous avons montré comment construire des ontologies afin de capturer la connaissance d'un domaine particulier. Ces ontologies sont représentées dans un langage formel doté d'une sémantique et permettent de construire des systèmes à base de connaissances sur lesquels il est possible de raisonner. Nous allons voir dans le chapitre suivant pourquoi et comment

21. `http://jena.sourceforge.net/index.html`.

nous proposons de mobiliser les ontologies pour décrire la structure et le contenu de documents audiovisuels.

Chapitre 4

Une architecture pour raisonner dans les descriptions documentaires

Sommaire

4.1 Description du problème par l'exemple 97
4.2 Langage documentaire ou langage de représentation des connaissances ? 100
4.2.1 La solution MPEG-7 + XML Schema 100
4.2.2 La solution OWL + RDF 106
4.2.3 Discussion 108
4.3 Une architecture duale 109
4.3.1 Présentation générale 109
4.3.2 Formaliser la structure dans une ontologie 110
4.3.3 Engendrer les structures génériques des émissions 111
4.3.4 La base des connaissances 118
4.4 Conclusion 118

Le chapitre 1 a permis de définir nos besoins en terme de descriptions documentaires. Nous avons ensuite présenté des modèles, des langages et des outils provenant de l'ingénierie documentaire (chapitre 2) et de l'ingénierie des connaissances (chapitre 3) pour décrire des documents multimédias.

Dans ce chapitre, nous allons commencer par rappeler notre problématique en l'illustrant par un exemple. Nous donnerons alors plus formellement les fonctionnalités attendues de notre système documentaire audiovisuel (section 4.1). Nous discuterons ensuite du langage adéquat pour représenter les descriptions des documents, en se basant sur l'état de l'art proposé dans les deux chapitres précédents (section 4.2). Nous soutenons alors qu'aucun des langages issus de ces deux paradigmes ne peut satisfaire seul les besoins que nous avons exprimés. Nous proposons donc une architecture générale qui combine les langages MPEG-7/XML Schema et OWL/RDF et qui permet de contrôler et de raisonner sur les descriptions documentaires (section 4.3).

4.1 Description du problème par l'exemple

La manipulation de contenus audiovisuels étant une tâche difficile dans l'absolu, notre hypothèse de travail est que celle-ci doit se faire par l'intermédiaire de la manipulation de *descriptions*

textuelles (au sens le plus général) de ces contenus plutôt que par la manipulation « directe » de ceux-ci. De plus, les descriptions documentaires ont tout intérêt à être représentées à l'aide d'un langage de marquage (XML ou l'un de ses dérivés). En effet, comme nous l'avons vu dans la section 2.1 (page 32), ce type de langage permet de structurer et de contrôler les descriptions des contenus. Or, ce contrôle est indispensable pour garantir la systématicité des descriptions, préalable à toute gestion d'un fond documentaire important en volume.

Depuis le début de ce mémoire, le problème que nous nous posons consiste à pouvoir retrouver, de manière « intelligente », des séquences audiovisuelles par leur *structure* et par leur *contenu*. Le système de recherche devant être accessible au plus grand nombre, il doit être capable de retrouver des séquences audiovisuelles dont la description ne coïnciderait pas exactement avec les termes employés pour formuler la requête. Pour illustrer ce point crucial, nous reproduisons ci-dessous l'exemple donné dans la section 1.2.3 (page 21). Nous utiliserons cet exemple tout au long de ce chapitre afin de montrer les apports de notre approche. Le tableau 4.1 contient la description de trois séquences particulières de l'émission `Stade 2` diffusée le 17 mars 2002.

11 [**Plateau extérieur invités** : Laurent Jalabert et Sandy Casar] à 18 :37 :56 :00 - 00 :02 :43 :00. - France 2 *En direct de Nice, Jean-René GODART interviewe Laurent JALABERT et Sandy CASAR au sujet de la dernière étape de la course cycliste Paris-Nice.*
12 [**Cyclisme** : Paris-Nice, les Français qui brillent] à 18 :40 :39 :00 - 00 :03 :17 :00 Rodolphe Gaudin. - France 2 *Reportage consacré à la course cycliste Paris-Nice. Pour les équipes françaises et les coureurs elle représente un enjeu important en prévision du Tour de France. Commentaires sur images de la course en alternance avec les interviews de Jean-Marie LEBLANC, Didier ROUS et Sandy CASAR.*
13 [**Plateau** : 6ème partie] à 18 :43 :56 :00 - 00 :09 :06 :00. - Eurosport *Plateau composé de la suite de l'interview en direct de Nice de Sandy CASAR par Jean-René GODART au sujet de la course cycliste Paris-Nice et d'une succession de brèves en images commentées par Alexandre BOYON et Laurent PUYAT.*

TAB. 4.1 – *Extrait d'une notice documentaire concernant l'émission* `Stade2`

Imaginons qu'un utilisateur du système souhaite « retrouver *les séquences audiovisuelles* `interview` *de* `Sandy Casar` *dans le cadre de la course cycliste* `Paris-Nice` ». Une recherche en "texte intégral" effectuée sur cette description avec les mots clés `Paris-Nice`, `Sandy Casar` et `interview` va bien retourner les trois séquences numérotées 11, 12 et 13, mais des limitations sont apparues, tant sur le modèle de découpage structurel que sur la description du contenu de l'émission, comme nous l'avons déjà montré dans la section 1.2.3, page 21. Ainsi, nous avons déjà constaté que le modèle de structuration consistant à ne considérer que l'émission dans sa globalité ou ses séquences n'était pas suffisant car il ne permettait pas de répondre précisément aux requêtes (par exemple, retourner une séquence contenue dans une autre séquence, ou encore, retourner une séquence virtuelle construite par mise bout à bout de deux séquences). De plus, la description en texte libre du contenu interdisait toute généralisation de la recherche puisqu'à la requête, « retrouver *toutes les séquences audiovisuelles* `dialogue` *d'un* `coureur cycliste` *dans le cadre d'une* `course à étapes` », la recherche en texte intégral sur cette description n'aurait donné aucun résultat alors que les séquences 11, 12 et 13 sont clairement pertinentes.

Pour remédier à ces problèmes, nous avons suggéré tout d'abord de disposer d'un modèle de représentation documentaire suffisamment riche pour pouvoir contraindre et décrire complètement la structure logique des documents. Nous avons également proposé d'utiliser le calcul d'inférences pour réduire la distance entre les requêtes formulées par des utilisateurs et les descriptions qui font appel au vocabulaire utilisé par les professionnels de l'audiovisuel. Les raisonnements conduits devraient s'appuyer sur une modélisation des connaissances qui sont mobilisées tout au long du processus de description.

Ces besoins peuvent se traduire sous la forme d'une spécification fonctionnelle du système de description et de recherche documentaire. Cette spécification ne préjuge pas des choix techniques à effectuer mais elle précise les fonctions à remplir par le système informatique à mettre en place. Nous répartissons ces fonctions selon deux axes : celles dédiée au système de description des documents audiovisuels et celles dédiées au système de recherche.

La description doit inclure des informations concernant la structure logique des documents audiovisuels et leur contenu. Elle doit être contrôlée tant sur le plan structurel que du point de vue du vocabulaire utilisé. Les fonctions à remplir d'un tel système sont donc les suivantes :

- **F1** : Permettre le contrôle de la représentation formelle de la structure logique des documents. Nous avons vu dans la section 1.1.3 (page 13) que les centres d'archives avaient l'habitude de regrouper les documents en collections, afin de décrire une fois pour toute les éléments communs à un ensemble d'émissions comme par exemple leur structure. Le système informatique doit donc être capable de valider la description d'une émission particulière vis-à-vis du modèle défini pour la collection dont elle fait partie.

- **F2** : Fournir un modèle permettant de représenter la sémantique des descripteurs utilisés pour exprimer la structure des documents, de manière à ce que la machine soit capable de faire des raisonnements sur cette partie de la description.

- **F3** : Fournir un modèle permettant de décrire le contenu des documents de manière à ce que la machine ait accès à leur sens pour pouvoir là encore effectuer des raisonnements.

Une fois la documentation effectuée, les documents doivent pouvoir être exploités, c'est-à-dire en premier lieu être retrouvés. Le système de recherche s'appuiera sur la modélisation des connaissances effectuée lors du processus de description tel que nous l'avons décrit précédemment afin d'effectuer des calculs d'inférence pour mieux satisfaire les requêtes. Ces calculs d'inférences s'effectueront à la fois sur les éléments structurels et les éléments sémantiques des descriptions. Les fonctions à remplir sont donc les suivantes :

- **F4** : Permettre la recherche sémantique de segments documentaires selon des critères de structure et de contenu en effectuant au besoin des inférences sur les descriptions pour améliorer la recherche.

- **F5** : Permettre la navigation dans la structure des documents retrouvés, et ainsi, retrouver le contexte de diffusion des séquences audiovisuelles.

Nous venons, dans cette section, de spécifier les fonctionnalités de notre système d'information documentaire audiovisuel. Sa mise en œuvre informatique permettra de réellement profiter des nouvelles possibilités apportées par la numérisation des documents, telles que nous les avons présentées dans le chapitre 1. Il reste désormais à étudier les solutions techniques permettant de le réaliser, et en particulier à voir quel langage est le plus adapté pour décrire les documents audiovisuels compte-tenu des fonctionnalités attendues du système. Ceci est l'objet de la section suivante.

4.2 Langage documentaire ou langage de représentation des connaissances ?

La description des documents audiovisuels nécessitent d'avoir des descripteurs qu'il faudra ensuite lier au flux. Nous allons étudier dans cette section les différentes possibilités permettant d'obtenir ces descripteurs et avec lesquels il est possible de raisonner : utiliser le langage MPEG-7 et ses mécanismes d'extension (section 4.2.1) ou utiliser un langage de représentation des connaissances (section 4.2.2). Nous verrons alors comment réconcilier ces deux approches (section 4.2.3).

4.2.1 La solution MPEG-7 + XML Schema

XML comme méta-langage documentaire

Nous avons étudié dans le chapitre 2 les modèles et langages issus du monde documentaire permettant de décrire des documents audiovisuels numériques. Aujourd'hui, ces approches prennent appui sur le paradigme des « documents structurés » et préconisent donc l'utilisation de langages de marquage pour décrire les documents. Nous avons déjà présenté les atouts du langage XML, successeur de la norme SGML, qui est incontournable pour représenter des données structurées. Les documents audiovisuels étant fortement structurés, choisir XML comme méta-langage documentaire pour les décrire paraît donc être une idée séduisante.

XML fournit, avec les DTDs et les schémas, des moyens pour contraindre la structure des documents. Nous pouvons ainsi représenter la structure logique d'un ensemble d'émissions et disposer de tous les outils développés autour de XML pour la valider, ce qui répond à la fonctionnalité F1 de notre système de description. Dans ce cas de figure, les relations temporelles entre les séquences audiovisuelles se traduiront en relations méréologiques entre les éléments XML correspondants dans un modèle de schéma de structure. Cependant, comme nous l'avons déjà signalé dans la section 1.2.1 (page 15), le média audiovisuel a la particularité d'être multidimensionnel, et cette solution ne permet de prendre en compte que la dimension temporelle du média, puisqu'il n'est alors plus possible de contraindre l'agencement, dans l'espace, des éléments composants l'image avec les seules relations méréologiques de XML. En pratique, l'usage a tendance à privilégier la dimension temporelle comme dimension d'analyse, c'est-à-dire à segmenter les programmes dans le temps. Une relation univoque peut alors être établie entre l'ordre temporel des séquences audiovisuelles composant une émission et la séquence des éléments XML composant la description de l'émission. Nous nous satisferons de cette solution en arguant donc que l'agencement spatial des composants de l'image pourra toujours être représenté au moyen de balises spécifiques, mais qu'il ne sera pas contrôlé directement dans les schémas.

Le langage XML est également fondamentalement hiérarchique et il ne permet pas de représenter deux objets localisés sur des segments documentaires qui se chevauchent [Auffret, 2000], ce qui est pourtant courant dans le domaine de l'audiovisuel, par exemple lorsque deux séquences s'enchaînent sur un fondu. Cette faiblesse trouve son explication dans l'origine de XML qui fait suite à un langage pensé et mis au point pour encoder du texte où l'on ne trouve jamais, par exemple, deux paragraphes localisés sur la même zone. La transposition à l'audiovisuel n'est donc pas aussi évidente qu'elle n'y paraissait au premier abord. Si le langage des schémas XML permet de représenter le contenu d'un élément comme une expression régulière d'autres éléments, les relations temporelles possibles entre deux séquence audiovisuelles sont bien plus riches comme l'a montré J.F. Allen [Allen, 1983]. Mais là encore, nous considérons que les relations de précédence et d'inclusion entre séquences – qui sont « naturellement » exprimables avec les schémas XML –

sont suffisantes dans la plupart des cas. Globalement, l'utilisation de XML comme méta-langage documentaire permet donc de satisfaire la fonctionnalité F1 de notre système de description.

MPEG-7 comme langage de description

Le langage MPEG-7 est une norme ISO reposant sur la syntaxe XML qui permet de décrire des documents multimédias [MPEG-7, 2001]. Nous avons présenté ce langage dans la section 2.3.2 et nous avons eu l'occasion de louer ses qualités pour décrire la structure des documents. Pour illustrer ce point, reprenons l'exemple présenté dans le tableau 4.1. Les trois séquences numérotées 11, 12 et 13 peuvent se représenter schématiquement par le diagramme temporel de la figure 4.1.

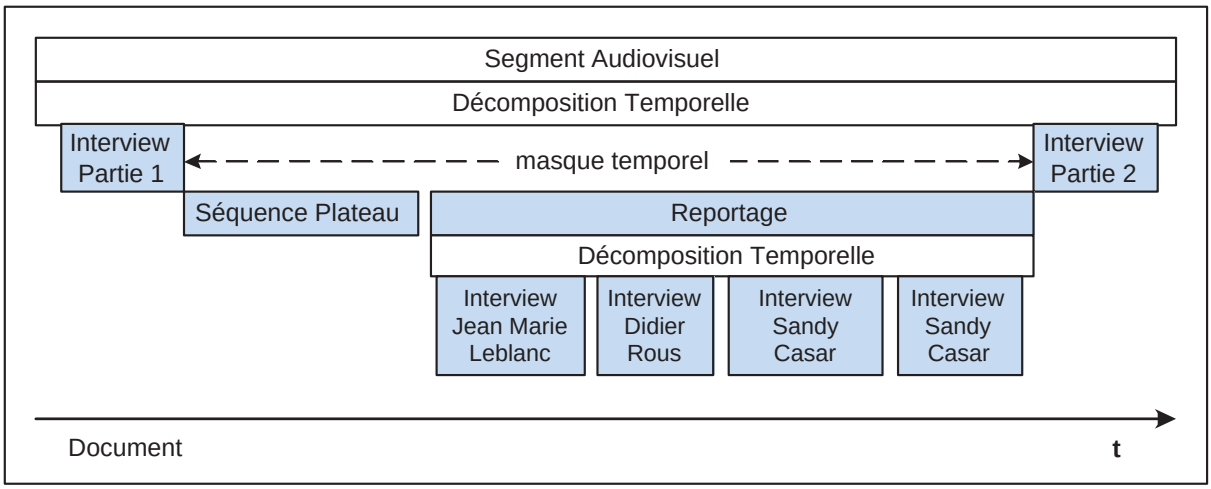


FIG. 4.1 – *Représentation structurelle des séquences décrites dans le tableau 4.1*

Le code 4.1 présente ensuite la représentation de cette structure en MPEG-7. Nous pouvons noter que les possibilités de structuration fines offertes par le langage sont pleinement mises à contribution. Ainsi, l'interview en duplex de Nice réalisée par `Jean-Marie Godart` et découpée en deux parties dans les séquences 11 et 13 est désormais réunie sous la forme d'un segment documentaire, mais qui n'est pas connecté temporellement. De même, le reportage (séquence 12) est finement découpé pour faire apparaître les quatre interview qu'il contient.

Le découpage des séquences audiovisuelles en segments est donc un succès, mais il n'est pas encore suffisant. En effet, comme nous l'avons spécifié dans la section 1.4 (page 27), la description de la structure doit en outre inclure une caractérisation des segments documentaires en terme de genre et de thématique générale. Nous avons déjà constaté que le langage MPEG-7 fournissait un ensemble conséquent de descripteurs – plus de 600 – mais que ceux-ci étaient, pour l'essentiel, liés aux caractéristiques physiques des informations audiovisuelles. Ainsi, le langage ne fournit pas nativement de descripteurs indiquant le genre audiovisuel des séquences (reportage, séquence plateau, interview...) ou permettant de caractériser leur thématique générale (sports, sciences, politique, économie...). Ces descripteurs étant nécessaires pour l'application que nous visons, nous allons voir si les mécanismes d'extension de MPEG-7 nous permettent de les obtenir tout en satisfaisant les deux autres fonctionnalités F2 et F3 de notre système de description.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<Mpeg7 xmlns="urn:mpeg:mpeg7:schema:2001"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xmlns:mpeg7="urn:mpeg:mpeg7:schema:2001"
       xsi:schemaLocation="urn:mpeg:mpeg7:schema:2001 mpeg7-2001-valid.xsd">
 <Description xsi:type="ContentEntityType">
  <MultimediaContent xsi:type="AudioVisualType">
   <AudioVisual id="a6a55234b-2562-4119-a41a-a5fe41e058b5">
    <MediaLocator>
     <MediaUri>file://E:/These/VideosNumeriques/Stade2_17-03-2002.mpg</MediaUri>
    </MediaLocator>
    <TemporalDecomposition>
     <AudioVisualSegment id="a4275a8f2-c17c-4f2f-af52-fd36a115b363">
      <TextAnnotation>
       <FreeTextAnnotation>Interview en duplex de Nice de Jean-Marie Godart.
        Interview en 2 parties de Laurent Jalabert et de Sandy Casar (présence
        d'images de la course et d'un dessin de Chenez)</FreeTextAnnotation>
      </TextAnnotation>
      <TemporalMask>
       <SubInterval>
        <MediaTimePoint>T00:19:43:2257920F14112000</MediaTimePoint>
        <MediaDuration>PT00H01M32S11289600N14112000F</MediaDuration>
       </SubInterval>
       <SubInterval>
        <MediaTimePoint>T00:25:00:10725120F14112000</MediaTimePoint>
        <MediaDuration>PT00H02M39S2257920N14112000F</MediaDuration>
       </SubInterval>
      </TemporalMask>
     </AudioVisualSegment>
     <AudioVisualSegment id="af5b7f531-9d7e-4088-8df6-784bffe2c3ac">
      <TextAnnotation>
       <FreeTextAnnotation>Séquence Plateau, Thème Cyclisme. Annonce du
        reportage concernant le Paris-Nice</FreeTextAnnotation>
      </TextAnnotation>
      <MediaTime>
       <MediaTimePoint>T00:21:15:13547520F14112000</MediaTimePoint>
       <MediaDuration>PT00H00M18S7338240N14112000F</MediaDuration>
      </MediaTime>
     </AudioVisualSegment>
     <AudioVisualSegment id="afb22c30b-8326-45cb-8654-4a4c760b1f35">
      <TextAnnotation>
       <FreeTextAnnotation>Reportage, Thème Cyclisme. Course Paris-Nice
       </FreeTextAnnotation>
      </TextAnnotation>
      <MediaTime>
       <MediaTimePoint>T00:21:34:6773760F14112000</MediaTimePoint>
       <MediaDuration>PT00H03M12S5644800N14112000F</MediaDuration>
      </MediaTime>
```

CODE 4.1 – *Description MPEG-7 de la structure schématisée dans la figure 4.1*

```
      <TemporalDecomposition>
       <AudioVisualSegment id="abe1649cc-232d-4b00-a37a-8e7b1de414d9">
        <TextAnnotation>
         <FreeTextAnnotation>Interview de Jean-Marie Leblanc, directeur
          du Tour de France</FreeTextAnnotation>
        </TextAnnotation>
        <MediaTime>
         <MediaTimePoint>T00:22:01:11854080F14112000</MediaTimePoint>
         <MediaDuration>PT00H00M15S10160640N14112000F</MediaDuration>
        </MediaTime>
       </AudioVisualSegment>
       <AudioVisualSegment id="a9f7fb357-6b3a-4313-a3ea-94a3d0c7d491">
        <TextAnnotation>
         <FreeTextAnnotation>Interview de Didier Rous, champion de France
         </FreeTextAnnotation>
        </TextAnnotation>
        <MediaTime>
         <MediaTimePoint>T00:22:35:564480F14112000</MediaTimePoint>
         <MediaDuration>PT00H00M16S1693440N14112000F</MediaDuration>
        </MediaTime>
       </AudioVisualSegment>
       <AudioVisualSegment id="ad2f17eaa-4228-408b-b564-0e75f068bf2d">
        <TextAnnotation>
         <FreeTextAnnotation>Interview de Sandy Casar avant l'étape
         </FreeTextAnnotation>
        </TextAnnotation>
        <MediaTime>
         <MediaTimePoint>T00:23:39:12983040F14112000</MediaTimePoint>
         <MediaDuration>PT00H00M11S2257920N14112000F</MediaDuration>
        </MediaTime>
       </AudioVisualSegment>
       <AudioVisualSegment id="adb23ab65-f8e7-4b2a-8c98-807197da600a">
        <TextAnnotation>
         <FreeTextAnnotation>Interview de Sandy Casar après l'étape
         </FreeTextAnnotation>
        </TextAnnotation>
        <MediaTime>
         <MediaTimePoint>T00:24:19:10725120F14112000</MediaTimePoint>
         <MediaDuration>PT00H00M07S9031680N14112000F</MediaDuration>
        </MediaTime>
       </AudioVisualSegment>
      </TemporalDecomposition>
     </AudioVisualSegment>
    </TemporalDecomposition>
   </AudioVisual>
  </MultimediaContent>
 </Description>
</Mpeg7>
```

Code 4.1 – *Description MPEG-7 de la structure schématisée dans la figure 4.1* **(suite)**

Les schémas de classification pour étendre le vocabulaire

Une première manière d'étendre la liste des descripteurs MPEG-7 consiste à utiliser les schémas de classification (CS) que nous avons déjà présenté dans la section 2.3.2. Ceux-ci permettent de définir ses propres terminologies dans lesquelles les termes sont principalement hiérarchisés via la relation de spécialisation (NT pour *Narrower Term*) ou de généralisation (BT pour *Broader Term*). Le code 4.2 montre ainsi comment les concepts `Débat` et `Interview` sont définis et rattachés au concept `Séquence Dialogue` qui est une séquence particulière.

```
<ClassificationScheme uri="urn:mpeg:mpeg7:cs:TypologieAVDomainCS">
 <Term termID="Sequence">
  <Name>Séquence</Name>
  <Term termID="SequenceDialogue" relation="NT">
   <Name>Séquence Dialogue</Name>
   <Term termID="Interview" relation="NT">
    <Name>Interview</Name>
    <Definition>Interview ou discussion privilégiant la relation binaire.
     L'interviewer peut être absent à l'image et ses questions supprimées au
     montage</Definition>
   </Term>
   <Term termID="Debat" relation="NT">
    <Name>Débat</Name>
    <Definition>Discussion organisée autour d'un thème et mettant en présence
     plusieurs interlocuteurs dont les arguments s'opposent</Definition>
   </Term>
  </Term>
 </Term>
</ClassificationScheme>
```

Code 4.2 – *Exemple de schéma de classification définissant les termes* `Interview` *et* `Débat` *comme des* `Séquence Dialogue`

Ce mécanisme nous permet de définir tous les descripteurs supplémentaires nécessaires à la caractérisation des segments documentaires pour décrire complètement la structure logique d'un document audiovisuel. Cependant, MPEG-7 limite l'utilisation des schémas de classification aux seuls documents instances puisque ceux-ci sont normalement destinés à la description. En fait, l'utilisation normale du langage, selon ses concepteurs, consiste à produire des descriptions de documents valides vis-à-vis des schémas XML déjà pré-définis dans MPEG-7. Mais dans notre cas, les descripteurs ajoutés au moyen des CS doivent pouvoir être utilisés dans des nouveaux schémas qui permettront de contraindre la structure logique d'un ensemble d'émissions, les descriptions de celles-ci devant par la suite être valides vis-à-vis de ces schémas. Compte tenu de cette limitation du langage, il est clair que nous ne pouvons pas utiliser le mécanismes des CS seul pour ajouter de nouveaux descripteurs. De plus, nous remarquons que les structures de connaissance produites par ces schémas de classification sont de type thésaurus, c'est-à-dire très légère du point de vue des connaissances comparativement aux ontologies que nous avons longuement décrites dans le chapitre 3. Les fonctionnalités F2 et F3 de notre système, qui doivent permettre le raisonnement sur les descriptions, ne seront donc pas complètement satisfaites.

XML Schema pour étendre le vocabulaire

Le langage de définition des descriptions (DDL) est une partie centrale de la norme MPEG-7 puisqu'il fournit les règles syntaxiques pour exprimer et combiner les descripteurs (Ds) et les schémas de description (DSs) [Terzi *et al.*, 2001]. C'est le langage XML Schema du W3C [XML Schema, 2001] qui a été retenu comme DDL pour la norme. Nous avons vu dans la section 2.1.2 que ce langage fournissait les mécanismes nécessaires pour étendre ou raffiner des types existants. Une autre manière d'accroître le nombre de descripteurs consiste donc à utiliser ce mécanisme de dérivation de type sur les descripteurs MPEG-7 pour construire ses propres schémas de description. Par exemple, le code 4.3 définit le type dédié au `Magazine Composite`, sorte d'émission qui contient une alternance de `séquences plateau` et de `reportages` et qui a un présentateur. Le type `Emission` peut lui même être dérivé d'un type MPEG-7 (par exemple, le type abstrait `DSType`).

```
<xsd:complexType name="MagazineCompositeType">
  <xsd:complexContent>
    <xsd:extension base="EmissionType">
      <xsd:sequence maxOccurs="unbounded">
        <xsd:element name="Plateau" type="PlateauType"/>
        <xsd:element name="Reportage" type="ReportageType"/>
      </xsd:sequence>
      <xsd:attribute name="présentateur" type="xsd:string" use="required"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

CODE 4.3 – *Exemple de définition du type* Magazine Composite *en XML Schema*

Ce mécanisme nous permet assurément de représenter la structure logique des documents audiovisuels et de contrôler celle-ci grâce à des schémas définis pour des collections d'émissions, ce qui répond à la fonctionnalité F1 de notre système de description. En revanche, la fonctionnalité F2 ne pourra pas être satisfaite car il n'est pas possible de raisonner sur la structure des descriptions. En effet, XML Schema nous permet d'ajouter de la structure mais il fournit très peu de moyens pour exprimer de la connaissance, autrement dit la sémantique de cette structure. Par exemple, il n'est pas possible de spécifier qu'une `EmissionPlateau` est exactement – *ou strictement équivalent à* – une sorte d'`EmissionSimple` dont toutes les séquences seraient des `SequencesPlateau`, ou encore qu'une `Emission` ne peut pas être à la fois une `EmissionSimple` et une `EmissionComposite`.

Synthèse

En guise de première conclusion, nous constatons que l'utilisation d'un langage de marquage comme méta-langage documentaire est satisfaisante pour notre application. Le langage XML Schema permet de contrôler la structure logique des documents audiovisuels à décrire, même si cela oblige à choisir une dimension d'analyse particulière sur laquelle s'effectuera la validation (dans notre cas, nous avons retenu la dimension temporelle). Le langage MPEG-7 fournit un bon modèle pour découper les documents et décrire leur structure logique, mais il faut pour cela étendre le nombre de descripteurs disponibles. Aucune des deux extensions possibles du langage ne permet en revanche de représenter en machine la sémantique de ces nouveaux descripteurs. Quand à la description sémantique du contenu proprement dit de chaque document, les schémas

de classification n'offrent qu'une expressivité réduite et limiteront fortement les inférences possibles en terme de raisonnement sur les descriptions. L'utilisation conjointe des langages MPEG-7 et XML Schema permet donc de satisfaire la fonctionnalité F1 de notre système de description, mais pas les fonctionnalités F2 et F3.

4.2.2 La solution OWL + RDF

Nous avons vu dans le chapitre 3 comment les ontologies permettent de modéliser les connaissances relatives à un domaine de manière à ce que la machine ait accès à la sémantique des informations. Nous avons également présenté les langages de représentation de connaissances en cours de développement pour faire évoluer le Web vers un « Web Sémantique » [Berners-Lee *et al.*, 2001], où des agents logiciels seront capable de raisonner sur les données. Ces tâches d'inférences étant un besoin clairement établi dans notre système de description, nous pouvons essayer d'utiliser les modèles et langages – définis pour le Web Sémantique – pour décrire les documents audiovisuels.

De la même manière que nous avions défini en XML Schema le type `Magazine Composite` (code 4.3), nous pouvons cette fois définir le concept `Magazine Composite` dans une ontologie. Le code 4.4 présente ainsi la définition de ce concept dans le langage OWL [OWL, 2004].

```
<owl:Class rdf:ID="MagazineComposite">
  <rdfs:subClassOf rdf:resource="#Emission"/>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="sequence"/>
      <owl:allValuesFrom>
        <owl:Class>
          <owl:unionOf rdf:parseType="Collection">
            <owl:Class rdf:about="#Plateau"/>
            <owl:Class rdf:about="#Reportage"/>
          </owl:unionOf>
        </owl:Class>
      </owl:allValuesFrom>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#présentateur"/>
      <owl:hasValue rdf:resource="&xsd;string"/>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
```

Code 4.4 – *Exemple de définition de la classe* Magazine Composite *en OWL*

Cette définition a l'avantage d'expliciter en machine la sémantique de ce qu'est un `Magazine Composite`, à savoir une émission composée de séquences plateau et de reportages et ayant un présentateur. Nous remarquons toutefois que cette définition n'est pas exactement équivalente à celle donnée en XML Schema dans le code 4.3. En effet, la notion d'alternance entre `séquences plateau` et `reportages` a disparu car si celle-ci était facilement modélisable dans le langage des schémas, elle l'est nettement moins en OWL puisque ce dernier ne dispose pas des expressions régulières. Le langage OWL permet également d'exprimer des axiomes pour spécifier par

exemple qu'une émission ne peut pas à la fois être simple et composite (grâce au connecteur : `owl:disjointWith`). Nous avons vu que toutes les connaissances ainsi encodées ont une traduction en triplets RDF [RDF, 2004] qui sont susceptibles d'alimenter une base de connaissances sur laquelle il est alors possible d'effectuer des raisonnements.

Grâce à une modélisation des connaissances, ce langage nous permet donc de produire des descriptions qui ont une sémantique pour la machine et sur lesquelles elle pourra faire des calculs d'inférence. Dans notre exemple, la sémantique des descripteurs permettant de représenter la structure des documents audiovisuels est explicitée et on pourra donc raisonner sur cette partie de la description ce qui correspond à la fonctionnalité F2 de notre système. Le même procédé peut être utilisé pour décrire le contenu des séquences audiovisuelles via une modélisation des connaissances du domaine concerné. Par exemple, la séquence 12 dans le tableau 4.1 pourrait se traduire par un ensemble d'assertions RDF dont le graphe est représenté dans la figure 4.2.

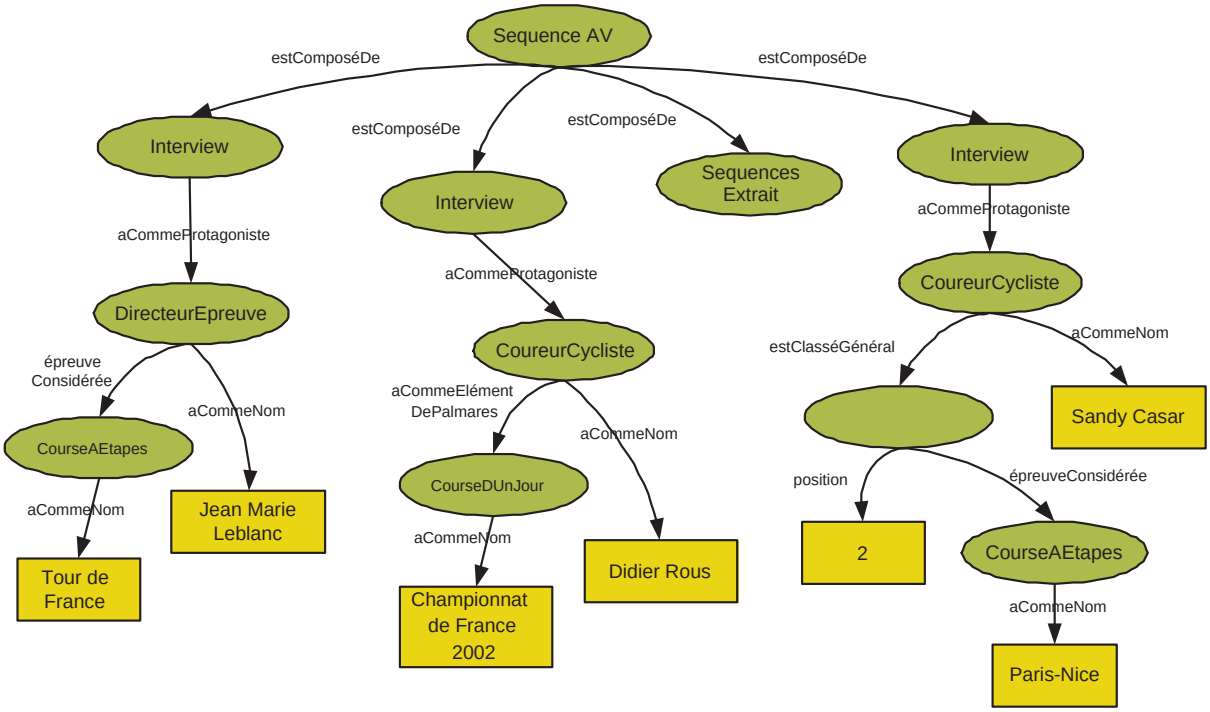


FIG. 4.2 – *Description sémantique de la séquence numérotée 12 du tableau 4.1 sous la forme d'un graphe RDF*

Ce graphe indique que la séquence audiovisuelle est composée de séquences d'extraits et de trois interview : celle du directeur du Tour de France `Jean-Marie Leblanc`, et celles des coureurs cyclistes `Didier Rous`, champion de France 2002, et `Sandy Casar` qui est deuxième au classement général de la course `Paris-Nice`. La modélisation des connaissances du domaine du cyclisme nous permet ici d'effectuer des inférences sur cette partie de la description et donc de répondre à la fonctionnalité F3 du système de description.

L'utilisation d'un langage de représentation des connaissances apparaît donc tout à fait satisfaisante pour répondre à notre besoin de pouvoir effectuer des raisonnements sur les descriptions. Cependant, **la structure du document est complètement perdue**. En effet, il n'est maintenant plus possible de retrouver la séquence qui précède, qui suit ou qui englobe une séquence

donnée, bref, son contexte. Par exemple, dans le graphe de la figure 4.2, on ne sait pas dans quelle ordre ont été diffusé les différentes interview. Or, comme le souligne [Auffret, 2000], le sens d'un document audiovisuel est dû à l'agencement de ses parties, qui témoigne d'une *structure intentionnelle* logique guidant les parcours interprétatifs des spectateurs.

Le problème, comme l'ont montré P.F. Patel-Schneider et J. Siméon dans [Patel-Schneider and Siméon, 2002], vient des fondements des langages utilisés : alors que XML (et XML Schema) sont basés sur un modèle d'arbre où les nœuds sont totalement ordonnés, OWL (et RDF) sont basés sur un modèle de graphe orienté où les arcs sont non ordonnés. En d'autres termes, la notion d'expression régulière utilisée pour contraindre la structure des documents XML est absente dans OWL. Patel-Schneider et Siméon ont alors proposé de représenter l'information d'ordre entre les éléments XML dans un langage nommé SWOL [Patel-Schneider and Siméon, 2002], au moment où le futur langage OWL commençait à être spécifié. Ces propositions n'ont pas été retenues dans le langage OWL et elles ne représentaient de toute façon qu'une solution partielle aux problèmes documentaires. En effet, l'ordre entre les séquences audiovisuelles d'une émission pourrait être représentée en introduisant des relations temporelles (de type de celles d'Allen [Allen, 1983]) directement dans la définition des concepts correspondant aux émissions. Mais la structure logique des documents ne seraient pas pour autant complètement représentée : les relations d'Allen seules ne sont pas suffisantes, par exemple, pour exprimer la notion *d'alternance entre deux séquences* n *fois* comme on l'a utilisé dans la définition de la structure d'un `Magazine Composite` (voir le code 4.3).

Finalement, l'utilisation des langages OWL et RDF permet bien de satisfaire les fonctionnalités F2 et F3 de notre système de description, mais pas la fonctionnalité F1, qui consiste à contraindre la structure logique des émissions en terme de type de séquences audiovisuelles. Il apparaît donc impératif de conserver les pouvoirs d'expression d'un langage documentaire pour décrire et contrôler cette structure.

4.2.3 Discussion

Des tentatives ont déjà eu lieu pour essayer de réconcilier les deux approches que nous venons de présenter. Par exemple, [Hunter and Iannella, 1998] proposent de décrire la sémantique du contenu avec une extension du Dublin Core (voir section 2.3.1) et la structure à l'aide de RDF. Plus précisément, le graphe orienté étiqueté de RDF est utilisé pour exprimer les relations méréologiques nécessaires à la description structurelle des documents. Mais comme l'avouent eux-mêmes les auteurs, cette solution est complexe puisqu'elle amène à étendre puis à utiliser le Dublin Core d'une manière non conforme à son esprit.

[Hunter and Iannella, 1998] proposent également de représenter la structure des documents audiovisuels en SMIL et la sémantique du contenu en RDF. Mais là encore, les possibilités d'expressivité pour décrire la structure sont pauvres puisque seuls les constructeurs `Seq` et `Par` sont disponibles dans le langage SMIL qui, de toute façon, a été élaboré pour décrire des structures de présentation et non pas la structure logique des documents.

Nous avons présenté, dans la section 2.4.2, le modèle des Strates-IA [Prié, 1999; Egyed-Zs., 2003] qui peut être également vu comme une tentative de réconciliation entre une approche documentaire et une approche orientée représentation des connaissances. C'est clairement dans cette direction que nous proposons d'aller, en insistant plus sur l'utilisation d'un langage formel pour décrire la sémantique des descripteurs, tout en conservant les pouvoirs d'expressivité des langages documentaires pour contrôler les descriptions.

Le but de notre approche consiste ainsi à utiliser tout d'abord l'inférence comme moyen de réduire la distance entre requêtes et descriptions, et donc de mieux répondre aux requêtes.

Cela pré-suppose une modélisation et une représentation des connaissances mobilisées lors des raisonnements. Nous avons appelé ontologies toute modélisation définissant des primitives permettant de représenter des connaissances pour ensuite les utiliser dans un traitement formel. Nous proposons donc d'utiliser les ontologies comme moyen pour structurer les connaissances tant documentaires que conceptuelles. Nous nous orientons ainsi vers une fusion des langages issus de ces deux paradigmes. Nous montrons dans la section suivante comment cette fusion peut être réalisée en présentant l'architecture générale que nous proposons.

4.3 Une architecture duale

Nous avons tenté dans la section précédente d'utiliser soit un langage documentaire, soit un langage de représentation des connaissances pour exprimer les descriptions de notre système documentaire audiovisuel. Nous avons alors démontré qu'aucune de ces deux approches ne permettaient, seules, de répondre aux trois fonctionnalités F1, F2 et F3 définies pour ce système.

Notre approche consiste, plutôt qu'à tenter de tirer parti d'une représentation minimale (et non redondante) qui, on l'a vu ne convient pas, à prendre conscience que les deux parties sont bien adaptées à leur tâche mais ne peuvent pas tout seules. On va donc utiliser la représentation de l'une pour enrichir celle de l'autre et automatiser le plus possible les passages entre ces deux représentations.

Dès lors, un certain nombre de problèmes à résoudre se posent :

1. Nous avons vu que les connaissances documentaires correspondent à ce que l'on peut connaître d'un document à travers sa structure ou ses éléments de contenu. Comment formaliser ces connaissances documentaires ? En particulier, quel langage faut-il utiliser pour contrôler et manipuler la structure des descriptions ?

2. Quelle(s) ontologie(s) construire ? S'il semble évident de disposer d'une ontologie propre à un domaine particulier, il semble également intéressant de pouvoir se référer à une ontologie de l'audiovisuel, capitalisant les connaissances des documentalistes, et capable ainsi de fixer la sémantique des descripteurs utilisés pour définir la structure des documents audiovisuels.

3. Comment définir une représentation des connaissances à partir d'une ontologie ou comment utiliser les primitives fournies par l'ontologie pour produire de véritables descriptions formalisées ? En particulier, quel langage opérationnel de représentation des connaissances faut-il utiliser ?

4. Comment articuler description du contenu, description documentaire et localisation de ces descriptions dans le flux audiovisuel ? La difficulté est de définir dans quelle mesure une notion générique définie dans l'ontologie renvoie à un type de localisation sur laquelle des inférences peuvent être menées pour en déduire des conséquences quant à la description, que ce soit pour l'instrumenter, pour préparer l'annotation manuelle, ou pour la compléter.

Nous allons répondre à ces questions dans les sections suivantes. Nous commençons par présenter l'architecture générale qui constitue le cœur de notre travail (section 4.3.1) avant de détailler chacune de ces composantes (sections 4.3.2, 4.3.3 et 4.3.4).

4.3.1 Présentation générale

Afin de prendre en compte les besoins d'expressivité liés à la définition d'un bon modèle documentaire d'une part, et ceux liés à la nécessité de pouvoir mener des inférences d'autre part,

nous proposons de construire une ontologie de l'audiovisuel dont on va traduire certaines parties en types XML Schema. En outre, pour faciliter l'échange des descriptions entre applications, nous proposons un moyen de raccorder ces types aux descripteurs MPEG-7 lorsqu'ils existent. La combinaison de ces types va permettre de construire des schémas de description qui seront à même de contraindre la structure logique des documents audiovisuels.

Le découpage temporel d'un programme particulier fournit alors une instance du modèle de documents auquel il est rattaché qui, lui, valide le squelette de la description. Les séquences d'intérêt y sont repérées temporellement (on dit encore *time-codées*). Mais ces séquences sont aussi des instances de concepts issus de l'ontologie de l'audiovisuel qu'il est possible de traduire en triplets RDF pour alimenter une base de connaissances. Le raisonnement sur les éléments de description est alors possible et le contexte de chaque séquence peut se retrouver à l'aide de la structure documentaire. Enfin, une autre ontologie d'un domaine particulier permet de représenter les connaissances conceptuelles à l'aide desquelles le contenu proprement dit de chacune des séquences sera décrit (section 4.3.4). La figure 4.3 illustre l'architecture générale du système que nous proposons et que nous détaillons dans les sections suivantes.

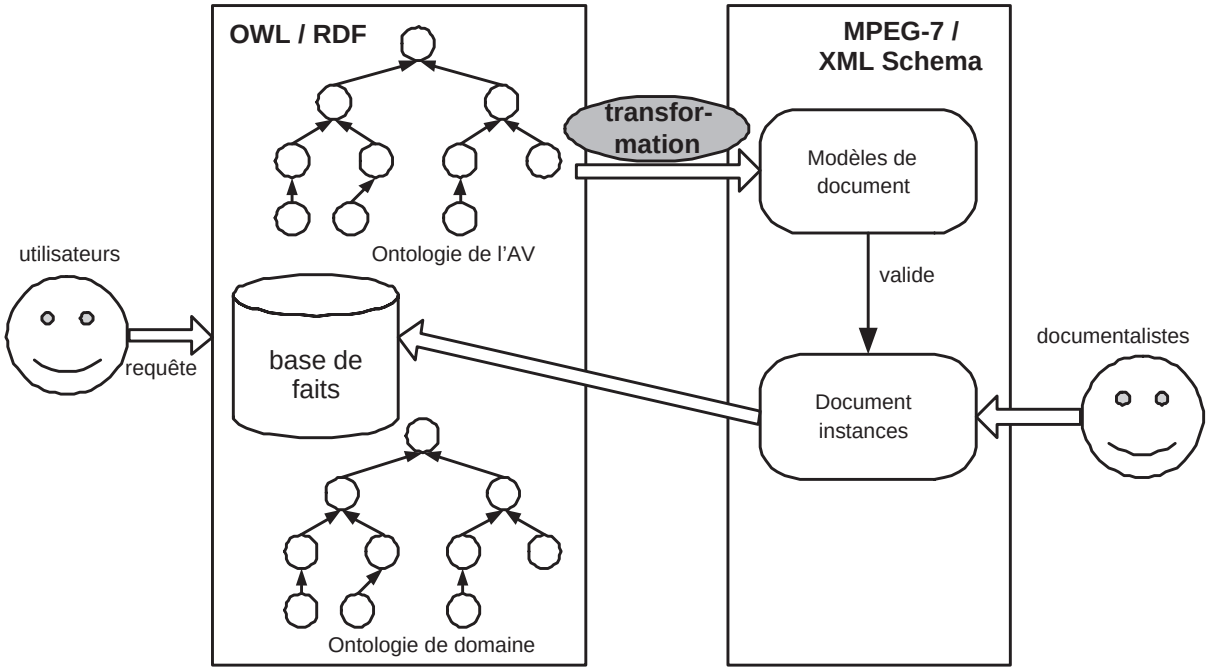


FIG. 4.3 – *Proposition d'architecture pour permettre le raisonnement dans les descriptions documentaires. L'ontologie de l'audiovisuel permet de formaliser les connaissances structurelles des descriptions. Elle est traduite en types XML Schema pour pouvoir exprimer des modèles de document et est instanciée lors de la description. Enfin, une autre ontologie de domaine, avec les faits qui y sont raccordés, permet d'exprimer la connaissance conceptuelle qui sera liée aux éléments structurels de la description.*

4.3.2 Formaliser la structure dans une ontologie

La fonctionnalité F2 de notre système documentaire est de fournir un modèle permettant de représenter la sémantique des descripteurs utilisés pour exprimer la structure logique des do-

cuments audiovisuels, de manière à ce que la machine soit capable de faire des raisonnements sur cette partie de la description. Nous proposons donc de construire une *ontologie de l'audiovisuel* qui va permettre de modéliser et de formaliser les connaissances métiers nécessaires à la description des documents. Plus précisément, cette ontologie va normaliser le libellé des termes couramment utilisés pour décrire la structure et la mise en forme des documents. Elle va également fixer leur sémantique en machine puisque les descripteurs renverront à des concepts formels manipulables informatiquement. La machine pourra ainsi conduire des raisonnements sur ceux-ci.

Reste désormais le problème du langage de représentation des connaissances à utiliser ! Il est nécessaire d'encoder notre ontologie dans un langage opérationnel pour lequel il existe des raisonneurs qui seront à même de réaliser les inférences souhaitées. Nous avons présenté dans la section 3.3 un état de l'art de ces langages et nous avons insisté sur le langage OWL, promu par le W3C pour être le standard pour représenter des ontologies. Les raisonneurs implémentant la sémantique de ce langage commencent à être disponibles et nous en avons présenté quelques uns dans la section 3.5.2. Nous avons donc formalisé l'ontologie de l'audiovisuel dans le langage OWL.

4.3.3 Engendrer les structures génériques des émissions

La fonctionnalité F1 de notre système concerne le contrôle et la représentation de la structure logique des documents. Nous avons vu dans la section 2.1.2 que le langage des schémas XML a été justement élaboré dans ce but. Nous proposons donc de traduire certains concepts de l'ontologie de l'audiovisuel en types XML Schema. La combinaison de ces types, *via* les primitives de modélisation du langage XML Schema, permet de construire des schémas de description qui contrôleront la description de la structure logique des documents.

Cette traduction s'effectue à partir de la définition formelle en OWL des concepts de l'ontologie. Ainsi, certaines *classes* OWL vont se transformer en *types complexes* XML Schema, la relation de *sous-classe* en une *extension* d'un type de base, la restriction sur une *propriété* en un *élément* du modèle de contenu, l'*union* de deux classes en un *choix* dans ce modèle de contenu... Nous réfléchissons actuellement à la manière d'effectuer ces transformations automatiquement (par exemple avec le langage de transformation XSLT), mais il n'est pas sûr que ce processus soit complètement déterministe. La définition des types est ensuite enrichie grâce aux primitives de modélisation du langage XML Schema qui permettent essentiellement de construire des modèles de contenu sous forme d'expressions régulières d'éléments.

Nous proposons également de lier les types XML Schema nouvellement construits aux descripteurs MPEG-7 d'une manière permise par ce dernier langage. Ainsi, les descriptions des documents audiovisuels seront interopérables avec les outils MPEG-7. Nous avons vu justement dans les sections 2.3.2 et 4.2.1 qu'il est possible d'étendre une partie de MPEG-7 à l'aide du langage XML Schema. Cette partie, nommée MDS (*Multimedia Description Schemes*), concerne les descripteurs permettant de décrire des documents multimédias et audiovisuels. En fait, deux types d'extension sont possibles :

- L'extension du MDS (ou *MDS extension*) : dans ce cas, une description utilise du vocabulaire originellement défini dans le MDS mais redéfini dans d'autres schémas en utilisant les mécanismes de dérivation du DDL, à savoir la dérivation de type par restriction et par extension vue dans la section 2.1.2. La validation MPEG-7 d'une telle description revient à vérifier que la syntaxe et la sémantique des éléments définis dans la partie MDS est respectée, mais pour tous les autres éléments, aucune validation n'est requise.

- L'encapsulation du MDS (ou *MDS embedding*) : dans ce cas, une description utilise un

vocabulaire propre qui peut partiellement être défini à l'aide de référence à des descripteurs et schémas de description du MDS. La validation MPEG-7 d'une telle description revient à vérifier que la syntaxe et la sémantique des éléments définis dans la partie MDS est respectée, mais pour tous les autres éléments, aucune validation n'est requise.

La conformité avec MPEG-7 des descriptions produites avec l'un de ces deux mécanismes d'extension est décrite dans la partie 7 de la norme [MPEG-7, 2001]. Cependant, cette partie est loin de résoudre les problèmes complexes posés, ce qui fait dire à certains que l'extension du langage via le DDL n'est que *théorique*. Les auteurs eux-mêmes concèdent l'ambiguïté actuelle mais repoussent la clarification de ces aspects à des futures versions de la norme. Ainsi, compte tenu des spécifications actuelles, une description peut être conforme à MPEG-7 sans être valide vis-à-vis de XML Schema, puisque aucune validation n'est requise sur les éléments non MPEG-7. Pour notre part, nous ajoutons donc une contrainte qui est que les descriptions doivent être également validables par les *parseurs* XML Schema. De plus, la norme recommande d'étendre le MDS plutôt que de l'encapsuler, puisque la première solution permet d'avoir un squelette de description en MPEG-7. C'est donc le mécanisme que nous avons adopté, qui a l'avantage d'assurer une interopérabilité de nos descriptions avec les autres systèmes implémentant le langage MPEG-7. Nous présentons dans la suite une manière générique d'étendre le langage prenant en compte notre ontologie de l'audiovisuel.

Une description MPEG-7 d'un document multimédia commence par spécifier le type de son contenu, dont nous avons rappelé, à l'aide de la figure 2.3 (page 48), les différentes valeurs possibles. Pour étendre le langage, nous commençons donc par ajouter un nouveau type de document multimédia, le type `TV`, qui correspond à une émission diffusée à la télévision. Une description de ce type devra spécifier le nom de l'émission et ses thématiques générales, ainsi que sa date, son heure et sa chaîne de diffusion. La figure 4.4 illustre notre nouvelle hiérarchie des types de description tandis que le code 4.5 donne la définition du type TV.

```
<!-- ************************************************************************
 Ajout d'un nouveau type de contenu multimédia, le type TV, qui correspond à
 une émission diffusée
************************************************************************ -->

<complexType name="TVType">
 <complexContent>
  <extension base="mpeg7:MultimediaContentType">
   <sequence>
    <element name="Emission" type="ina:GenreType"/>
    <element name="Thematique" type="string"/>
   </sequence>
   <attribute name="nom" type="string" use="required"/>
   <attribute name="dateDiffusion" type="date" use="required"/>
   <attribute name="heureDiffusion" type="time" use="required"/>
   <attribute name="chaineDiffusion" type="string" use="required"/>
  </extension>
 </complexContent>
</complexType>
```

Code 4.5 – *Définition du type* TV*, considéré comme un nouveau type de contenu multimédia*

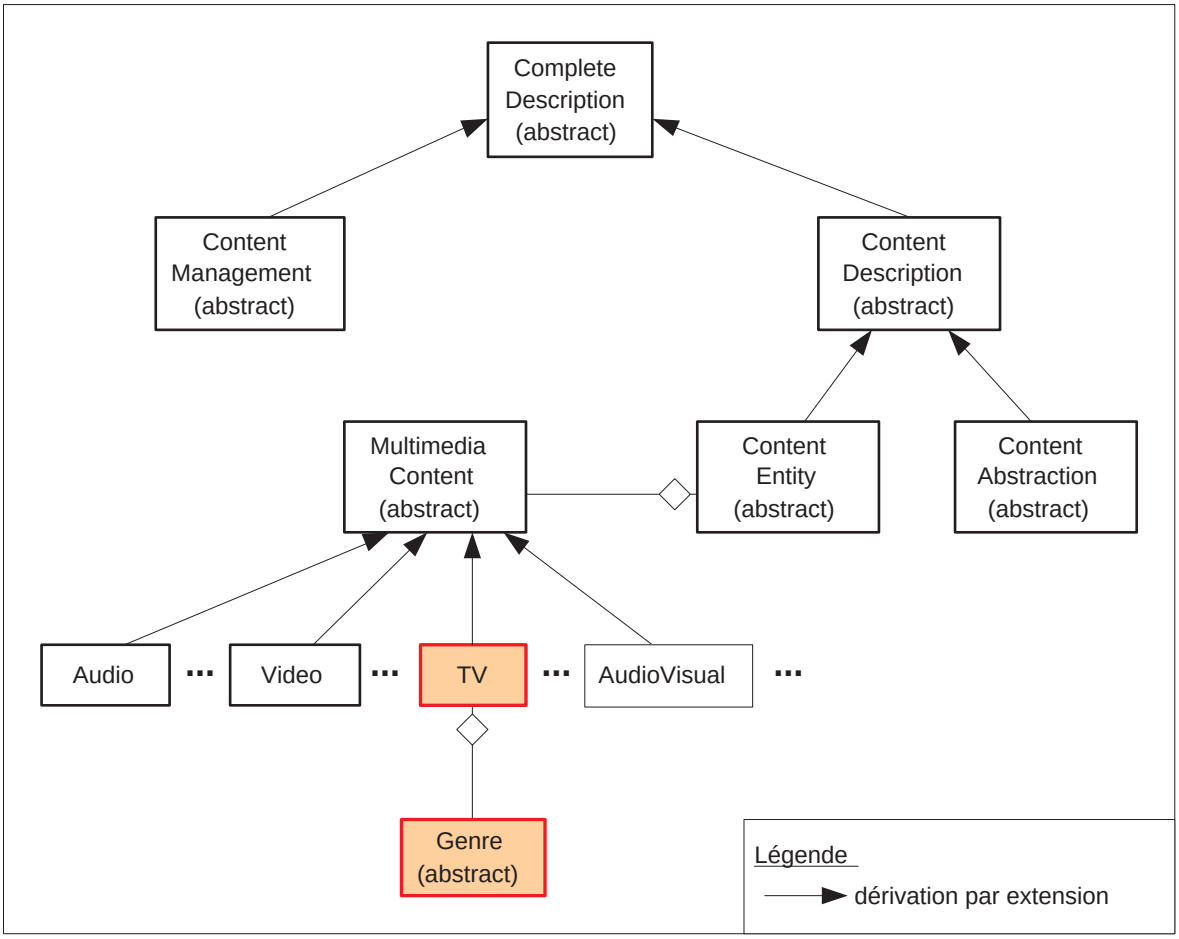


FIG. 4.4 – *Nouvelle hiérarchie des types de description (en MPEG-7 étendu)*

Nous avons vu que dans la norme MPEG-7, l'unité de base de structuration est le *segment*, c'est-à-dire une portion de contenu qui peut être spatial, temporel, ou spatio-temporel. L'émission diffusée est donc tout d'abord un segment particulier qui pourra se décomposer. Elle est ensuite caractérisée par un genre audiovisuel. C'est pourquoi, nous définissons le type abstrait `Genre` comme une extension du type MPEG-7 `Segment` qui possède en outre l'élément `MonMediaLocator` permettant d'associer un URI à une émission, et qui peut se décomposer temporellement. De la même façon, nous définissons le type abstrait `Sequence` comme une extension du type MPEG-7 `Segment` qui peut également se décomposer temporellement. Finalement, une séquence est localisée temporellement au moyen de l'élément `MediaTime` si elle est connectée temporellement, ou de l'élément `TemporalMask` sinon. Le code 4.6 donne la définition de ces deux types.

Le fait d'avoir défini ces deux types comme abstraits interdit de les utiliser directement dans les descriptions : ils doivent alors être spécialisés. Nous allons donc définir, pour chaque genre audiovisuel (dont la sémantique est donnée dans l'ontologie de l'audiovisuel), un nouveau type qui va restreindre le type `Genre`. La restriction va porter sur le type de décomposition que le genre audiovisuel admettra. Cette technique permet de créer de véritables schémas de description personnalisés à chaque genre audiovisuel. À titre d'exemple, le code 4.7 donne la définition du type `MagazineComposite` qui est un genre particulier.

De la même façon, l'ontologie de l'audiovisuel contenant de nombreux types de séquences telles que *reportage* ou *séquence plateau*, nous allons créer pour chacune d'entre elles le type correspondant. Ces derniers sont définis comme des extensions du type `Sequence`, dans la mesure où ils peuvent ajouter des informations spécifiques caractérisant la séquence audiovisuelle. Le code 4.8 donne ainsi la définition des types correspondants aux séquences *reportage*, *interview* et *séquence plateau*.

La figure 4.5 illustre notre nouvelle hiérarchie des types de segments utilisables pour décrire des contenus télévisuels. Elle est à comparer avec la figure 2.4 (page 49) qui donnait la hiérarchie originelle des types de segments disponibles dans le MDS.

Nous avons défini un type pour chaque genre audiovisuel. Chacun d'entre eux admet une décomposition temporelle particulière en terme de type de séquences qu'il peut contenir. Cette décomposition est en fait le schéma de description de chacune des émissions puisqu'elle spécifie l'ordre et l'agencement de leurs différentes séquences. Nous définissons donc tout d'abord le type abstrait – donc devant être spécialisé – `GeneralDecomposition` comme une restriction du type MPEG-7 `SegmentDecomposition` (code 4.9). La figure 4.6 donne la nouvelle hiérarchie des différentes décompositions structurelles applicables aux segments.

Cette restriction est un peu particulière puisqu'elle ne retient rien de son type de base : autrement dit, toute instance de ce type aura un contenu vide. Cela n'est possible que grâce à la définition du type MPEG-7 `SegmentDecomposition` qui a tous ses éléments optionnels. Nous pouvons alors définir les schémas de description correspondant à chaque genre d'émission. Par exemple, le code 4.10 définit le type `MagazineCompositeDecomposition` comme une extension du type `GeneralDecomposition`. Il indique qu'un *magazine composite* ne contient que des *séquences plateau* et des *reportages*. Le seul intérêt de passer par le type intermédiaire `GeneralDecomposition` est d'assurer la validité des schémas. Nous retrouvons ici un des défauts du langage XML Schema déjà mis en avant dans la section 2.1.2 et lié à son mécanisme d'héritage. En effet, la définition du type `MagazineComposite` contient une restriction qui porte sur le type de décomposition qu'elle admet. Mais cette décomposition est définie par extension puisqu'elle ajoute les séquences autorisées dans ce genre d'émission. Comme il faut que cette décomposition soit à un moment définie comme une restriction du type MPEG-7 `SegmentDecomposition` pour conserver la validité des schémas, l'astuce consiste à introduire ce type intermédiaire.

Les séquences peuvent bien entendu contenir d'autres séquences. Elles se décomposent donc

```
<!-- ************************************************************************
 Ajout de deux types, correspondant au genre d'émission et au type de
 séquence, obtenus par extension du type MPEG-7 abstrait 'mpeg7:SegmentType'
************************************************************************** -->

<element name="DecompositionTemporelle"
         type="mpeg7:SegmentDecompositionType" abstract="true"/>

<complexType name="GenreType" abstract="true">
 <complexContent>
  <extension base="mpeg7:SegmentType">
   <sequence>
    <element name="Annotation" type="mpeg7:TextualType" minOccurs="0"/>
    <element name="MonMediaLocator" type="mpeg7:MediaLocatorType"/>
    <choice minOccurs="0" maxOccurs="unbounded">
     <element ref="ina:DecompositionTemporelle"/>
    </choice>
   </sequence>
  </extension>
 </complexContent>
</complexType>

<complexType name="SequenceType" abstract="true">
 <complexContent>
  <extension base="mpeg7:SegmentType">
   <sequence>
    <element name="Annotation" type="mpeg7:TextualType" minOccurs="0"/>
    <element name="Thematique" type="string" minOccurs="0" maxOccurs="3"/>
    <choice minOccurs="0">
     <element name="MediaTime" type="mpeg7:MediaTimeType"/>
     <element name="TemporalMask" type="mpeg7:TemporalMaskType"/>
    </choice>
    <choice minOccurs="0" maxOccurs="unbounded">
     <element ref="ina:DecompositionTemporelle"/>
    </choice>
   </sequence>
  </extension>
 </complexContent>
</complexType>
```

CODE 4.6 – *Définition des types* Genre *et* Sequence*, qui sont tous deux des segments qui peuvent se décomposer temporellement*

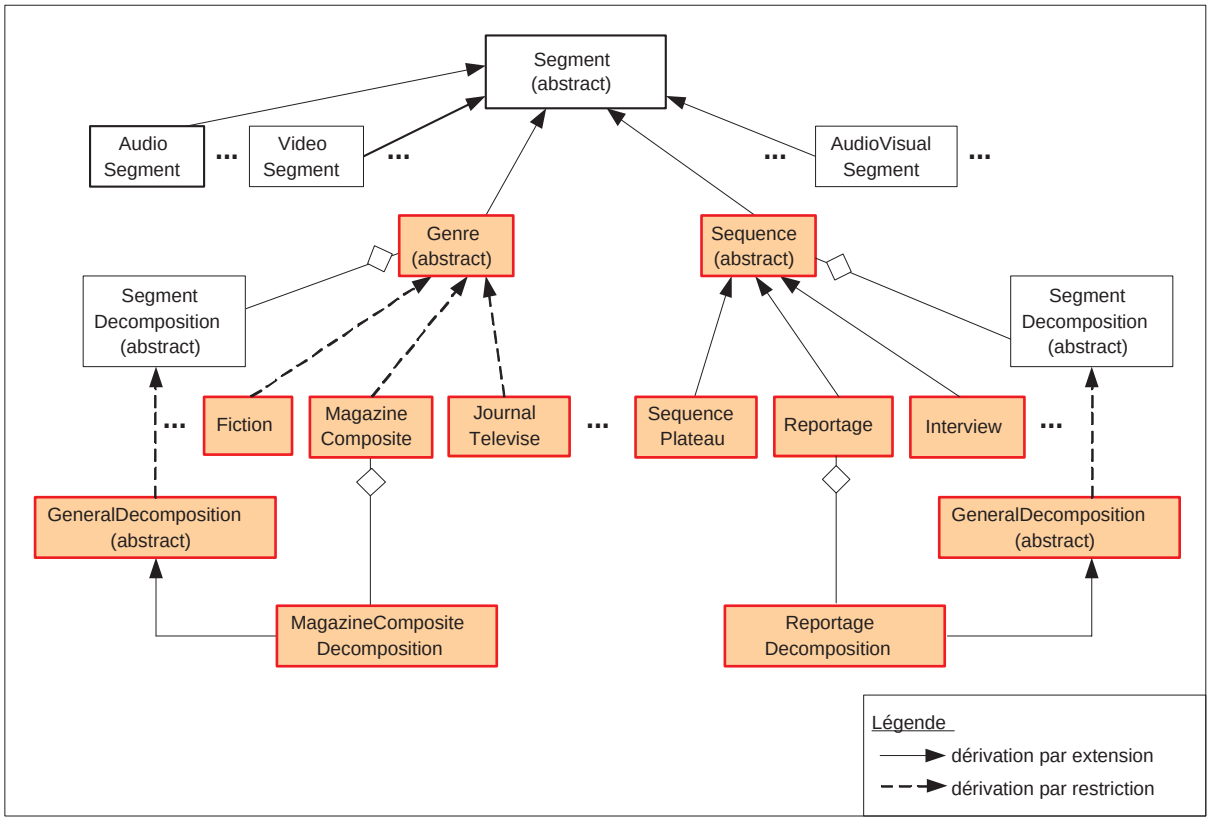


FIG. 4.5 – *Nouvelle hiérarchie des types de segments (en MPEG-7 étendu) : les différents genres et séquences proviennent de l'ontologie de l'audiovisuel*

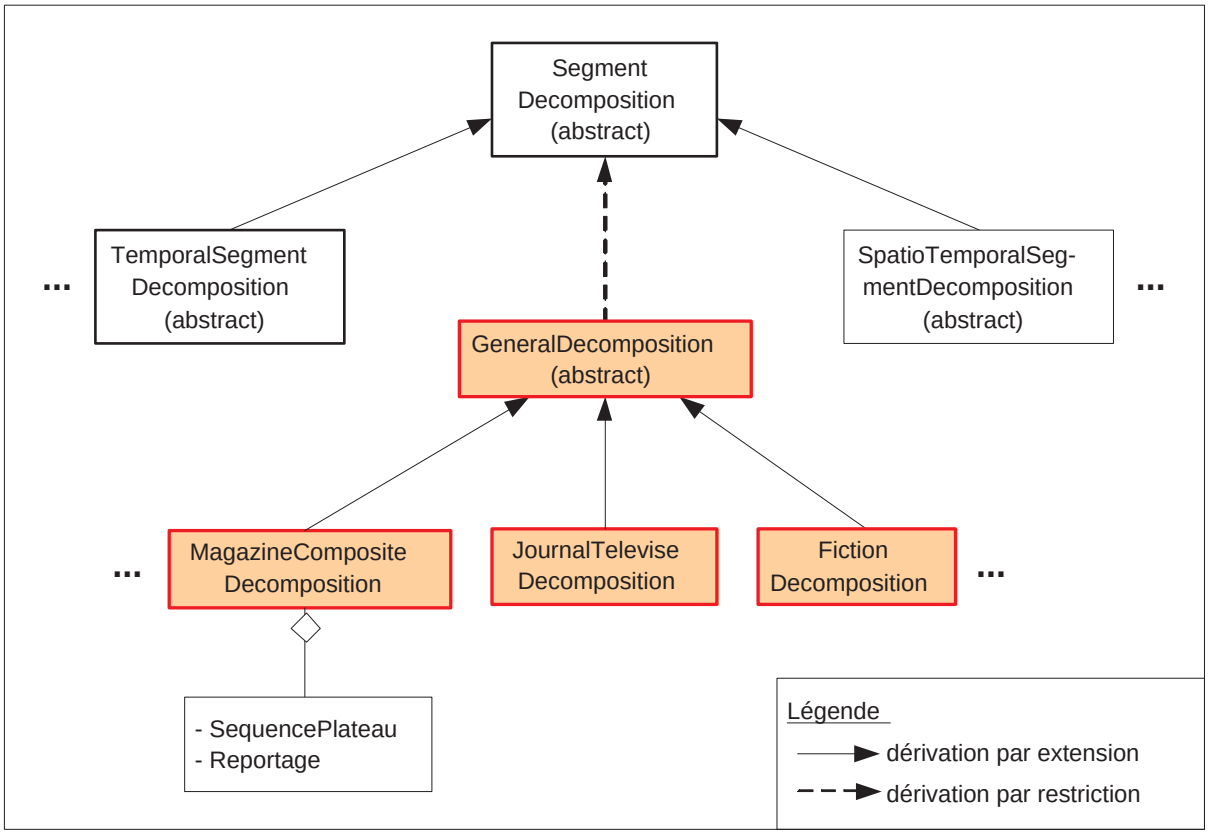


FIG. 4.6 – *Nouvelle hiérarchie des différentes décompositions structurelles applicables aux segments (en MPEG-7 étendu)*

```
<!--*************************************************************************
 Pour chaque sous-concept 'x' du concept 'Emission' de l'ontologie de l'AV, il
 existe un type 'xType' dérivé par restriction du type 'ina:GenreType'
************************************************************************** -->

<element name="MagazineCompositeDecompositionTemporelle"
         type="ina:MagazineCompositeDecompositionType"
         substitutionGroup="ina:DecompositionTemporelle"/>

<complexType name="MagazineCompositeType">
 <complexContent>
  <restriction base="ina:GenreType">
   <sequence>
    <element name="Annotation" type="mpeg7:TextualType" minOccurs="0"/>
    <element name="MonMediaLocator" type="mpeg7:MediaLocatorType"/>
    <choice minOccurs="0" maxOccurs="unbounded">
     <element ref="ina:MagazineCompositeDecompositionTemporelle"/>
    </choice>
   </sequence>
  </restriction>
 </complexContent>
</complexType>
```

Code 4.7 – *Définition du type* MagazineComposite *qui est un genre particulier*

également temporellement et il est possible de contraindre cette décomposition par le même mécanisme. Ainsi, le code 4.11 définit le type `ReportageDecomposition` et spécifie qu'un *reportage* peut contenir des *interview*.

4.3.4 La base des connaissances

La fonctionnalité F3 de notre système documentaire audiovisuel est finalement de fournir un modèle permettant de décrire le contenu des documents de manière à ce que la machine ait accès à leur sens pour pouvoir là encore effectuer des raisonnements. Nous proposons donc de construire une *ontologie de domaine* qui va modéliser et formaliser les connaissances nécessaires à la description du contenu des documents. Les instances des concepts et des relations de cette ontologie sont des assertions utilisées pour la description du contenu.

Ces assertions seront exprimées à l'aide du langage RDF, que nous avons présenté dans la section 3.3.2. Ce langage vient en complément du langage OWL, chargé d'encoder les ontologies, qui a également une traduction en RDF. Finalement, la base de connaissances est un ensemble d'assertions représentées sous forme de triplets RDF. Elle contient les connaissances ontologiques issues des deux ontologies modélisées, et des connaissances assertionnelles fournies par les descriptions.

4.4 Conclusion

Dans ce chapitre, nous avons commencé par rappeler le problème que nous nous posons en l'illustrant, à savoir pouvoir décrire et contrôler la structure logique des documents audiovisuels tout en représentant la sémantique de cette structure et du contenu de manière à ce qu'un

```
<!-- ************************************************************************
 Pour chaque sous-concept 'x' du concept 'Sequence' de l'ontologie de l'AV, il
 existe un type 'xType' dérivé par extension du type 'ina:SequenceType'
************************************************************************** -->

<complexType name="SequencePlateauType">
 <complexContent>
  <extension base="ina:SequenceType">
   <sequence minOccurs="1" maxOccurs="unbounded">
    <element name="Journaliste" type="string"/>
   </sequence>
  </extension>
 </complexContent>
</complexType>

<element name="ReportageDecompositionTemporelle"
         type="ina:ReportageDecompositionType"
         substitutionGroup="ina:DecompositionTemporelle"/>

<complexType name="ReportageType">
 <complexContent>
  <extension base="ina:SequenceType">
   <sequence>
    <element name="Lieux" type="string"/>
   </sequence>
  </extension>
 </complexContent>
</complexType>

<complexType name="InterviewType">
 <complexContent>
  <extension base="ina:SequenceType">
   <sequence>
    <element name="Interviewer" type="string" minOccurs="0"/>
    <element name="Interviewe" type="string" maxOccurs="unbounded"/>
   </sequence>
  </extension>
 </complexContent>
</complexType>
```

Code 4.8 – *Définition des types* SequencePlateau, Reportage *et* Interview

```
<complexType name="GeneralDecompositionType" abstract="true">
 <complexContent>
  <restriction base="mpeg7:SegmentDecompositionType"/>
 </complexContent>
</complexType>
```

Code 4.9 – *Définition du type* GeneralDecomposition *qui permet de décomposer des segments*

```
<!-- ************************************************************************
 Pour chaque sous-concept 'x' du concept 'Emission' de l'ontologie de l'AV, il
 existe un type 'xDecompositionType' dérivé par extension du type
 'ina:GeneralDecompositionType'
********************************************************************** -->

<complexType name="MagazineCompositeDecompositionType">
 <complexContent>
  <extension base="ina:GeneralDecompositionType">
   <!-- Un magazine composite est fait de séquence plateau et de reportage -->
   <choice minOccurs="0" maxOccurs="unbounded">
    <element name="Plateau" type="ina:SequencePlateauType"/>
    <element name="Reportage" type="ina:ReportageType"/>
   </choice>
  </extension>
 </complexContent>
</complexType>
```

Code 4.10 – *Définition du type* MagazineCompositeDecomposition *qui permet de décomposer temporellement les* MagazineComposite

```
<!-- ************************************************************************
 Pour chaque sous-concept 'x' du concept 'Sequence' de l'ontologie de l'AV, il
 peut exister un type 'xDecompositionType' dérivé par extension du type
 'ina:GeneralDecompositionType'
********************************************************************** -->

<complexType name="ReportageDecompositionType">
 <complexContent>
  <extension base="ina:GeneralDecompositionType">
   <!-- Un reportage peut contenir des séquences interview -->
   <choice minOccurs="0" maxOccurs="unbounded">
    <element name="Interview" type="ina:InterviewType"/>
   </choice>
  </extension>
 </complexContent>
</complexType>
```

Code 4.11 – *Définition du type* ReportageDecomposition *qui permet de décomposer temporellement les* Reportage

système informatique puisse conduire des raisonnements sur l'ensemble des descriptions. Nous avons alors discuté des langages possibles pour représenter ces descriptions en se basant sur l'état de l'art de la représentation des documents structurés (chapitre 2) et de l'ingénierie des connaissances (chapitre 3). Sur le constat qu'aucun des langages issus de ces deux paradigmes ne pouvait satisfaire seul les besoins que nous avons exprimés, nous avons proposé une architecture générale permettant de les combiner.

Cette architecture a été élaborée pour décrire des documents audiovisuels mais elle pourrait s'appliquer pour décrire tous types de *documents très structurés*. En effet, la description de ce type de document est aujourd'hui généralement exprimée à l'aide de langages documentaires parce que la structure est prépondérante. Mais notre architecture permet de représenter, en sus, la sémantique des descripteurs utilisés pour décrire la structure sans perdre pour autant les facilités de contrôle apportées par l'ingénierie documentaire. Notre contribution dépasse donc le cadre de l'audiovisuel et peut se transposer à d'autres types de documents structurés. Il suffit pour cela de remplacer l'ontologie de l'audiovisuel par une autre ontologie contenant les concepts nécessaires à la description structurelle. Cela pourra être par exemple les notions de *strophe*, *vers*, *refrain*... pour décrire des poèmes.

Nous allons voir maintenant comment il est possible de mettre en œuvre cette architecture en présentant l'ontologie de l'audiovisuel et les schémas de descriptions que nous avons réalisé (chapitre 5) afin de les utiliser ensuite dans l'indexation de documents concernant un domaine particulier : le cyclisme (chapitre 6).

Chapitre 5

Ontologie de l'audiovisuel et schémas de description

Sommaire

5.1 Construction d'une ontologie de l'audiovisuel 123
5.1.1 Conceptualisation 124
5.1.2 Formalisation 130
5.1.3 Ontologie de l'audiovisuel et Ontologie de MPEG-7 132
5.2 Schémas de description et description de la structure 132
5.2.1 Les schémas de description 132
5.2.2 La description d'une émission particulière : l'exemple du magazine `Stade2` 134
5.3 Expérimentations 138
5.3.1 La dérivation de types en XML Schema 138
5.3.2 Expérimentations avec le langage MPEG-7 139
5.4 Conclusion 147

Dans ce chapitre, nous présentons les réalisations et les expérimentations menées autour de la modélisation d'une ontologie de l'audiovisuel. Nous commençons par détailler le contenu de cette ontologie compte tenu de la méthodologie de construction que nous avons suivie. Nous montrons ainsi comment l'ontologie a été conceptualisée puis formalisée et finalement traduite dans le langage OWL (section 5.1). Nous présentons alors comment les schémas de description sont construits à partir de cette ontologie et nous donnons un exemple d'instance d'un schéma particulier contraignant la structure logique des magazines sportifs composites (section 5.2). Nous menons finalement diverses expérimentations sur les schémas produits pour tester leur validité au sens XML. Nous montrons ainsi comment nous avons été amenés à modifier les schémas MPEG-7 – pourtant normalisés – pour qu'ils puissent être schéma-valides (section 5.3).

5.1 Construction d'une ontologie de l'audiovisuel

L'ontologie de l'audiovisuel constitue la pierre angulaire de l'architecture générale que nous avons proposée dans le chapitre précédent. Nous allons décrire comment nous avons modélisé cette ontologie dans cette section. Elle a été bâtie sur des pratiques établies par les professionnels de l'audiovisuel et peut donc être qualifiée d'« ontologie métier ». Nous avons suivi la méthodologie de construction proposée par B. Bachimont, décrite dans la section 3.4.2, page 88. Nous décrivons

donc comment l'ontologie a été conceptualisée (section 5.1.1) puis formalisée (section 5.1.2). Nous concluons cette section en montrant comment l'ontologie construite vient compléter une ontologie de MPEG-7 proposée par J. Hunter dans des travaux antérieurs [Hunter, 2001] (section 5.1.3).

5.1.1 Conceptualisation

L'objet audiovisuel

Nous avons déjà étudié le cycle de vie des documents audiovisuels (voir section 1.2.2, page 17). Le document télévisuel traverse toute une série d'étapes avant d'être capté et archivé. Les contenus audiovisuels sont d'abord produits pour être vendus à des diffuseurs qui en font des programmes. Ceux-ci s'inscrivent alors dans une grille des programmes (résultat d'une politique éditoriale) qui se transforme en un flux d'images et de sons qui parvient aux téléspectateurs. Notre ontologie de l'audiovisuel commence donc par distinguer l'objet audiovisuel selon la place qu'il occupe dans ce cycle de vie :

- l'*objet de production* peut être une séquence ou une émission complète à structure simple ou composite ;
- l'*objet de diffusion* permet d'inclure le programme dans une tranche horaire, et de spécifier le statut (première diffusion, multi-diffusion...), le mode de diffusion (direct, duplex, liaison téléphonique...) ou encore la périodicité (quotidien, mensuel, aléatoire...) ;
- l'*objet d'archivage* est assimilé à la description du programme et peut s'inscrire dans une collection.

La figure 5.1 donne les axes différentiels adoptés pour conceptualiser l'objet audiovisuel, tandis que la figure 5.2 montre un extrait de la taxinomie avec les principes différentiels associés au concept `EmissionComposite` dans l'éditeur **DOE**.

L'objet de production (émission ou séquence) est ensuite caractérisé en termes de genre audiovisuel et de thématique générale. Pour ce faire, nous avons utilisé différentes typologies (ou vocabulaires formalisés) existantes :

- la typologie Médiamétrie,
- et la typologie INA basée sur des genres, des thèmes et des publics.

Les typologies Médiamétrie et INA

La typologie Médiamétrie est caractérisée par une liste de termes pré-coordonnés et hiérarchisés. Par exemple, la figure 5.3 montre la branche de l'arbre correspondant à une émission de type *sport*. Le code `FBM` indique alors que l'on a un programme de type *sport, magazine, multisports* (par exemple : *Stade 2*). Chaque branche est donc une expression qui s'utilise seule.

Cette typologie est en fait fortement liée à l'usage des publicitaires puisque son objectif est de fournir une classification toujours plus précise aux diffuseurs et aux annonceurs. Elle suit donc une logique fondée sur la programmation, mais elle n'est pas utilisable telle quelle pour la description. Nous l'avons donc utilisé en combinaison avec une autre typologie élaborée par l'INA.

La typologie INA est, au contraire de celle de Médiamétrie, un système à facettes composé de plusieurs listes : des *genres*, des *thèmes* et des *publics*. Chaque liste comporte un nombre

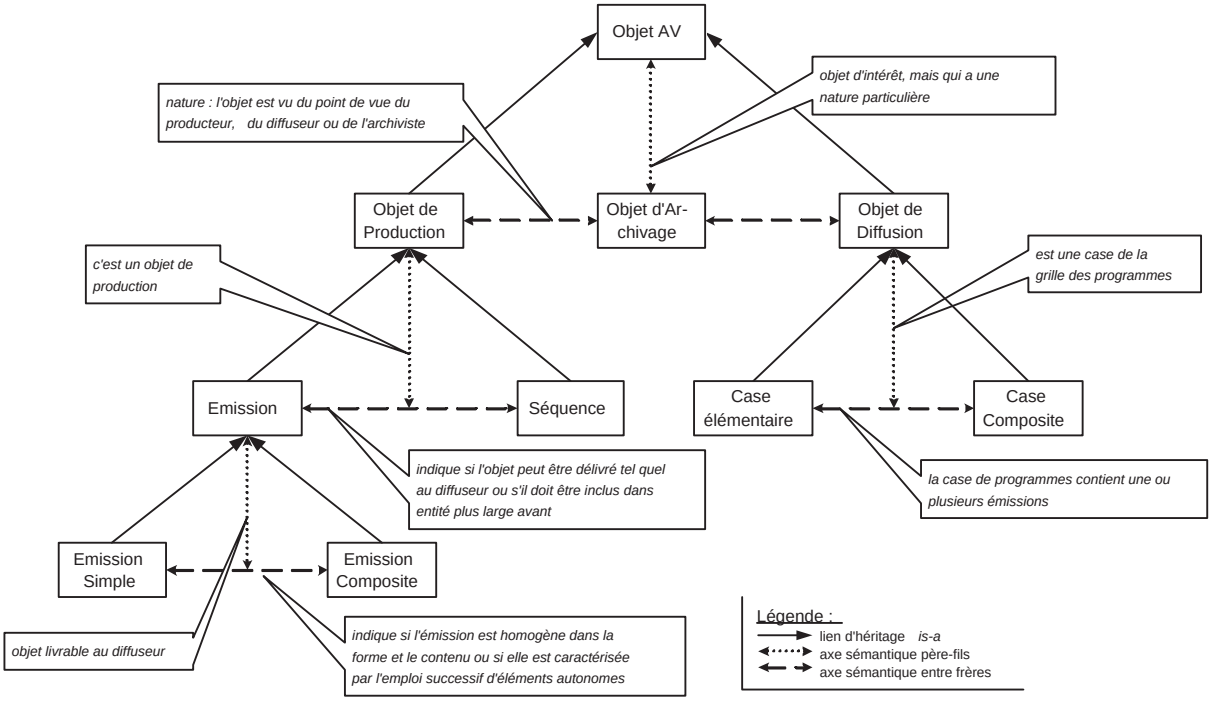


FIG. 5.1 – *Axes différentiels pour les différents types d'objets audiovisuels*

restreint de termes. Un objet de production peut alors être caractérisé par une combinaison de valeurs provenant de ces listes. Ce choix de départ présente comme avantage une grande souplesse liée à la capacité à prendre en compte des genres nouveaux en raison de la multiplicité des combinaisons possibles. L'inconvénient majeur vient du fait qu'il est impossible de hiérarchiser ces combinaisons. De plus, la typologie qui en résulte est globalement moins raffinée que celle provenant de Médiamétrie.

L'ontologie de l'audiovisuel s'inspire de ces deux typologies en prenant la richesse de la première et le système à facettes de la seconde. Les genres audiovisuels deviennent des types d'émissions ou de séquences et spécialisent donc les objets de production. Les thématiques générales et les publics deviennent des propriétés (*qualia*) liées aux objets de production et de diffusion.

Les genres audiovisuels

Les genres correspondent, en règle générale, à la forme audiovisuelle d'une émission ou d'une séquence. Notre point de départ a été la liste des genres possibles pour la télévision donnée par l'INA [Pichon, 1996], que nous énumérons dans le tableau 5.1).

Nous constatons que tous ces genres ne sont pas à mettre sur le même niveau. Si la plupart des termes correspondent bien à une forme audiovisuelle, certains sont dédiés à la caractérisation des séquences (par exemple, `extrait`) tandis que d'autres vont s'appliquer à une émission (par exemple, `journal télévisé`). De plus, on trouve des valeurs qui codent le mode de diffusion (`retransmission`) ou de réalisation (`réalisation TV dans un lieu public`), le statut de l'objet diffusé (`création télévisuelle`), un type de programmation (`tranche horaire`), l'appartenance à un ensemble (`série`), le média d'origine (`long métrage` ou `téléfilm`)...

Nous avons donc restructuré complètement cette liste pour en faire une taxinomie de concepts,

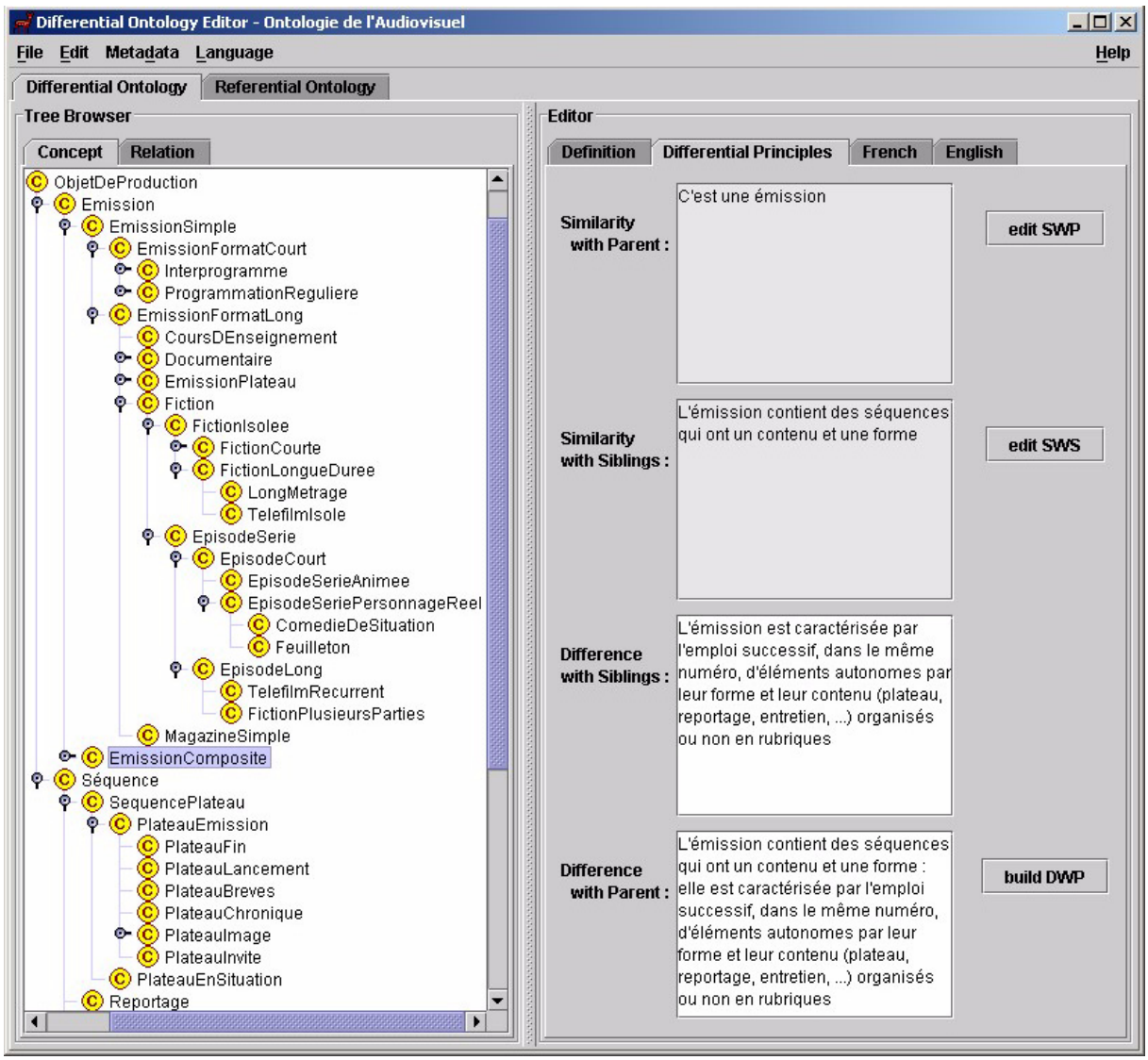


FIG. 5.2 – *Les principes différentiels associés au concept* EmissionComposite *dans l'outil DOE*

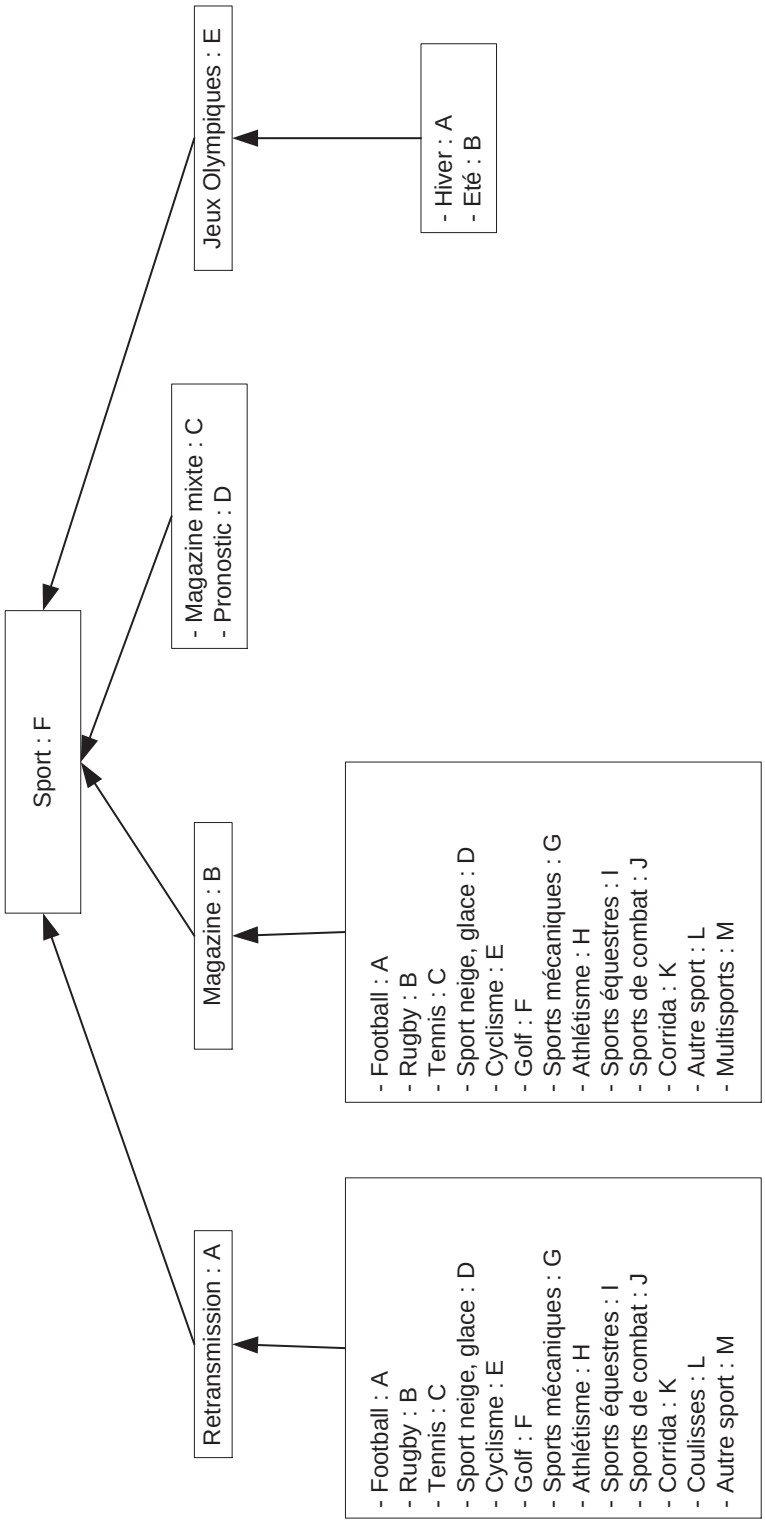


FIG. 5.3 – *Branche de l'arbre correspondant aux émissions de type sport dans la typologie Médiamétrie : le code* `FBM` *désigne une émission de type* `sport, magazine, multisports`

Libellé	Libellé	Libellé
Adaptation	Feuilleton	Programme à base de clips
Animation	Interlude	Programme atypique
Bande annonce	Jeu	Réalisation dans un lieu public
Best of	Journal télévisé	Reality show
Brève	Journée témoin	Récit, Portrait
Comédie de situation	Long métrage	Reportage
Conférence de presse	Magazine	Retransmission
Cours d'enseignement	Message info	Rétrospective
Court métrage	Message publicitaire	Revue de presse
Création télévisuelle	Micro-trottoir	Série
Débat	Mini-programme	Spectacle TV
Déclaration	Montage d'archives	Téléfilm
Documentaire	Oeuvre enregistrée en studio	Témoignage
Entretien, Interview	Plateau en situation	Tranche horaire
Extrait	Presse filmée	Vidéoclip

TAB. 5.1 – *Les genres possibles pour la télévision selon l'INA*

en nous aidant des principes dictés par la méthodologie de B. Bachimont. Des informations supplémentaires disponibles dans [Pichon, 1996] nous ont permis de catégoriser ces concepts comme des émissions simples ou composites, ou des séquences. Ainsi, `Magazine`, `JournalTelevise` et `BestOf` spécialisent le concept `EmissionComposite` tandis que `Documentaire` et `Fiction` spécialisent le concept `EmissionSimple`. De même, nous avons distingué parmi les séquences, les concepts `Reportage` et `SequenceArchive` qui sont des `SequenceMontee`, des concepts `SequencePlateau` et `SequenceExtrait` qui sont des `SequenceNonMontee`. Les résultats de cette conceptualisation sont donnés sous forme de taxinomies capturées par l'outil DOE dans l'annexe B.1. L'annexe B.2 fournit la liste alphabétique des concepts de l'ontologie différentielle avec pour chacun d'eux, les principes différentiels issus de notre méthodologie.

Les propriétés

Outre les genres audiovisuels, notre ontologie contient des propriétés (branche `Qualia`). Celles-ci concernent soit un objet de diffusion, soit un objet de production. Parmi les propriétés de diffusion, on va trouver les concepts relatifs :

- au mode de diffusion (par exemple, `Direct`, `Differe`),
- au statut de diffusion (par exemple, `PremiereDiffusion`, `Multidiffusion`),
- à la périodicité du programme (par exemple, `WeekEnd`, `BiMensuel`) ou encore
- au public visé (par exemple, `ProgrammeJeunesse`, `PlusDe16Ans`).

Les propriétés des objets de production sont en fait des thématiques générales qui sont considérées sous l'un ou l'autre des angles suivants :

- le contenu proprement parlé de l'émission c'est-à-dire ce dont on parle ;
- son type, c'est-à-dire la référence du domaine dans laquelle elle s'inscrit (spectacle, sciences, information, vie quotidienne...).

[Pichon, 1996] fournit une liste à plat de ces thématiques que nous reproduisons dans le tableau 5.2. Nous constatons donc que ces termes peuvent qualifier non seulement le contenu,

Libellé	Libellé	Libellé	Libellé
Spectacle	Humour	Arts	Politique
Théâtre	Érotisme	Littérature	Economie
Fiction	Sciences	Sciences humaines	Société
Fantastique	Sciences naturelles	Histoire	Sports
Cinéma	Sciences exactes	Religion	Informations pratiques
Média	Médecine, Santé	Ethnologie	Loisirs
Intrigue policière	Technologie	Sociologie	Tourisme
Musique	Industrie	Psychologie	Aventure, Exploration
Danse	Environnement	Géographie	Insertion sociale
Variétés	Artisanat	Philosophie	Vie professionnelle
Cirque	Faune	Information	

TAB. 5.2 – *Les thèmes possibles pour la télévision selon l'INA*

mais également :

- le type de l'œuvre dans le cas des fictions (`intrigues policières`), des diffusions intégrales de spectacles (`théâtre, opéra, cirque, musique, variétés`...) et par extension des retransmissions sportives (`sport`) ;
- un domaine de connaissance pour tous les termes qualifiant les sciences humaines, les arts ou les sciences ;
- la tonalité de l'émission pour le code « `humour` ».

De la même manière que pour les genres, nous avons structuré cette liste en taxinomie en nous aidant des principes dictés par notre méthodologie. Nous l'avons également largement complétée en nous aidant d'une structure hiérarchique de thèmes standardisés par le consortium TV Anytime [Pfeiffer and Srinivasan, 2000] que nous avons présenté dans la section 2.3.3, page 51. Finalement, notre ontologie contient 112 thèmes pour caractériser aussi bien les émissions que les séquences.

Les activités

La dernière partie de notre ontologie contient les concepts permettant de décrire les différentes activités impliquées dans la production d'un programme. Ces activités se répartissent entre le tournage (`PriseDeSon`, `MouvementDeCaméra`...), le montage (`MontageAudio`, `Volet`, `Fondu`...) et la post-production (`EffetsSpeciaux`, `SousTitre`...). Une branche de l'arbre permet également de spécialiser le concept `Personne` en distinguant le personnel de production (`Producteur`, `Realisateur`...) des éventuels participants à une émission (`Chroniqueur`, `Artiste`...).

Cette partie de l'ontologie reprend pour l'essentiel un travail antérieur réalisé par T. Dechilly [Dechilly and Bachimont, 2000]. Nous l'avons légèrement remanié afin de prendre en considération les travaux de H. Zettl [Zettl, 1999]. Dans son ouvrage *Sight Sound Motion, Applied Media Aesthetics*, H. Zettl, qui a été lui-même réalisateur avant d'enseigner la production télévisuelle, procède à une analyse des dimensions « esthétiques » des supports de communication audiovisuelle, télévision et cinéma. Dans une démarche qu'on peut qualifier de sémiotique, il propose un modèle des éléments constituant l'objet audiovisuel et explique comment ces éléments génèrent des effets perceptifs désirés par le réalisateur.

Synthèse

Nous avons utilisé l'outil **DOE** – que nous avons nous même développé – pour mener à bien toute la conceptualisation de l'ontologie de l'audiovisuel. L'ontologie différentielle obtenue est disponible sur le Web à `http://opales.ina.fr/public/ontologies/`.

Il est difficile d'évaluer globalement l'ontologie produite mais l'application de la méthodologie proposée par B. Bachimont apporte clairement une aide précieuse pendant la phase de conceptualisation. L'ontologie est plus lisible puisque les choix de modélisation son explicités par les principes différentiels. Nous remarquons également un effet de bord dû à la méthodologie : celle-ci a tendance à augmenter le nombre total de concepts en ajoutant des concepts non-terminologiques (c'est-à-dire que l'on ne peut pas trouver en corpus) dans l'ontologie. En effet, pour préserver à la fois la structure arborescente de l'ontologie et le principe d'axe différentiel unique le long de la hiérarchie, la méthodologie amène à introduire des concepts tout à fait artificiellement. Par exemple, pour distinguer les différents types de `SequencePlateau`, nous avons défini les deux concepts `PlateauCaracteriseParSaForme` et `PlateauCaracteriseParSaPosition`. Les concepts `PlateauImage` et `PlateauInvite` vont alors spécialiser le premier tandis que les concepts `PlateauDebut`, `PlateauLancement` et `PlateauFin` vont spécialiser le second. Nous évaluons finalement à un tiers le nombre de ces concepts ajoutés dans l'ontologie, ce qui peut paraître important. Mais en pratique, ces concepts ne sont pas complètement inutiles, et ils permettent par exemple de généraliser des requêtes assez facilement.

La phase de conceptualisation est ensuite suivie par une phase de formalisation de l'ontologie. Nous la détaillons dans la section suivante.

5.1.2 Formalisation

Dans l'ontologie formelle, les concepts ne sont plus définis par des principes différentiels mais par des mécanismes de composition de sens fournis par une sémantique référentielle dans un langage formel. En pratique, il s'agit de reprendre l'ontologie différentielle modélisée dans l'étape précédente et de définir le plus possible les concepts – avec des conditions nécessaires et suffisantes – par des combinaisons booléennes d'autres concepts, ou par l'expression de contraintes relationnelles. Nous avons vu dans la section 3.3 différents langages pour représenter des ontologies sur le Web, et en particulier le langage OWL qui sera très prochainement le standard promu par le W3C. Nous avons donc représenté l'ontologie de l'audiovisuel dans ce langage. Ainsi, le code 5.1 définit (au sens des logiques de descriptions) dans le langage OWL le concept `EmissionPlateau` comme étant exactement une `EmissionSimple` dont toutes les séquences sont des `SequencePlateau`.

```
<owl:Class rdf:ID="EmissionPlateau">
  <owl:intersectionOf rdf:parseType="Collection">
    <owl:Class rdf:about="#EmissionSimple"/>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#sequence"/>
      <owl:allValuesFrom rdf:resource="#SequencePlateau"/>
    </owl:Restriction>
  </owl:intersectionOf>
</owl:Class>
```

CODE 5.1 – *Définition du concept* EmissionPlateau *en OWL*

La formalisation de l'ontologie permet également de spécifier les domaines des relations et d'ajouter des axiomes logiques selon les possibilités d'expressivité du langage de représentation dans lequel sera encodée l'ontologie. L'ontologie formelle de l'audiovisuel contient donc quelques axiomes qu'il est possible de représenter en OWL DL (voir section 3.3.3). Par exemple, nous avons spécifié qu'une émission ne pouvait pas être à la fois simple et composite (code 5.2).

```
<owl:Class rdf:ID="EmissionComposite">
  <rdfs:subClassOf rdf:resource="#Emission"/>
  <owl:disjointWith rdf:resource="#EmissionSimple"/>
</owl:Class>
```

Code 5.2 – *Définition d'un axiome en OWL DL*

L'absence de types primitifs dans le langage OWL rend toutefois certaines définitions de concepts plus délicates. En effet, dans sa version actuelle, le langage OWL délègue la gestion des types primitifs au langage XML Schema en arguant que celui-ci dispose déjà de tous les mécanismes nécessaires à leur construction. Le problème est que cette décision oblige à *construire* et à *nommer* de nombreux types intermédiaires qui seront ensuite référencés dans la définition des concepts. Ainsi, dans l'ontologie de l'audiovisuel, le concept `EpisodeLong` est défini comme étant un épisode d'une série de fiction dont la durée minimale est de 52 minutes et la durée maximale de 90 minutes. Pour exprimer cette définition en OWL, il faut donc construire auparavant un type dont les valeurs permises seront comprises dans l'intervalle [52 min ; 90 min] (code 5.3).

```
<xsd:simpleType name="DureeCompriseEntre52Et90">
  <xsd:restriction base="xsd:duration">
    <xsd:minInclusive value="P52M"/>
    <xsd:naxInclusive value="P90M"/>
  </xsf:restriction>
</xsd:simpleType>

<owl:Class rdf:ID="EpisodeLong">
  <owl:intersectionOf rdf:parseType="Collection">
    <owl:Class rdf:about="#EpisodeSerie"/>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#duree"/>
      <owl:allValuesFrom rdf:datatype="&xsd;DureeCompriseEntre52Et90"/>
    </owl:Restriction>
  </owl:intersectionOf>
</owl:Class>
```

Code 5.3 – *Définition du concept* EpisodeLong *en OWL à l'aide du type* DureeCompriseEntre-52Et90 *défini en XML Schema*

Cette solution est évidemment contraignante : il faut nommer explicitement un type pour pouvoir définir ce concept. Il serait donc souhaitable de disposer de mécanismes de construction de types primitifs directement dans le langage d'ontologies. Cette remarque rejoint les travaux de J.Z. Pan et I. Horrocks qui présentent dans [Pan and Horrocks, 2003] une manière d'ajouter de tels constructeurs dans OWL. L'approche proposée permet également de maîtriser la complexité des algorithmes permettant de raisonner avec cette extension, et les auteurs espèrent fortement que ces nouveaux constructeurs pourront être intégrés dans une future version du langage OWL.

Nous avons vu dans la section 3.4.4 que notre éditeur d'ontologies **DOE** ne prenait pas en charge tout le processus de formalisation présenté ci-dessus. Pour formaliser l'ontologie de l'audiovisuel, nous avons donc eu recours à un autre outil, **OilEd** [Bechhofer *et al.*, 2001], que nous avons présenté dans la section 3.4.3. L'interopérabilité entre ces deux outils est assuré en transformant l'ontologie modélisée dans DOE en RDFS. Nous avons choisi d'utiliser ce langage intermédiaire car il est à la fois proche du modèle de représentation de notre outil, et en même temps supporté par tous les autres environnements de construction [Isaac *et al.*, 2003].

L'ontologie de l'audiovisuel ainsi formalisée peut être raccrochée à une autre ontologie concernant le langage MPEG-7 et proposée par J. Hunter [Hunter, 2001]. Nous montrons comment dans la section suivante.

5.1.3 Ontologie de l'audiovisuel et Ontologie de MPEG-7

Pour pallier le manque de sémantique de MPEG-7 que nous avons constaté dans la section 2.3.2 (voir page 46), J. Hunter a déjà proposé une ontologie qui conceptualise et formalise les principaux descripteurs de ce langage [Hunter, 2001]. Ainsi, elle a déjà construit une sorte d'ontologie de MPEG-7, encodée dans le langage DAML+OIL, qui définit les descripteurs permettant de caractériser les segments selon le média, leur décomposition possible ou encore les entités basiques non multimédias (personne, rôle, lieu, instrument...). La figure 5.4 montre cette ontologie une fois importée dans l'outil **OilEd**.

L'ontologie proposée reprend pour l'essentiel la définition des types donnée dans la norme MPEG-7, en traduisant la dérivation des types en une relation de spécialisation. Le résultat obtenu n'est clairement pas une « ontologie différentielle » et l'héritage multiple est d'ailleurs largement utilisé. En revanche, nous pouvons lier cette ontologie à l'ontologie formelle de l'audiovisuel qui n'impose plus d'avoir une hiérarchie strictement arborescente de concepts. Les objets de production définis dans l'ontologie de l'audiovisuel deviennent ainsi un nouveau type de contenu multimédia et viennent spécialiser le concept `MultimediaContent`.

Nous montrons dans la section suivante comment l'ontologie de l'audiovisuel permet d'engendrer des schémas de description pour contraindre la structure logique des émissions.

5.2 Schémas de description et description de la structure

L'ontologie de l'audiovisuel décrite dans la section précédente a permis de modéliser et de formaliser les connaissances métiers nécessaires à la description de la structure des documents dans un langage de représentation des connaissances. Mais, nous avons vu également que les langages documentaires sont plus adaptés pour exprimer et contrôler cette structure. C'est pourquoi, nous avons proposé de traduire certains concepts de l'ontologie en types XML Schema pour construire des schémas de description. Nous allons donc montrer comment ceux-ci sont construits (section 5.2.1) avant de donner un exemple de description d'une émission particulière qui sera une instance d'un de ces schémas (section 5.2.2).

5.2.1 Les schémas de description

Nous avons vu dans la section 4.3.3 une manière de traduire les concepts de l'ontologie exprimée en OWL en types XML Schema. Ces types doivent ensuite être combinés pour constituer un schéma de description qui contrôlera la structure générale d'une classe d'émissions. Mais quels doivent être les concepts concernés par cette traduction ? Dans notre cas, la structure d'une émission consiste à la décomposer temporellement en segments qu'on cherche ensuite à

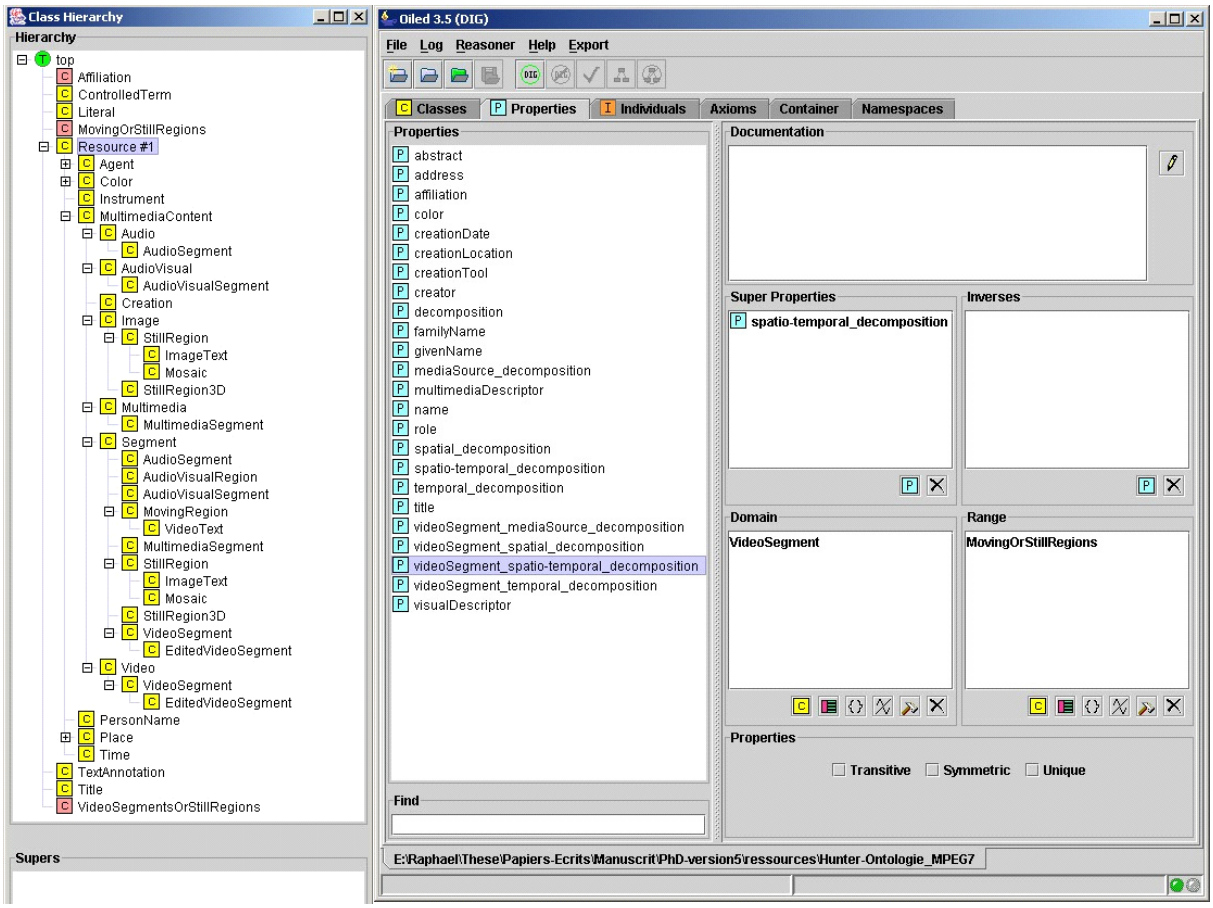


FIG. 5.4 – *L'ontologie MPEG-7, d'après [Hunter, 2001], importée dans l'outil OilEd : à gauche, la taxinomie des concepts et à droite la liste des propriétés avec leur définition*

caractériser en termes de genres audiovisuels. Ce sont donc ces concepts que nous avons traduis en types XML Schema. Dans l'ontologie de l'audiovisuel, il s'agit de la branche `ObjetDeProduction` qui regroupe l'ensemble des types d'émissions et de séquences.

On observe alors une certaine redondance d'information entre les concepts de l'ontologie et les types XML Schema obtenus après traduction. En premier lieu, la spécialisation des concepts est recréée avec l'héritage de types. Les attributs des concepts deviennent également des attributs de ces types. Les propriétés attachées aux concepts deviennent, eux, des éléments dans la structure des types. La combinaison de ces types ajoutent en revanche de la connaissance qu'il n'était pas possible de modéliser dans l'ontologie et qui justifie justement l'utilisation d'un langage documentaire. Ce sera par exemple l'utilisation d'une expression régulière combinant des types de séquences audiovisuelles pour exprimer la structure d'une classe d'émissions. Ainsi, le tableau 5.3 donne, sous forme EBNF, la structure méréologique des émissions classifiées comme des *magazines sportifs*. Ce schéma de description indique qu'un magazine sportif commence toujours par une séquence `PlateauDebut`, suivi par un certain nombre de séquences qui sont soit une séquence `PlateauCaracteriseParSaForme`, soit un enchaînement `PlateauLancement`, `Reportage`, et se termine par une séquence `PlateauFin`. De plus, la hiérarchie des types conserve la modélisation ontologique du domaine de l'audiovisuel et nous indique donc que les types `PlateauImage` et `PlateauInvite` peuvent se substituer au type `PlateauCaracteriseParSaForme`. Finalement, les `Reportage` peuvent contenir des `Interview` et des `SequenceExtrait`.

```
                MagazineSportif  :=  ( PlateauDebut,
                                       ( PlateauCaracteriseParSaForme |
                                         (PlateauLancement, Reportage) )+,
                                      PlateauFin )
PlateauCaracteriseParSaForme  :=  (SequenceExtrait)*
                    Reportage  :=  (Interview | SequenceExtrait)*
```

TAB. 5.3 – *Schéma de description pour contrôler la structure d'un* Magazine Sportif *(forme EBNF)*

Nous avons également proposé de raccorder tous ces nouveaux types à des descripteurs MPEG-7 existants et nous avons donné dans la section 4.3.3 une manière générique de le faire. Nous avons vu que le modèle de structuration MPEG-7 était basé sur la notion de segment qui acceptait des décompositions en d'autres segments. Ainsi, pour reprendre l'exemple précédent, le type `MagazineSportif` va hériter du type `Genre`, et les différentes sortes de séquences vont hériter du type `Sequence` tel que que nous les avons défini dans le code 4.6, page 115. Ceux-ci vont avoir dans leur modèle de contenu des décompositions temporelles particulières afin de refléter la grammaire EBNF définie dans le tableau 5.3.

Nous avons ainsi défini le schéma de description de toutes les émissions cataloguées comme des *magazines sportifs*. La description d'une émission particulière de ce genre sera une instance du schéma. Cette description est représentée partiellement en *pur* MPEG-7 ce qui va faciliter son échange entre applications. Nous donnons dans la section suivante un exemple complet d'une telle description.

5.2.2 La description d'une émission particulière : l'exemple du magazine `Stade2`

La description d'un document audiovisuel commence par la localisation de ses entités d'intérêts. En ce qui nous concerne, il s'agit de repérer dans le temps des segments dont on va

caractériser la forme et décrire le contenu. Pour cela, nous avons utilisé l'outil **SegmenTool** développé à la direction de la recherche de l'INA. Cet outil permet de découper temporellement des émissions et produit un squelette de description MPEG-7 (figure 5.5).

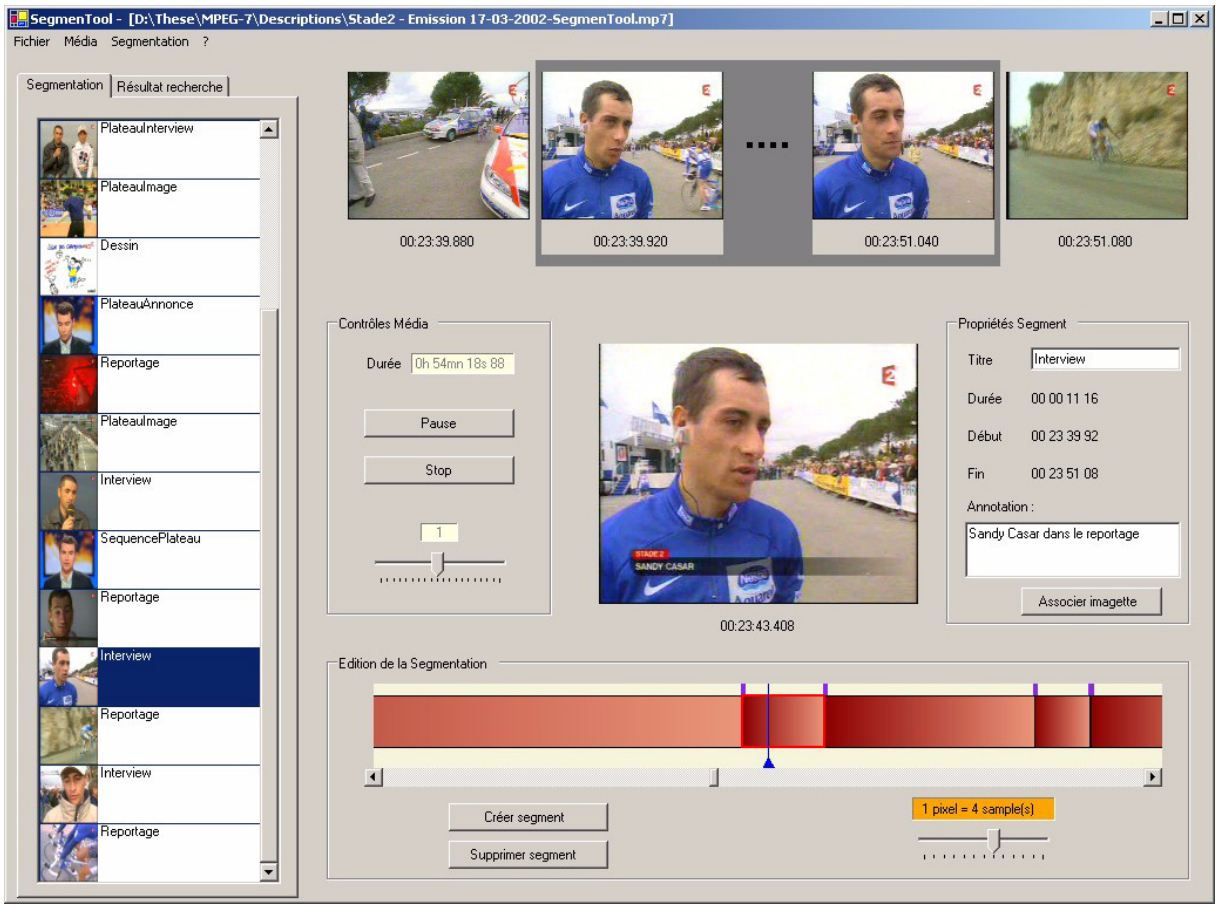


FIG. 5.5 – *Découpage temporel des documents audiovisuels avec l'outil* SegmenTool

Nous modifions ensuite ce squelette afin de typer chacun des segments en fonction de son genre et conformément à l'extension du langage MPEG-7 que nous avons proposé. La description doit alors être « schéma valide » vis-à-vis du schéma de description dont elle est l'instance. Elle reflète ainsi la structure logique de l'émission. Des thématiques générales peuvent également être ajoutées pour caractériser plus en avant les séquences. Le code 5.4 montre le code XML correspondant à un extrait de la description de l'émission *Stade 2* diffusée le 17 mars 2002. Cette description est valide vis-à-vis du schéma définissant la structure logique des *magazines sportifs*. Dans cet extrait, on a localisé temporellement les `interview` de *Jean-Marie Leblanc*, *Didier Rous* et *Sandy Casar* dans un `reportage` de l'émission, ce qui correspond à la séquence numérotée 12 dans le tableau 4.1.

Chaque séquence est maintenant *time-codée* et caractérisée en termes de genre audiovisuel et éventuellement d'une thématique générale dans la description. Comme les descripteurs utilisés ont leur correspondance dans l'ontologie de l'audiovisuel, nous pouvons générer des instances des concepts de cette ontologie. Ainsi, le code 5.5 donne un exemple d'assertion (on dit encore *fait*) qu'il est possible de construire automatiquement à partir de la description donnée dans le code

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<Mpeg7 xmlns="urn:mpeg:mpeg7:schema:2001"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xmlns:mpeg7="urn:mpeg:mpeg7:schema:2001"
       xmlns:ina="http://www.ina.fr/schema/"
       xsi:schemaLocation="urn:mpeg:mpeg7:schema:2001 mpeg7-2001-valid.xsd
                           http://www.ina.fr/schema/ SchemaGenerique.xsd">

 <Description xsi:type="ContentEntityType">
  <MultimediaContent xsi:type="ina:TVType"
                     nom="Stade2" dateDiffusion="2002-03-17"
                     heureDiffusion="18:50:00" chaineDiffusion="France2">
   <ina:Emission xsi:type="ina:MagazineCompositeType">
    <ina:MonMediaLocator>
     <MediaUri>file://E:/Raphael/These/VideosNumeriques/Stade2_17-03-2002.mpg
     </MediaUri>
    </ina:MonMediaLocator>
    <ina:MagazineCompositeDecompositionTemporelle
        xsi:type="ina:MagazineCompositeDecompositionType"
        gap="false" overlap="false">
     <ina:Reportage>
      <ina:Annotation>Reportage, Thème Cyclisme. Course Paris-Nice
      </ina:Annotation>
      <ina:Thematique>Cyclisme</ina:Thematique>
      <ina:MediaTime>
       <MediaTimePoint>T00:21:34:6773760F14112000</MediaTimePoint>
       <MediaDuration>PT00H03M12S5644800N14112000F</MediaDuration>
      </ina:MediaTime>
      <ina:ReportageDecompositionTemporelle
          xsi:type="ina:ReportageDecompositionType"
          gap="true" overlap="false">
       <ina:Interview>
        <ina:Annotation>Interview de Jean-Marie Leblanc, directeur du Tour de
        France</ina:Annotation>
        <ina:MediaTime>
         <MediaTimePoint>T00:22:01:11854080F14112000</MediaTimePoint>
         <MediaDuration>PT00H00M15S10160640N14112000F</MediaDuration>
        </ina:MediaTime>
        <ina:Interviewe>Jean-Marie Leblanc</ina:Interviewe>
       </ina:Interview>
       <ina:Interview>
        <ina:Annotation>Interview de Didier Rous, champion de France
        </ina:Annotation>
        <ina:MediaTime>
         <MediaTimePoint>T00:22:35:564480F14112000</MediaTimePoint>
         <MediaDuration>PT00H00M16S1693440N14112000F</MediaDuration>
        </ina:MediaTime>
        <ina:Interviewe>Didier Rousc</ina:Interviewe>
       </ina:Interview>
```

Code 5.4 – *Description instance d'une séquence particulière en MPEG-7 étendu*

```
      <ina:Interview>
       <ina:Annotation>Interview de Sandy Casar avant l'étape</ina:Annotation>
       <ina:MediaTime>
        <MediaTimePoint>T00:23:39:12983040F14112000</MediaTimePoint>
        <MediaDuration>PT00H00M11S2257920N14112000F</MediaDuration>
       </ina:MediaTime>
       <ina:Interviewe>Sandy Casar</ina:Interviewe>
      </ina:Interview>
      <ina:Interview>
       <ina:Annotation>Interview de Sandy Casar après l'étape</ina:Annotation>
       <ina:MediaTime>
        <MediaTimePoint>T00:24:19:10725120F14112000</MediaTimePoint>
        <MediaDuration>PT00H00M07S9031680N14112000F</MediaDuration>
       </ina:MediaTime>
       <ina:Interviewe>Sandy Casar</ina:Interviewe>
      </ina:Interview>
     </ina:ReportageDecompositionTemporelle>
     <ina:Lieux>Nice</ina:Lieux>
    </ina:Reportage>
   </ina:MagazineCompositeDecompositionTemporelle>
  </ina:Emission>
  <ina:Thematique>Multisports</ina:Thematique>
 </MultimediaContent>
</Description>
</Mpeg7>
```

Code 5.4 – *Description instance d'une séquence particulière en MPEG-7 étendu* **(suite)**

5.4. Cet exemple indique qu'il existe une instance du concept `Interview` et donne une valeur aux différentes propriétés attachées au concept.

```
<Interview rdf:ID="interview4">
  <rdfs:comment>Interview de Sandy Casar après l'étape</rdfs:comment>
  <hasStartTime rdf:datatype="&xsd;string">T00:24:19</hasStartTime>
  <hasDuration rdf:datatype="&xsd;string">PT00H00M07S9</hasDuration>
  <hasThematique rdf:resource="#Cyclisme"/>
  ...
</Interview>
```

CODE 5.5 – *Exemple de triplets RDF/XML construits automatiquement à partir de la description*

L'ensemble des assertions a une traduction immédiate en triplets RDF qui viennent alimenter une base de connaissances sur laquelle on pourra effectuer des inférences. Ainsi, la séquence `interview4` répond à la requête « *trouver les séquences de type* `SequenceDialogue` *qui ont comme thématique le* `Sport` » puisque dans l'ontologie de l'audiovisuel, `SequenceDialogue` subsume `Interview` et `Sport` subsume `Cyclisme`.

Dans la section suivante, nous présentons quelques expérimentations menées autour des schémas de description que nous avons produis. En particulier, tout testons la validité de ces schémas.

5.3 Expérimentations

Nous avons montré dans ce chapitre comment nous avons construit une ontologie de l'audiovisuel et comment nous en avons traduit une partie en types XML Schema. La combinaison de ces types nous a permis de définir des schémas de descriptions qui valident la structure logique de collection d'émissions. Cette dernière section présente quelques expérimentations réalisées autour de ces schémas de descriptions. Nous étudions en particulier les problèmes posés par la dérivation de types en XML Schema (section 5.3.1) avant de se pencher sur le problème de la validité (au sens XML Schema) des descriptions MPEG-7 (section 5.3.2).

5.3.1 La dérivation de types en XML Schema

Nous avons vu dans la section 2.1.2 (page 33) les deux manières de dériver des types (*restriction* et *extension*) dans le langage XML Schema. Nous avons également constaté que cette dérivation posait quelques problèmes et qu'elle n'était pas équivalente au mécanisme d'héritage que l'on trouve dans les langages de programmation par objets, ni à celui de la représentation de connaissances à objets. Pour illustrer ces problèmes, prenons l'exemple suivant : une `Emission` peut être représentée de manière très simple par la structure suivante : elle est identifiée par deux attributs qui sont le nom de l'émission et la chaîne où elle a été diffusée, et elle contient un ensemble (éventuellement vide) de séquences. Une `Sequence` possède aussi deux attributs pour indiquer les positions temporelles de début et de fin de la séquence. Imaginons maintenant que l'on souhaite définir d'une part une `SequencePlateau` comme étant une spécialisation d'une `Sequence` qui possède en plus un nom de présentateur, et d'autre part un `MagazinePlateau` comme étant une spécialisation d'une émission dans la mesure où les seules séquences autorisées sont des `SequencePlateau`. Cette structure peut être schématisée comme représenté dans la figure 5.6.

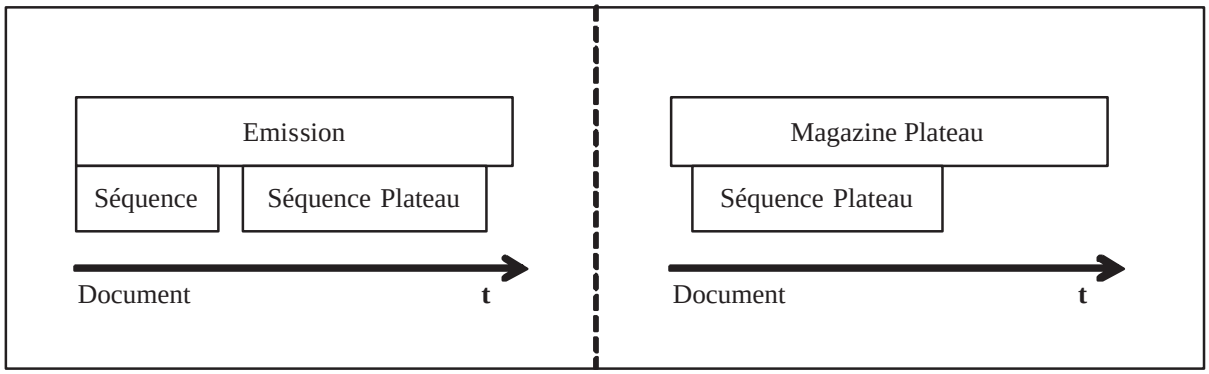


FIG. 5.6 – *Décomposition possible d'une* `Emission` *et d'un* `MagazinePlateau`

Pour représenter cette situation dans le langage XML Schema, il faudrait étendre le type `Sequence` en lui ajoutant un élément contenant le nom d'un présentateur pour avoir une `SequencePlateau`. Par contre, il faudrait restreindre le type `Emission` pour avoir un `MagazinePlateau` puisque les séquences admises doivent être uniquement des `SequencePlateau` et non pas n'importe quel type de `Sequence`. Le problème est qu'il est impossible de restreindre directement l'`Emission` pour qu'elle ne contienne que des `SequencePlateau`. En effet, ce dernier type est dérivé par extension de `Sequence`, il est donc moins spécifique (au sens des schémas) que celui-ci et ne peut être utilisé pour le remplacer dans une dérivation par restriction.

Une possibilité pour contourner ce problème consiste à passer d'abord par un type intermédiaire qui restreint l'`Emission` en ne conservant que ses attributs, puis à étendre ce type intermédiaire en lui ajoutant un ensemble de `SequencePlateau` [Brunie and Troncy, 2003]. On utilise ainsi le mécanisme de restriction pour « couper » ce que nous ne voulons pas dans les descriptions, puis le mécanisme d'extension pour ajouter ce que nous voulons en plus. Le schéma résultat est donné dans le code 5.6.

Il est à noter que cette astuce n'est valable que grâce à la définition extrêmement souple du type `Emission` : celui-ci contient en effet un ensemble *optionnel* de séquences. Ainsi, la restriction de ce type en ne gardant quasiment rien est possible.

La structure d'une `Emission` peut alors être composée de `Sequence` et de `SequencePlateau` (voir le code 5.7), alors que la structure d'un `MagazinePlateau` contiendra obligatoirement des `SequencePlateau` (voir le code 5.8).

5.3.2 Expérimentations avec le langage MPEG-7

Après avoir clarifié les ambiguïtés portées par la dérivation de types en XML Schema, penchons nous maintenant sur la validité XML des schémas MPEG-7. Tout d'abord, pour pouvoir utiliser ce langage, il est nécessaire de collecter les éléments de schémas qui sont publiés dans la version définitive de la norme. En effet, pendant sa phase d'élaboration, des schémas provisoires définissant le langage ont circulé sur le Web, mais ceux-ci présentent quelques différences avec la version publiée du standard, et il est donc sage de ne pas les utiliser.

Délibérément, la norme MPEG-7 n'inclue donc pas de schémas *finaux*, et nous avons d'ailleurs trouvé un certain nombre d'erreurs attestant le fait que personne n'a jamais essayé de les valider. Selon nous, la justification de cette décision est que le langage MPEG-7 n'est pas vraiment utilisable en tant que tel, mais qu'il doit servir de base à la construction de sous-ensembles plus

```
<?xml version="1.0" encoding="iso-8859-1"?>

<schema xmlns="http://www.w3.org/2001/XMLSchema"
        xmlns:ina="http://www.ina.fr/schema/"
        targetNamespace="http://www.ina.fr/schema/"
        elementFormDefault="qualified" attributeFormDefault="unqualified">

  <element name="Emission" type="ina:EmissionType"/>

  <complexType name="EmissionType">
    <sequence minOccurs="0" maxOccurs="unbounded">
      <element name="Sequence" type="ina:SequenceType"/>
    </sequence>
    <attribute name="nom" type="string" use="required"/>
    <attribute name="chaineDiffusion" type="string" use="required"/>
  </complexType>

  <complexType name="SequenceType">
    <attribute name="tcDebut" type="string" use="required"/>
    <attribute name="tcFin" type="string" use="required"/>
  </complexType>

  <complexType name="SequencePlateauType">
    <complexContent>
      <extension base="ina:SequenceType">
        <sequence>
          <element name="Presentateur" type="string"/>
        </sequence>
      </extension>
    </complexContent>
  </complexType>

  <complexType name="intern_MagazinePlateauType">
    <complexContent>
      <restriction base="ina:EmissionType">
        <attribute name="nom" type="string" use="required"/>
        <attribute name="chaineDiffusion" type="string" use="required"/>
      </restriction>
    </complexContent>
  </complexType>

  <complexType name="MagazinePlateauType">
    <complexContent>
      <extension base="ina:intern_MagazinePlateauType">
        <sequence minOccurs="0" maxOccurs="unbounded">
          <element name="Sequence" type="ina:SequencePlateauType"/>
        </sequence>
      </extension>
    </complexContent>
  </complexType>
</schema>
```

CODE 5.6 – *Structure possible d'une* `Emission` *et d'un* `MagazinePlateau` *en XML Schema*

```
<?xml version="1.0" encoding="iso-8859-1"?>

<ina:Emission xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
              xmlns:ina="http://www.ina.fr/schema/"
              xsi:schemaLocation="http://www.ina.fr/schema/ emissionPlateau.xsd"
              nom="Stade2" chaineDiffusion="France2">

  <ina:Sequence tcDebut="00H01M32S24F" tcFin="00H19M07S17F"/>
  <ina:Sequence xsi:type="ina:SequencePlateauType"
                tcDebut="00H19M07S17F" tcFin="00H24M46S4F">
    <ina:Presentateur>Gérard Holtz</ina:Presentateur>
  </ina:Sequence>
</ina:Emission>
```

CODE 5.7 – *Description valide d'une* `Emission`

```
<?xml version="1.0" encoding="iso-8859-1"?>

<ina:Emission xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
              xmlns:ina="http://www.ina.fr/schema/"
              xsi:schemaLocation="http://www.ina.fr/schema/ emissionPlateau.xsd"
              nom="ToutLeMondeEnParle" chaineDiffusion="France2"
              xsi:type="ina:MagazinePlateauType">

  <ina:Sequence tcDebut="00H01M32S24F" tcFin="00H19M07S17F">
    <ina:Presentateur>Thierry Ardisson</ina:Presentateur>
  </ina:Sequence>
</ina:Emission>
```

CODE 5.8 – *Description valide d'un* `MagazinePlateau`

spécifiques et dépendants d'une application identifiée, comme le fait le consortium TV Anytime (voir la section 2.3.3, page 51).

Parmi les erreurs que nous constaté et dû corriger, la plupart sont d'ordre typographiques [Brunie and Troncy, 2003] :

- le type `RegionLocatorType` ne peut pas être défini comme *final* (`final="#all"`) puisque il est ensuite dérivé pour produire le type `InkMediaInformationType` ;
- le type `InkMediaInformationType` possède deux attributs de type entier, et leur valeur par défaut ne peuvent donc pas être un réel ;
- le type `mimeType` contient une erreur dans sa définition et l'expression régulière doit être remplacée par : `<pattern value="[a-z]+/[_a-zA-Z0-9\-\.$]+"/>` ;
- le type `GaussianMixtureModelType` est défini comme une séquence d'éléments, mais le nombre maximal d'éléments (`maxOccurs`) est défini deux fois ; le premier est donc le nombre minimal (`minOccurs`).

Un autre cas d'erreur est lié à la définition de types en utilisant une grammaire ambiguë (les quatre types MPEG-7 suivants sont concernés : `EditedVideoEditingTemporalDecomposition-Type`, `ShotEditingTemporalDecompositionType`, `CompositionShotEditingTemporalDecompositionType` et `IntraCompositionShotEditingTemporalDecompositionType`). Or, cette pratique est explicitement interdite par la recommandation XML Schema [XML Schema, 2001] et la contrainte nommée UPA (*Unique Particle Attribution*).

Le code donné dans le tableau 5.9 illustre cette situation pour le type `EditedVideoEditing-TemporalDecompositionType`.

Les quatre types incriminés ont un modèle de contenu de la forme schématisée dans la figure 5.7. Par exemple, pour `EditedVideoEditingTemporalDecompositionType`, les éléments **S** correspondent à un choix entre un élément nommé `Shot` et un élément nommé `ShotRef`, alors que les éléments **T** correspondent eux à un choix entre `GlobalTransition` et `GlobalTransitionRef`.

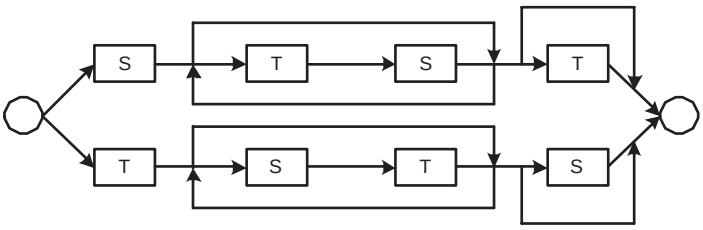


FIG. 5.7 – *Modèle de contenu ambigu dans MPEG-7*

Les phases générées par cette grammaire sont une alternance stricte de **T** et de **S**, commençant et finissant par l'un ou l'autre.

Pour désambiguïser cette grammaire, nous utilisons les résultats obtenus autour des langages rationnels et nous commençons par traduire le modèle de contenu de la figure 5.7 en un automate d'états finis qui sera plus facile à déterminiser [Benzaken, 1991].

Définition 1 *On appelle* automate fini *la donnée d'un quintuplet* $\mathcal{A} = (V, Q, I, T, F)$ *où* V *est l'alphabet (fini),* Q *est un ensemble fini dont les éléments s'appellent les* états *de l'automate et où :* $I \subseteq Q$ *(et* $T \subseteq Q$*) désignent l'ensemble des* états initiaux *(*terminaux*) de* $\mathcal{A}$*.* $F \subseteq Q^*V^*Q$ *désigne l'ensemble des* flèches *(ou* transitions*) de* $\mathcal{A}$*.*

```
<complexType name="EditedVideoEditingTemporalDecompositionType">
  <complexContent>
   <extension base="mpeg7:TemporalSegmentDecompositionType">
    <sequence>
     <choice>
      <sequence>
       <choice>
        <element name="Shot" type="mpeg7:ShotType"/>
        <element name="ShotRef" type="mpeg7:ReferenceType"/>
       </choice>
       <sequence minOccurs="0" maxOccurs="unbounded">
        <choice>
         <element name="GlobalTransition" type="mpeg7:GlobalTransitionType"/>
         <element name="GlobalTransitionRef" type="mpeg7:ReferenceType"/>
        </choice>
        <choice>
         <element name="Shot" type="mpeg7:ShotType"/>
         <element name="ShotRef" type="mpeg7:ReferenceType"/>
        </choice>
       </sequence>
       <choice minOccurs="0">
        <element name="GlobalTransition" type="mpeg7:GlobalTransitionType"/>
        <element name="GlobalTransitionRef" type="mpeg7:ReferenceType"/>
       </choice>
      </sequence>
      <sequence>
       <choice>
        <element name="GlobalTransition" type="mpeg7:GlobalTransitionType"/>
        <element name="GlobalTransitionRef" type="mpeg7:ReferenceType"/>
       </choice>
       <sequence minOccurs="0" maxOccurs="unbounded">
        <choice>
         <element name="Shot" type="mpeg7:ShotType"/>
         <element name="ShotRef" type="mpeg7:ReferenceType"/>
        </choice>
        <choice>
         <element name="GlobalTransition" type="mpeg7:GlobalTransitionType"/>
         <element name="GlobalTransitionRef" type="mpeg7:ReferenceType"/>
        </choice>
       </sequence>
       <choice minOccurs="0">
        <element name="Shot" type="mpeg7:ShotType"/>
        <element name="ShotRef" type="mpeg7:ReferenceType"/>
       </choice>
      </sequence>
     </choice>
    </sequence>
   </extension>
  </complexContent>
</complexType>
```

CODE 5.9 – *Définition ambiguë du type* `EditedVideoEditingTemporalDecompositionType` *dans MPEG-7*

Définition 2 *Un automate fini est dit* déterministe *lorsqu'il satisfait aux deux conditions :*

- $(q,\ a,\ q')$ *et* $(q,\ a,\ q'') \in F \Rightarrow q' = q''\ \ (\forall q,\ q',\ q'' \in Q$ *et* $\forall a \in V)$ *;*
- *l'ensemble I n'a qu'un état.*

Définition 3 *Un automate fini est dit* complet *lorsque :*

- $\forall q \in Q,\ \ \forall a \in V,\ \ \exists f \in F$ *tel que* $f = (q,\ a,\ q')$

On peut représenter un automate par un graphe orienté étiqueté. Chaque état est représenté par un sommet (dessiné par un rond) et chaque flèche (q, a, q') se représente par un arc (orienté) d'étiquette *a* joignant q à q'. Les sommets correspondant aux états initiaux (resp. terminaux) seront assortis d'une demi-flèche d'entrée (resp. de sortie). Ainsi, le modèle de contenu représenté dans la figure 5.7 peut se représenter par un automate dont le graphe est donné dans la figure 5.8 et dont la représentation formelle est : $\mathcal{A}$ = (V={s,t}, Q={0,1,2,3,4,5,6,7,8,9}, I={0,5}, T={4,9}, F={0s1, 0s4, 1t2, 1t4, 2s1, 2s3, 2s4, 3t4, 5t6, 5t9, 6s7, 6s9, 7t6, 7t8, 7t9, 8s9}).

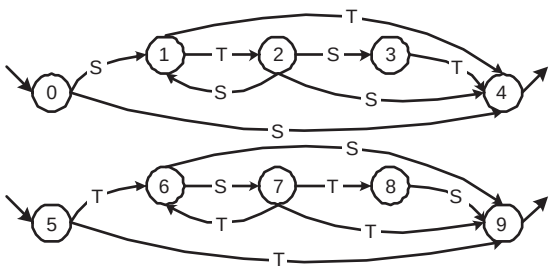


FIG. 5.8 – *Automate décrivant le modèle de contenu donné dans la figure 5.7*

Le théorème de Kleene [Kleene, 1971] nous permet d'affirmer que le langage L reconnu par cet automate fini est également reconnu par une grammaire hors contexte (ici droite) élargie et par une expression régulière. Déterminisons tout d'abord cet automate. Soit $\mathcal{A}$ = (V, Q, I, T, F) un automate quelconque. Pour chaque partie S de Q, et pour chaque a $\in$ V, on peut définir la partie S.a de Q de la manière suivante :

S.a = q | $\exists$ s $\in$ S tel que (s, a, q) $\in$ F ; en particulier $\emptyset$.a = $\emptyset$.

Considérons alors l'automate $\mathcal{PA}$ = (V, $\mathcal{Q}$, I, $\mathcal{T}$, $\mathcal{F}$) où :

- $\mathcal{Q}$ est l'ensemble de toutes les parties de Q ;
- $\mathcal{T}$ est l'ensemble de toutes les parties S de Q telles que S $\cap$ T $\neq \emptyset$;
- $\mathcal{F}$ est l'ensemble de toutes les transitions (S, a, S.a).

Théorème 1 *L'automate* $\mathcal{PA}$ *est déterministe complet et est équivalent à* $\mathcal{A}$

Ainsi, l'automate précédent (figure 5.8) est équivalent à l'automate fini, déterministe, complet et minimal donné dans la figure 5.9.

Le langage L reconnu par cet automate est finalement équivalent à celui reconnu par l'expression régulière suivante :

$$L = [S.\ (T.(ST)^* + (TS)^*)] + [T.\ (S.(TS)^* + (ST)^*]$$

```
<complexType name="EditedVideoEditingTemporalDecompositionType">
  <complexContent>
   <extension base="mpeg7:TemporalSegmentDecompositionType">
    <choice>
     <sequence>
      <choice>
       <element name="Shot" type="mpeg7:ShotType"/>
       <element name="ShotRef" type="mpeg7:ReferenceType"/>
      </choice>
      <choice>
       <sequence>
        <choice>
         <element name="GlobalTransition" type="mpeg7:GlobalTransitionType"/>
         <element name="GlobalTransitionRef" type="mpeg7:ReferenceType"/>
        </choice>
        <sequence minOccurs="0" maxOccurs="unbounded">
         <choice>
          <element name="Shot" type="mpeg7:ShotType"/>
          <element name="ShotRef" type="mpeg7:ReferenceType"/>
         </choice>
         <choice>
          <element name="GlobalTransition" type="mpeg7:GlobalTransitionType"/>
          <element name="GlobalTransitionRef" type="mpeg7:ReferenceType"/>
         </choice>
        </sequence>
       </sequence>
       <sequence minOccurs="0" maxOccurs="unbounded">
        <choice>
         <element name="GlobalTransition" type="mpeg7:GlobalTransitionType"/>
         <element name="GlobalTransitionRef" type="mpeg7:ReferenceType"/>
        </choice>
        <choice>
         <element name="Shot" type="mpeg7:ShotType"/>
         <element name="ShotRef" type="mpeg7:ReferenceType"/>
        </choice>
       </sequence>
      </choice>
     </sequence>
     <sequence>
      <choice>
       <element name="GlobalTransition" type="mpeg7:GlobalTransitionType"/>
       <element name="GlobalTransitionRef" type="mpeg7:ReferenceType"/>
      </choice>
      <choice>
       <sequence>
        <choice>
         <element name="Shot" type="mpeg7:ShotType"/>
         <element name="ShotRef" type="mpeg7:ReferenceType"/>
        </choice>
```

Code 5.10 – *Définition* corrigée *du type MPEG-7* `EditedVideoEditingTemporalDecompositionType`

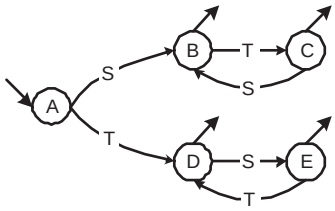


FIG. 5.9 – *Automate déterminisé décrivant le modèle de contenu donné dans la figure 5.7*

```
<sequence minOccurs="0" maxOccurs="unbounded">
         <choice>
          <element name="GlobalTransition" type="mpeg7:GlobalTransitionType"/>
          <element name="GlobalTransitionRef" type="mpeg7:ReferenceType"/>
         </choice>
         <choice>
          <element name="Shot" type="mpeg7:ShotType"/>
          <element name="ShotRef" type="mpeg7:ReferenceType"/>
         </choice>
        </sequence>
       </sequence>
       <sequence minOccurs="0" maxOccurs="unbounded">
        <choice>
         <element name="Shot" type="mpeg7:ShotType"/>
         <element name="ShotRef" type="mpeg7:ReferenceType"/>
        </choice>
        <choice>
         <element name="GlobalTransition" type="mpeg7:GlobalTransitionType"/>
         <element name="GlobalTransitionRef" type="mpeg7:ReferenceType"/>
        </choice>
       </sequence>
      </choice>
     </sequence>
    </choice>
   </extension>
  </complexContent>
</complexType>
```

CODE 5.10 – *Définition* corrigée *du type MPEG-7* `EditedVideoEditingTemporalDecompositionType` **(suite)**

Il est maintenant plus aisé, à partir de cette expression régulière, de retraduire en XML Schema la définition du type `EditedVideoEditingTemporalDecompositionType` (code 5.10).

La correction des erreurs typographiques listées ci-dessus et la désambiguïsation de la grammaire définissant les quatre types incriminés permet d'obtenir un schéma MPEG-7 valide vis-à-vis de la norme XML Schema [XML Schema, 2001].

5.4 Conclusion

Dans ce chapitre, nous avons détaillé le contenu de l'ontologie de l'audiovisuel. Celle-ci permet de fixer formellement la sémantique des descripteurs utilisés pour exprimer la structure des documents audiovisuels. Nous avons alors montré comment nous transformons une partie de cette ontologie (les genres et les thèmes) pour créer des types XML Schema, qui, une fois combinés, permettent de créer des schémas de description qui vont contrôler la structure logique des documents. Nous avons donné un exemple d'un tel schéma pour les magazines sportifs dits composites, et un exemple de description pour une émission particulière instanciant ce schéma. Nous avons finalement mené des expérimentations autour de ces schémas afin de tester leur validité XML et nous avons vu comment nous avons dû modifier les schémas MPEG-7 « officiels » pour qu'ils soient opérationnels. Moyennant les modifications présentées, ces derniers peuvent être validés avec des outils informatiques et donc intégrés dans une application.

Il reste maintenant à compléter la description de la structure avec la description du contenu des séquences audiovisuelles. Nous avons choisi un domaine d'application particulier, le *cyclisme*, et nous présentons nos réalisations et nos expérimentations avec ce domaine dans le chapitre suivant.

Chapitre 6

Ontologie de domaine : application au domaine du cyclisme

Sommaire

6.1 Construction d'une ontologie du cyclisme 150

6.1.1 Conceptualisation 150

6.1.2 Formalisation et opérationnalisation 153

6.1.3 L'ontologie de haut niveau 154

6.2 Génération d'une base de faits 156

6.3 Expérimentations 160

6.3.1 Expérimentations avec le système *SeBOR* 161

6.3.2 Expérimentations avec le système *Racer* 163

6.3.3 Visionner les séquences audiovisuelles 164

6.4 Conclusion 166

Après avoir détaillé dans le chapitre précédent la construction de l'ontologie de l'audiovisuel qui permet d'exprimer la structure des documents audiovisuels, nous présentons dans ce chapitre la construction d'une ontologie de domaine qui permet de décrire le contenu de ces documents. Comme il fallait choisir un domaine de connaissances, nous avons opté pour un sport particulier, le *cyclisme*. La modélisation d'une ontologie du cyclisme doit nous permettre de décrire les différentes séquences audiovisuelles consacrées à ce sport, qu'elles soient diffusées dans le cadre de journaux télévisés ou de magazines sportifs. Mais elle permettra surtout de décrire les longues retransmissions du *Tour de France* qui est un événement médiatique majeur dans le monde juste après les *Jeux Olympiques* et la *Coupe du Monde de Football*. Il y a donc potentiellement de nombreuses séquences audiovisuelles à décrire et susceptibles d'intéresser le plus grand nombre.

Le choix de ce domaine présente quelques avantages. Le cyclisme utilise un vocabulaire spécialisé au contour bien défini, ce qui a rendu la modélisation assez aisée sans que nous soyons pour autant un « expert », mais juste un téléspectateur assidu. Le *Tour de France* est une manifestation annuelle et l'ontologie pourra donc être ré-utilisée chaque année moyennant quelques très légères modifications. C'est également un événement très médiatique et de nombreuses ressources textuelles le couvrant sont disponibles, ce qui est une aide précieuse au moment de la construction de l'ontologie et de la base de faits. Enfin, nous remarquons que cette épreuve est très liée à l'histoire de la télévision puisque le tout premier journal télévisé, diffusé en juillet 1949

et présenté par Pierre Sabbagh, débutait par l'ouverture du prologue du *Tour de France* de cette année.

Nous allons donc voir comment nous avons construit l'ontologie du cyclisme (section 6.1). Nous montrons ensuite comment cette ontologie sert de ressource principale au système **Seigo** [Le Roux, 2003] qui permet de construire semi-automatiquement une base de faits formalisés, candidats à la description du contenu des séquences audiovisuelles (section 6.2). La suite du chapitre est alors consacrée à des expérimentations dans des systèmes à base de connaissances implémentés (section 6.3). Nous montrons que nous sommes capables de répondre à des requêtes mêlant structure et contenu grâce à l'architecture générale présentée dans le chapitre 4.

6.1 Construction d'une ontologie du cyclisme

Tout comme l'ontologie de l'audiovisuel, l'ontologie du cyclisme a été construite selon la méthodologie proposée par B. Bachimont, décrite dans la section 3.4.2, page 88. Nous décrivons donc dans la suite comment l'ontologie a été conceptualisée (section 6.1.1) puis formalisée (section 6.1.2). Nous concluons cette section par quelques réflexions menées autour du haut niveau des deux ontologies que nous avons modélisé (section 6.1.3).

6.1.1 Conceptualisation

Contrairement à l'ontologie de l'audiovisuel qui avait été construite à partir d'une ressource unique (un manuel d'indexation utilisé par des documentalistes de l'audiovisuel), l'ontologie du cyclisme a été modélisée à partir d'un corpus textuel de plus de 550 000 mots que nous avons constitué pour l'occasion. Ce corpus est composé d'environ 1200 articles issus de la presse généraliste (*Le Monde*, *Libération*, *Le Parisien*), de la presse spécialisée (*L'Équipe*) ou encore de dépêches d'agences de presse (*AFP*) qui ont couvert le Tour de France 1999 [Troncy and Isaac, 2002; Le Roux, 2003]. Il est plutôt homogène, tant sur le plan stylistique que sur les différents thèmes abordés (descriptions géographique, vie de la course, classements, dopage...).

Nous avons ensuite utilisé sur ce corpus des procédés d'extraction terminologique afin de trouver des « candidats-termes » pour devenir des concepts de l'ontologie. La construction de cette ontologie ayant débuté il y a deux ans, l'outil **Lexter** [Bourigault, 1994] avait été utilisé à l'époque mais il gagnerait aujourd'hui a être remplacé par les outils **Syntex**, **Upery** et **Yakwa** (voir section 3.4.1, page 86). À titre d'exemple, l'analyse sur une partie de ce corpus a fait ressortir un certain nombre de termes qualifiant une personne humaine (tableau 6.1), mais qui ne jouent manifestement pas tous le même rôle dans une épreuve cycliste.

coureur cycliste - spectateur - directeur sportif - médecin équipe - journaliste - commissaire de course - grimpeur - rouleur - sprinteur

Tab. 6.1 – *Quelques termes désignant une personne*

Après avoir isolé un certain nombre de libellés grâce à ce procédé, il faut maintenant veiller à leurs donner une signification précise, ce qui est précisément l'objectif de la phase de conceptualisation dans notre méthodologie. Comme nous l'avons déjà signalé, la méthodologie repose sur la sémantique différentielle présentée par François Rastier dans [Rastier *et al.*, 1994]. Dans ce paradigme, le sens est intra-linguistique : il se construit par des relations d'opposition entre les unités du système linguistique. En fait, il s'agit d'attribuer un sens aux termes grâce à des

traits sémantiques ou *sèmes*. Ces sèmes que l'on va associer à une unité sont regroupables en deux catégories :

- les *sèmes génériques* qui permettent de regrouper les unités entre elles ;
- les *sèmes spécifiques* qui permettent à une unité de se distinguer de celles avec qui on l'a regroupée.

Par exemple, dans le domaine du cyclisme, on va pouvoir associer le sème « *monte bien les côtes dures et longues en montagne* » au signifiant *grimpeur*, et le sème « *endurant et rapide dont les qualités se révèlent surtout dans les courses sur terrain plat* » au signifiant *rouleur*. Ces deux sèmes permettent bien de distinguer ces deux unités l'une de l'autre. En revanche, elles vont partager le même sème « *homme dont le métier est de participer aux courses cyclistes* » – associé au signifiant *coureur cycliste*. Ces deux types de sèmes permettent donc de définir une unité par les identités et les différences qu'elle entretient avec ces plus proches voisins. C'est cette position dans le système qui donne un sens à l'unité considérée.

La définition des quatre principes différentiels permet alors d'expliciter la signification de chacun des concepts. Pour reprendre l'exemple donné dans le tableau 6.1, on s'aperçoit que les termes `Grimpeur`, `Rouleur` et `Sprinter` renvoient à un `Coureur Cycliste`. De plus, toutes les personnes habituellement présentes sur le Tour de France n'ayant pas le même statut, on peut les regrouper selon les raisons qui les ont poussées à y participer. Ainsi, la notion `Personne` peut se spécialiser en trois nouvelles notions – `Personnel Épreuve`, `Personnel Équipe` et `Spectateur` – selon les principes différentiels présentés dans le tableau 6.2. Le sous-arbre complet de la notion `Personne` est donné dans la figure 6.1.

⟶ `Personnel Épreuve`	
SWP :	c'est un individu humain
SWS :	une propriété précise la raison de la personne sur les lieux de l'épreuve
DWS :	est accrédité par la direction de l'épreuve
DWP :	joue un rôle particulier par rapport à l'épreuve : la personne est accréditée par la direction de celle-ci
⟶ `Personnel Équipe`	
SWP :	c'est un individu humain
SWS :	une propriété précise la raison de la personne sur les lieux de l'épreuve
DWS :	est employé par une des équipes participantes
DWP :	joue un rôle particulier par rapport à l'épreuve : la personne est employée par une équipe participante à l'épreuve
⟶ `Spectateur`	
SWP :	c'est un individu humain
SWS :	une propriété précise la raison de la personne sur les lieux de l'épreuve
DWS :	n'est ni accrédité par la direction de l'épreuve, ni employé par une des équipes participantes
DWP :	joue un rôle particulier par rapport à l'épreuve : se contente d'y assister

TAB. 6.2 – *Principes différentiels associés aux spécialisations directes de* Personne

Outre les différents acteurs apparaissant dans les courses cyclistes, nous avons également modélisé la notion de course cycliste, selon qu'elle se déroule en plusieurs étapes ou pas et selon les caractéristiques des étapes. Une partie de l'ontologie concerne également les différents événements susceptibles de se produire pendant la course, qu'il soient programmés, tels que les

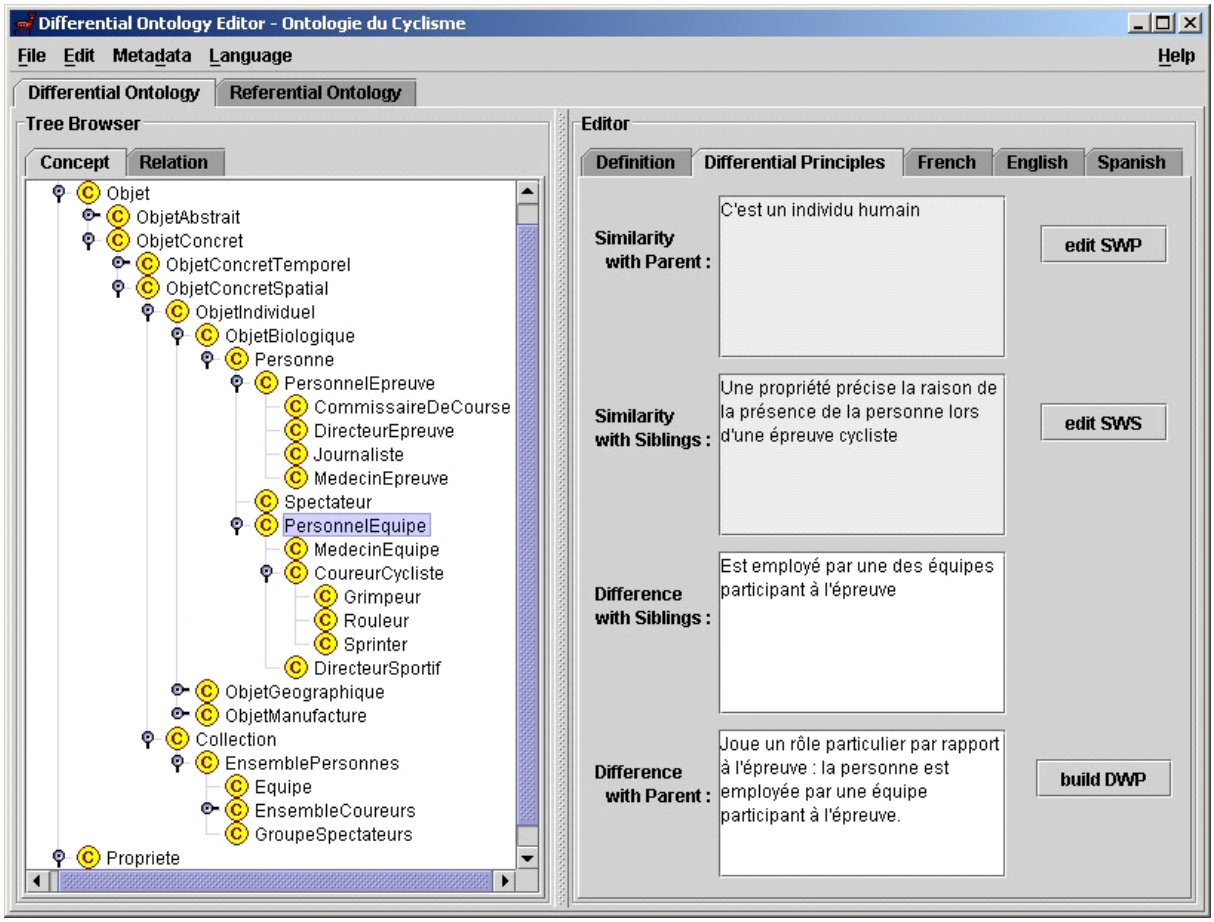


FIG. 6.1 – *Les principes différentiels de la notion* Personnel Équipe *dans* DOE

sprints ou les *ascensions de cols*, ou imprévus, tels que les *échappées*, les *chutes*. La taxinomie obtenue (environ 90 concepts) et les axes différentiels choisis sont donnés dans l'annexe C.1.

L'ontologie contient également 60 relations qui ont elles aussi été structurées selon des principes différentiels. Celles-ci permettent de localiser des personnes ou des événements dans le temps et l'espace, de donner un classement provisoire d'un coureur cycliste (ou d'une équipe) pour une épreuve donnée ou de rappeler son palmarès, de caractériser les épreuves en indiquant le départ, l'arrivée et la distance kilométrique à parcourir... L'arbre des relations et les axes différentiels choisis sont donnés dans l'annexe C.2.

Nous avons une fois encore utilisé l'outil **DOE** pour mener à bien toute la conceptualisation de l'ontologie du cyclisme [Troncy and Isaac, 2002]. L'ontologie différentielle obtenue est disponible sur le Web à `http://opales.ina.fr/public/ontologies/`. La phase de conceptualisation est suivie par une phase de formalisation de l'ontologie que nous détaillons dans la section suivante.

6.1.2 Formalisation et opérationnalisation

L'arbre ontologique construit dans la première étape a permis de fixer un certain nombre de notions en normalisant leurs sens. La seconde étape de notre méthodologie consiste à passer par une sémantique extensionnelle pour lier ces notions à un ensemble de référents dans le monde. L'*ontologie référentielle* obtenue est alors composée de *concepts* qui vont agir comme des primitives formelles. Chaque concept étant lié par référence à un ensemble d'objets du domaine (son *extension*), on peut avoir recours à des opérations de composition de sens qui utilisent les opérations qui existent pour les ensembles (réunion, intersection ou complémentaire).

La comparaison des extensions permet de définir une relation d'héritage extensionnelle entre les concepts : un concept sera subsumé par un autre si et seulement si son extension est incluse dans celle de son parent. On peut alors se poser la question de la structure de la hiérarchie de subsomption obtenue. S'il paraît naturel d'affirmer que les relations d'héritage définies dans l'ontologie différentielle tiendront toujours, l'ajout de nouveaux nœuds va modifier la structure arborescente. En effet, les « transcriptions » des notions différentielles en concepts formels peuvent admettre des extensions qui ont un sous-ensemble commun. Ainsi, `Rouleur` et `Grimpeur` sont bien des notions (différentielles) qui s'excluent, mais les concepts formels correspondants ont des extensions qui peuvent avoir en commun plusieurs individus : par exemple, l'individu `Lance Armstrong` est habituellement répertorié dans ces deux catégories. On peut donc définir dans l'ontologie référentielle un concept formel `Grimpeur_Rouleur` dont la référence sera l'intersection des extensions des concepts `Grimpeur` et `Rouleur`, et dont `Lance Armstrong` sera un élément.

La formalisation de l'ontologie du cyclisme permet ainsi de définir (au sens des logiques de description) certains concepts. Par exemple, le code 6.1 donne la définition dans le langage OWL du concept `Prologue` qui est toujours, à la fois un `contre la montre individuel` et en même temps l'étape 0 d'une `course à étapes`. Nous pouvons noter que cette définition requiert un constructeur qui n'est disponible que dans les langages OWL DL et OWL Full (voir section 3.3.3).

L'ontologie formelle du cyclisme contient également la spécification des domaines des relations et la définition d'un certain nombre d'axiomes logiques. Par exemple, les concepts `Personnel Épreuve`, `Personnel Équipe` et `Spectateur` forment une partition disjointe du concept `Personne`. On exprime ainsi le fait qu'une personne est soit membre d'une équipe, soit employée par la direction d'une épreuve, soit spectateur de la course. Le code 6.2 donne la définition de cet axiome dans le langage DAML+OIL [1].

1. On peut évidemment exprimer la même assertion dans les langages OWL DL et OWL Full à l'aide des constructeurs `owl:disjointWith`, `owl:unionOf` et `rdfs:subClassOf`, mais le résultat est assez verbeux en com-

```
<owl:Class rdf:ID="Prologue">
  <rdfs:subClassOf rdf:resource="#CLMIndividuel"/>
  <owl:Restriction>
      <owl:onProperty rdf:resource="#aCommeNumero"/>
      <owl:hasValue rdf:datatype="&xsd;nonNegativeInteger">0</owl:hasValue>
  </owl:Restriction>
</owl:Class>
```

CODE 6.1 – *Définition du concept* Prologue *en OWL DL*

```
<daml:Class rdf:about="Personne">
  <daml:disjointUnionOf rdf:parseType="daml:collection">
    <daml:Class rdf:about="#PersonnelEpreuve"/>
    <daml:Class rdf:about="#PersonnelEquipe"/>
    <daml:Class rdf:about="#Spectateur"/>
  </daml:disjointUnionOf>
</daml:Class>
```

CODE 6.2 – *Définition d'une partition disjointe en DAML+OIL*

La définition des axiomes concerne finalement les relations auxquelles on doit attacher des propriétés algébriques. Par exemple, les relations de localisation (`traverse`) sont définies comme transitives alors que les relations `precedeTemporellement` et `succedeTemporellement` seront notées comme inverse l'une de l'autre.

Nous avons vu dans la section 3.4.4 que notre éditeur d'ontologies **DOE** ne prenait pas en charge tout le processus de formalisation présenté ci-dessus, mais nous avons une fois encore utilisé l'outil **OilEd** [Bechhofer *et al.*, 2001] pour mener à bien ce processus. L'interopérabilité entre ces deux outils est assuré en transformant l'ontologie modélisée dans DOE en RDFS. En revanche, notre outil s'est avéré très utile pour la modélisation du haut niveau de l'ontologie que nous détaillons dans la section suivante.

6.1.3 L'ontologie de haut niveau

Lors de la construction de l'ontologie du cyclisme, nous nous sommes heurtés à un problème de poids : comment exprimer les principes différentiels pour les notions de haut niveau de l'ontologie ? En effet, la modélisation d'un domaine fait rapidement ressortir un ensemble de concepts « majeurs », d'assez haut niveau, que l'on a spécialisés pour former des sous-arbres dans la taxinomie. Ainsi, dans le cyclisme, émergent des concepts tels que `Personne`, `Épreuve` ou encore `Incident`. Mais, d'après la méthodologie exposée ci-dessus, ces concepts ne peuvent pas appartenir à une même fratrie dont le père serait le concept le plus général `AnyConcept` : il est en effet impossible de dégager un axe sémantique permettant de les réunir et par conséquent d'exprimer les principes différentiels censés justifier cet arbre. La solution passe donc par l'introduction de nouveaux concepts permettant d'effectuer cette jonction entre le sommet de l'ontologie et les concepts propres au domaine modélisé.

On retrouve ici les problèmes liés à ce qu'on appelle les *top-level ontologies*. Quelques tentatives comme *UpperCyc* et dernièrement l'initiative du SUO[2] sous l'égide de l'IEEE ont pour

paraison avec celui obtenu grâce au constructeur `daml:disjointUnionOf` disponible en DAML+OIL.

2. SUO : *Standard Upper Ontology*, `http://suo.ieee.org/`.

but de standardiser les concepts d'une telle ontologie afin que celle-ci puisse resservir à tous ceux qui en ont besoin. Nous nous sommes donc inspirés de ces travaux pour compléter l'ontologie du cyclisme.

Ce travail de modélisation est difficile car il exige des connaissances qui relèvent plus de la philosophie que du domaine étudié. La manipulation des principes différentiels s'avère en effet ardue dans les premières ramifications de l'arbre. La figure 6.2 montre partiellement le résultat obtenu dans les taxinomies des concepts et des relations. Peut-être critiquables, les choix qui ont été effectués ont au moins le mérite de pouvoir être expliqués par l'énonciation des principes différentiels pour chacun des nœuds de l'arbre. De plus, le gain en termes de ré-utilisabilité est indéniable : d'autres ontologies développée à l'INA profitent de l'effort effectué en reprenant des parties hautes de l'ontologie.

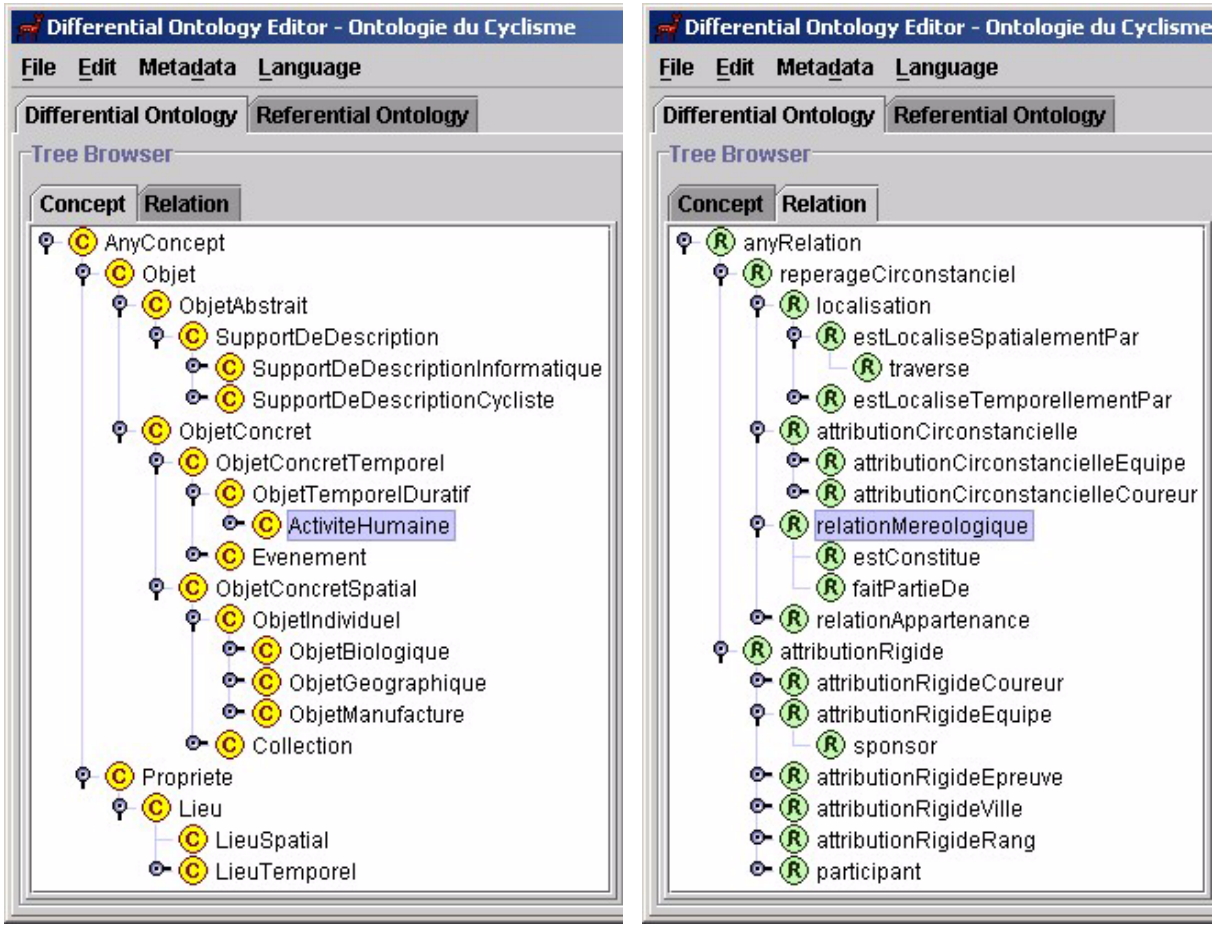


FIG. 6.2 – *Les* concepts *et les* relations *de haut niveau utilisés dans l'ontologie du cyclisme*

L'ontologie du cyclisme nous fournit maintenant tout le vocabulaire formalisé nécessaire pour décrire le contenu des séquences audiovisuelles traitant de ce sport. Il reste donc maintenant à annoter les séquences avec des instances des concepts et des relations de cette ontologie. Cette tâche étant longue et difficile dans l'absolu, nous avons utilisé un système de génération automatique de faits qui utilise notre ontologie en entrée. Nous présentons cet outil dans la section suivante.

6.2 Génération d'une base de faits

Seigo est un système d'extraction d'information guidée par une ontologie développé par E. Le Roux dans le cadre de sa thèse [Le Roux, 2003]. L'hypothèse de base sur laquelle repose ce système est que les informations pertinentes pour l'indexation audiovisuelle peuvent être extraites des textes relatant les événements télévisés.

Architecture générale

Seigo est donc un prototype d'extraction d'information qui prend en entrée trois types de données (figure 6.3) :

- Des articles de presse écrite ayant un rapport avec l'événement visé dans les documents audiovisuels à décrire : il s'agit du corpus d'articles que nous avons déjà rassemblé pour construire l'ontologie.
- Une ontologie de domaine : il s'agit ici de celle du cyclisme.
- Un dictionnaire sémantique qui contient en fait des instances de concepts permettant d'amorcer le processus d'extraction d'information. Ce dictionnaire regroupe par exemple les protagonistes du Tour de France (coureurs cyclistes, directeurs d'équipes), les noms des équipes, les noms des villes étapes... Il peut être assimilé à de la connaissance de contexte nécessaire à l'appréhension du domaine [Euzenat, 2002].

Le système Seigo fonctionne en trois phases : une analyse morpho-syntaxique, la reconnaissance des entités nommées et l'application de patrons du domaine [Le Roux, 2003]. L'analyse morpho-syntaxique est faite à l'aide du logiciel *Cordial 8* [3]. Celui-ci permet de découper les textes en phrases et en mots et de lemmatiser les mots du corpus. Les erreurs d'étiquetage ont ensuite été corrigées à la main. La sortie de cette analyse va permettre de créer par la suite les patrons d'extraction d'information. La reconnaissance des entités nommées s'effectue de deux manières : en utilisant le dictionnaire sémantique construit à partir du corpus de textes et de l'ontologie du cyclisme, et en utilisant des patrons spécialement créés pour retrouver certaines entités nommées. Ces patrons font intervenir à la fois la syntaxe et les étiquettes conceptuelles fournies par le dictionnaire. Finalement, la dernière phase du système consiste à appliquer des patrons d'extraction du domaine. Cette étape est la plus complexe. Elle fait intervenir les concepts et les relations de l'ontologie et a pour but de fournir un ensemble de faits formalisés qui pourront être ancrés à des segments de documents audiovisuels pour les annoter.

Quels types d'information extraire ?

Un des problèmes rencontrés lors de l'élaboration du système Seigo a été ensuite d'identifier les informations à extraire des corpus textuels et qui vont donc décider des patrons d'extraction à construire. Pour cela, différentes enquêtes ont été menées auprès de documentalistes de l'INA, du service des sports de France Télévision, et de la société du Tour de France [Le Roux, 2003]. Les résultats nous renseignent sur le type d'information habituellement décrite car recherchée. Pour les documentalistes de l'INA, les événements suivants sont ainsi jugés pertinents pour la description :

- le descriptif des images telles qu'elles sont ;

3. Le logiciel Cordial est développé par Synapse Développement, `http://wwww.synapse-fr.com`.

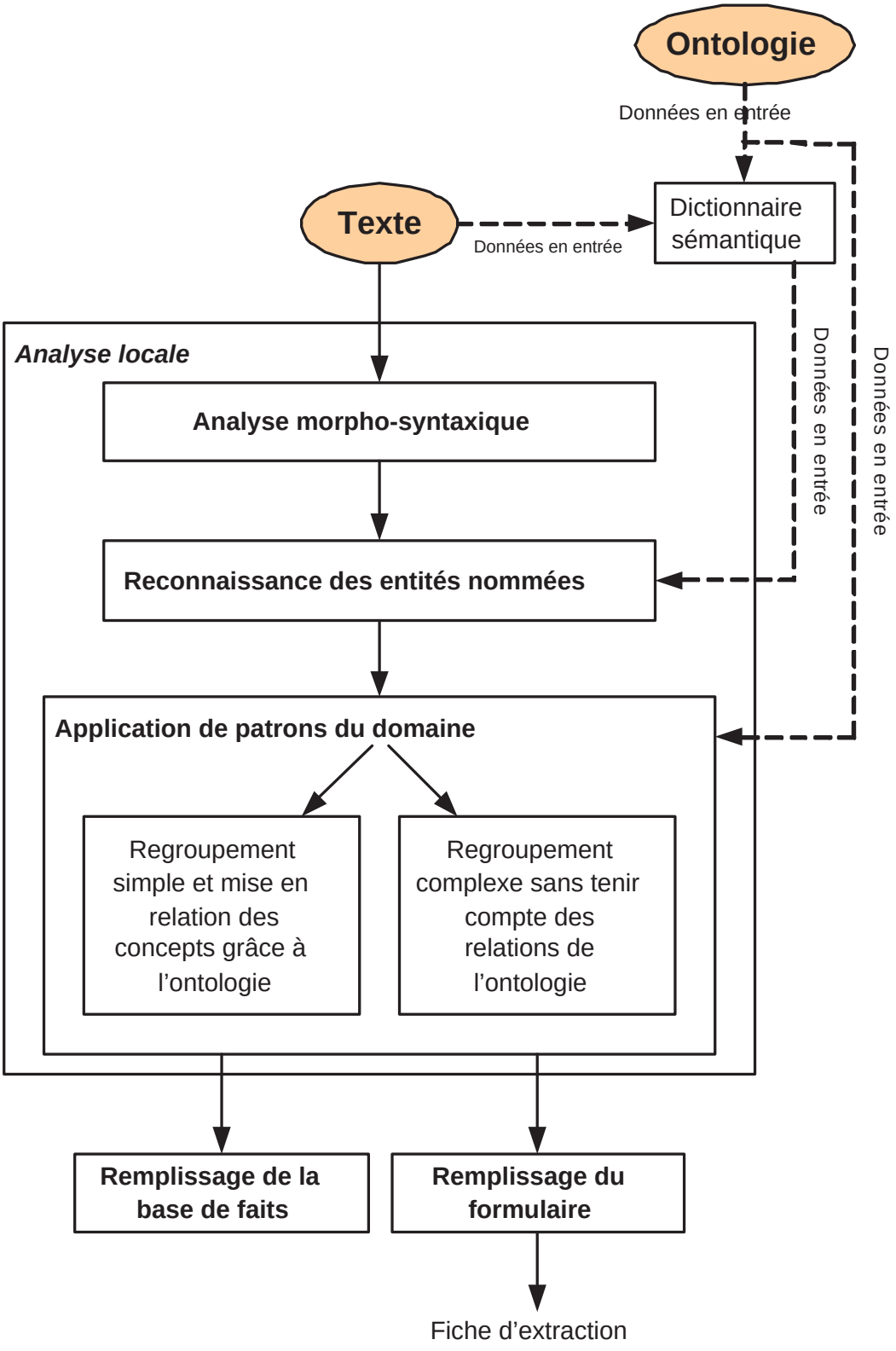


FIG. 6.3 – *Architecture du système* SEIGO, *d'après [Le Roux, 2003]*

- ce qui est factuel ;
- tout ce qui est inhabituel ;
- l'ergonomie des vélos ;
- les personnes qui apparaissent à l'écran (les supporters, les médecins...) ;
- la remise des maillots ;
- la liste des numéros des dossards avec le nom des coureurs ;
- les horaires des coureurs lors des contre-la-montre ;
- toutes les chutes ;
- tous les passages de col ;
- toutes les crevaisons ;
- toutes les échappées ;
- la santé des coureurs.

Les documentalistes de France Télévision distinguent, elles, les informations qui sont indexées le jour de l'étape pour lequel elles décrivent tout de qui est diffusé à l'antenne, et les informations qui vont permettre de préparer les sujets des magazines comme *Stade 2*. Le tableau 6.3 regroupe ces différentes informations.

Informations intéressantes prises en direct	**Informations pour les magazines**
- le vainqueur - les maillots (vert, jaune, à pois...) - les classements - les victoires - les chutes - les abandons - les arrivées d'étape - *etc.*	- le dopage - le portrait des vainqueurs - les résultats des coureurs dans les autres courses - les coureurs en contre-emploi - les coureurs dans le village du Tour - les coureurs lors de la pause de la dernière étape - le vainqueur de l'étape du 14 juillet - les animaux qui se font « remarquer » au cours de l'étape (par exemple, un cheval qui galope à côté des coureurs) - les bords de route - les débuts de la télévision avec le sport - tout ce qui est récurrent - prendre toutes les informations sur les coureurs les plus populaires - *etc.*

TAB. 6.3 – *Les informations intéressantes pour France 2 lors d'un Tour de France*

Les résultats de ces enquêtes ont présidé à l'élaboration du système d'extraction d'information SEIGO. Mais celui-ci étant un prototype de recherche, seules les séquences correspondant aux *échappées* et aux *chutes* ont été étudiées et implantées.

Exemple du système Seigo

Pour mieux illustrer les résultats fournis par le système d'extraction d'information, prenons l'exemple suivant. La phrase

> *Au terme d'une échappée, l'espagnol David Etxbarria (Once) s'est imposé, vendredi 16 juillet, lors de la 12e étape entre Saint-Galmier et Saint-Flour (202 km), devançant le champion de France François Simon (CA).*

provient d'une dépêche AFP et a été extraite de notre corpus d'articles de presse. Elle est donnée en entrée au système Seigo, qui l'analyse et fournit en sortie un ensemble d'instances de concepts et de relations de l'ontologie du cyclisme qui peuvent se représenter sous la forme du graphe de la figure 6.4.

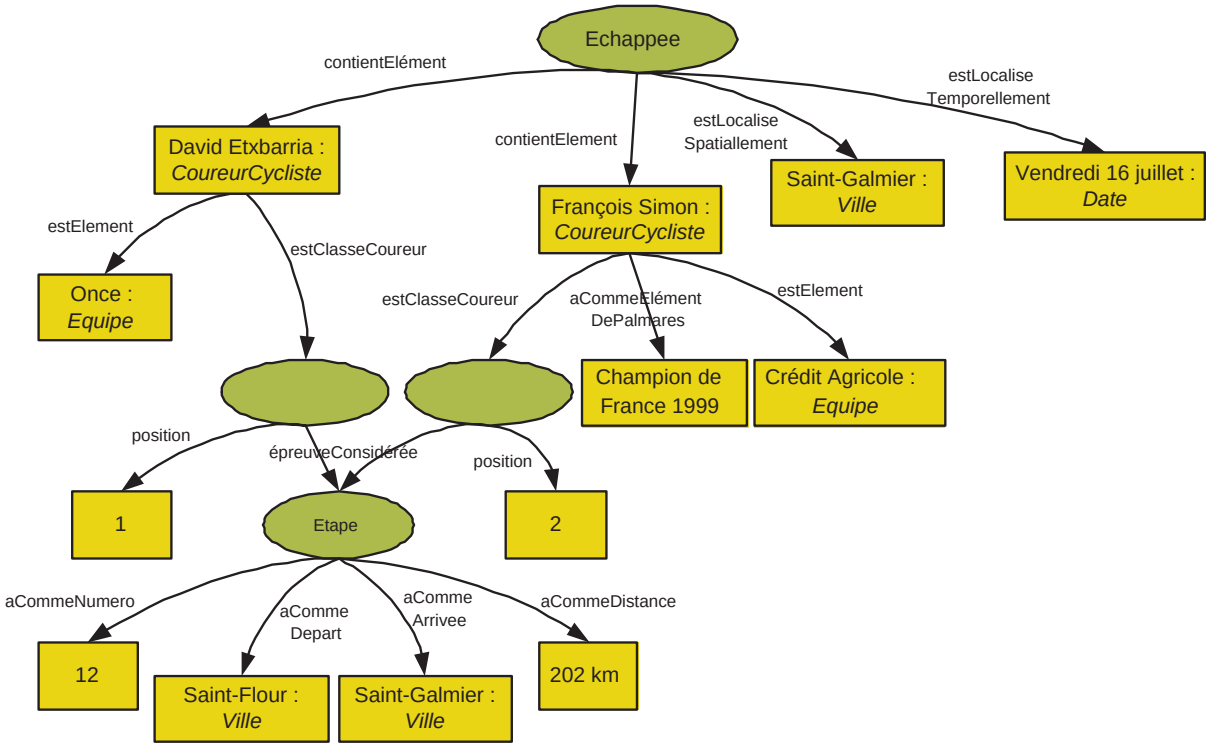


Fig. 6.4 – *Représentation sous forme de graphe d'une phrase analysée par le système* Seigo

Lier les annotations conceptuelles à la description structurelle des documents audiovisuels

Nous avons déjà vu dans le chapitre précédent comment le découpage temporel d'un programme particulier fournit l'instanciation de la structure de la description et par conséquent les connaissances assertionnelles liées au domaine de l'audiovisuel. Les assertions formalisées provenant du système Seigo représentent la sémantique des informations contenues dans les séquences. Il reste donc maintenant à lier ces faits avec les segments audiovisuels, ce qui constitue la phase d'annotation proprement dite des documents.

Pour cela, nous proposons de revenir à la description de la structure des documents. Cette description est un document XML (plus précisément, MPEG-7 étendu) qui identifie parfaitement

chacune des séquences à décrire grâce à des *time-code*. Chacune des parties de ce document XML est accessible via une expression XPATH [XPath, 1999]. Nous proposons donc d'utiliser ce langage pour lier les faits à la description de la structure des documents qui renvoie, elle, aux séquences audiovisuelles. Cette technique permet de délocaliser la description de la structure de celle du contenu. Cela permet ainsi de faire cohabiter de multiples descriptions pour le même segment reflétant ainsi les multiples points de vue de description que l'on peut avoir pour une séquence audiovisuelle donnée.

Par exemple, la figure 6.5 indique que *le coureur cycliste Sandy Casar est désormais classé 2e du classement général de la course Paris-Nice* et que cette assertion est liée – via une relation XPATH – à un segment particulier d'une description d'un document audiovisuel. La description structurelle XML visée par cette expression est donnée dans le code 5.4. Ces faits ont également une traduction en RDF et peuvent à leur tour alimenter notre base des connaissances.

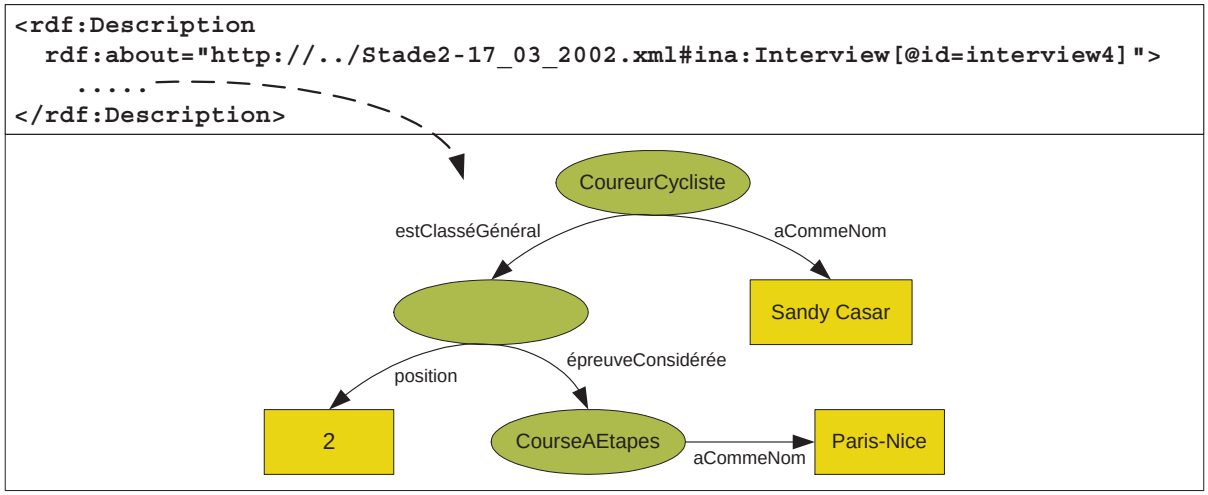


FIG. 6.5 – *Graphe exemple d'une assertion en XML/RDF*

La construction de l'ontologie du cyclisme et l'utilisation du système SEIGO permet de disposer d'un ensemble de faits que l'on va utiliser pour annoter formellement le contenu de séquences audiovisuelles. À titre d'expérimentation, nous avons ainsi annoté des séquences provenant du magazine sportif *Stade 2*. Reste à savoir maintenant si nous sommes capables de retrouver les séquences audiovisuelles annotées, selon leur structure et leur contenu. Nous présentons dans la section suivante les résultats des tests effectués à cette occasion.

6.3 Expérimentations

Une fois les séquences audiovisuelles annotées, il reste à vérifier que nous sommes effectivement capables de les retrouver en interrogeant notre base de descriptions. Nous avons ainsi défini dans la section 4.1 les fonctionnalités attendues du système de recherche que nous rappelons ci-dessous :

– **F4** : Permettre la recherche sémantique de segments documentaires selon des critères de structure et de contenu en effectuant au besoin des inférences sur les descriptions pour améliorer la recherche.

– **F5** : Permettre la navigation dans la structure des documents retrouvés, et ainsi, retrouver le contexte de diffusion des séquences audiovisuelles.

Dans cette section, nous présentons deux expérimentations qui montrent comment ces fonctionnalités sont satisfaites grâce à l'architecture générale que nous avons proposé. La première expérimentation utilise l'architecture de stockage RDF **SeBOR** (section 6.3.1), et la seconde, l'outil **Racer** (section 6.3.2). Nous montrons finalement comment nous revenons à la description de la structure des documents pour jouer les séquences vidéos retournées par le système de recherche (section 6.3.3).

6.3.1 Expérimentations avec le système *SeBOR*

La première expérimentation que nous avons menée utilise l'architecture **Sesame** [Broekstra, 2002], qui permet de stocker et d'interroger l'ensemble des données ontologiques et assertionnelles à l'aide des langages de requêtes RQL et RDQL (voir section 3.5.1). Cette architecture utilise le moteur d'inférence **RDF-MT** qui implémente totalement la sémantique du langage RDFS. Le modèle de stockage est orienté « triplets » puisque toutes les données sont rangées sous forme de triplets RDF. Lorsque les schémas d'ontologies sont entrés dans la base, le système infère tous les triplets possibles compte tenu de la sémantique du langage RDFS, à la manière de Datalog. La tableau 6.4 donne la taille de notre base de connaissances avec les ontologies de l'audiovisuel et du cyclisme avant et après saturation de la base.

	Nb triplets bruts	**Nb triplets inférés**	**Total**
Ontologie de l'audiovisuel	1550	3028	4578
Ontologie du cyclisme	737	1084	1821
Total (avec les triplets dus à RDFS)	2287	4226	6513

TAB. 6.4 – *Nombre de triplets contenus dans la base de connaissances* Sesame *avec les ontologies de l'audiovisuel et du cyclisme*

Reprenons maintenant l'exemple donné au début du chapitre 4 (section 4.1, page 97) et suivi jusqu'ici : nous souhaitons

> « *retrouver toutes les séquences audiovisuelles* `dialogue` *d'un* `coureur cycliste` *qui se serait classé 2e dans le cadre d'une* `course à étapes` ».

Dans le langage RDQL, cette requête se traduit comme indiqué dans le code 6.3.

La figure 6.6 montre le résultat de cette requête, une fois appliquée dans l'architecture Sesame. Une des réponses fournie par le système est la quatrième interview du reportage correspondant à la séquence 12 dans le tableau 4.1 (voir page 98). En effet, grâce à l'ontologie de l'audiovisuel, le système a inféré qu'une `interview` était une `séquence dialogue`, et grâce à l'ontologie du cyclisme, que la course `Paris-Nice` était une `course à étapes` et que `Sandy Casar` était bien un `coureur cycliste`.

L'architecture Sesame n'implémente pour l'instant que la sémantique de RDF Schema comme langage d'ontologies, mais des raisonnements plus poussés sont possibles grâce au raisonneur **BOR** [Simov and Jordanov, 2002] qui a été intégré à Sesame dans le cadre du projet On-To-Knowledge. En fait, BOR implémente la sémantique du langage DAML+OIL, et il est donc proche de ce qui pourra être obtenu avec les futurs moteurs d'inférence OWL (Lite ou DL). Avec

```
SELECT ?x, ?y, ?u, ?v
WHERE (?x, <rdf:type>, <avs:SequenceDialogue>),
      (?x, <avs:aCommeParticipant>, ?y),
      (?x, <avs:aCommeThematique>, <avs:Cyclisme>),
      (?y, <rdf:type>, <cycs:CoureurCycliste>),
      (?y, <cycs:estClasseCoureurGeneral>, ?u),
      (?u, <cycs:epreuveConsideree>, ?v),
      (?u, <cycs:position>, 2),
      (?v, <rdf:type>, <cycs:CourseAEtapes>)
USING rdf FOR <http://www.w3.org/1999/02/22-rdf-syntax-ns#>,
      avs FOR <http://www.ina.fr/audiovisuel-schema/>,
      cycs FOR <http://www.ina.fr/cyclisme-schema/>
```

CODE 6.3 – *Requête n°1 exprimée dans le langage RDQL*

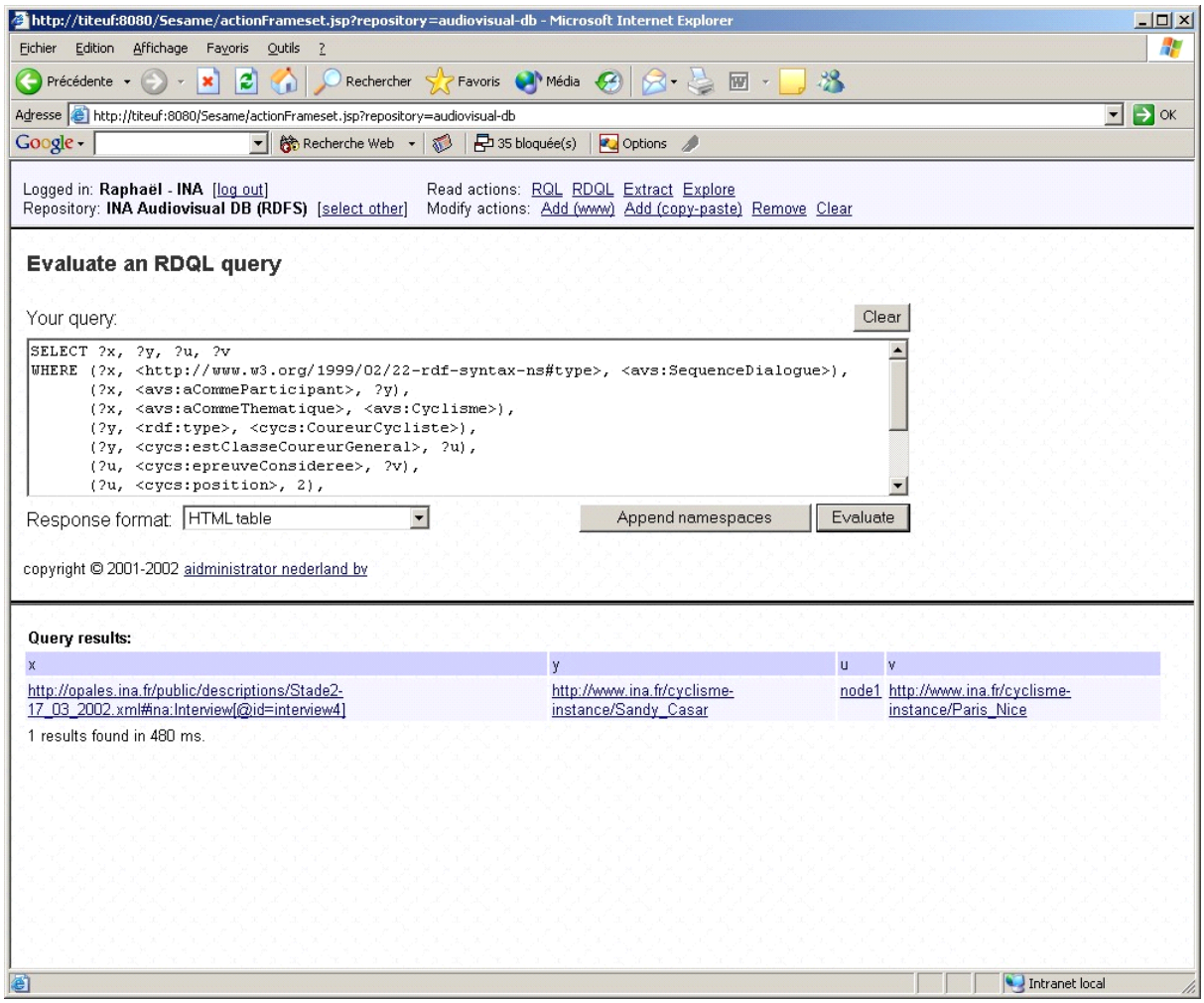


FIG. 6.6 – *Exemple d'une requête RDQL dans le système* SESAME

cette architecture couplée Sesame-BOR, des raisonnements encore plus intéressants peuvent être menés. Ainsi, le système pourra répondre à la question suivante :

> « *retrouver toutes les séquences audiovisuelles où une* `échappée` *est en train de passer un* `col` *d'au moins de* 2e `catégorie` »

qui se traduit en RDQL par le code 6.4.

```
SELECT ?x, ?y, ?z
WHERE (?x, <rdf:type>, <avs:Sequence>),
      (?x, <avs:aCommeParticipant>, ?y),
      (?x, <avs:aCommeThematique>, <avs:Cyclisme>),
      (?y, <rdf:type>, <cycs:Echappee>),
      (?y, <cycs:estLocaliseSpatialementPar>, ?z),
      (?z, <rdf:type>, <cycs:Col21HCCategorie>)
USING rdf FOR <http://www.w3.org/1999/02/22-rdf-syntax-ns#>,
      avs FOR <http://www.ina.fr/audiovisuel-schema/>,
      cycs FOR <http://www.ina.fr/cyclisme-schema/>
```

CODE 6.4 – *Requête n° 2 exprimée dans le langage RDQL. Le concept* `Col21HCategorie` *est défini comme équivalent à l'union des concepts* `Col2Categorie`, `Col1Categorie` *et* `ColHCCategorie`

Il pourra retourner les séquences dont la description indique qu'un groupe de coureurs cyclistes est en train de passer un col classé comme étant de difficulté de 2e, de 1e ou hors catégorie, et que ce groupe de coureurs est en tête de la course.

6.3.2 Expérimentations avec le système *Racer*

La seconde expérimentation que nous avons mené utilise le classifieur de logiques de descriptions **Racer** [Haarslev and Möller, 2001]. Comme nous l'avons vu dans la section 3.5.2, cet outil fournit une implémentation très efficace la $T-Box$ et la $A-Box$ de la logique de descriptions $SHIQ$, c'est-à-dire du langage OWL-Lite. De plus, il implémente le raisonnement sur les types concrets (ou primitifs), et il est donc possible de spécifier dans les requêtes des contraintes linéaires sur les nombres entiers, réels et complexes, ou encore des (in)égalités sur les chaînes de caractères.

Outre les requêtes précédentes, nous pouvons donc établir des requêtes plus complexes tirant partie de cette nouvelle expressivité. Par exemple, la requête

> « *retrouver toutes les séquences audiovisuelles* `interview` *d'un* `coureur cycliste` *dont la durée est supérieure à* `30 secondes` ».

se traduit dans le langage **Racer** par le code 6.5.

```
(define-concept x (and Interview
                        (aCommeParticipant CoureurCycliste)
                        (aCommeDuree duree)
                        (>= duree 30)))
   (concept-instances x)
```

CODE 6.5 – *Requête n° 3 exprimée dans le langage Racer*

L'interface graphique RICE [Möller *et al.*, 2003] permet de visualiser les $T-Box$ et $A-Box$ de notre système à base de connaissances et retourne les résultats des requêtes (figure 6.7).

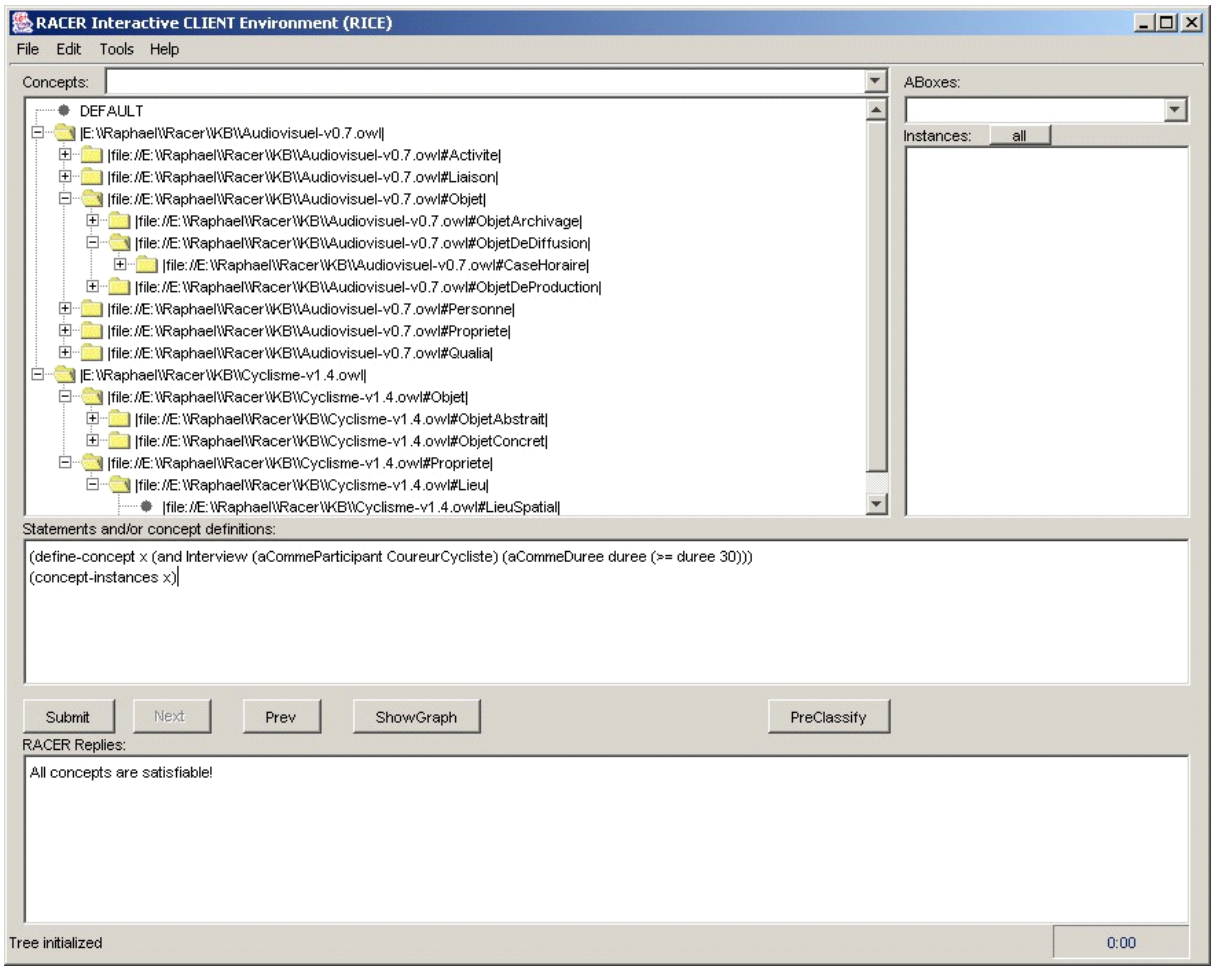


FIG. 6.7 – *L'outil* RICE, *interface graphique du classifieur* Racer

6.3.3 Visionner les séquences audiovisuelles

Les systèmes à base de connaissances permettent donc de retrouver les instances des concepts figurant dans la requête (dans notre cas, des instances du concept `Sequence`), mais il reste maintenant à visionner les séquences audiovisuelles correspondantes. Pour cela, nous revenons à la description de la structure des documents. En effet, chaque description de séquence est une expression XPath qui pointe vers un élément de la description structurelle de l'émission dans laquelle elle se trouve. Il suffit donc de résoudre cette expression XPath pour avoir accès au *time-code* de la séquence à jouer.

Nous avons donc développé une petite feuille de style qui génère une présentation HTML+TIME à partir de la description de la structure MPEG7 étendue des émissions. Cette feuille de style a été écrit dans le langage XSLT [XSLT, 1999]. La figure 6.8 montre le résultat de l'application de cette feuille de style sur la description détaillée dans le code 5.4. La structure est rappelée par l'indentation des tableaux HTML : ainsi, l'interview de *Jean-Marie Leblanc* est une séquence

d'un reportage du magazine *Stade 2*.

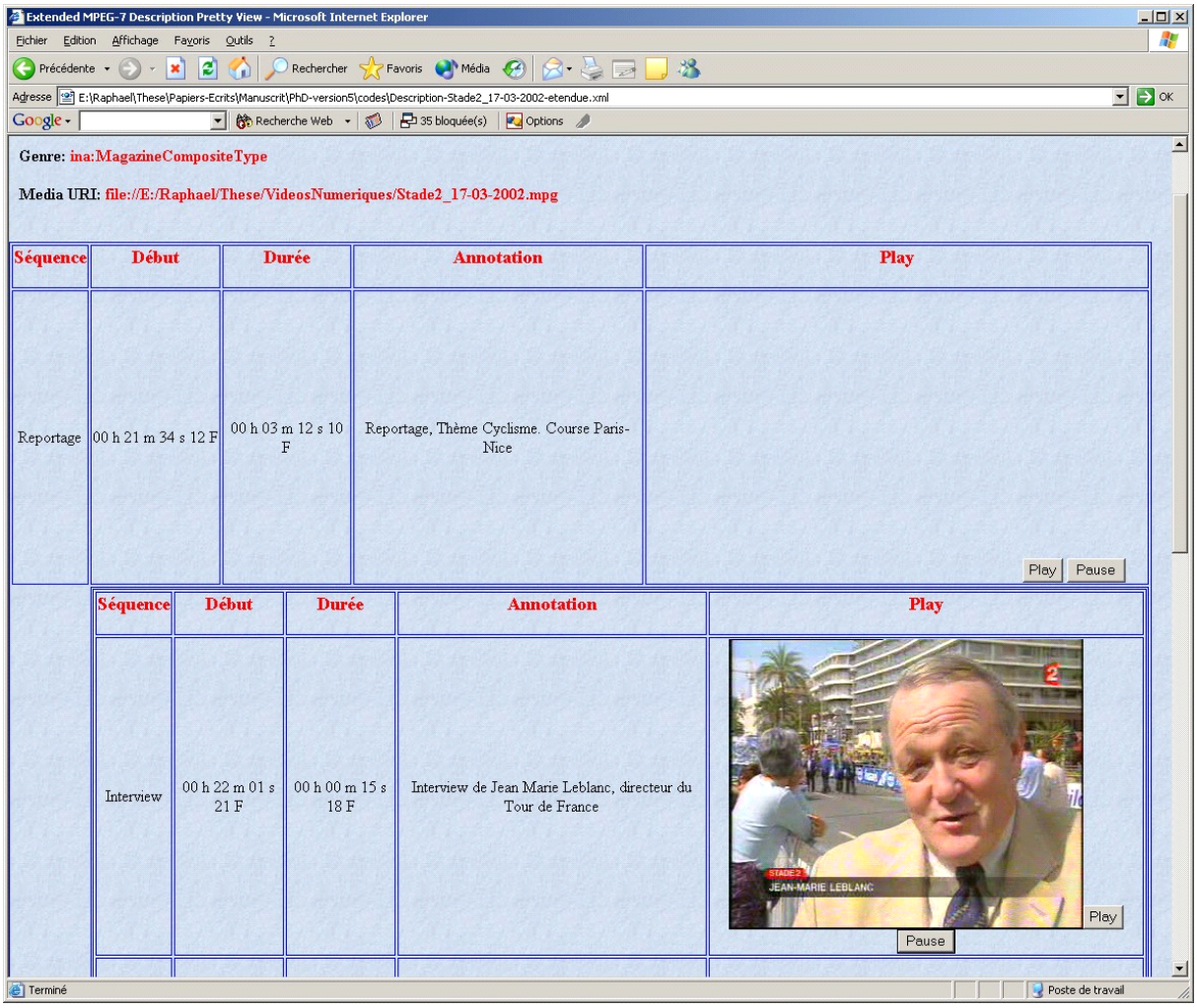


FIG. 6.8 – *Visualisation des séquences vidéos dans* Internet Explorer. *La description de la structure est transformée en une présentation HTML+TIME grâce à une feuille de style XSLT*

Le fait de générer cette présentation dans le langage HTML+TIME (plutôt que SMIL2 par exemple) permet de disposer du logiciel *Windows Media Player* qui s'intègre dans *Internet Explorer*, et qui permet surtout de jouer parfaitement tout types de fichier vidéos. En revanche, cela contraint à utiliser notre système sur une plate-forme Windows.

Finalement, le fait de revenir à la description de la structure des documents audiovisuels permet de revenir au contexte de diffusion des séquences retournées par le systèmes à base de connaissances. Il est désormais très aisé de retrouver la séquence qui suit, qui précède ou qui englobe une séquence donnée, puisque ce type de requête peut se traduire en une expression XPath qui sera résolu par le navigateur Web.

6.4 Conclusion

Dans ce chapitre, nous avons détaillé la construction d'une ontologie du cyclisme, le domaine de connaissances que nous avons choisi pour illustrer notre architecture générale de description. Cette ontologie fournit les concepts et les relations nécessaires à la description du contenu des séquences audiovisuelles traitant de ce sport. Nous avons également montré comment l'ontologie servait de ressource au système SEIGO qui permet de construire semi-automatiquement une base de faits à partir d'un corpus textuel. Nous avons finalement présenté deux expérimentations montrant la pertinence de notre approche puisqu'il est désormais possible d'interroger la base des descriptions selon la structure et la sémantique du contenu des émissions simultanément.

Le chapitre suivant conclut notre travail de thèse mais permet surtout de dresser des perspectives à la recherche entreprise.

Conclusion générale

Bilan de la recherche

À l'heure où les documents multimédias sont de plus en plus présents, nous constatons toute la difficulté à manipuler ce genre de documents. Cette tâche est à la fois complexe et spécifique et nécessite le plus souvent de recourir à des représentations médiatrices. Ainsi, il est d'usage de décrire les documents audiovisuels à l'aide de la paraphrase textuelle. La numérisation des documents audiovisuels permet d'envisager de nouvelles exploitations des contenus grâce à l'instrumentation de leur description. La description fine de la structure va ainsi autoriser de multiples parcours de navigation, ou rendra possible la recomposition dynamique de documents. L'interrogation « intelligente » ou la personnalisation de l'accès sont possibles grâce à l'ajout de connaissances formalisées dans les descriptions. Le système technique qui rend ces services doit alors être intégré et faire le lien entre le contenu et sa description.

Nous avons soutenu qu'une représentation de la *structure* et du *contenu* des documents est nécessaire. Par structure, nous entendons la structure documentaire, c'est-à-dire l'organisation méréologique des éléments qui composent le document, tandis que le contenu est une structure conceptuelle, c'est-à-dire une caractérisation de ces éléments. Le problème que nous nous posons est alors de trouver le formalisme adéquat, à la fois expressif et optimal en terme de manipulations, pour représenter ces descriptions.

Nous avons effectué une revue des propositions actuelles de modélisation des documents audiovisuels, issues de l'ingénierie documentaire et de l'ingénierie des connaissances. L'ingénierie documentaire fournit des modèles et des langages de balisage pertinents pour décrire la structure logique des documents structurés. Le langage MPEG-7 basé sur XML est en particulier le nouveau standard pour décrire les documents multimédias. Nous avons montré ainsi que ce langage permettait raisonnablement d'exprimer et de contrôler la structure des documents audiovisuels, mais qu'il ne permettait pas de décrire formellement la sémantique du contenu. L'ingénierie des connaissances fournit elle des modèles (l'ontologie) et des langages (de représentation des connaissances) pour décrire formellement la sémantique des contenus. Il est alors possible de conduire des raisonnements sur les descriptions dans des systèmes à base de connaissances implémentés. Mais nous avons montré que ce type de langage échoue pour décrire et contrôler la structure des documents puisqu'il ne permet pas d'exprimer des motifs sous forme d'expressions régulières de séquences composant les documents. Nous soutenons donc qu'aucun des langages étudiés ne permet de traiter seul les deux aspects – structure et sémantique – de manière satisfaisante.

En conséquence, nous avons proposé une architecture générale permettant le raisonnement dans les descriptions de documents audiovisuels. Cette architecture se compose d'une ontologie

de l'audiovisuel qui permet de définir formellement la sémantique des descripteurs utiles à la description de la structure des documents. Une partie de cette ontologie est ensuite traduite en types XML Schema qui, une fois combinés, permettent d'exprimer des schémas de description venant valider cette structure. Une autre ontologie d'un domaine de connaissance particulier permet alors de décrire formellement la sémantique du contenu des séquences audiovisuelles. Décrire une émission particulière revient alors à la découper temporellement et à instancier ces schémas de description, ce qui nous fournit les instances des concepts de l'ontologie de l'audiovisuel et peut donc alimenter notre base de connaissances sur laquelle nous pourrons effectuer des inférences.

Nous avons modélisé une ontologie de l'audiovisuel, qui contient des notions de genres, de thèmes et de procédés techniques permettant la production de programmes, en fonction de pratiques professionnelles bien établies. Pour cela, nous avons justifié l'utilisation d'une méthodologie de construction proposée par B. Bachimont. Cette méthodologie n'étant pas outillée, nous avons développé l'éditeur d'ontologies **DOE** qui permet essentiellement de conceptualiser les ontologies. La formalisation peut alors s'effectuer dans d'autres environnements puisque notre outil est interopérable avec eux. Nous avons également traduit une partie de cette ontologie en types XML Schema, liés à des descripteurs MPEG-7, pouvant ainsi exprimer des schémas de description qui contraignent la structure logique d'un ensemble d'émissions. Nous avons modélisé une autre ontologie pour un domaine particulier, le cyclisme. Nous avons pu alors mener quelques expérimentations autour d'un corpus de séquences vidéos annotées provenant du magazine sportif *Stade 2*. Les descriptions formalisées sont stockées dans une base de connaissances implémentée et nous avons montré que l'on pouvait retrouver des séquences par leur structure et leur contenu.

L'approche développée est générique dans le sens où elle peut s'appliquer à tout types de documents très structurés (dont les documents audiovisuels). Mais elle a également ses limites : le coût de construction des ontologies. En effet, l'architecture propose de modéliser une ontologie par domaine de connaissance traité dans les séquences audiovisuelles. L'ontologie de l'audiovisuel est elle générique, mais elle doit tout de même être maintenue pour prendre en compte l'évolution rapide de la télévision (par exemple, les nouveaux genres d'émissions). Toutefois, ces limites peuvent être atténuées pour deux raisons :

1. Compte tenu des efforts actuels, il est raisonnable de penser qu'un nombre croissant d'ontologies de domaine seront disponibles sur le Web et pourront donc être ré-utilisées moyennant quelques adaptations.

2. De nombreux travaux tentent également de proposer des méthodes et des outils pour construire ces ontologies semi-automatiquement à partir de ressources textuelles.

Ces efforts permettent de réduire fortement le coût de construction des ontologies. De plus, nous avons montré comment il est possible d'utiliser un système d'extraction d'information guidé par une ontologie pour proposer une base de faits formalisés pouvant décrire le contenu des séquences. Ces outils terminologiques doivent encore être mieux intégrés aux outils permettant la manipulation formelle des connaissances, mais cela sera indéniablement un domaine de recherche porteur dans un proche avenir dans lequel nous comptons jouer un rôle actif.

Perspectives

La recherche de séquences audiovisuelles semble maintenant plus performante et plus en phase avec les nouvelles possibilités apportées par la numérisation des documents, mais cette amélio-

ration doit encore être évaluée. Cela nécessite de disposer d'un corpus conséquent de séquences annotées, et d'observer les sessions de recherche d'un échantillon d'utilisateurs spécialistes et non spécialistes de la documentation audiovisuelle afin de mesurer le gain obtenu en terme de recherche.

Une évolution normale de notre travail est ensuite de l'appliquer à un autre domaine que le sport ou le cyclisme pour vérifier la généricité de la méthode proposée. S'agira-t-il seulement de construire une autre ontologie de domaine ou l'architecture devra-t-elle être revue ? L'évolution du Web actuel vers le Web Sémantique devrait rendre disponible un nombre croissant d'ontologies même si celles-ci seront probablement de qualités très diverses. Il est donc probable de trouver plusieurs ontologies conceptualisant le même domaine. À l'inverse, les documents audiovisuels abordent souvent des thématiques à l'intersection de plusieurs domaines. Dans tous les cas, l'architecture devrait pouvoir gérer le fait que plusieurs ontologies co-existent pour décrire le(s) domaine(s). Cette problématique rejoint celle de l'interopérabilité (ou de l'alignement) entre ontologies.

Un autre problème intéressant à aborder consiste à évaluer l'apport de l'ontologie formelle de l'audiovisuel pour la recherche de séquences audiovisuelles d'un grand fonds documentaire comme celui de l'INA, mais sans changer les descriptions existantes. En effet, les expérimentations menées jusqu'à présent pré-supposaient de revoir le processus de description afin de prendre en compte l'architecture que nous avons proposée. Il s'agirait cette fois de conserver l'ensemble des descriptions telles quelles, et d'effectuer une correspondance entre d'une part les termes utilisés actuellement dans ces descriptions et provenant de typologies, et d'autre part les concepts formels de l'ontologie de l'audiovisuel. Il sera alors possible de conduire des inférences sur cette partie de la description – qui est la plus utilisée – et donc d'interroger « intelligemment » une grosse base documentaire.

Finalement, nous avons longuement recherché dans notre travail le langage de description « idéal » permettant de décrire la structure et la sémantique du contenu de documents audiovisuels. Nous avons justifié l'utilisation conjointe de langages documentaires (MPEG-7 et XML Schema) et de langages de représentation des connaissances (OWL et RDF) pour à la fois contraindre la structure logique de collection d'émissions et décrire leur contenu. Mais nous avons également mis en avant les inconvénients de cette solution et la quête d'un « meilleur » langage de description pour l'audiovisuel est toujours d'actualité.

Cette problématique rejoint celle d'un nouveau projet intégré, **aceMedia**[4] (*Autonomous Content Entity*), financé par la communauté européenne à l'occasion du 6ème programme cadre européen de la recherche, et dont l'objectif est : « *Integrating knowledge, semantics and content for user-centred intelligent media services* ». Ce projet a donc pour ambition de fournir des outils centrés sur des besoins utilisateurs et permettant la manipulation de contenus multimédias, incluant la recherche « intelligente » de contenus, la personnalisation de leur accès ou encore leur adaptation à des terminaux de réception différents. Il devrait donc fournir un terrain de jeu intéressant pour confronter les idées développées dans cette thèse avec les recherches entreprises dans le cadre du projet.

4. `http://www.acemedia.org`.

Annexe A

Complément sur l'INA

A.1 Les différentes missions confiées à l'INA

Issu de l'éclatement de l'ORTF en 1974, l'INA, Établissement Public à caractère Industriel et Commercial (EPIC), conserve et exploite les archives audiovisuelles des télévisions et radios nationales et gère le dépôt légal des documents sonores et audiovisuels radio et télédiffusés. Cette mission patrimoniale trouve sa réalisation à travers deux départements distincts :

- Le Département Droits et Archives (DDA), héritier du fonds des archives de l'ORTF, s'occupe de l'archivage professionnel. Celui-ci concerne uniquement les chaînes publiques de radio et de télévision et est destiné à un usage professionnel : journalistes, réalisateurs, producteurs, programmateurs et éditeurs de l'audiovisuel et du multimédia. L'INA doit ainsi assurer la conservation, la gestion et la commercialisation de plus de 1 100 000 heures de programmes réparties sur près de 100 km de rayonnages (Figure A.1).

- L'Inathèque de France, instituée le 1er Janvier 1995, est en charge de la mise en œuvre du dépôt légal de la radio et de la télévision défini par la loi du 20 Juin 1992. L'Inathèque recense ainsi tous les programmes des diffuseurs nationaux hertziens de télévision, de 13 chaînes du câble et du satellite et de 17 chaînes de radio (Figure A.2). L'accès en est ouvert aux étudiants et aux chercheurs sur le site de la bibliothèque François Mitterrand (Tolbiac). Plus de 950 000 heures de programmes y sont conservées depuis sa création s'étalant sur près de 14 km de rayonnages.

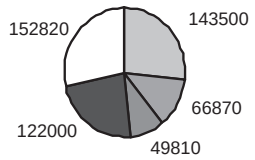
Ces fonds colossaux de plus de deux millions d'heures de radio et de télévision font de l'INA le premier centre d'archives audiovisuelles au monde mais également la première banque d'images numériques en Europe grâce à sa collection photographique de plus d'un million de documents.

Attachée au pôle patrimoine, se trouve la Direction Recherche et Expérimentation (DRE) qui concentre ses travaux autour de trois thèmes principaux :

- le traitement des objets sonores, pour leur production et leur création, à travers le Groupe de Recherches Musicales (GRM) ;

- le traitement des programmes audiovisuels, pour la restauration numérique et la protection des données, à travers l'équipe Traitement Technique de l'Audiovisuel (TTA) ;

- la recherche et l'expérimentation de nouveaux moyens d'exploitation des contenus audiovisuels en s'appuyant sur la manipulation de leurs descriptions, à travers l'équipe Description des Contenus Audiovisuel (DCA) au sein duquel cette recherche s'est effectuée.

	Origine des Fonds	Période Couverte	Type de Contenu	Heures de Programmes par Support
Cinéma	Leclerc Actualités françaises Office français d'information cinématographique Coopérative générale du cinéma	1914-1934 1940-1969 1940-1944 1944-1963	Sujets d'act ualité Sujets d'actualité et documentaires Sujets d'actualité tournés par les alliés Longs et courts métrage, dont *La Bataille du Rail*	 Film et Kinescope (1914-1982) Bandes vidéo 2 pouces (1962-1982) Bandes vidéo 1 pouce (1977-1989) Cassettes ¾ pouce (1977-1990) Cassettes ½ pouce Béta (1989-) **Soit 535 000 heures de programmes**
Télévision	RTF, ORTF TF1 Antenne 2, France 2 FR3, France 3 France 5 Divers : La Cinq, TV Emploi, Ministère de la jus tice, Arménie…	1949-1975 1975-1982 1982-2000 1975- 1975- 2000- *	Fonds public Fonds privé	
Radio	Paris PTT Radio Paris, Radiodiffusion nationale (Vichy), BBC … RDF, RTF, ORTF Radio France Sorafom, Ocora Radio France Internationale Divers : Voix de l'Amérique, AFP audio, Radios locales privées	1933-1940 1940-1944 1945-1975 1975- 1965-1969 1975- *	Programmes vers l'Outre - Mer	 Disque 78 tous à gravure directe (1933-1955) Bandes magnétiques (1950-) Cassettes DAT (1990-) CD Audio (1999-) **Soit 575 000 heures de programmes**

* Période non renseignée

FIG. A.1 – *Aperçu des volumes et origine des fonds du Département Droits et Archives au 1er janvier 2003 (source : communication interne de l'INA)*

		Chaînes concernées	Heures de programme conservées	Heures de programme captés
Télévision	**Câble et Satellite**	TV5, TMC, RTL 9, Canal J, Paris Première, Planète, EuroNews, MCM, Jimmy, Serieclub, Eurosport France, LCI	431 610 heures	56 609 heures/an
	Hertzienne	TF1, France 2, France 3, Canal+, France 5, arte, M6		
Radio	**FM**	Europe 1, Nostalgie, RTL, RMC Info Talk Sport, Skyrock, Fun Radio, NRJ, RTL2, RFM, Chérie FM, Europe 2, rfi	502 414 heures	113 376 heures/an
	France	France Culture, France Inter, France Musiques, France Info, France Bleu		
Total heures conservées à l'Inathèque : 934 024 heures Total linéaire de stockage de supports physiques conservés : 14 km Total volume annuel de programmes captés : 169 985 heures / an				

FIG. A.2 – *Chaînes concernées par le Dépôt Légal et aperçu des fonds à l'Inathèque de France au 1er janvier 2003 (source : site Web de l'INA)*

Parallèlement à ces travaux de recherche, la DRE développe des projets expérimentaux, mène des études donnant lieu à des publications professionnelles ou publie encore de nombreux ouvrages consacrés à l'univers audiovisuel et multimédia. Enfin, et pour être tout à fait complet, l'INA est aussi le premier centre européen de formation professionnelle aux métiers de l'audiovisuel et du multimédia, et le lieu de production d'œuvres documentaires ou de fiction et de programmes hypermédias.

L'équipe DCA positionne ses recherches sur les nouveaux moyens d'exploitation des contenus audiovisuels (AV), que ce soit pour l'INA ou pour la communauté audiovisuelle en générale. Ainsi, elle se propose de mener des activités de recherche sur :

- la structuration des contenus pour les nouveaux moyens de diffusion ;
- la production multimédia intégrant des contenus audiovisuels ;
- l'industrialisation des moyens de production multimédia.

L'hypothèse de travail forte et fondatrice du thème de recherche est que la manipulation des contenus doit se faire par l'intermédiaire de la manipulation de *descriptions* de ces contenus plutôt que par la manipulation « directe » de ceux-ci. Cette hypothèse fournit donc comme objet de recherche central la *description des contenus audiovisuels* d'où découle un certain nombre de problématiques de recherche. Une bonne manière de les présenter consiste à revenir sur le cycle de vie de ces descriptions. Celui-ci peut se décomposer en trois phases – la *production*, la *représentation* et l'*utilisation* – qui se déclinent elles-mêmes en différents objectifs de recherche.

L'INA a également lancé en 1999 un Plan de Sauvegarde et de Numérisation afin de sauvegarder les documents déjà archivés sur support magnétique ou film. Quand on sait que 60% des émissions de télévision et 90% des émissions de radio sont enregistrées sur un support unique (et doivent donc faire face au problème de la disparition des machines de lecture) et se dégradent inexorablement au fil du temps (problèmes liés au "syndrome du vinaigre" pour les supports films ou pour les bandes magnétiques, ou à l'inflammation spontanée des pellicules films en nitrate), on s'aperçoit que c'est à une véritable *course contre la montre* que s'est engagé l'Institut.

Enfin, l'Institut a entrepris de structurer ses fonds d'archives en ensembles thématiques afin d'en améliorer la lisibilité et d'en faciliter l'exploitation. Ces ensembles d'archives télévisuelles, radio, et photographiques permettent de constituer des offres commerciales, après analyse et libération des droits, restauration des supports et création de supports de communication. Aujourd'hui, environ 130 corpus thématisés sont en cours de réalisation ou prêts à être exploités. Ils sont construits à partir d'un événement (un ensemble de faits d'actualités marquants qui ne sont ni prévisibles ni récurrents, par exemple, la tempête de 1999 ou l'assassinat du préfet Claude Érignac), d'un thème (au contraire d'un événement, un ensemble de faits récurrents, par exemple, les départs en vacances, la coupe du monde de football) ou d'une personnalité (biographie de la personne telle qu'elle a été traitée par la télévision, par exemple, pour Charles Aznavour, sa carrière de chanteur et d'acteur, mais aussi son engagement en faveur de l'Arménie).

A.2 Notices documentaires produites par les documentalistes de l'INA

Titre propre	Stade 2
Chaîne de diffusion	France 2
Date de diffusion	28.12.1975
Jour	dimanche
Statut de diffusion	Première diffusion
Heure de diffusion	17 :50 :00
Heure de fin de diffusion	18 :50 :00
Durée	01 :00 :00
Thématique	Sports
Genre	Magazine
Type de description	Collection d'émissions composites
Médiamétrie	Sport, magazine, multisports
Public	Présence d'un public in situ
Descripteurs	émission (Stade 2) ; rencontre sportive ; diffusion en direct (duplex)

Chapeau

Magazine sportif multisports hebdomadaire diffusé en direct le dimanche en fin d'après-midi aux alentours de 17h50. Alternance de plateaux et de sujets filmés (reportages ou résumés en images de rencontres sportives). Pierre SLED, présentateur principal de l'émission, est entouré de plusieurs journalistes de la rédaction des sports de France 2 qui développent l'actualité sportive du week-end et de la semaine précédente.

Résumé

• *Principe de l'émission* : Pierre SLED est entouré de plusieurs journalistes qui interviennent successivement en fonction de leurs spécialités respectives. Des sportifs ou des personnalités sont fréquemment interviewés en duplex ou en plateau à l'occasion d'un événement particulier. Un journaliste est présent en régie pour commenter des informations en bref. Un public, composé principalement de jeunes gens est présent sur le plateau de l'émission.

• *Déroulement chronologique* : Pierre SLED présente (en commentaire off) le sommaire en images de l'émission. Il annonce ensuite la rubrique "Dédicace" qui pendant la première demi-heure de l'émission alterne avec les sujets en images. Il s'agit pour les téléspectateurs de retrouver l'identité d'un sportif à partir de sa signature et de quelques indices divulgués au cours de l'émission. Les sujets sont introduits par Pierre SLED qui assure également les transitions avec les journalistes qui se succèdent sur le plateau.

Dispositifs

• *Générique*

- Début : réalisé grâce aux techniques de l'infographie, le générique est précédé par une séquence de sponsoring. Il débute par une animation avec les chiffres "2" et "3" encerclés dans un rond noir ou blanc ; rythme rapide créant l'illusion d'un clignotement. Cette animation fait place à des silhouettes de sportifs (tennisman en action, athlète...) qui évoluent à partir du centre de l'écran sur une musique électronique rythmée. Ces silhouettes se fondent pour former le mot "Sport" en lettres noires. Ce mot s'insère entre les chiffres 2 (en rouge) et 3 (en bleu) dans un rectangle blanc qui rétrécit progressivement pour laisser apparaître en arrière plan une vue plongeante des projecteurs du plateau. Cet effet est réalisé grâce à un zoom arrière. Le générique se conclut par le titre "Stade 2" plein écran, plaqué sur le dessin d'un chronomètre. Le titre se pixelise avant de disparaître rapidement de l'écran, laissant place à une vue générale du plateau.

suite sur la page suivante

suite de la page précédente

- Fin : début du générique en surimpression sur images du plateau dans la partie gauche de l'écran. Un chronomètre occupe la partie droite. Musique électronique rythmée et "tic tac" de chronomètre. Suite du générique se déroulant en bas de l'écran de droite à gauche. Carton final avec le logo "Sport 2-3", les deux chiffres encadrant en haut et en bas le mot "Sport". un écran de sponsoring suit le générique.

• *Dispositifs généraux*

- Dispositifs techniques : Multicaméra, 5 caméras : deux caméras fixes situées de part et d'autre du plateau, deux caméras devant la table des présentateurs. Une louma.
- Organisation spatiale : Les journalistes se succèdent autour d'une table triangulaire en Plexiglas. Du début à la fin de l'émission, Pierre SLED reste debout devant la table. La table est située au centre du plateau. Le public est assis sur des gradins et des sièges à droite, à gauche et devant l'espace occupé par les journalistes.
- Décor et mobilier : La table autour de laquelle sont assis les journalistes est composée de trois consoles en Plexiglas translucide disposées en triangle. L'espace dans lequel se situent les présentateurs est délimité au sol par un plancher en bois. Les sièges en bois sont dotés d'un pied métallique. Au fond et sur les côtés du plateau, des cloisons blanches avec de grandes fenêtres laissent passer une lumière bleutée. Sous ces cloisons, des portes de placards à claire voie laissent entrevoir des objets et vêtements de sport : ballons, gants de boxe... Les vêtements sont sur des cintres dans des penderies. Un écran géant occupe le panneau situé en hauteur, derrière Pierre SLED. Il comporte la mention "Stade 2" et un chronomètre. Il peut servir au lancement et à la conclusion des reportages. Des écrans de contrôle sont présents au pied des caméras fixes. Un écran est situé à proximité de Pierre SLED. A partir de janvier 1997, l'écran géant est absent du décor.
- Ambiance : Couleurs dominantes : blanc, bleu, bois clair. Lumière franche. Son direct.
- Intervenants : L'ambiance est conviviale. Le langage employé est châtié. Les enchaînements sont fluides sans improvisation. Le présentateur principal et les journalistes sont vêtus de façon décontractée (type sportswear). Certains sont en costume cravate.

• *Dispositifs éléments*

- Le sommaire en images est accompagné d'un bandeau situé en bas de l'écran et comportant la mention "à venir". La rubrique "dédicace" adopte toujours le même principe : la signature d'un sportif apparaît en surimpression sur une image floue de sa silhouette. Lorsque le nom du sportif est découvert, celui-ci s'adresse brièvement aux téléspectateurs au cours d'une séquence préalablement enregistrée.
- Les sujets filmés sont commentés off. Ce commentaire peut être assuré par les journalistes présents sur le plateau ou enregistré. Les images des reportages sont siglées "Stade 2". En cas d'achat d'images, une mention précise l'origine de production. En incrustation sur les résumés en images, mentions des scores ainsi que des noms des équipes. Les résumés en images se concluent souvent par un carton récapitulatif des résultats. Pendant les reportages, un bandeau fixe est présent en bas de l'écran avec en surimpression en lettres bleu pastel la mention "Stade 2". Le logo "Sport 2-3" figure à droite. Un générique propre est associé à ces reportages. Certains reportages ou portraits peuvent adopter un ton humoristique.
- Les interventions d'invités en duplex sont fonction de l'actualité sportive (avant ou après un événement). L'invité se trouve dans un studio d'enregistrement en région ou à l'étranger. Il est interrogé sur place par un journaliste de la rédaction de France 2. Le duplex permet à Pierre SLED et à ses collaborateurs de dialoguer avec eux. Les images peuvent laisser voir : le journaliste et l'invité dans le studio ; Pierre SLED et ses collaborateurs sur le plateau ; ou encore, les visages de Pierre SLED et de l'invité dans des médaillons situés en bas d'écran sur les images du plateau.

suite sur la page suivante

suite de la page précédente	
- Les brèves sont exposées en direct par un journaliste présent en régie. • *Construction générale* Les plateaux sont de longueur inégale selon la présence ou non d'invités. La transition plateaux-sujets filmés s'effectue par fondu enchaîné ou par un volet se déroulant de bas en haut. La louma est utilisée principalement au début et à la fin de l'émission pour les vues générales du plateau. Il y a peu de mouvements de caméra, le champ contre champ (généralement utilisé lors des discussions entre journalistes) étant assuré par les caméras fixes. Nombreuses vues générales du public souvent filmées depuis la gauche du plateau. Pierre SLED est souvent filmé en plan de taille.	
Notes • *Version décrite* : émissions de 1996 et 1997. • *Programmation* : en direct, le dimanche en fin d'après-midi. • *Diffusion* : à partir de décembre 1975 : diffusion de 19h à 20h le dimanche ; à partir de 1986 : diffusion de 18h30 à 19h30 ; à partir de septembre 1995 : diffusion de 17h50 à 18h50. • *Contexte de réalisation* : alternance de plusieurs réalisateurs : Renée PERE CHAMPAGNE, Pascal GERVAIS, Jean Marc L'HENORET. Depuis les débuts de l'émission, Gérard Van der GUCHT, Françoise BOULLAIN, Jean Pierre OUALID ont également réalisé l'émission.	
Résumé producteur On ne change pas une formule qui gagne. Depuis maintenant 23 ans, "Stade 2" demeure la seule émission d'actualité multisports et non thématique. Sports traditionnels mais aussi ouverture aux nouvelles disciplines qui font fureur chez les jeunes, évolutions dans le traitement de l'information, tant du point de vue technique (apports de nouveau instruments comme le visiophone) que de l'écriture (nouveau ton grâce à de jeunes journalistes), "Stade 2" reste le magazine sportif de référence et la tribune des champions avec des invités prestigieux. Le tout en public emmené par Pierre Sled sous la houlette de Pierre Salviac.	
Société de programmes	France 2
Nature de production	Production propre
Type de traitement	Indexation formelle

TAB. A.1 – *Extrait de la notice documentaire de la fiche collection, modifiée en 1996, concernant l'émission* `Stade2`

Annexe B

Ontologie de l'audiovisuel

B.1 Taxinomies des concepts

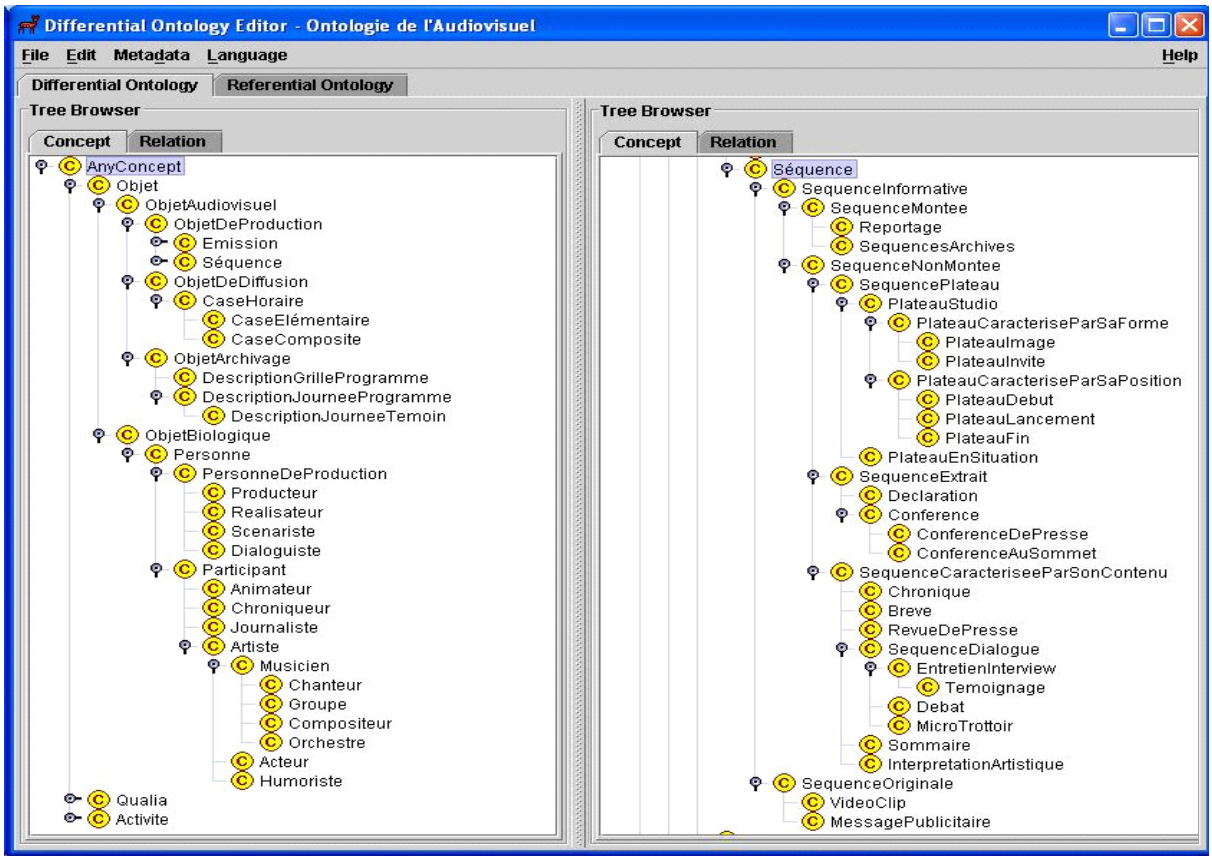


FIG. B.1 – *Taxinomie des concepts dans l'ontologie de l'audiovisuel : le haut niveau et les différents types de séquences*

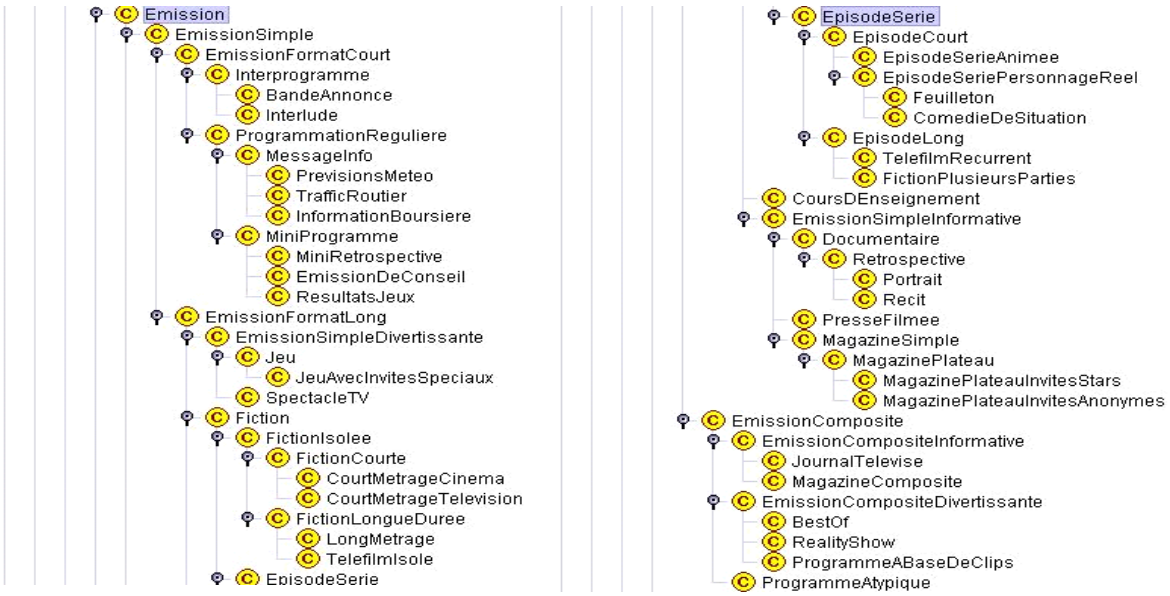


FIG. B.2 – *Taxinomie des concepts dans l'ontologie de l'audiovisuel : les différents types d'émissions*

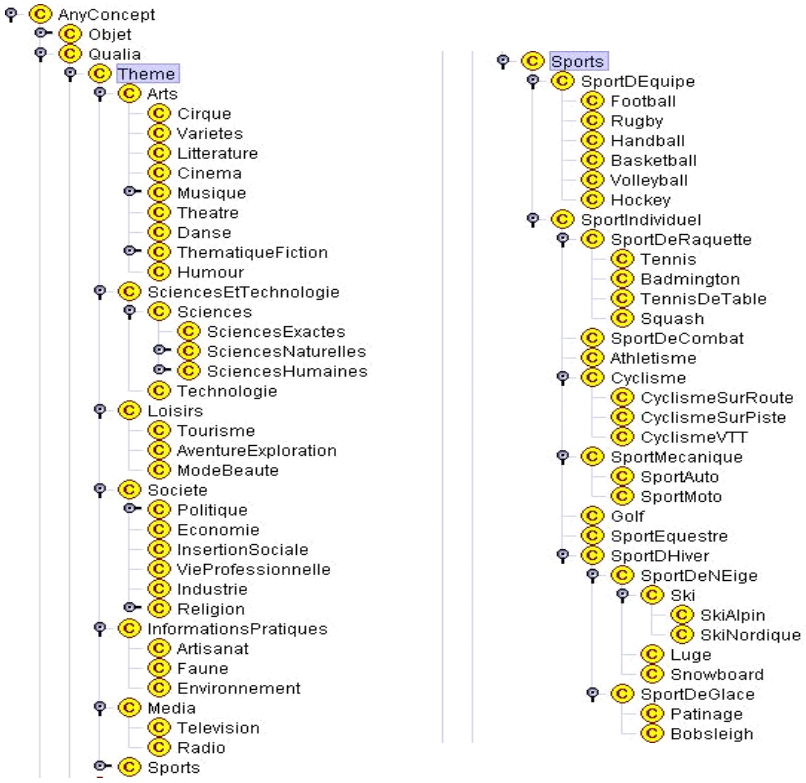


FIG. B.3 – *Taxinomie des concepts dans l'ontologie de l'audiovisuel : les thématiques générales*

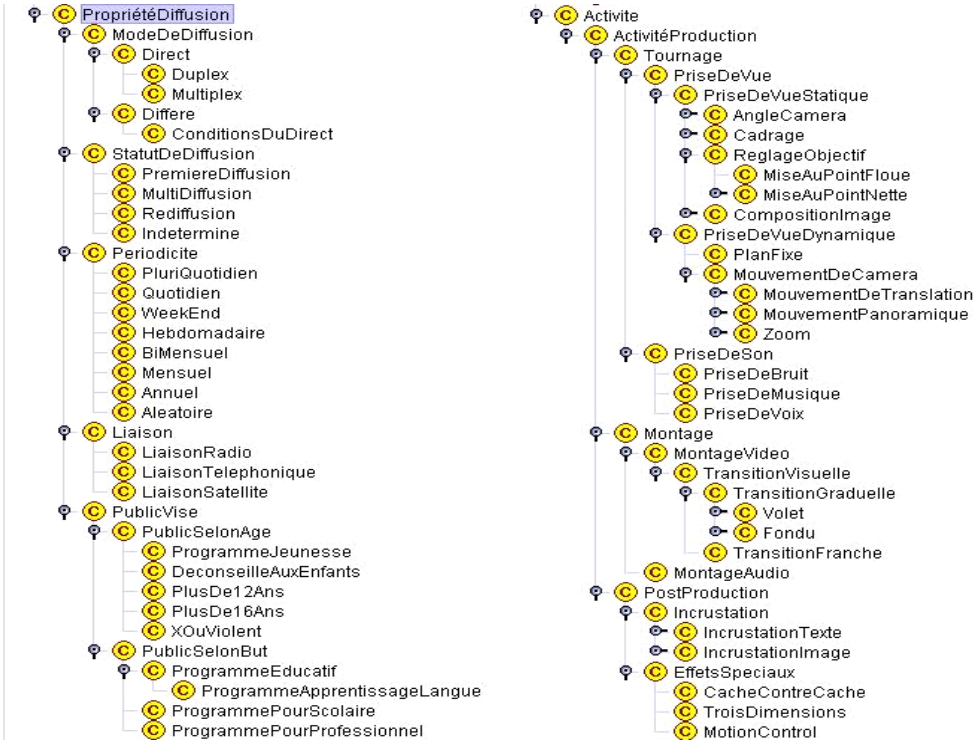


FIG. B.4 – *Taxinomie des concepts dans l'ontologie de l'audiovisuel : les propriétés et les activités audiovisuelles*

B.2 Ontologie différentielle : liste alphabétique des concepts

Activite

- *Concept Parent* : AnyConcept

ActivitéProduction

- *Concept Parent* : Activite

AnyConcept

BandeAnnonce

- *Concept Parent* : Interprogramme
- *SWP* : C'est une émission de courte durée qui n'a pas une programmation régulière
- *SWS* : Le programme a un objectif particulier
- *DWS* : C'est un montage de courts extraits annonçant soit de futurs programmes télévisuels (une émission ou un programme exceptionnel), soit tout autre événement qui exste indépendamment du média (film, manifestation extérieure, etc.)
- *DWP* : Le programme a un objectif particulier : C'est un montage de courts extraits annonçant soit de futurs programmes télévisuels (une émission ou un programme exceptionnel), soit tout autre événement qui exste indépendamment du média (film, manifestation extérieure, etc.)

BestOf

- *Concept Parent* : EmissionCompositeDivertissante
- *Définition Encyclopédique* : Reprise des meilleurs moments (ou morceaux choisis) sélectionnés dans plusieurs numéros d'une collection. Le best of peut être régulier, par exemple tous les mois ou tous les ans, ou unique. Il inclut donc une notion de montage afin de relier les différents éléments, reportages ou séquences, déjà diffusés. Le montage peut être un simple bout à bout ou comprendre des séquences originales (tournées pour le best of), le plus souvent des plateaux. Il peut avoir une thématique et un générique, des invités par exemple, propres. Il peut porter des noms différents en fonction des chaînes ou des émissions : best of, remix, ...

  Attention : ne pas confondre avec le remix de type M6 qui est le plus souvent un reformatage !
- *Synonyme* : Remix

Breve

- *Concept Parent* : SequenceCaracteriseeParSonContenu

Chronique

- *Concept Parent* : SequenceCaracteriseeParSonContenu

ComedieDeSituation

- *Concept Parent* : EpisodeSeriePersonnageReel
- *SWP* : C'est un épisode de moins de 52 minutes avec des personnages réels
- *SWS* : Les épisodes de la série sont diffusables isolément ou pas
- *DWS* : L'épisode est clos, c'est-à-dire qu'il forme un tout cohérent et est diffusable isolément
- *DWP* : Les épisodes de la série sont diffusables isolément ou pas : L'épisode est clos, c'est-à-dire qu'il forme un tout cohérent et est diffusable isolément
- *Définition Encyclopédique* : Version française de Sitcom
- *Synonyme* : Sitcom

Conference

- *Concept Parent* : SequenceExtrait

ConferenceAuSommet

- *Concept Parent* : Conference
- *Définition Encyclopédique* : Utiliser uniquement pour les réunions de chefs d'état ou de gouvernement

ConferenceDePresse

- *Concept Parent* : Conference

- *Définition Encyclopédique* : Rencontre organisée avec des journalistes par une personne physique ou morale, pour leur tenir un discours ou répondre à des questions. Uniquement pour une émission faite pour la TV

  Attention : ne pas confondre avec "Declaration". La notion d'organisation est prépondérante pour la conférence de presse !

  Attention : ne convient pas aux sujets consacrés à une conférence de presse s'ils sont montés !

CoursDEnseignement

- *Concept Parent* : EmissionFormatLong
- *SWP* : C'est une émission de longue durée
- *SWS* : L'émission est caractérisée par un statut particulier
- *DWS* : C'est une émission éducative
- *DWP* : L'émission est caractérisée par un statut particulier : C'est une émission éducative
- *Définition Encyclopédique* : Terme à utiliser pour tous les cours enregistrés et diffusés notamment sur France 5 (cours magistraux, cours techniques, travaux pratiques ...)

CourtMetrageCinema

- *Concept Parent* : FictionCourte
- *SWP* : C'est une oeuvre de fiction de moins de 60 minutes
- *SWS* : L'oeuvre de fiction est conçue originalement pour un média particulier
- *DWS* : L'oeuvre est d'abord conçue pour le cinéma
- *DWP* : L'oeuvre de fiction est conçue originalement pour un média particulier : L'oeuvre est d'abord conçue pour le cinéma
- *Définition Encyclopédique* : Oeuvre de moins de 60 minutes et diffusée soit isolément soit dans le cadre d'une tranche de programmation

CourtMetrageTelevision

- *Concept Parent* : FictionCourte
- *SWP* : C'est une oeuvre de fiction de moins de 60 minutes
- *SWS* : L'oeuvre de fiction est conçue originalement pour un média particulier
- *DWS* : L'oeuvre est d'abord conçue pour la télévision
- *DWP* : L'oeuvre de fiction est conçue originalement pour un média particulier : L'oeuvre est d'abord conçue pour la télévision

Debat

- *Concept Parent* : SequenceDialogue
- *Définition Encyclopédique* : Discussion organisée autour d'un thème et mettant en présence plusieurs interlocuteurs dont les arguments s'opposent

Declaration

- *Concept Parent* : SequenceExtrait
- *Définition Encyclopédique* : Uniquement pour une déclaration faite pour le média

  Attention : ne pas confondre avec le genre "Retransmission" !

Documentaire

- *Concept Parent* : EmissionSimpleInformative
- *SWP* : C'est une émission informative
- *SWS* : L'émission est typique d'une époque
- *DWS* : L'émission est contemporaine
- *DWP* : L'émission est typique d'une époque : L'émission est contemporaine
- *Définition Encyclopédique* : Document à structure simple dont la caractéristique première est d'être traité par un réalisateur et non par un journaliste comme le reportage. Le documentaire est informatif. Il aborde tous les sujets : problèmes de société, faits historiques, scientifiques ou artistiques. Le documentaire prend souvent une distance plus grande que le reportage avec le sujet traité

  Attention : combinaison impossible avec "Reportage" !

Emission

- *Concept Parent* : ObjetDeProduction

EmissionComposite

- *Concept Parent* : Emission
- *SWP* : C'est une émission
- *SWS* : L'émission contient des séquences qui ont un contenu et une forme
- *DWS* : L'émission est caractérisée par l'emploi successif, dans le même numéro, d'éléments autonomes par leur forme et leur contenu (plateau, reportage, entretien, ...) organisés ou non en rubriques
- *DWP* : L'émission contient des séquences qui ont un contenu et une forme : elle est caractérisée par l'emploi successif, dans le même numéro, d'éléments autonomes par leur forme et leur contenu (plateau, reportage, entretien, ...) organisés ou non en rubriques

EmissionCompositeDivertissante

- *Concept Parent* : EmissionComposite

EmissionCompositeInformative

- *Concept Parent* : EmissionComposite

EmissionDeConseil

- *Concept Parent* : MiniProgramme
- *SWP* : C'est un programme court et divertissant
- *SWS* : Le programme a un objectif particulier
- *DWS* : C'est une émission de conseil jardinage, bricolage, santé, monnaie ...
- *DWP* : Le programme a un objectif particulier : C'est une émission de conseil jardinage, bricolage, santé, monnaie ...

EmissionFormatCourt

- *Concept Parent* : EmissionSimple
- *SWP* : C'est une émission homogène dans sa forme et dans son contenu
- *SWS* : L'émission est caractérisée par sa durée
- *DWS* : Il s'agit toujours d'un programme court : quelques minutes
- *DWP* : L'émission est caractérisée par sa durée : Il s'agit toujours d'un programme court : quelques minutes

EmissionFormatLong

- *Concept Parent* : EmissionSimple
- *SWP* : C'est une émission homogène dans sa forme et dans son contenu
- *SWS* : L'émission est caractérisée par sa durée
- *DWS* : Il s'agit toujours d'un programme long : plusieurs dizaines de minutes
- *DWP* : L'émission est caractérisée par sa durée : Il s'agit toujours d'un programme long : plusieurs dizaines de minutes

EmissionSimple

- *Concept Parent* : Emission
- *SWP* : C'est une émission
- *SWS* : L'émission contient des séquences qui ont un contenu et une forme
- *DWS* : L'émission est caractérisée par l'homogénéité de la forme et du contenu quelle que soit le mode de structuration des séquences : montage cut, fondu, rubriques ...
- *DWP* : L'émission contient des séquences qui ont un contenu et une forme : elle est caractérisée par l'homogénéité de la forme et du contenu quelle que soit le mode de structuration des séquences : montage cut, fondu, rubriques ...

EmissionSimpleDivertissante

- *Concept Parent* : EmissionFormatLong

- *SWP* : C'est une émission de longue durée
- *SWS* : L'émission est caractérisée par un statut particulier
- *DWS* : C'est une émission divertissante
- *DWP* : L'émission est caractérisée par un statut particulier : C'est une émission divertissante

EmissionSimpleInformative

- *Concept Parent* : EmissionFormatLong
- *SWP* : C'est une émission de longue durée
- *SWS* : L'émission est caractérisée par un statut particulier
- *DWS* : C'est une émission informative
- *DWP* : L'émission est caractérisée par un statut particulier : C'est une émission informative

EntretienInterview

- *Concept Parent* : SequenceDialogue
- *Définition Encyclopédique* : Interview ou discussion privilégiant la relation binaire. L'interviewer peut être absent à l'image et ses questions supprimées au montage

  Attention : ne pas combiner avec "Débat" !

EpisodeCourt

- *Concept Parent* : EpisodeSerie
- *SWP* : C'est un épisode d'une série
- *SWS* : L'épisode a une durée
- *DWS* : L'épisode a une durée maximale de 52 minutes. Durée ordinaire : 13, 26 ou 52 minutes
- *DWP* : L'épisode a une durée : L'épisode a une durée maximale de 52 minutes. Durée ordinaire : 13, 26 ou 52 minutes
- *Définition Encyclopédique* : Il a une durée maximale de 52 minutes. De plus, la série a un caractère répétitif et elle est caractérisée par la convention et la répétition propre à chaque suite des personnages, de la mise en image, des décors ET des situations. Durée ordinaire : 13, 26 ou 52 minutes

EpisodeLong

- *Concept Parent* : EpisodeSerie
- *SWP* : C'est un épisode d'une série
- *SWS* : L'épisode a une durée
- *DWS* : L'épisode a une durée minimale de 52 minutes (jusqu'à 90 minutes)
- *DWP* : L'épisode a une durée : L'épisode a une durée minimale de 52 minutes (jusqu'à 90 minutes)
- *Définition Encyclopédique* : Il a une durée minimale de 52 minutes (jusqu'à 90 minutes). De plus, la série est caractérisée par des personnages ou des situations récurrents, des décors variés en studio et en extérieur. Ces séries sont diffusées en général en début de soirée (prime time)

  Attribut : Adaptation –>boolean def : émission réalisée pour la TV, à partir d'une oeuvre pré-existante (oeuvre écrite) quelque soit le genre de cette oeuvre

EpisodeSerie

- *Concept Parent* : Fiction
- *SWP* : C'est une oeuvre de fiction
- *SWS* : L'oeuvre est isolée ou participe d'une série
- *DWS* : L'oeuvre participe d'une série. Il s'agit d'une suite fermée, c'est-à-dire un ensemble dont le nombre d'émissions est défini a priori et est strictement supérieur à 3 !
- *DWP* : L'oeuvre est isolée ou participe d'une série : L'oeuvre participe d'une série. Il s'agit d'une suite fermée, c'est-à-dire un ensemble dont le nombre d'émissions est défini a priori et est strictement supérieur à 3 !

EpisodeSerieAnimee

- *Concept Parent* : EpisodeCourt

- *SWP* : C'est un épisode de moins de 52 minutes
- *SWS* : L'épisode comporte des personnages
- *DWS* : Les personnages sont animés (pour tout type d'animation : dessin animé, marionette, images de synthèse)
- *DWP* : L'épisode comporte des personnages : Les personnages sont animés (pour tout type d'animation : dessin animé, marionette, images de synthèse)
- *Définition Encyclopédique* : Pour tout type d'animation : dessin animé, marionette, images de synthèse. Ce terme couvre notamment la notion de "NouvellesImages" qui est un terme archivé
- *Synonyme* : NouvellesImages

EpisodeSeriePersonnageReel

- *Concept Parent* : EpisodeCourt
- *SWP* : C'est un épisode de moins de 52 minutes
- *SWS* : L'épisode comporte des personnages
- *DWS* : Les personnages sont réels
- *DWP* : L'épisode comporte des personnages : Les personnages sont réels

Feuilleton

- *Concept Parent* : EpisodeSeriePersonnageReel
- *SWP* : C'est un épisode de moins de 52 minutes avec des personnages réels
- *SWS* : Les épisodes de la série sont diffusables isolément ou pas
- *DWS* : Les épisodes sont ouverts, c'est-à-dire qu'ils ne peuvent pas être diffusés isolément : les numéros sont conçus pour être diffusés dans un ordre donné
- *DWP* : Les épisodes de la série sont diffusables isolément ou pas : Les épisodes sont ouverts, c'est-à-dire qu'ils ne peuvent pas être diffusés isolément : les numéros sont conçus pour être diffusés dans un ordre donné
- *Définition Encyclopédique* : Attribut : Adaptation –>boolean def : émission réalisée pour la TV, à partir d'une oeuvre pré-existante (oeuvre écrite) quelque soit le genre de cette oeuvre

Fiction

- *Concept Parent* : EmissionFormatLong
- *SWP* : C'est une émission de longue durée
- *SWS* : L'émission est caractérisée par un statut particulier
- *DWS* : C'est une oeuvre de fiction
- *DWP* : L'émission est caractérisée par un statut particulier : C'est une oeuvre de fiction

FictionCourte

- *Concept Parent* : FictionIsolee
- *SWP* : C'est une fiction isolée
- *SWS* : L'oeuvre de fiction a une durée
- *DWS* : L'oeuvre a une durée inférieure à 60 minutes
- *DWP* : L'oeuvre de fiction a une durée : L'oeuvre a une durée inférieure à 60 minutes

FictionIsolee

- *Concept Parent* : Fiction
- *SWP* : C'est une oeuvre de fiction
- *SWS* : L'oeuvre est isolée ou participe d'une série
- *DWS* : L'oeuvre est isolée et elle doit être diffusée en 3 parties au maximum
- *DWP* : L'oeuvre est isolée ou participe d'une série : L'oeuvre est isolée et elle doit être diffusée en 3 parties au maximum

- *Définition Encyclopédique* : La fiction isolée est une émission caractérisée par l'originalité de son scénario et de ses personnages. Il peut s'agir d'un court ou d'un long métrage, mais elle doit être diffusée en 3 parties maximum

  Attribut : Adaptation –>boolean def : émission réalisée pour la TV, à partir d'une oeuvre pré-existante (oeuvre écrite) quelque soit le genre de cette oeuvre

FictionLongueDuree

- *Concept Parent* : FictionIsolee
- *SWP* : C'est une fiction isolée
- *SWS* : L'oeuvre de fiction a une durée
- *DWS* : L'oeuvre a une durée supérieure à 60 minutes
- *DWP* : L'oeuvre de fiction a une durée : L'oeuvre a une durée supérieure à 60 minutes

FictionPlusieursParties

- *Concept Parent* : EpisodeLong
- *SWP* : C'est un épisode de plus de 52 minutes
- *SWS* : Les épisodes de la série sont diffusables isolément ou pas
- *DWS* : L'épisode est ouvert, c'est-à-dire qu'il ne peut pas être diffusé isolément : les numéros sont conçus pour être diffusés dans un ordre donné
- *DWP* : Les épisodes de la série sont diffusables isolément ou pas : L'épisode est ouvert, c'est-à-dire qu'il ne peut pas être diffusé isolément : les numéros sont conçus pour être diffusés dans un ordre donné
- *Définition Encyclopédique* : La série comprend au moins 4 parties (à l'inverse de la "FictionIsolee")
- *Synonyme* : Vidéogramme

InformationBoursiere

- *Concept Parent* : MessageInfo
- *SWP* : C'est un programme court et informatif
- *SWS* : Le programme donne une certaine information
- *DWS* : L'information concerne le cours des différentes places boursières
- *DWP* : Le programme donne une certaine information : L'information concerne le cours des différentes places boursières

Interlude

- *Concept Parent* : Interprogramme
- *SWP* : C'est une émission de courte durée qui n'a pas une programmation régulière
- *SWS* : Le programme a un objectif particulier
- *DWS* : C'est un programme court servant à recaler l'antenne
- *DWP* : Le programme a un objectif particulier : C'est un programme court servant à recaler l'antenne

InterpretationArtistique

- *Concept Parent* : SequenceCaracteriseeParSonContenu

Interprogramme

- *Concept Parent* : EmissionFormatCourt
- *SWP* : C'est une émission de courte durée
- *SWS* : L'émission est caractérisée par sa programmation
- *DWS* : L'émission n'a pas une programmation régulière : c'est plutôt un élément ou un ensemble d'éléments appartenant à l'habillage de la chaîne ou à sa politique de programmation diffusé(s) entre deux émissions ou à l'intérieur d'une émission
- *DWP* : L'émission est caractérisée par sa programmation : L'émission n'a pas une programmation régulière : c'est plutôt un élément ou un ensemble d'éléments appartenant à l'habillage de la chaîne ou à sa politique de programmation diffusé(s) entre deux émissions ou à l'intérieur d'une émission

Jeu

- *Concept Parent* : EmissionSimpleDivertissante
- *SWP* : C'est une émission divertissante
- *SWS* : L'émission a un genre particulier
- *DWS* : L'émission est un jeu
- *DWP* : L'émission a un genre particulier : L'émission est un jeu

JeuAvecInvitesSpeciaux

- *Concept Parent* : Jeu
- *SWP* : C'est un jeux
- *DWP* : Les participants au jeu sont connus

JournalTelevise

- *Concept Parent* : EmissionCompositeInformative

LongMetrage

- *Concept Parent* : FictionLongueDuree
- *SWP* : C'est une oeuvre de fiction de plus de 60 minutes
- *SWS* : L'oeuvre de fiction est conçue originalement pour un média particulier
- *DWS* : L'oeuvre est d'abord destinée pour le cinéma
- *DWP* : L'oeuvre de fiction est conçue originalement pour un média particulier : L'oeuvre est d'abord destinée pour le cinéma
- *Définition Encyclopédique* : Oeuvre de Cinéma
- *Synonyme* : LongMetrageCinema

MagazineComposite

- *Concept Parent* : EmissionCompositeInformative
- *Définition Encyclopédique* : Au sens large : toute émission à structure et périodicité régulière, autour d'un ou plusieurs thèmes. Contenu informatif, lié à une actualité quelconque. A la différence du "Magazine Simple", le "Magazine Composite" contient des reportages en plus de séquence plateau

  Note : Les magazines composés uniquement de rubriques plateau avec, éventuellement, des inserts d'images d'illustration, ne sont pas des émissions composites

MagazinePlateau

- *Concept Parent* : MagazineSimple
- *SWP* : C'est une émission divertissante de type Magazine
- *DWP* : Le magazine se déroule presque exclusivement en plateau
- *Définition Encyclopédique* : Contient typiquement des séquences "Debat" ou "EntretienInterview"

MagazinePlateauInvitesAnonymes

- *Concept Parent* : MagazinePlateau
- *SWP* : C'est un magazine qui se déroule presque exclusivement en plateau
- *SWS* : Des invités participent à l'émission
- *DWS* : Les invités sont anonymes
- *DWP* : Des invités participent à l'émission : Les invités sont anonymes

MagazinePlateauInvitesStars

- *Concept Parent* : MagazinePlateau
- *SWP* : C'est un magazine qui se déroule presque exclusivement en plateau
- *SWS* : Des invités participent à l'émission
- *DWS* : Les invités sont connus
- *DWP* : Des invités participent à l'émission : Les invités sont connus

MagazineSimple

- *Concept Parent* : EmissionSimpleInformative
- *SWP* : C'est une émission informative
- *SWS* : L'émission est typique d'une époque
- *Définition Encyclopédique* : Attention : ne contient pas de "Reportage"
  L'émission est un magazine, construit autour d'un ou de plusieurs thèmes, et le contenu est plutôt informatif ou lié à une actualité quelconque

MessageInfo

- *Concept Parent* : ProgrammationReguliere
- *SWP* : C'est une émission de courte durée qui a une programmation régulière
- *SWS* : Le programme a un statut particulier
- *DWS* : C'est un programme informatif
- *DWP* : Le programme a un statut particulier : C'est un programme informatif
- *Définition Encyclopédique* : Attention : cela peut aussi caractériser les flashs d'information si la chaîne les identifie ainsi.
- *Synonyme* : BulletinInformation

MessagePublicitaire

- *Concept Parent* : SequenceOriginale
- *Définition Encyclopédique* : A n'utiliser que pour décrire une séquence. Pour les films réalisés dans le cadre des campagnes d'information institutionnelles, ayant une composante importante de création, et réalisés par des personnalités parfois extérieures à la publicité, utiliser "Court Métrage" et "Message Info" Exemple : "Scénarios sur la drogue", "3000 scénarios contre un virus" sont des "Court Métrage" Exemple : "spot sur la sécurité routière", "spot d'associations humanitaires" sont des "Message Info"

MicroTrottoir

- *Concept Parent* : SequenceDialogue
- *Définition Encyclopédique* : Succession de courtes interviews enregistrées dans un lieu public (en général dans la rue) pour recueillir les impressions ou les réactions spontanées de personnes, généralement des anonymes

MiniProgramme

- *Concept Parent* : ProgrammationReguliere
- *SWP* : C'est une émission de courte durée qui a une programmation régulière
- *SWS* : Le programme a un statut particulier
- *DWS* : C'est un programme divertissant
- *DWP* : Le programme a un statut particulier : C'est un programme divertissant

MiniRetrospective

- *Concept Parent* : MiniProgramme
- *SWP* : C'est un programme court et divertissant
- *SWS* : Le programme a un objectif particulier
- *DWS* : C'est une micro-rétrospective sur un événement (une épreuve sportive, une course au large, un festival, l'an 2000 ...)
- *DWP* : Le programme a un objectif particulier : C'est une micro-rétrospective sur un événement (une épreuve sportive, une course au large, un festival, l'an 2000 ...)

Montage

- *Concept Parent* : ActivitéProduction
- *SWP* : C'est une activité particulière dans le processus de production
- *SWS* : Constitue une étape clé dans le processus de fabrication de l'objet audiovisuel
- *DWS* : Cette étape fait suite au tournage

- *DWP* : Constitue une étape clé dans le processus de fabrication de l'objet audiovisuel : Cette étape fait suite au tournage

MontageAudio

- *Concept Parent* : Montage

MontageVideo

- *Concept Parent* : Montage

Objet

- *Concept Parent* : AnyConcept

ObjetArchivage

- *Concept Parent* : ObjetAudiovisuel

ObjetAudiovisuel

- *Concept Parent* : Objet

ObjetBiologique

- *Concept Parent* : Objet

ObjetDeDiffusion

- *Concept Parent* : ObjetAudiovisuel
- *Synonyme* : Case dans la grille de programmes
- *Synonyme* : ELP

ObjetDeProduction

- *Concept Parent* : ObjetAudiovisuel

Personne

- *Concept Parent* : ObjetBiologique

PlateauCaracteriseParSaForme

- *Concept Parent* : PlateauStudio

PlateauCaracteriseParSaPosition

- *Concept Parent* : PlateauStudio

PlateauDebut

- *Concept Parent* : PlateauCaracteriseParSaPosition

PlateauEnSituation

- *Concept Parent* : SequencePlateau
- *Définition Encyclopédique* : Le journaliste ou le présentateur est filmé dans un studio autre que celui de l'émission, ou en dehors d'un studio (intérieur ou extérieur)
- *Synonyme* : PlateauExterieur

PlateauFin

- *Concept Parent* : PlateauCaracteriseParSaPosition
- *Synonyme* : DernierPlateau

PlateauImage

- *Concept Parent* : PlateauCaracteriseParSaForme

PlateauInvite

- *Concept Parent* : PlateauCaracteriseParSaForme

PlateauLancement

- *Concept Parent* : PlateauCaracteriseParSaPosition

PlateauStudio

- *Concept Parent* : SequencePlateau

Portrait

- *Concept Parent* : Retrospective
- *SWP* : L'émission est une rétrospective
- *SWS* : L'émission récapitule des faits marquants pour un type d'objet particulier
- *DWS* : L'émission brosse le portrait d'une personne
- *DWP* : L'émission récapitule des faits marquants pour un type d'objet particulier : L'émission brosse le portrait d'une personne

PostProduction

- *Concept Parent* : ActivitéProduction
- *SWP* : C'est une activité particulière dans le processus de production
- *SWS* : Constitue une étape clé dans le processus de fabrication de l'objet audiovisuel
- *DWS* : Cette étape conclue le processus de fabrication de l'objet audiovisuel
- *DWP* : Constitue une étape clé dans le processus de fabrication de l'objet audiovisuel : Cette étape conclue le processus de fabrication de l'objet audiovisuel

PresseFilmee

- *Concept Parent* : EmissionSimpleInformative
- *SWP* : C'est une émission informative
- *SWS* : L'émission est typique d'une époque
- *DWS* : L'émission est ancienne et n'a plus cours
- *DWP* : L'émission est typique d'une époque : L'émission est ancienne et n'a plus cours
- *Définition Encyclopédique* : Typiquement, les Actualités Françaises
- *Synonyme* : ActualitesFrancaises

PrevisionsMeteo

- *Concept Parent* : MessageInfo
- *SWP* : C'est un programme court et informatif
- *SWS* : Le programme donne une certaine information
- *DWS* : L'information concerne les prévisions Météo et l'éphéméride
- *DWP* : Le programme donne une certaine information : L'information concerne les prévisions Météo et l'éphéméride

PriseDeSon

- *Concept Parent* : Tournage
- *SWP* : Fait partie des activités de tournage dans la production
- *SWS* : Consiste à saisir/capter une des composantes de l'objet audiovisuel
- *DWS* : Concerne uniquement les éléments de l'objet audiovisuel pris avec un micro : rapport à l'ambiance sonore
- *DWP* : Consiste à saisir/capter une des composantes de l'objet audiovisuel : Concerne uniquement les éléments de l'objet audiovisuel pris avec un micro : rapport à l'ambiance sonore

PriseDeVue

- *Concept Parent* : Tournage
- *SWP* : Fait partie des activités de tournage dans la production
- *SWS* : Consiste à saisir/capter une des composantes de l'objet audiovisuel
- *DWS* : Concerne uniquement les éléments de l'objet audiovisuel pris avec une caméra : rapport à l'image
- *DWP* : Consiste à saisir/capter une des composantes de l'objet audiovisuel : Concerne uniquement les éléments de l'objet audiovisuel pris avec une caméra : rapport à l'image

ProgrammationReguliere

- *Concept Parent* : EmissionFormatCourt

- *SWP* : C'est une émission de courte durée
- *SWS* : L'émission est caractérisée par sa programmation
- *DWS* : L'émission a une programmation régulière, à heure et à jour fixe
- *DWP* : L'émission est caractérisée par sa programmation : L'émission a une programmation régulière, à heure et à jour fixe

ProgrammeABaseDeClips

- *Concept Parent* : EmissionCompositeDivertissante
- *Définition Encyclopédique* : Emission composée pour l'essentiel de clips intégraux ou d'extraits avec, éventuellement, une présentation et de brèves interventions d'invités

ProgrammeAtypique

- *Concept Parent* : EmissionComposite
- *Définition Encyclopédique* : Emission qui ne relève pas d'un autre genre. Caractéristique des émissions qui mélangent les genres et notamment des émissions qui mêlent spectacle, débat, l'interview et sujets ou reportages. C'est notamment le cas des programmes spéciaux comme les soirées caritatives, les remises de prix, les nuits consacrées à un sport, à un art du spectacle, à la publicité, au cinéma..., les émissions de fin d'année
- *Synonyme* : Création Originale

PropriétéDiffusion

- *Concept Parent* : Qualia

Qualia

- *Concept Parent* : AnyConcept

RealityShow

- *Concept Parent* : EmissionCompositeDivertissante
- *Définition Encyclopédique* : Mise en spectacle par reconstitution visuelle ou verbale d'événements de personnes privées. Elle peut comprendre : - des reconstitutions de faits divers interprétés par leurs protagonistes ou par des acteurs, - des témoignages de citoyens ordinaires exposant des situations difficiles, illustrés ou non par des reportages

Recit

- *Concept Parent* : Retrospective
- *SWP* : L'émission est une rétrospective
- *SWS* : L'émission récapitule des faits marquants pour un type d'objet particulier
- *DWS* : L'émission retrace le récit d'un événement. Le conteur est présent ou non à l'image
- *DWP* : L'émission récapitule des faits marquants pour un type d'objet particulier : L'émission retrace le récit d'un événement. Le conteur est présent ou non à l'image

Reportage

- *Concept Parent* : SequenceMontee
- *Définition Encyclopédique* : Oeuvre exclusive d'un journaliste qui couvre un événement sur place ou rend compte d'une enquête sur un sujet de société, économique, politique, social ou lié à la vie quotidienne. Le reportage doit être monté

  Attention : ne pas combiner avec "Documentaire" !

ResultatsJeux

- *Concept Parent* : MiniProgramme
- *SWP* : C'est un programme court et divertissant
- *SWS* : Le programme a un objectif particulier
- *DWS* : Le programme donne les résultats des jeux de la Française des Jeux
- *DWP* : Le programme a un objectif particulier : Le programme donne les résultats des jeux de la Française des Jeux

Retrospective

- *Concept Parent* : Documentaire
- *SWP* : L'émission est de genre documentaire
- *DWP* : L'émission récapitule les principaux faits ou étapes marquant le déroulement de la vie d'une personne physique ou morale, ou d'un événement

RevueDePresse

- *Concept Parent* : SequenceCaracteriseeParSonContenu
- *Définition Encyclopédique* : Présentation de l'éventail des opinions et commentaires récemment diffusés par différents médias (presse écrite généralement), sur un fait d'actualité ou dans un domaine de l'actualité

Séquence

- *Concept Parent* : ObjetDeProduction

SequenceCaracteriseeParSonContenu

- *Concept Parent* : SequenceNonMontee

SequenceDialogue

- *Concept Parent* : SequenceCaracteriseeParSonContenu

SequenceExtrait

- *Concept Parent* : SequenceNonMontee

SequenceInformative

- *Concept Parent* : Séquence

SequenceMontee

- *Concept Parent* : SequenceInformative

SequenceNonMontee

- *Concept Parent* : SequenceInformative

SequenceOriginale

- *Concept Parent* : Séquence

SequencePlateau

- *Concept Parent* : SequenceNonMontee

SequencesArchives

- *Concept Parent* : SequenceMontee

Sommaire

- *Concept Parent* : SequenceCaracteriseeParSonContenu
- *Synonyme* : PlateauDebut

SpectacleTV

- *Concept Parent* : EmissionSimpleDivertissante
- *SWP* : C'est une émission divertissante
- *SWS* : L'émission a un genre particulier
- *DWS* : L'émission est un spectacle conçu et organisé pour la télévision
- *DWP* : L'émission a un genre particulier : L'émission est un spectacle conçu et organisé pour la télévision
- *Définition Encyclopédique* : Note : il implique la présence d'un public

TelefilmIsole

- *Concept Parent* : FictionLongueDuree
- *SWP* : C'est une oeuvre de fiction de plus de 60 minutes
- *SWS* : L'oeuvre de fiction est conçue originalement pour un média particulier
- *DWS* : L'oeuvre est d'abord destinée pour la télévision

- *DWP* : L'oeuvre de fiction est conçue originalement pour un média particulier : L'oeuvre est d'abord destinée pour la télévision
- *Définition Encyclopédique* : Oeuvre de Télévision Fiction basée sur des problèmes de société ou mettant en scène un milieu particulier, et diffusée en général en début de soirée (prime time)

TelefilmRecurrent

- *Concept Parent* : EpisodeLong
- *SWP* : C'est un épisode de plus de 52 minutes
- *SWS* : Les épisodes de la série sont diffusables isolément ou pas
- *DWS* : L'épisode est clos, c'est-à-dire qu'il forme un tout cohérent et est diffusable isolément. Il y a une thématique propre à chaque numéro, basée sur des problèmes de société ou sur un personnage récurrent ou mettant en scène un milieu particulier
- *DWP* : Les épisodes de la série sont diffusables isolément ou pas : L'épisode est clos, c'est-à-dire qu'il forme un tout cohérent et est diffusable isolément. Il y a une thématique propre à chaque numéro, basée sur des problèmes de société ou sur un personnage récurrent ou mettant en scène un milieu particulier

Temoignage

- *Concept Parent* : EntretienInterview
- *Définition Encyclopédique* : Interview d'une (ou plusieurs) personne(s) sur un événement qu'elle(s) a (ont) personnellement vu, entendu ou vécu. Dans certains cas, l'événement est en cours.

  Attention : le terme peut être combiné avec "MicroTrottoir" !

Theme

- *Concept Parent* : Qualia

Tournage

- *Concept Parent* : ActivitéProduction
- *SWP* : C'est une activité particulière dans le processus de production
- *SWS* : Constitue une étape clé dans le processus de fabrication de l'objet audiovisuel
- *DWS* : Cette étape est un préalable aux suivantes
- *DWP* : Constitue une étape clé dans le processus de fabrication de l'objet audiovisuel : Cette étape est un préalable aux suivantes

TrafficRoutier

- *Concept Parent* : MessageInfo
- *SWP* : C'est un programme court et informatif
- *SWS* : Le programme donne une certaine information
- *DWS* : L'information concerne l'état du traffic routier
- *DWP* : Le programme donne une certaine information : L'information concerne l'état du traffic routier

VideoClip

- *Concept Parent* : SequenceOriginale
- *Définition Encyclopédique* : Chanson ou morceau de musique filmés

Annexe C

Ontologie du cyclisme

C.1 Taxinomies des concepts avec les axes différentiels

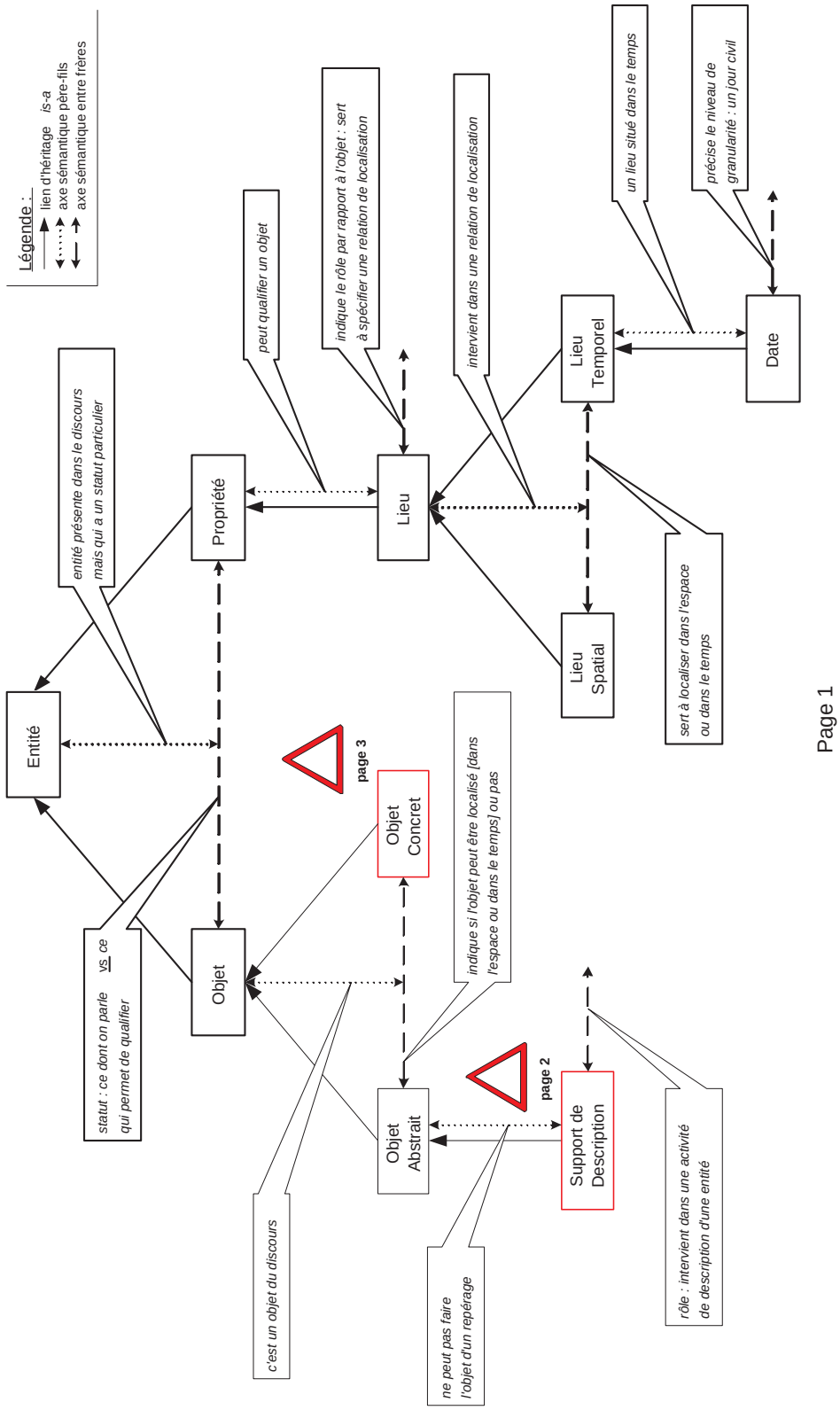


FIG. C.1 – *Taxinomie des concepts dans l'ontologie du cyclisme - Page 1/7*

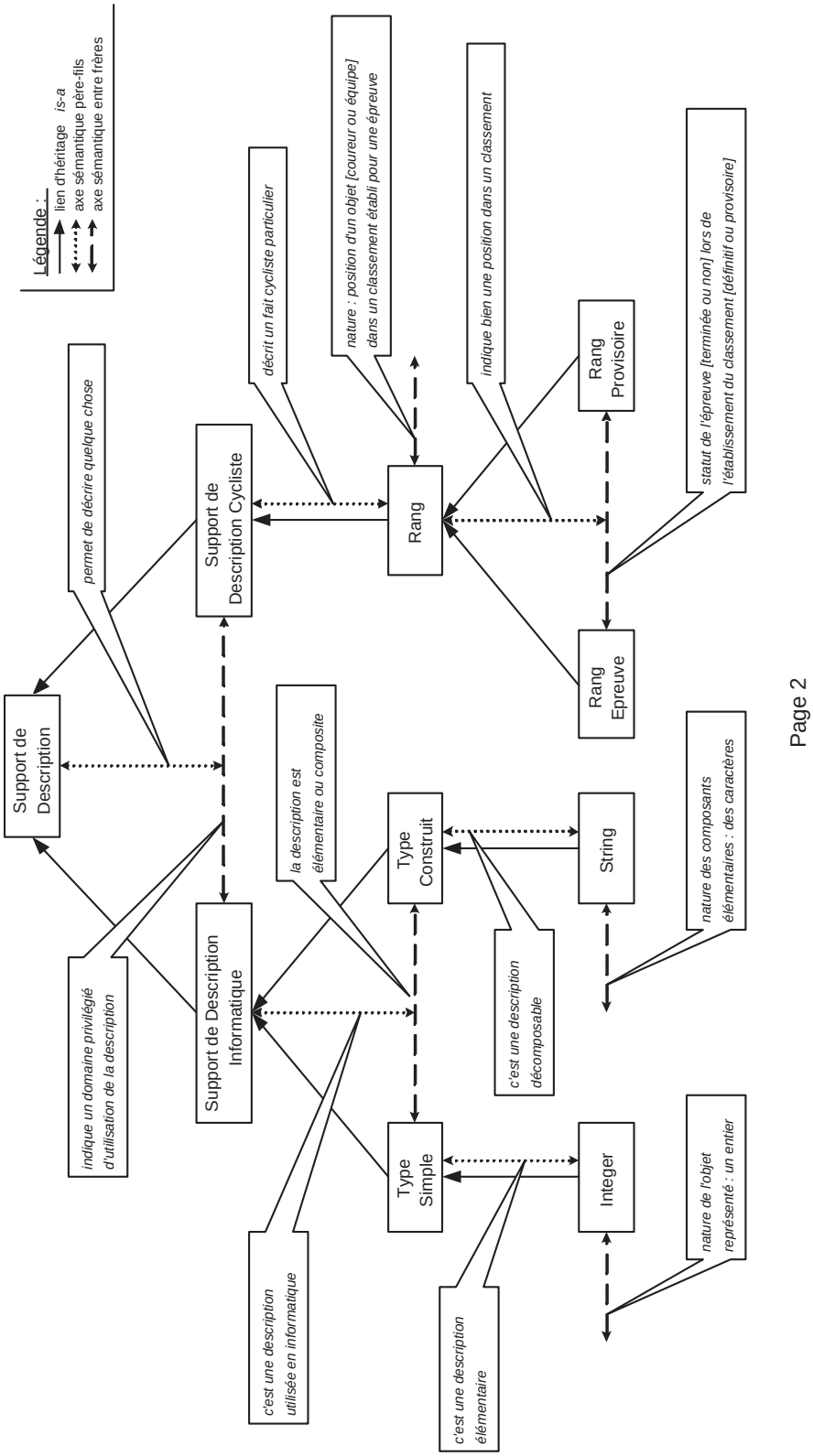


FIG. C.2 – *Taxinomie des concepts dans l'ontologie du cyclisme - Page 2/7*

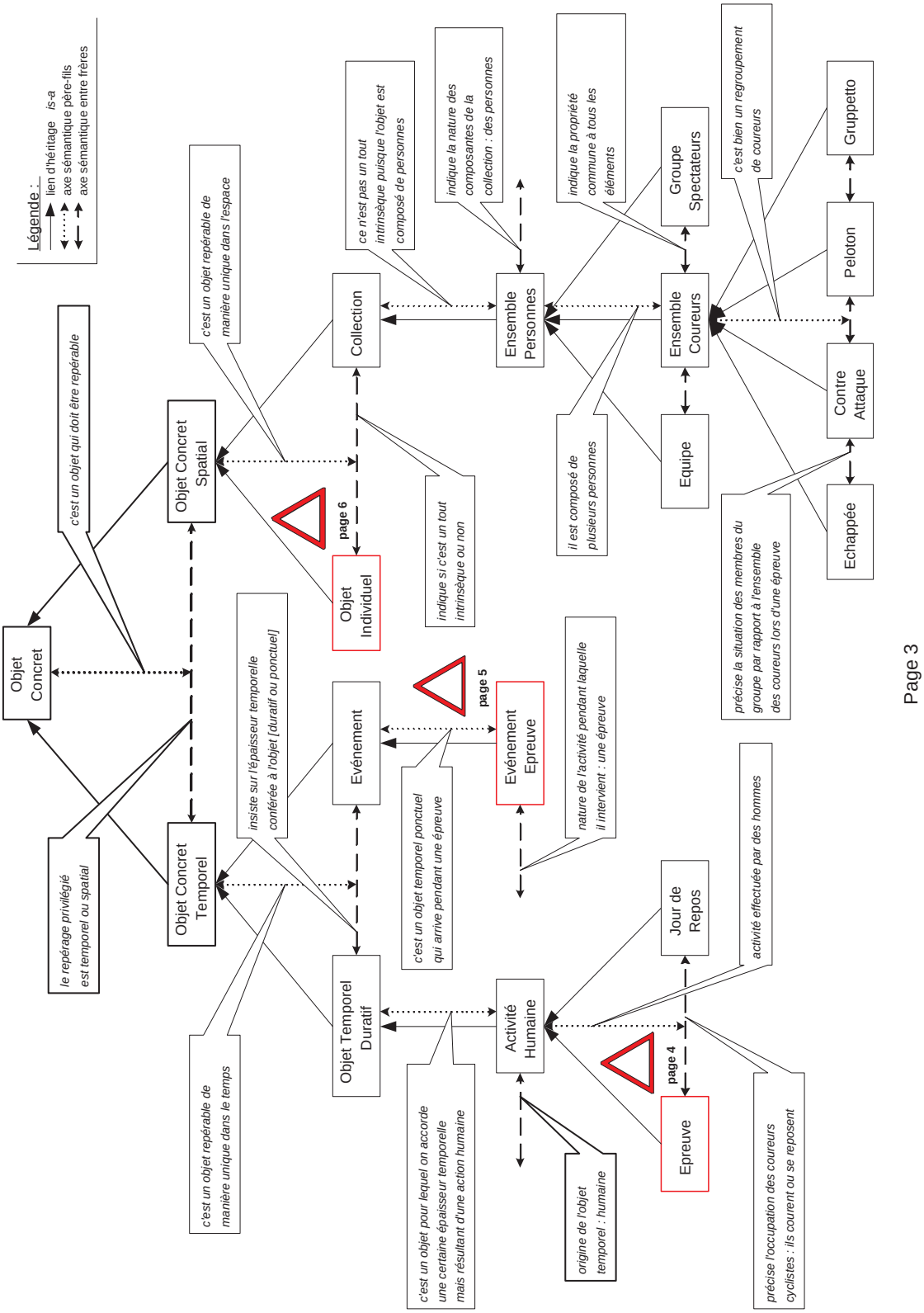


FIG. C.3 – *Taxinomie des concepts dans l'ontologie du cyclisme - Page 3/7*

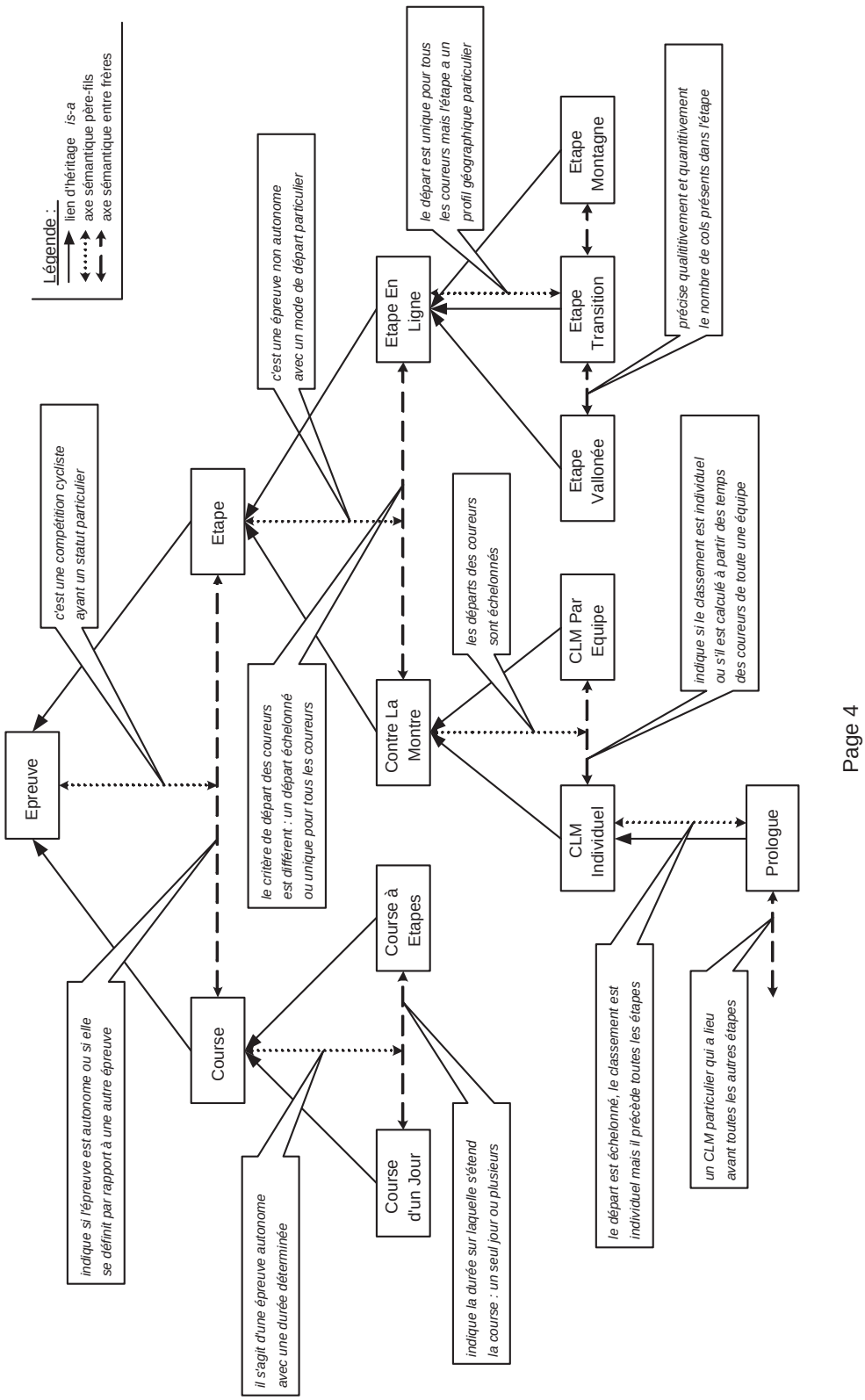


FIG. C.4 – *Taxinomie des concepts dans l'ontologie du cyclisme - Page 4/7*

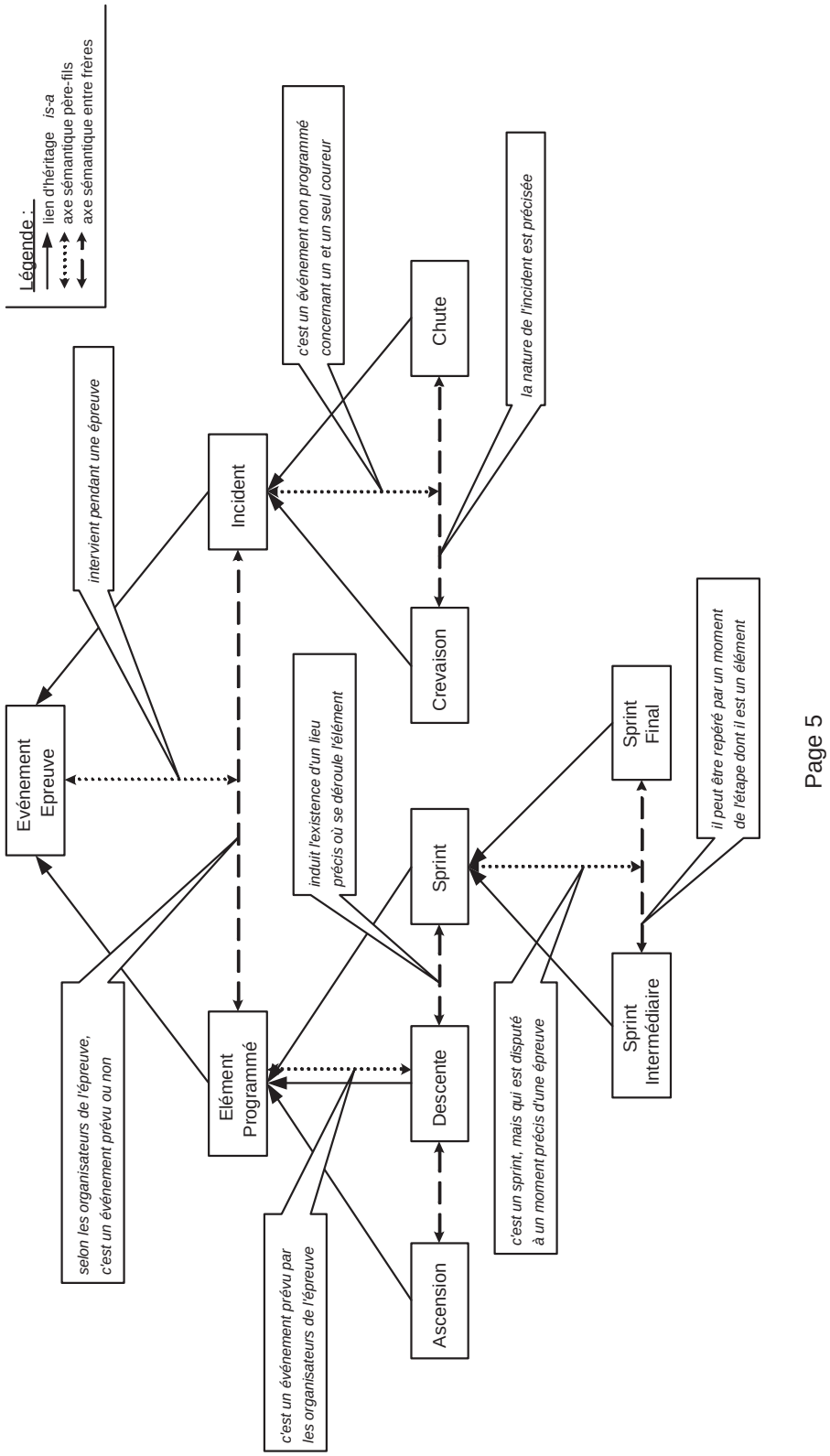


FIG. C.5 – *Taxinomie des concepts dans l'ontologie du cyclisme - Page 5/7*

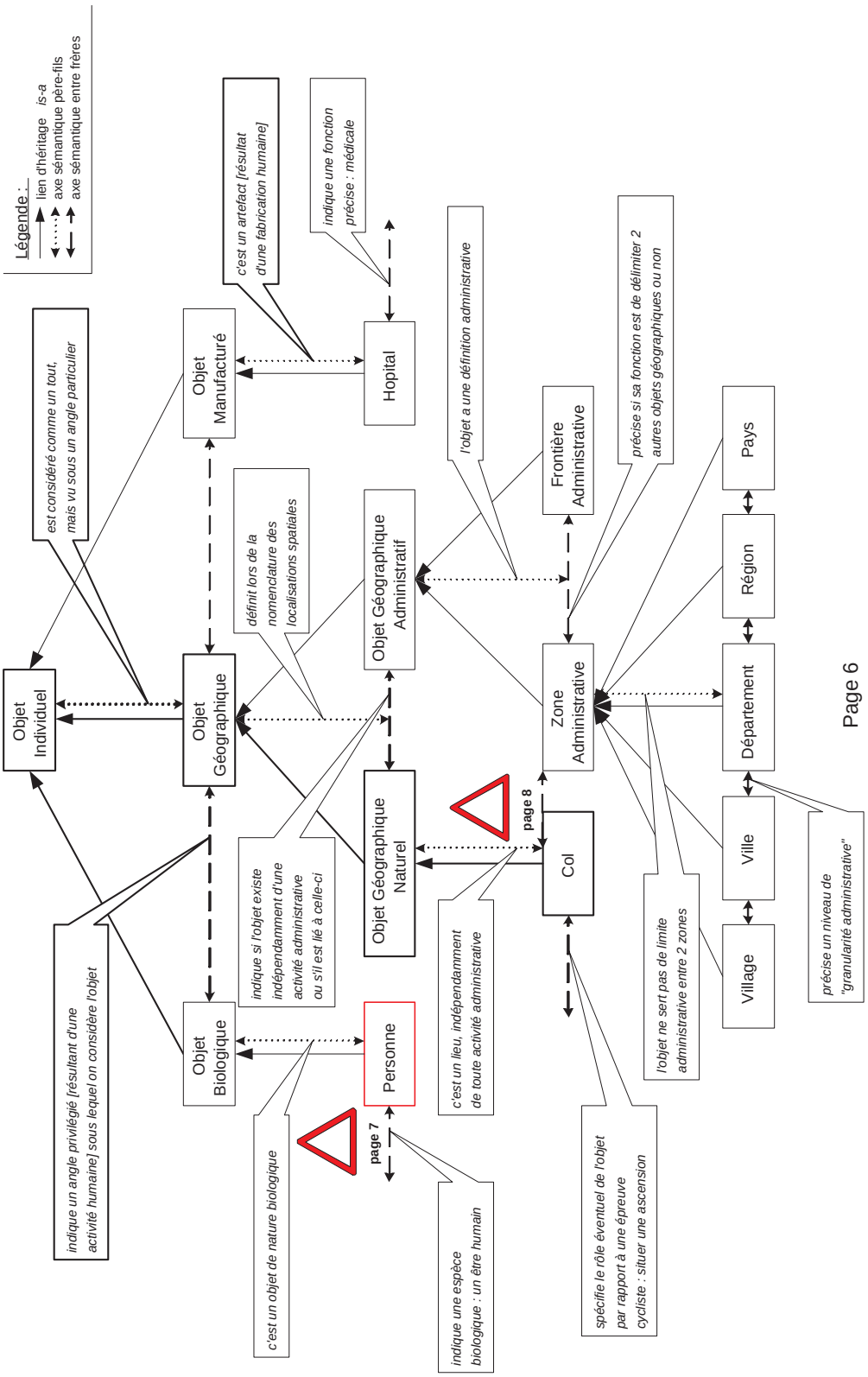


FIG. C.6 – *Taxinomie des concepts dans l'ontologie du cyclisme - Page 6/7*

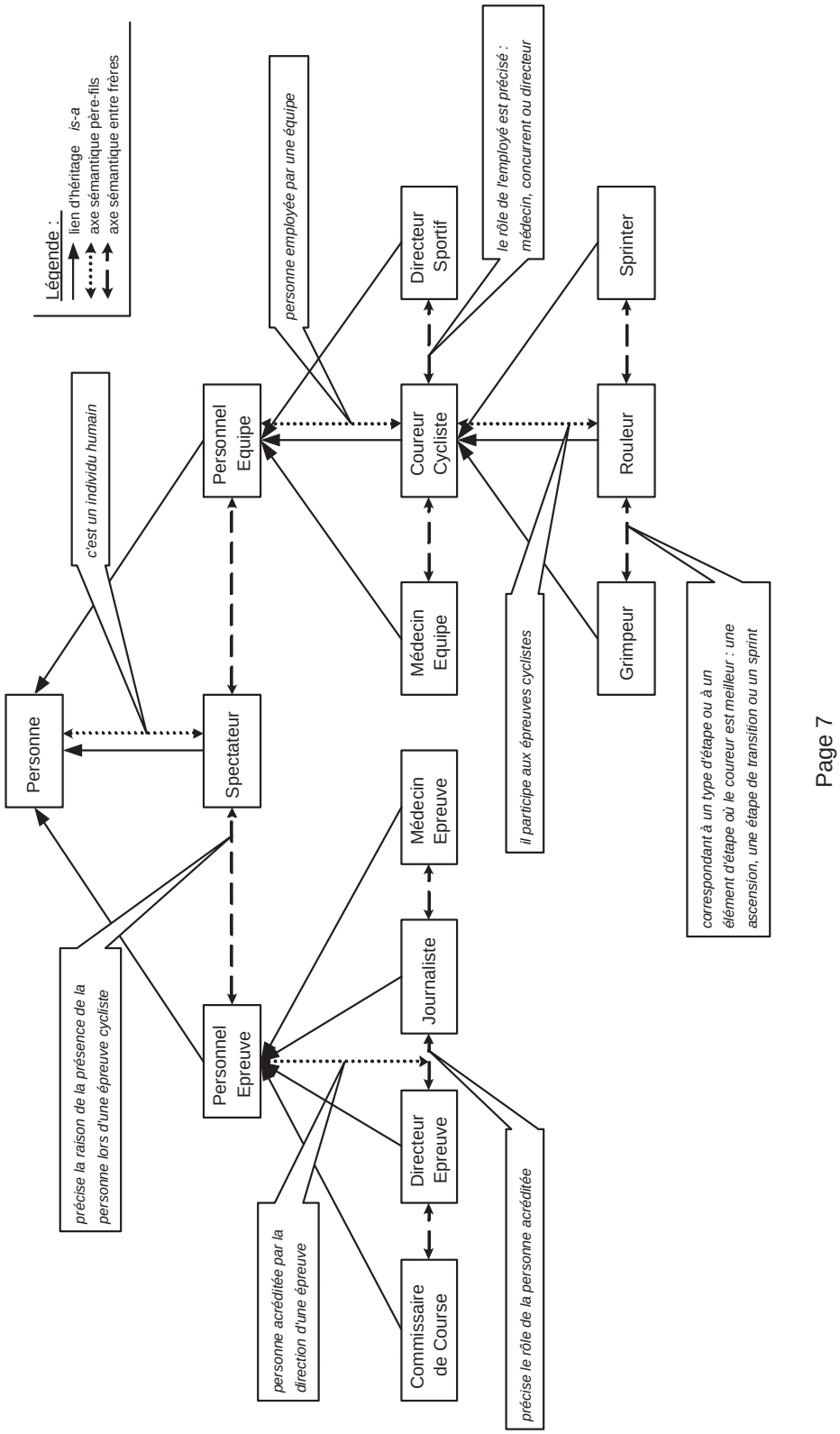


FIG. C.7 – *Taxinomie des concepts dans l'ontologie du cyclisme - Page 7/7*

C.2 Taxinomies des relations avec les axes différentiels

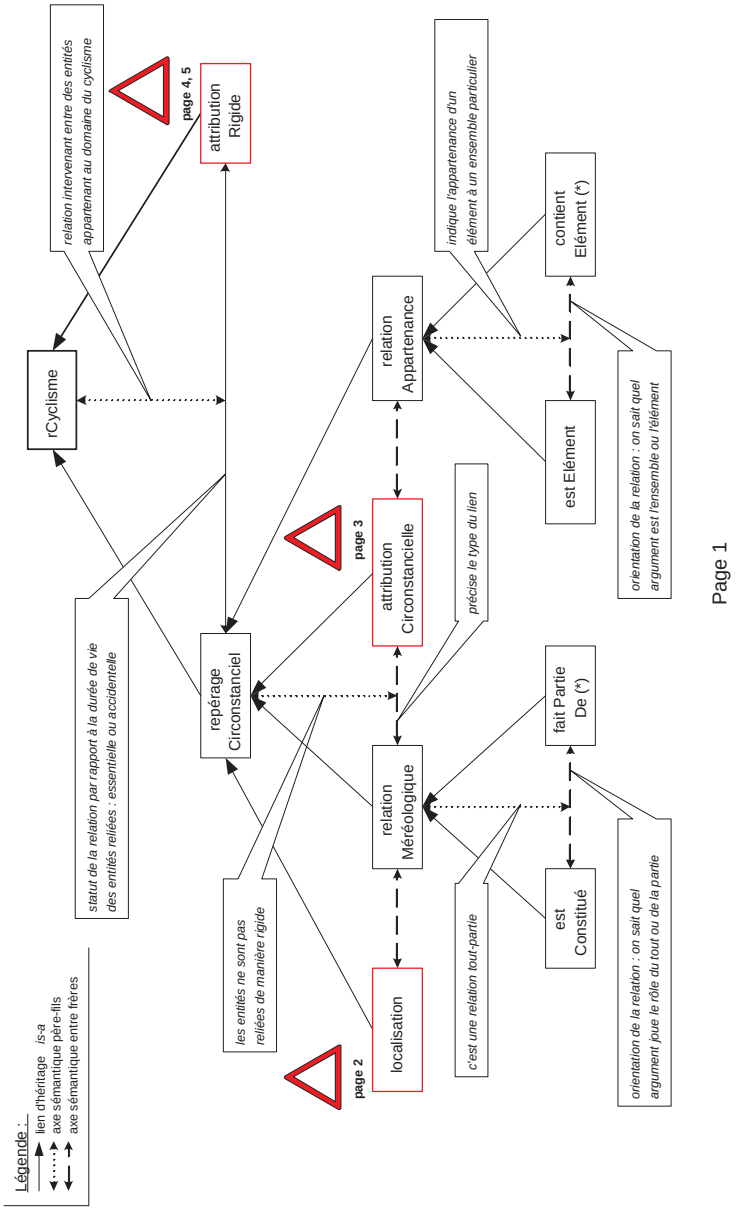


FIG. C.8 – *Taxinomie des relations dans l'ontologie du cyclisme - Page 1/5*

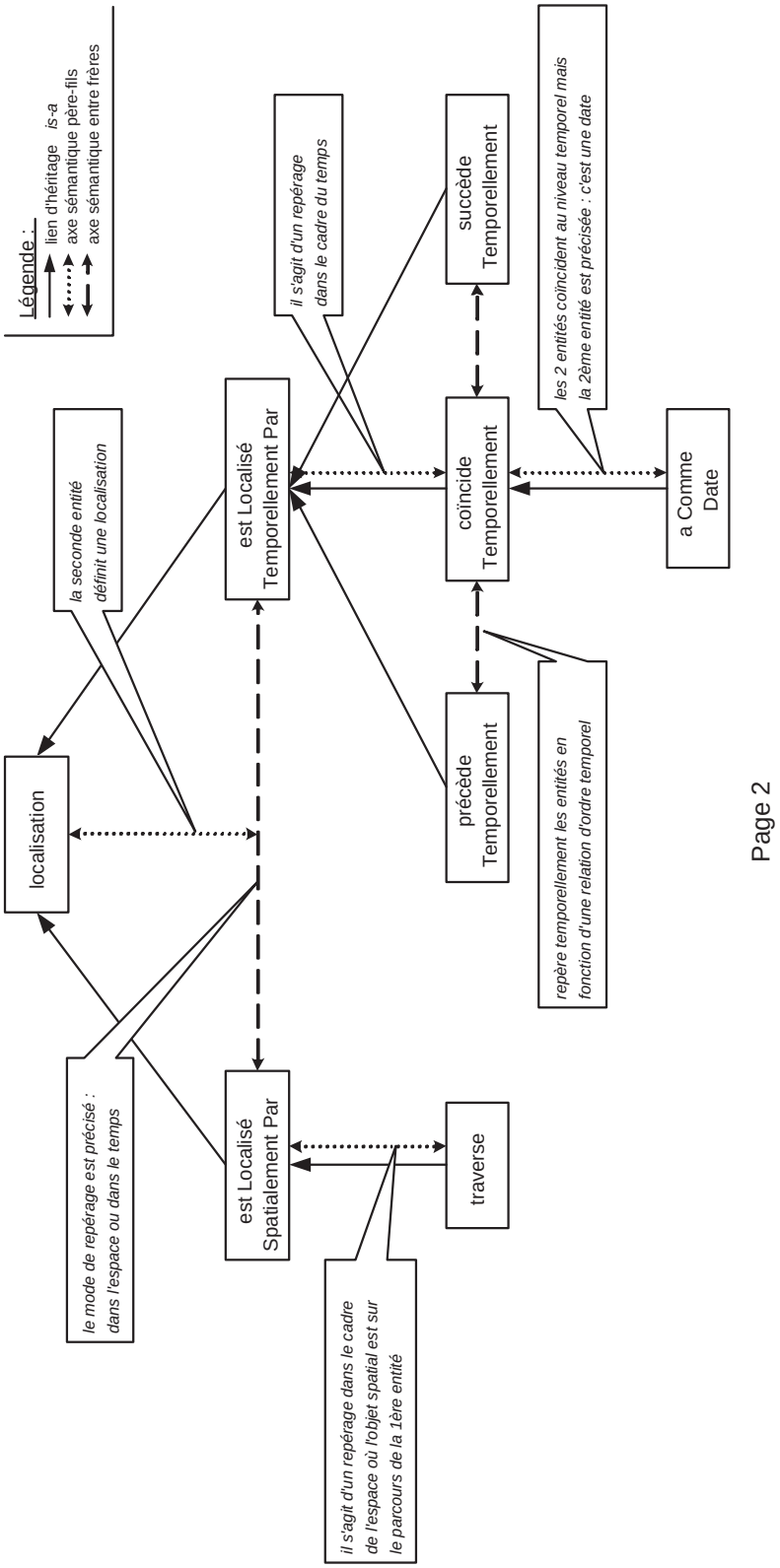


FIG. C.9 – *Taxinomie des relations dans l'ontologie du cyclisme - Page 2/5*

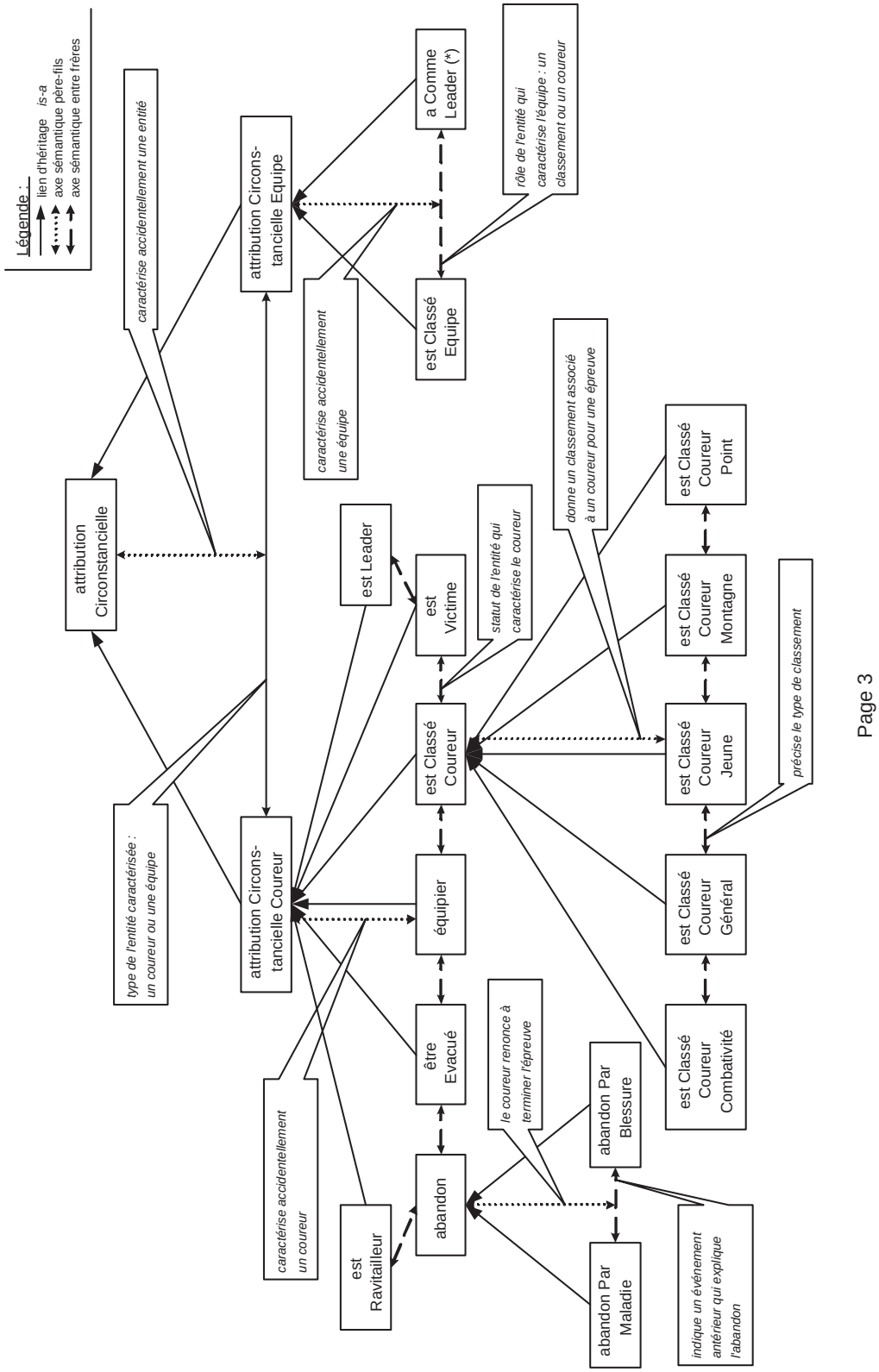


FIG. C.10 – *Taxinomie des relations dans l'ontologie du cyclisme - Page 3/5*

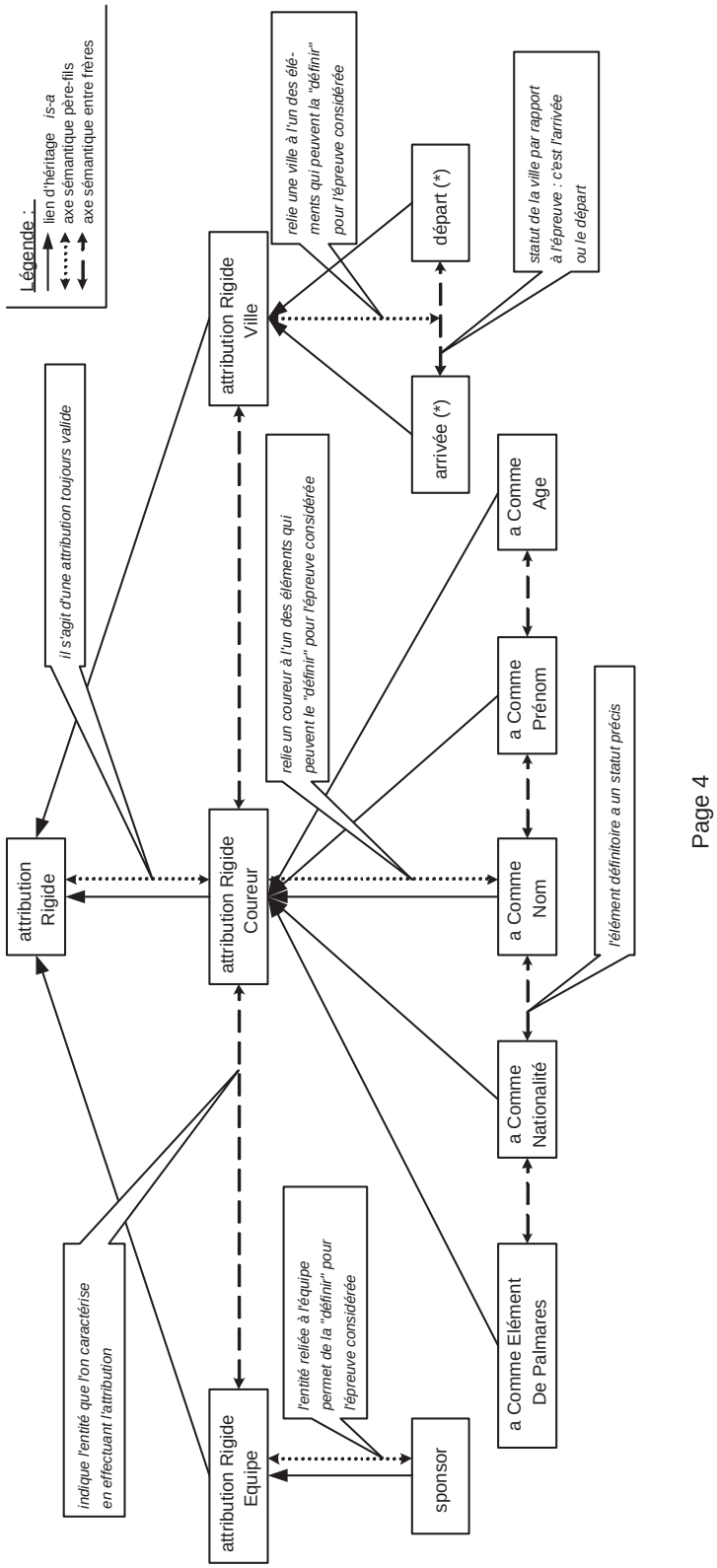


FIG. C.11 – *Taxinomie des relations dans l'ontologie du cyclisme - Page 4/5*

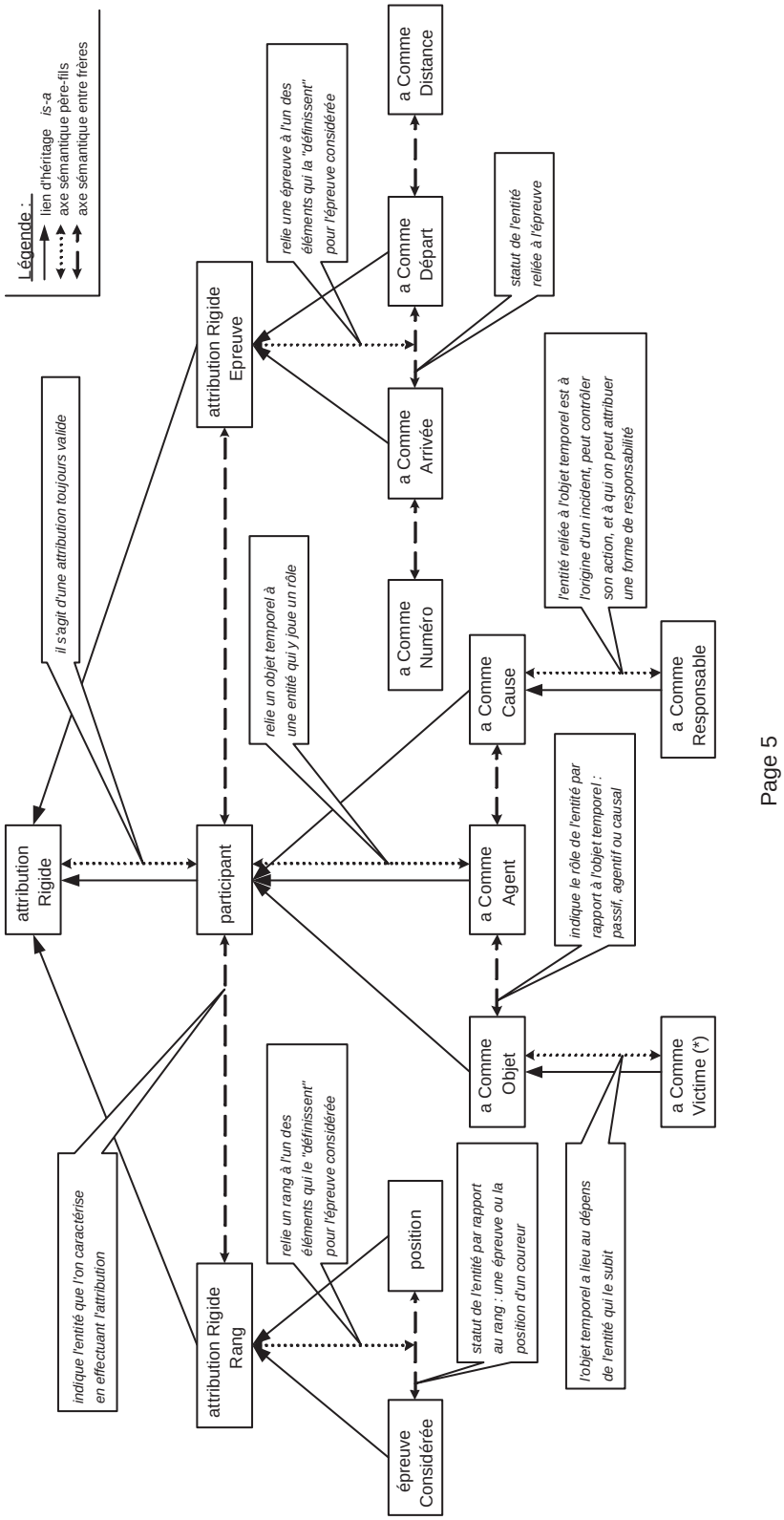


FIG. C.12 – *Taxinomie des relations dans l'ontologie du cyclisme - Page 5/5*

C.3 Ontologie formalisée

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<rdf:RDF xmlns:dc="http://purl.org/dc/elements/1.1/"
    xmlns:owl="http://www.w3.org/2002/07/owl#"
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
    xmlns:xsd="http://www.w3.org/2000/10/XMLSchema#">
    <owl:Ontology rdf:about="">
        <dc:title>Ontologie du cyclisme</dc:title>
        <dc:creator>Raphaël Troncy</dc:creator>
        <dc:description>Ontologie pour le Tour de France Cycliste de 1999
        </dc:description>
        <dc:subject>Classes, Roles, Individus et Axiomes nécessaires</dc:subject>
    </owl:Ontology>
    <owl:Class rdf:about="#Journaliste">
        <rdfs:label>Journaliste</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#PersonnelEpreuve"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#PersonnelEquipe">
        <rdfs:label>PersonnelEquipe</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#Personne"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#SprintIntermediaire">
        <rdfs:label>SprintIntermediaire</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#Sprint"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#CourseAEtapes">
        <rdfs:label>CourseAEtapes</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#Course"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#ObjetIndividuel">
        <rdfs:label>ObjetIndividuel</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ObjetConcretSpatial"/>
        </rdfs:subClassOf>
    </owl:Class>
```

Code C.6 – *Ontologie formelle du cyclisme en OWL : les concepts*

```
<owl:Class rdf:about="#ObjetConcret">
        <rdfs:label>ObjetConcret</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#Objet"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#EnsemblePersonnes">
        <rdfs:label>EnsemblePersonnes</rdfs:label>
        <owl:sameClassAs>
            <rdfs:Class>
                <owl:intersectionOf rdf:parseType="owl:collection">
                    <owl:Class rdf:about="#Collection"/>
                    <owl:Restriction>
                        <owl:onProperty rdf:resource="#contientElement"/>
                        <owl:minCardinality>1</owl:minCardinality>
                    </owl:Restriction>
                </owl:intersectionOf>
            </rdfs:Class>
        </owl:sameClassAs>
    </owl:Class>
    <owl:Class rdf:about="#ObjetTemporelDuratif">
        <rdfs:label>ObjetTemporelDuratif</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ObjetConcretTemporel"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Sprinter">
        <rdfs:label>Sprinter</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#CoureurCycliste"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#CommissaireDeCourse">
        <rdfs:label>CommissaireDeCourse</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#PersonnelEpreuve"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#FrontiereAdministrative">
        <rdfs:label>FrontiereAdministrative</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ObjetGeographiqueAdministratif"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#ObjetAbstrait">
        <rdfs:label>ObjetAbstrait</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#Objet"/>
        </rdfs:subClassOf>
    </owl:Class>
```

Code C.6 – *Ontologie formelle du cyclisme en OWL : les concepts* **(suite)**

```
<owl:Class rdf:about="#Village">
        <rdfs:label>Village</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ZoneAdministrative"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Departement">
        <rdfs:label>Departement</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ZoneAdministrative"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#ContreLaMontre">
        <rdfs:label>ContreLaMontre</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#Etape"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#ObjetGeographiqueAdministratif">
        <rdfs:label>ObjetGeographiqueAdministratif</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ObjetGeographique"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#EtapeVallonee">
        <rdfs:label>EtapeVallonee</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#EtapeEnLigne"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#ActiviteHumaine">
        <rdfs:label>ActiviteHumaine</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ObjetTemporelDuratif"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Course">
        <rdfs:label>Course</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#Epreuve"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Crevaison">
        <rdfs:label>Crevaison</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#Incident"/>
        </rdfs:subClassOf>
    </owl:Class>
```

Code C.6 – *Ontologie formelle du cyclisme en OWL : les concepts* **(suite)**

```
<owl:Class rdf:about="#Maillot">
        <rdfs:label>Maillot</rdfs:label>
        <owl:sameClassAs>
            <rdfs:Class>
                <owl:intersectionOf rdf:parseType="owl:collection">
                    <owl:Class rdf:about="#CoureurCycliste"/>
                    <owl:Restriction>
                        <owl:onProperty rdf:resource="#estClasseCoureur"/>
                        <owl:someValuesFrom>
                            <owl:Class>
                                <owl:intersectionOf parseType="owl:collection"/>
                            </owl:Class>
                        </owl:someValuesFrom>
                    </owl:Restriction>
                </owl:intersectionOf>
            </rdfs:Class>
        </owl:sameClassAs>
    </owl:Class>
    <owl:Class rdf:about="#Entite">
        <rdfs:label>Entite</rdfs:label>
    </owl:Class>
    <owl:Class rdf:about="#Equipe">
        <rdfs:label>Equipe</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#EnsemblePersonnes"/>
        </rdfs:subClassOf>
        <rdfs:subClassOf>
            <owl:Restriction>
                <owl:onProperty rdf:resource="#sponsor"/>
                <owl:cardinality>1</owl:cardinality>
            </owl:Restriction>
        </rdfs:subClassOf>
        <rdfs:subClassOf>
            <owl:Restriction>
                <owl:onProperty rdf:resource="#aCommeLeader"/>
                <owl:cardinality>1</owl:cardinality>
            </owl:Restriction>
        </rdfs:subClassOf>
        <rdfs:subClassOf>
            <owl:Restriction>
                <owl:onProperty rdf:resource="#contientElement"/>
                <owl:cardinality>9</owl:cardinality>
            </owl:Restriction>
        </rdfs:subClassOf>
    </owl:Class>
```

Code C.6 – *Ontologie formelle du cyclisme en OWL : les concepts* **(suite)**

```
<owl:Class rdf:about="#Date">
        <rdfs:label>Date</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#LieuTemporel"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Lieu">
        <rdfs:label>Lieu</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#Propriete"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Pays">
        <rdfs:label>Pays</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ZoneAdministrative"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Rang">
        <rdfs:label>Rang</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#SupportDeDescriptionCycliste"/>
        </rdfs:subClassOf>
        <rdfs:subClassOf>
            <owl:Restriction>
                <owl:onProperty rdf:resource="#position"/>
                <owl:cardinality>1</owl:cardinality>
            </owl:Restriction>
        </rdfs:subClassOf>
        <rdfs:subClassOf>
            <owl:Restriction>
                <owl:onProperty rdf:resource="#epreuveConsideree"/>
                <owl:cardinality>1</owl:cardinality>
            </owl:Restriction>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#EtapeTransition">
        <rdfs:label>EtapeTransition</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#EtapeEnLigne"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Prologue">
        <rdfs:label>Prologue</rdfs:label>
        <owl:sameClassAs>
            <owl:Class rdf:about="#CLMIndividuel"/>
        </owl:sameClassAs>
    </owl:Class>
```

Code C.6 – *Ontologie formelle du cyclisme en OWL : les concepts* **(suite)**

```
<owl:Class rdf:about="#DirecteurSportif">
        <rdfs:label>DirecteurSportif</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#PersonnelEquipe"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Hopital">
        <rdfs:label>Hopital</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ObjetManufacture"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Region">
        <rdfs:label>Region</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ZoneAdministrative"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#CoureurCycliste">
        <rdfs:label>CoureurCycliste</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#PersonnelEquipe"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Sprint">
        <rdfs:label>Sprint</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ElementProgramme"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#String">
        <rdfs:label>String</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#TypeConstruit"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#EtapeEnLigne">
        <rdfs:label>EtapeEnLigne</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#Etape"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#SupportDeDescription">
        <rdfs:label>SupportDeDescription</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ObjetAbstrait"/>
        </rdfs:subClassOf>
    </owl:Class>
```

Code C.6 – *Ontologie formelle du cyclisme en OWL : les concepts* **(suite)**

```
<owl:Class rdf:about="#Rouleur">
        <rdfs:label>Rouleur</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#CoureurCycliste"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#PersonnelEpreuve">
        <rdfs:label>PersonnelEpreuve</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#Personne"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#JourDeRepos">
        <rdfs:label>JourDeRepos</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ActiviteHumaine"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#ElementProgramme">
        <rdfs:label>ElementProgramme</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#EvenementEpreuve"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Collection">
        <rdfs:label>Collection</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ObjetConcretSpatial"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#DirecteurEpreuve">
        <rdfs:label>DirecteurEpreuve</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#PersonnelEpreuve"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#MaillotBlanc">
        <rdfs:label>MaillotBlanc</rdfs:label>
        <owl:sameClassAs>
            <rdfs:Class>
                <owl:intersectionOf rdf:parseType="owl:collection">
                    <owl:Class rdf:about="#Maillot"/>
                    <owl:Restriction>
                        <owl:onProperty rdf:resource="#estClasseCoureurJeune"/>
                        <owl:someValuesFrom>
                            <owl:Class>
                                <owl:intersectionOf parseType="owl:collection"/>
```

Code C.6 – *Ontologie formelle du cyclisme en OWL : les concepts* **(suite)**

```
</owl:Class>
                        </owl:someValuesFrom>
                    </owl:Restriction>
                </owl:intersectionOf>
            </rdfs:Class>
        </owl:sameClassAs>
    </owl:Class>
    <owl:Class rdf:about="#MaillotJaune">
        <rdfs:label>MaillotJaune</rdfs:label>
        <owl:sameClassAs>
            <rdfs:Class>
                <owl:intersectionOf rdf:parseType="owl:collection">
                    <owl:Class rdf:about="#Maillot"/>
                    <owl:Restriction>
                        <owl:onProperty rdf:resource="#estClasseCoureurGeneral"/>
                        <owl:someValuesFrom>
                            <owl:Class>
                                <owl:intersectionOf parseType="owl:collection"/>
                            </owl:Class>
                        </owl:someValuesFrom>
                    </owl:Restriction>
                </owl:intersectionOf>
            </rdfs:Class>
        </owl:sameClassAs>
    </owl:Class>
    <owl:Class rdf:about="#GroupeSpectateurs">
        <rdfs:label>GroupeSpectateurs</rdfs:label>
        <owl:sameClassAs>
            <rdfs:Class>
                <owl:intersectionOf rdf:parseType="owl:collection">
                    <owl:Class rdf:about="#EnsemblePersonnes"/>
                    <owl:Restriction>
                        <owl:onProperty rdf:resource="#contientElement"/>
                        <owl:minCardinality>2</owl:minCardinality>
                    </owl:Restriction>
                    <owl:Restriction>
                        <owl:onProperty rdf:resource="#contientElement"/>
                        <owl:alLValuesFrom>
                            <owl:Class rdf:about="#Spectateur"/>
                        </owl:alLValuesFrom>
                    </owl:Restriction>
                </owl:intersectionOf>
            </rdfs:Class>
        </owl:sameClassAs>
    </owl:Class>
```

Code C.6 – *Ontologie formelle du cyclisme en OWL : les concepts* **(suite)**

```
<owl:Class rdf:about="#RangEpreuve">
        <rdfs:label>RangEpreuve</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#Rang"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#MedecinEquipe">
        <rdfs:label>MedecinEquipe</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#PersonnelEquipe"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#TypeSimple">
        <rdfs:label>TypeSimple</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#SupportDeDescriptionInformatique"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#EtapeMontagne">
        <rdfs:label>EtapeMontagne</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#EtapeEnLigne"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#ObjetManufacture">
        <rdfs:label>ObjetManufacture</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ObjetIndividuel"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#SupportDeDescriptionCycliste">
        <rdfs:label>SupportDeDescriptionCycliste</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#SupportDeDescription"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Chute">
        <rdfs:label>Chute</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#Incident"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Etape">
        <rdfs:label>Etape</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#Epreuve"/>
        </rdfs:subClassOf>
    </owl:Class>
```

Code C.6 – *Ontologie formelle du cyclisme en OWL : les concepts* **(suite)**

```
<owl:Class rdf:about="#Objet">
        <rdfs:label>Objet</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#Entite"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Ville">
        <rdfs:label>Ville</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ZoneAdministrative"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Integer">
        <rdfs:label>Integer</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#TypeSimple"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#EvenementEpreuve">
        <rdfs:label>EvenementEpreuve</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#Evenement"/>
        </rdfs:subClassOf>
        <rdfs:subClassOf>
            <owl:Restriction>
                <owl:onProperty rdf:resource="#localisation"/>
                <owl:cardinality>1</owl:cardinality>
            </owl:Restriction>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#ObjetConcretSpatial">
        <rdfs:label>ObjetConcretSpatial</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ObjetConcret"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#ObjetBiologique">
        <rdfs:label>ObjetBiologique</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ObjetIndividuel"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Incident">
        <rdfs:label>Incident</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#EvenementEpreuve"/>
        </rdfs:subClassOf>
    </owl:Class>
```

Code C.6 – *Ontologie formelle du cyclisme en OWL : les concepts* **(suite)**

```
<owl:Class rdf:about="#ObjetConcretTemporel">
        <rdfs:label>ObjetConcretTemporel</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ObjetConcret"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#ObjetGeographiqueNaturel">
        <rdfs:label>ObjetGeographiqueNaturel</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ObjetGeographique"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#LieuTemporel">
        <rdfs:label>LieuTemporel</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#Lieu"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Grimpeur">
        <rdfs:label>Grimpeur</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#CoureurCycliste"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Personne">
        <rdfs:label>Personne</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ObjetBiologique"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#ContreAttaque">
        <rdfs:label>ContreAttaque</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#EnsembleCoureurs"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#RangProvisoire">
        <rdfs:label>RangProvisoire</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#Rang"/>
        </rdfs:subClassOf>
        <rdfs:subClassOf>
            <owl:Restriction>
                <owl:onProperty rdf:resource="#localisation"/>
                <owl:cardinality>1</owl:cardinality>
            </owl:Restriction>
        </rdfs:subClassOf>
    </owl:Class>
```

Code C.6 – *Ontologie formelle du cyclisme en OWL : les concepts* **(suite)**

```
<owl:Class rdf:about="#Spectateur">
        <rdfs:label>Spectateur</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#Personne"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Col">
        <rdfs:label>Col</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ObjetGeographiqueNaturel"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Echappee">
        <rdfs:label>Echappee</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#EnsembleCoureurs"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Epreuve">
        <rdfs:label>Epreuve</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ActiviteHumaine"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#SupportDeDescriptionInformatique">
        <rdfs:label>SupportDeDescriptionInformatique</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#SupportDeDescription"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#MaillotVert">
        <rdfs:label>MaillotVert</rdfs:label>
        <owl:sameClassAs>
            <rdfs:Class>
                <owl:intersectionOf rdf:parseType="owl:collection">
                    <owl:Class rdf:about="#Maillot"/>
                    <owl:Restriction>
                        <owl:onProperty rdf:resource="#estClasseCoureurPoint"/>
                        <owl:someValuesFrom>
                            <owl:Class>
                                <owl:intersectionOf parseType="owl:collection"/>
                            </owl:Class>
                        </owl:someValuesFrom>
                    </owl:Restriction>
                </owl:intersectionOf>
            </rdfs:Class>
        </owl:sameClassAs>
    </owl:Class>
```

Code C.6 – *Ontologie formelle du cyclisme en OWL : les concepts* **(suite)**

```
<owl:Class rdf:about="#Descente">
        <rdfs:label>Descente</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ElementProgramme"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#ObjetGeographique">
        <rdfs:label>ObjetGeographique</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ObjetIndividuel"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#TypeConstruit">
        <rdfs:label>TypeConstruit</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#SupportDeDescriptionInformatique"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#MaillotAPoisRouge">
        <rdfs:label>MaillotAPoisRouge</rdfs:label>
        <owl:sameClassAs>
            <rdfs:Class>
                <owl:intersectionOf rdf:parseType="owl:collection">
                    <owl:Class rdf:about="#Maillot"/>
                    <owl:Restriction>
                        <owl:onProperty rdf:resource="#estClasseCoureurMontagne"/>
                        <owl:someValuesFrom>
                            <owl:Class>
                                <owl:intersectionOf parseType="owl:collection"/>
                            </owl:Class>
                        </owl:someValuesFrom>
                    </owl:Restriction>
                </owl:intersectionOf>
            </rdfs:Class>
        </owl:sameClassAs>
    </owl:Class>
    <owl:Class rdf:about="#Propriete">
        <rdfs:label>Propriete</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#Entite"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#CourseDUnJour">
        <rdfs:label>CourseDUnJour</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#Course"/>
        </rdfs:subClassOf>
    </owl:Class>
```

Code C.6 – *Ontologie formelle du cyclisme en OWL : les concepts* **(suite)**

```
<owl:Class rdf:about="#EnsembleCoureurs">
        <rdfs:label>EnsembleCoureurs</rdfs:label>
        <owl:sameClassAs>
            <rdfs:Class>
                <owl:intersectionOf rdf:parseType="owl:collection">
                    <owl:Class rdf:about="#EnsemblePersonnes"/>
                    <owl:Restriction>
                        <owl:onProperty rdf:resource="#contientElement"/>
                        <owl:alLValuesFrom>
                            <owl:Class rdf:about="#CoureurCycliste"/>
                        </owl:alLValuesFrom>
                    </owl:Restriction>
                </owl:intersectionOf>
            </rdfs:Class>
        </owl:sameClassAs>
    </owl:Class>
    <owl:Class rdf:about="#CLMParEquipe">
        <rdfs:label>CLMParEquipe</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ContreLaMontre"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Gruppetto">
        <rdfs:label>Gruppetto</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#EnsembleCoureurs"/>
        </rdfs:subClassOf>
        <rdfs:subClassOf>
            <owl:Restriction>
                <owl:onProperty rdf:resource="#contientElement"/>
                <owl:minCardinality>2</owl:minCardinality>
            </owl:Restriction>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#ZoneAdministrative">
        <rdfs:label>ZoneAdministrative</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ObjetGeographiqueAdministratif"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Peloton">
        <rdfs:label>Peloton</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#EnsembleCoureurs"/>
        </rdfs:subClassOf>
```

Code C.6 – *Ontologie formelle du cyclisme en OWL : les concepts* **(suite)**

```
<rdfs:subClassOf>
            <owl:Restriction>
                <owl:onProperty rdf:resource="#contientElement"/>
                <owl:minCardinality>2</owl:minCardinality>
            </owl:Restriction>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#CLMIndividuel">
        <rdfs:label>CLMIndividuel</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ContreLaMontre"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#MedecinEpreuve">
        <rdfs:label>MedecinEpreuve</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#PersonnelEpreuve"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#LieuSpatial">
        <rdfs:label>LieuSpatial</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#Lieu"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#SprintFinal">
        <rdfs:label>SprintFinal</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#Sprint"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Ascension">
        <rdfs:label>Ascension</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ElementProgramme"/>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Evenement">
        <rdfs:label>Evenement</rdfs:label>
        <rdfs:subClassOf>
            <owl:Class rdf:about="#ObjetConcretTemporel"/>
        </rdfs:subClassOf>
    </owl:Class>
```

Code C.6 – *Ontologie formelle du cyclisme en OWL : les concepts* **(suite)**

```
<owl:ObjectProperty rdf:about="#aCommeNumero">
        <rdfs:label>aCommeNumero</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionRigideEpreuve"/>
        <rdfs:range>
            <xsd:integer/>
        </rdfs:range>
    </owl:ObjectProperty>
    <owl:FunctionalProperty rdf:about="#aCommeNumero"/>
    <owl:ObjectProperty rdf:about="#aCommePrenom">
        <rdfs:label>aCommePrenom</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionRigideCoureur"/>
        <rdfs:range>
            <xsd:string/>
        </rdfs:range>
    </owl:ObjectProperty>
    <owl:FunctionalProperty rdf:about="#aCommePrenom"/>
    <owl:ObjectProperty rdf:about="#aCommeDate">
        <rdfs:label>aCommeDate</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#coincideTemporellementPar"/>
        <rdfs:range>
            <owl:Class rdf:about="#Date"/>
        </rdfs:range>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#attributionRigideRang">
        <rdfs:label>attributionRigideRang</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionRigide"/>
        <rdfs:domain>
            <owl:Class rdf:about="#Rang"/>
        </rdfs:domain>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#contientElement">
        <rdfs:label>contientElement</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#relationAppartenance"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#localisation">
        <rdfs:label>localisation</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#reperageCirconstanciel"/>
        <rdfs:domain>
            <owl:Class rdf:about="#Objet"/>
        </rdfs:domain>
        <rdfs:range>
            <owl:Class rdf:about="#Propriete"/>
        </rdfs:range>
    </owl:ObjectProperty>
```

Code C.7 – *Ontologie formelle du cyclisme en OWL : les propriétés*

```
<owl:ObjectProperty rdf:about="#estLeader">
        <rdfs:label>estLeader</rdfs:label>
        <owl:inverseOf rdf:resource="#aCommeLeader"/>
        <rdfs:subPropertyOf rdf:resource="#attributionCirconstancielleCoureur"/>
        <rdfs:range>
            <owl:Class rdf:about="#Equipe"/>
        </rdfs:range>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#relationMereologique">
        <rdfs:label>relationMereologique</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#reperageCirconstanciel"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#estClasseEquipe">
        <rdfs:label>estClasseEquipe</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionCirconstancielleEquipe"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#succedeTemporellementPar">
        <rdfs:label>succedeTemporellementPar</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#estLocaliseTemporellementPar"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#sponsor">
        <rdfs:label>sponsor</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionRigideEquipe"/>
        <rdfs:range>
            <xsd:string/>
        </rdfs:range>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#aCommeArrivee">
        <rdfs:label>aCommeArrivee</rdfs:label>
        <owl:inverseOf rdf:resource="#arrivee"/>
        <rdfs:subPropertyOf rdf:resource="#attributionRigideEpreuve"/>
        <rdfs:range>
            <owl:Class rdf:about="#Ville"/>
        </rdfs:range>
    </owl:ObjectProperty>
    <owl:FunctionalProperty rdf:about="#aCommeArrivee"/>
    <owl:ObjectProperty rdf:about="#estRavitailleur">
        <rdfs:label>estRavitailleur</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionCirconstancielleCoureur"/>
        <rdfs:range>
            <owl:Class>
                <owl:unionOf parseType="owl:collection">
                    <owl:Class rdf:about="#EnsembleCoureurs"/>
                    <owl:Class rdf:about="#Equipe"/>
                </owl:unionOf>
            </owl:Class>
        </rdfs:range>
    </owl:ObjectProperty>
```

Code C.7 – *Ontologie formelle du cyclisme en OWL : les propriétés* **(suite)**

```
<owl:ObjectProperty rdf:about="#aCommeNationalite">
        <rdfs:label>aCommeNationalite</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionRigideCoureur"/>
        <rdfs:range>
            <owl:Class rdf:about="#Pays"/>
        </rdfs:range>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#attributionRigide">
        <rdfs:label>attributionRigide</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#rCyclisme"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#estConstitue">
        <rdfs:label>estConstitue</rdfs:label>
        <owl:inverseOf rdf:resource="#faitPartieDe"/>
        <rdfs:subPropertyOf rdf:resource="#relationMereologique"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#estClasseCoureurMontagne">
        <rdfs:label>estClasseCoureurMontagne</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#estClasseCoureur"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#abandon">
        <rdfs:label>abandon</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionCirconstancielleCoureur"/>
        <rdfs:range>
            <owl:Class rdf:about="#Epreuve"/>
        </rdfs:range>
    </owl:ObjectProperty>
    <owl:FunctionalProperty rdf:about="#abandon"/>
    <owl:ObjectProperty rdf:about="#rCyclisme">
        <rdfs:label>rCyclisme</rdfs:label>
        <rdfs:domain>
            <owl:Class rdf:about="#Entite"/>
        </rdfs:domain>
        <rdfs:range>
            <owl:Class rdf:about="#Entite"/>
        </rdfs:range>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#estElement">
        <rdfs:label>estElement</rdfs:label>
        <owl:inverseOf rdf:resource="#contientElement"/>
        <rdfs:subPropertyOf rdf:resource="#relationAppartenance"/>
        <rdfs:domain>
            <owl:Class rdf:about="#ObjetIndividuel"/>
        </rdfs:domain>
        <rdfs:range>
            <owl:Class rdf:about="#Collection"/>
        </rdfs:range>
    </owl:ObjectProperty>
```

Code C.7 – *Ontologie formelle du cyclisme en OWL : les propriétés* **(suite)**

```
<owl:ObjectProperty rdf:about="#precedeTemporellementPar">
        <rdfs:label>precedeTemporellementPar</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#estLocaliseTemporellementPar"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#abandonParMaladie">
        <rdfs:label>abandonParMaladie</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#abandon"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#attributionRigideCoureur">
        <rdfs:label>attributionRigideCoureur</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionRigide"/>
        <rdfs:domain>
            <owl:Class rdf:about="#CoureurCycliste"/>
        </rdfs:domain>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#depart">
        <rdfs:label>depart</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionRigideVille"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#attributionCirconstancielleCoureur">
        <rdfs:label>attributionCirconstancielleCoureur</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionCirconstancielle"/>
        <rdfs:domain>
            <owl:Class rdf:about="#CoureurCycliste"/>
        </rdfs:domain>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#arrivee">
        <rdfs:label>arrivee</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionRigideVille"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#etreEvacue">
        <rdfs:label>etreEvacue</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionCirconstancielleCoureur"/>
        <rdfs:range>
            <owl:Class rdf:about="#Hopital"/>
        </rdfs:range>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#aCommeDistance">
        <rdfs:label>aCommeDistance</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionRigideEpreuve"/>
        <rdfs:range>
            <xsd:integer/>
        </rdfs:range>
    </owl:ObjectProperty>
    <owl:FunctionalProperty rdf:about="#aCommeDistance"/>
```

Code C.7 – *Ontologie formelle du cyclisme en OWL : les propriétés* **(suite)**

```
<owl:ObjectProperty rdf:about="#aCommeVictime">
        <rdfs:label>aCommeVictime</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#aCommeObjet"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#faitPartieDe">
        <rdfs:label>faitPartieDe</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#relationMereologique"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#aCommeAge">
        <rdfs:label>aCommeAge</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionRigideCoureur"/>
        <rdfs:range>
            <xsd:integer/>
        </rdfs:range>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#aCommeNom">
        <rdfs:label>aCommeNom</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionRigideCoureur"/>
        <rdfs:range>
            <xsd:string/>
        </rdfs:range>
    </owl:ObjectProperty>
    <owl:FunctionalProperty rdf:about="#aCommeNom"/>
    <owl:ObjectProperty rdf:about="#estClasseCoureurJeune">
        <rdfs:label>estClasseCoureurJeune</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#estClasseCoureur"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#estClasseCoureurPoint">
        <rdfs:label>estClasseCoureurPoint</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#estClasseCoureur"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#abandonParBlessure">
        <rdfs:label>abandonParBlessure</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#abandon"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#equipier">
        <rdfs:label>equipier</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionCirconstancielleCoureur"/>
        <rdfs:range>
            <owl:Class rdf:about="#CoureurCycliste"/>
        </rdfs:range>
    </owl:ObjectProperty>
    <owl:TransitiveProperty rdf:about="#equipier"/>
    <owl:ObjectProperty rdf:about="#position">
        <rdfs:label>position</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionRigideRang"/>
        <rdfs:range><xsd:integer/></rdfs:range>
    </owl:ObjectProperty>
```

Code C.7 – *Ontologie formelle du cyclisme en OWL : les propriétés* **(suite)**

```
<owl:FunctionalProperty rdf:about="#position"/>
    <owl:ObjectProperty rdf:about="#aCommeResponsable">
        <rdfs:label>aCommeResponsable</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#aCommeCause"/>
        <rdfs:range>
            <owl:Class>
                <owl:unionOf parseType="owl:collection">
                    <owl:Class rdf:about="#Personne"/>
                    <owl:Class rdf:about="#EnsemblePersonnes"/>
                </owl:unionOf>
            </owl:Class>
        </rdfs:range>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#aCommeAgent">
        <rdfs:label>aCommeAgent</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#participant"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#aCommeCause">
        <rdfs:label>aCommeCause</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#participant"/>
        <rdfs:domain>
            <owl:Class rdf:about="#Incident"/>
        </rdfs:domain>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#aCommeObjet">
        <rdfs:label>aCommeObjet</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#participant"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#estClasseCoureurCombativite">
        <rdfs:label>estClasseCoureurCombativite</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#estClasseCoureur"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#estClasseCoureurGeneral">
        <rdfs:label>estClasseCoureurGeneral</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#estClasseCoureur"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#relationAppartenance">
        <rdfs:label>relationAppartenance</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#reperageCirconstanciel"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#traverse">
        <rdfs:label>traverse</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#estLocaliseSpatialementPar"/>
    </owl:ObjectProperty>
```

Code C.7 – *Ontologie formelle du cyclisme en OWL : les propriétés* **(suite)**

```
<owl:ObjectProperty rdf:about="#attributionRigideEquipe">
        <rdfs:label>attributionRigideEquipe</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionRigide"/>
        <rdfs:domain>
            <owl:Class rdf:about="#Equipe"/>
        </rdfs:domain>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#attributionRigideVille">
        <rdfs:label>attributionRigideVille</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionRigide"/>
        <rdfs:domain>
            <owl:Class rdf:about="#Ville"/>
        </rdfs:domain>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#estClasseCoureur">
        <rdfs:label>estClasseCoureur</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionCirconstancielleCoureur"/>
        <rdfs:range>
            <owl:Class rdf:about="#Rang"/>
        </rdfs:range>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#estLocaliseTemporellementPar">
        <rdfs:label>estLocaliseTemporellementPar</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#localisation"/>
        <rdfs:domain>
            <owl:Class rdf:about="#ObjetConcretTemporel"/>
        </rdfs:domain>
        <rdfs:range>
            <owl:Class rdf:about="#LieuTemporel"/>
        </rdfs:range>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#attributionCirconstancielleEquipe">
        <rdfs:label>attributionCirconstancielleEquipe</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionCirconstancielle"/>
        <rdfs:domain>
            <owl:Class rdf:about="#Equipe"/>
        </rdfs:domain>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#participant">
        <rdfs:label>participant</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionRigide"/>
        <rdfs:domain>
            <owl:Class rdf:about="#ObjetConcretTemporel"/>
        </rdfs:domain>
    </owl:ObjectProperty>
```

Code C.7 – *Ontologie formelle du cyclisme en OWL : les propriétés* **(suite)**

```
<owl:ObjectProperty rdf:about="#coincideTemporellementPar">
        <rdfs:label>coincideTemporellementPar</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#estLocaliseTemporellementPar"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#attributionRigideEpreuve">
        <rdfs:label>attributionRigideEpreuve</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionRigide"/>
        <rdfs:domain>
            <owl:Class rdf:about="#Epreuve"/>
        </rdfs:domain>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#epreuveConsideree">
        <rdfs:label>epreuveConsideree</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionRigideRang"/>
        <rdfs:range>
            <owl:Class rdf:about="#Epreuve"/>
        </rdfs:range>
    </owl:ObjectProperty>
    <owl:FunctionalProperty rdf:about="#epreuveConsideree"/>
    <owl:ObjectProperty rdf:about="#estVictime">
        <rdfs:label>estVictime</rdfs:label>
        <owl:inverseOf rdf:resource="#aCommeVictime"/>
        <rdfs:subPropertyOf rdf:resource="#attributionCirconstancielleCoureur"/>
        <rdfs:range>
            <owl:Class rdf:about="#Incident"/>
        </rdfs:range>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#aCommeDepart">
        <rdfs:label>aCommeDepart</rdfs:label>
        <owl:inverseOf rdf:resource="#depart"/>
        <rdfs:subPropertyOf rdf:resource="#attributionRigideEpreuve"/>
        <rdfs:range>
            <owl:Class rdf:about="#Ville"/>
        </rdfs:range>
    </owl:ObjectProperty>
    <owl:FunctionalProperty rdf:about="#aCommeDepart"/>
    <owl:ObjectProperty rdf:about="#estLocaliseSpatialementPar">
        <rdfs:label>estLocaliseSpatialementPar</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#localisation"/>
        <rdfs:domain>
            <owl:Class rdf:about="#ObjetConcretSpatial"/>
        </rdfs:domain>
        <rdfs:range>
            <owl:Class rdf:about="#LieuSpatial"/>
        </rdfs:range>
    </owl:ObjectProperty>
```

Code C.7 – *Ontologie formelle du cyclisme en OWL : les propriétés* **(suite)**

```
<owl:ObjectProperty rdf:about="#aCommeElementDePalmares">
        <rdfs:label>aCommeElementDePalmares</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionRigideCoureur"/>
        <rdfs:range>
            <owl:Class rdf:about="#RangEpreuve"/>
        </rdfs:range>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#attributionCirconstancielle">
        <rdfs:label>attributionCirconstancielle</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#reperageCirconstanciel"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#reperageCirconstanciel">
        <rdfs:label>reperageCirconstanciel</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#rCyclisme"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:about="#aCommeLeader">
        <rdfs:label>aCommeLeader</rdfs:label>
        <rdfs:subPropertyOf rdf:resource="#attributionCirconstancielleEquipe"/>
    </owl:ObjectProperty>
```

Code C.7 – *Ontologie formelle du cyclisme en OWL : les propriétés* **(suite)**

```
<owl:Class rdf:about="#Entite">
        <rdfs:subClassOf><rdfs:Class>
            <owl:disjointUnionOf parseType="owl:collection">
                <owl:Class rdf:about="#Objet"/>
                <owl:Class rdf:about="#Propriete"/>
            </owl:disjointUnionOf>
        </rdfs:Class></rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Objet">
        <rdfs:subClassOf>
            <rdfs:Class>
                <owl:disjointUnionOf parseType="owl:collection">
                    <owl:Class rdf:about="#ObjetAbstrait"/>
                    <owl:Class rdf:about="#ObjetConcret"/>
                </owl:disjointUnionOf>
            </rdfs:Class>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#SupportDeDescriptionInformatique">
        <rdfs:subClassOf>
            <rdfs:Class>
                <owl:disjointUnionOf parseType="owl:collection">
                    <owl:Class rdf:about="#TypeSimple"/>
                    <owl:Class rdf:about="#TypeConstruit"/>
                </owl:disjointUnionOf>
            </rdfs:Class>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Rang">
        <rdfs:subClassOf>
            <rdfs:Class>
                <owl:disjointUnionOf parseType="owl:collection">
                    <owl:Class rdf:about="#RangEpreuve"/>
                    <owl:Class rdf:about="#RangProvisoire"/>
                </owl:disjointUnionOf>
            </rdfs:Class>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#ObjetConcret">
        <rdfs:subClassOf>
            <rdfs:Class>
                <owl:disjointUnionOf parseType="owl:collection">
                    <owl:Class rdf:about="#ObjetConcretSpatial"/>
                    <owl:Class rdf:about="#ObjetConcretTemporel"/>
                </owl:disjointUnionOf>
            </rdfs:Class>
        </rdfs:subClassOf>
    </owl:Class>
```

Code C.8 – *Ontologie formelle du cyclisme en OWL : les axiomes*

```
<owl:Class rdf:about="#ObjetConcretSpatial">
        <rdfs:subClassOf><rdfs:Class>
            <owl:disjointUnionOf parseType="owl:collection">
                <owl:Class rdf:about="#ObjetIndividuel"/>
                <owl:Class rdf:about="#Collection"/>
            </owl:disjointUnionOf>
        </rdfs:Class></rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Personne">
        <rdfs:subClassOf><rdfs:Class>
            <owl:disjointUnionOf parseType="owl:collection">
                <owl:Class rdf:about="#PersonnelEpreuve"/>
                <owl:Class rdf:about="#PersonnelEquipe"/>
                <owl:Class rdf:about="#Spectateur"/>
            </owl:disjointUnionOf>
        </rdfs:Class></rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#ObjetGeographique">
        <rdfs:subClassOf>
            <rdfs:Class>
                <owl:disjointUnionOf parseType="owl:collection">
                    <owl:Class rdf:about="#ObjetGeographiqueAdministratif"/>
                    <owl:Class rdf:about="#ObjetGeographiqueNaturel"/>
                </owl:disjointUnionOf>
            </rdfs:Class>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#ObjetGeographiqueAdministratif">
        <rdfs:subClassOf>
            <rdfs:Class>
                <owl:disjointUnionOf parseType="owl:collection">
                    <owl:Class rdf:about="#FrontiereAdministrative"/>
                    <owl:Class rdf:about="#ZoneAdministrative"/>
                </owl:disjointUnionOf>
            </rdfs:Class>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Epreuve">
        <rdfs:subClassOf>
            <rdfs:Class>
                <owl:disjointUnionOf parseType="owl:collection">
                    <owl:Class rdf:about="#Course"/>
                    <owl:Class rdf:about="#Etape"/>
                </owl:disjointUnionOf>
            </rdfs:Class>
        </rdfs:subClassOf>
    </owl:Class>
```

Code C.8 – *Ontologie formelle du cyclisme en OWL : les axiomes* **(suite)**

```
<owl:Class rdf:about="#Course">
        <rdfs:subClassOf><rdfs:Class>
            <owl:disjointUnionOf parseType="owl:collection">
                <owl:Class rdf:about="#CourseAEtapes"/>
                <owl:Class rdf:about="#CourseDUnJour"/>
            </owl:disjointUnionOf>
        </rdfs:Class></rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Etape">
        <rdfs:subClassOf><rdfs:Class>
            <owl:disjointUnionOf parseType="owl:collection">
                <owl:Class rdf:about="#ContreLaMontre"/>
                <owl:Class rdf:about="#EtapeEnLigne"/>
            </owl:disjointUnionOf>
        </rdfs:Class></rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#ContreLaMontre">
        <rdfs:subClassOf>
            <rdfs:Class>
                <owl:disjointUnionOf parseType="owl:collection">
                    <owl:Class rdf:about="#CLMIndividuel"/>
                    <owl:Class rdf:about="#CLMParEquipe"/>
                </owl:disjointUnionOf>
            </rdfs:Class>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Sprint">
        <rdfs:subClassOf>
            <rdfs:Class>
                <owl:disjointUnionOf parseType="owl:collection">
                    <owl:Class rdf:about="#SprintIntermediaire"/>
                    <owl:Class rdf:about="#SprintFinal"/>
                </owl:disjointUnionOf>
            </rdfs:Class>
        </rdfs:subClassOf>
    </owl:Class>
    <owl:Class rdf:about="#Lieu">
        <rdfs:subClassOf>
            <rdfs:Class>
                <owl:disjointUnionOf parseType="owl:collection">
                    <owl:Class rdf:about="#LieuSpatial"/>
                    <owl:Class rdf:about="#LieuTemporel"/>
                </owl:disjointUnionOf>
            </rdfs:Class>
        </rdfs:subClassOf>
    </owl:Class>
</rdf:RDF>
```

Code C.8 – *Ontologie formelle du cyclisme en OWL : les axiomes* **(suite)**

Table des figures

1.1 Le cycle de vie d'un document audiovisuel d'après [Auffret, 2000] 19
1.2 Les différentes étapes du processus de description des documents audiovisuels . . 28
1.3 Structuration logique et description sémantique du contenu des documents audiovisuels 28

2.1 Structure hiérarchique avec ses attributs exprimés selon le Dublin Core pour la description de documents vidéo, d'après [Hunter, 1999] 45
2.2 Schéma général des outils de description disponibles dans le MDS 47
2.3 Hiérarchie des types de description MPEG-7 48
2.4 Hiérarchie des types de segments disponibles dans le MDS pour décrire le contenu multimédia 49
2.5 Hiérarchie des différentes décompositions structurelles applicables aux segments . 50
2.6 Annotation de séquences vidéos dans l'outil *VideoAnnEx*. La vidéo (en haut à droite) est découpée en plans repérés par des images clés (en bas) et annotée grâce au vocabulaire contrôlé appartenant à l'une des trois catégories : arrière-plan, événement et objet (en haut à gauche) 53
2.7 Interface de segmentation de *Log Creator* 54
2.8 Interface d'annotation de *Log Creator* 55
2.9 Interface de l'environnement auteur *Mdéfi* : vues hiérarchique (à gauche) et temporelle (en bas à droite) du contenu 56
2.10 Interface de l'outil *VIDETO*. La vidéo (en haut à gauche) est découpée en plans (à droite) et est annotée en fonction d'un schéma (en bas à gauche) 58
2.11 Structuration des annotations dans le modèle Strates-IA 60
2.12 Exemple d'annotation d'une vidéo dans le système OPALES : l'annotation est un graphe conceptuel (à droite) dont le support est généré par l'éditeur d'ontologie *DOE* (en bas à gauche) 63

3.1 Recensement des objets qui ont pu, à un moment donné, être classifié comme des ontologies, d'après [Smith and Welty, 2001] 68
3.2 Exemple de support avec le treillis de types et un graphe étoile 71
3.3 Exemples de graphe sur le support de la Figure 3.2 73
3.4 Architecture générale d'Ontobroker 79
3.5 Exemple de graphe RDF représentant l'assertion du tableau 3.1 81
3.6 *The Layer Cake* - d'après Tim Berners Lee [28/01/2003] 85
3.7 Les 3 étapes de la méthode de construction d'ontologies selon B. Bachimont . . . 89

4.1 Représentation structurelle des séquences décrites dans le tableau 4.1 101

4.2 Description sémantique de la séquence numérotée 12 du tableau 4.1 sous la forme d'un graphe RDF . . . 107
4.3 Proposition d'architecture pour permettre le raisonnement dans les descriptions documentaires. L'ontologie de l'audiovisuel permet de formaliser les connaissances structurelles des descriptions. Elle est traduite en types XML Schema pour pouvoir exprimer des modèles de document et est instanciée lors de la description. Enfin, une autre ontologie de domaine, avec les faits qui y sont raccordés, permet d'exprimer la connaissance conceptuelle qui sera liée aux éléments structurels de la description. . . . 110
4.4 Nouvelle hiérarchie des types de description (en MPEG-7 étendu) . . . 113
4.5 Nouvelle hiérarchie des types de segments (en MPEG-7 étendu) : les différents genres et séquences proviennent de l'ontologie de l'audiovisuel . . . 116
4.6 Nouvelle hiérarchie des différentes décompositions structurelles applicables aux segments (en MPEG-7 étendu) . . . 117

5.1 Axes différentiels pour les différents types d'objets audiovisuels . . . 125
5.2 Les principes différentiels associés au concept *EmissionComposite* dans l'outil DOE 126
5.3 Branche de l'arbre correspondant aux émissions de type sport dans la typologie Médiamétrie : le code `FBM` désigne une émission de type `sport, magazine, multisports` . . . 127
5.4 L'ontologie MPEG-7, d'après [Hunter, 2001], importée dans l'outil OilEd : à gauche, la taxinomie des concepts et à droite la liste des propriétés avec leur définition . . 133
5.5 Découpage temporel des documents audiovisuels avec l'outil *SegmenTool* . . . 135
5.6 Décomposition possible d'une `Emission` et d'un `MagazinePlateau` . . . 139
5.7 Modèle de contenu ambigu dans MPEG-7 . . . 142
5.8 Automate décrivant le modèle de contenu donné dans la figure 5.7 . . . 144
5.9 Automate déterminisé décrivant le modèle de contenu donné dans la figure 5.7 . . 146

6.1 Les principes différentiels de la notion *Personnel Équipe* dans *DOE* . . . 152
6.2 Les *concepts* et les *relations* de haut niveau utilisés dans l'ontologie du cyclisme . 155
6.3 Architecture du système Seigo, d'après [Le Roux, 2003] . . . 157
6.4 Représentation sous forme de graphe d'une phrase analysée par le système Seigo 159
6.5 Graphe exemple d'une assertion en XML/RDF . . . 160
6.6 Exemple d'une requête RDQL dans le système Sesame . . . 162
6.7 L'outil *RICE*, interface graphique du classifieur *Racer* . . . 164
6.8 Visualisation des séquences vidéos dans *Internet Explorer*. La description de la structure est transformée en une présentation HTML+TIME grâce à une feuille de style XSLT . . . 165

A.1 Aperçu des volumes et origine des fonds du Département Droits et Archives au 1er janvier 2003 (source : communication interne de l'INA) . . . 172
A.2 Chaînes concernées par le Dépôt Légal et aperçu des fonds à l'Inathèque de France au 1er janvier 2003 (source : site Web de l'INA) . . . 172

B.1 Taxinomie des concepts dans l'ontologie de l'audiovisuel : le haut niveau et les différents types de séquences . . . 177
B.2 Taxinomie des concepts dans l'ontologie de l'audiovisuel : les différents types d'émissions . . . 178
B.3 Taxinomie des concepts dans l'ontologie de l'audiovisuel : les thématiques générales 178

B.4 Taxinomie des concepts dans l'ontologie de l'audiovisuel : les propriétés et les activités audiovisuelles 179

C.1 Taxinomie des concepts dans l'ontologie du cyclisme - Page 1/7 194
C.2 Taxinomie des concepts dans l'ontologie du cyclisme - Page 2/7 195
C.3 Taxinomie des concepts dans l'ontologie du cyclisme - Page 3/7 196
C.4 Taxinomie des concepts dans l'ontologie du cyclisme - Page 4/7 197
C.5 Taxinomie des concepts dans l'ontologie du cyclisme - Page 5/7 198
C.6 Taxinomie des concepts dans l'ontologie du cyclisme - Page 6/7 199
C.7 Taxinomie des concepts dans l'ontologie du cyclisme - Page 7/7 200
C.8 Taxinomie des relations dans l'ontologie du cyclisme - Page 1/5 201
C.9 Taxinomie des relations dans l'ontologie du cyclisme - Page 2/5 202
C.10 Taxinomie des relations dans l'ontologie du cyclisme - Page 3/5 203
C.11 Taxinomie des relations dans l'ontologie du cyclisme - Page 4/5 204
C.12 Taxinomie des relations dans l'ontologie du cyclisme - Page 5/5 205

Liste des tableaux

1.1 Extrait d'une notice documentaire rédigée par l'Inathèque et concernant l'émission `Stade2` diffusée le 17 mars 2002 18
1.2 Extrait d'une notice documentaire rédigée par les Archives et concernant une séquence de l'émission `Stade2` diffusée le 17 mars 2002 20

2.1 Liste des éléments du *Dublin Core* 43

3.1 Interprétation des termes structurés du langage $\mathcal{FLN}$ 76
3.2 Exemples de règles d'inférence induites par la sémantique de RDF/RDFS (d'après [RDF-MT, 2004], section 4) 82
3.3 Liste des constructeurs disponibles dans le langage *OWL Lite* 84
3.4 Liste des constructeurs supplémentaires disponibles dans les langages *OWL DL* et *OWL Full* 84

4.1 Extrait d'une notice documentaire concernant l'émission `Stade2` 98

5.1 Les genres possibles pour la télévision selon l'INA 128
5.2 Les thèmes possibles pour la télévision selon l'INA 129
5.3 Schéma de description pour contrôler la structure d'un *Magazine Sportif* (forme EBNF) 134

6.1 Quelques termes désignant une personne 150
6.2 Principes différentiels associés aux spécialisations directes de *Personne* 151
6.3 Les informations intéressantes pour France 2 lors d'un Tour de France 158
6.4 Nombre de triplets contenus dans la base de connaissances *Sesame* avec les ontologies de l'audiovisuel et du cyclisme 161

A.1 Extrait de la notice documentaire de la fiche collection, modifiée en 1996, concernant l'émission `Stade2` 176

Liste des codes sources

2.1 Exemple de document XML 33
2.2 Exemple de DTD 34
2.3 Exemple de Schéma XML 36
3.1 Exemple de description dans la syntaxe RDF/XML 80
4.1 Description MPEG-7 de la structure schématisée dans la figure 4.1 102
4.2 Exemple de schéma de classification définissant les termes `Interview` et `Débat` comme des `Séquence Dialogue` 104
4.3 Exemple de définition du type *Magazine Composite* en XML Schema 105
4.4 Exemple de définition de la classe *Magazine Composite* en OWL 106
4.5 Définition du type *TV*, considéré comme un nouveau type de contenu multimédia 112
4.6 Définition des types *Genre* et *Sequence*, qui sont tous deux des segments qui peuvent se décomposer temporellement 115
4.7 Définition du type *MagazineComposite* qui est un genre particulier 118
4.8 Définition des types *SequencePlateau*, *Reportage* et *Interview* 119
4.9 Définition du type *GeneralDecomposition* qui permet de décomposer des segments 119
4.10 Définition du type *MagazineCompositeDecomposition* qui permet de décomposer temporellement les *MagazineComposite* 120
4.11 Définition du type *ReportageDecomposition* qui permet de décomposer temporellement les *Reportage* 120
5.1 Définition du concept *EmissionPlateau* en OWL 130
5.2 Définition d'un axiome en OWL DL 131
5.3 Définition du concept *EpisodeLong* en OWL à l'aide du type *DureeComprise Entre-52Et90* défini en XML Schema 131
5.4 Description instance d'une séquence particulière en MPEG-7 étendu 136
5.5 Exemple de triplets RDF/XML construits automatiquement à partir de la description 138
5.6 Structure possible d'une `Emission` et d'un `MagazinePlateau` en XML Schema . . 140
5.7 Description valide d'une `Emission` 141
5.8 Description valide d'un `MagazinePlateau` 141
5.9 Définition ambiguë du type `EditedVideoEditingTemporalDecompositionType` dans MPEG-7 143
5.10 Définition *corrigée* du type MPEG-7 `EditedVideoEditingTemporalDecompositionType` 145
6.1 Définition du concept *Prologue* en OWL DL 154
6.2 Définition d'une partition disjointe en DAML+OIL 154
6.3 Requête n°1 exprimée dans le langage RDQL 162

6.4 Requête n°2 exprimée dans le langage RDQL. Le concept `Col21HCategorie` est défini comme équivalent à l'union des concepts `Col2Categorie`, `Col1Categorie` et `ColHCCategorie` 163
6.5 Requête n°3 exprimée dans le langage Racer 163
C.6 Ontologie formelle du cyclisme en OWL : les concepts 207
C.7 Ontologie formelle du cyclisme en OWL : les propriétés 222
C.8 Ontologie formelle du cyclisme en OWL : les axiomes 231

Bibliographie

[Alexaki *et al.*, 2001] S. Alexaki, V. Christophides, G. Karvounarakis, D. Plexousakis, and K. Tolle. The ICS-Forth Suite : Managing Voluminous RDF Description Bases. In *2nd International Workshop on the Semantic Web (SemWeb'01)*, pages 1–13, Honk Kong, Chine, 1er Mai 2001.

[Allauzen and Gauvain, 2003] A. Allauzen and J.L. Gauvain. Adaptation automatique du modèle de langage d'un système de transcription. *Traitement Automatique des Langues*, 44(1) :11–31, 2003. Hermes.

[Allen, 1983] J.F. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11) :832–843, Novembre 1983.

[Angele and Sure, 2002] J. Angele and York Sure, editors. *First International Workshop Evaluation of Ontology-based Tools (EON'02)*, volume (62) of *CEUR-WS*, Sigüenza, Espagne, 30 Septembre 2002. `http://CEUR-WS.org/Vol-62/`.

[Arpirez *et al.*, 2001] J. Arpirez, O. Corcho, M. Fernández-López, and A. Gómez-Pérez. WebODE : a Workbench for Ontological Engineering. In *First international Conference on Knowledge Capture (K-CAP'01)*, pages 6–13, Victoria, Canada, 21-23 Octobre 2001. ACM.

[Auffret and Bachimont, 1999] G. Auffret and B. Bachimont. Audivisual cultural heritage : From TV and radio archiving to hypermedia publishing. In S. Abiteboul and A. M. Vercoustre, editors, *3rd European Conference on Research and Advanced Technology for Digital Libraries*, volume (1696) of *Lecture Notes in Computer Science*, pages 58–75, Paris, France, Septembre 1999. Springer Verlag.

[Auffret, 2000] G. Auffret. *Structuration de documents audiovisuels et publication électronique*. PhD thesis, Université de Technologie de Compiègne, 2000.

[Aussenac-Gilles *et al.*, 2003] N. Aussenac-Gilles, B. Biebow, and S. Szulman. D'une méthode à un guide pratique de modélisation des connaissances à partir de textes. In *5es journées Terminologie et Intelligence Artificielle*, pages 41–53, Strasbourg, France, 2003.

[Baader *et al.*, 2003] F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P.F. Patel-Schneider. *The Description Logic Handbook : Theory, Implementation and Applications*. Cambridge University Press, 2003.

[Bachimont and Vignaux, 2002] B. Bachimont and G. Vignaux. Modélisation sémantique et indexation multimédia. Action Spécifique n°50, Département STIC du CNRS, 2002. `http://www.cnrs.fr/STIC/actions/as/as/as50.htm`.

[Bachimont *et al.*, 2002] B. Bachimont, A. Isaac, and R. Troncy. Semantic Commitment for Designing Ontologies : A Proposal. In A. Gomez-Pérez and V.R. Benjamins, editors, *13th International Conference on Knowledge Engineering and Knowledge Management (EKAW'02)*, volume (2473) of *Lecture Notes in Artificial Intelligence*, pages 114–121, Sigüenza, Espagne, 1-4 Octobre 2002. Springer Verlag.

[Bachimont, 1996] B. Bachimont. *Herméneutique matérielle et artéfacture : critique du formalisme en intelligence artificielle. Des machines qui pensent aux machines qui donnent à penser.* PhD thesis, Ecole Polytechnique, 1996.

[Bachimont, 1999] B. Bachimont. Bibliothèques numériques audiovisuelles : des enjeux scientifiques et techniques. *Document Numérique, Numéro Spécial "Les Bibliothèques Numériques"*, 2(3-4) :219–242, 1999.

[Bachimont, 2000a] B. Bachimont. Engagement sémantique et engagement ontologique : conception et réalisation d'ontologies en ingénierie des connaissances. In J. Charlet, M. Zacklad, G. Kassel, and D. Bourigault, editors, *Ingénierie des Connaissances : Evolutions récentes et nouveaux défis*. Eyrolles, 2000.

[Bachimont, 2000b] B. Bachimont. Indexation audiovisuelle : une problématique en pleine évolution. *L'objet - Objets et multimédia*, 6(2) :171–191, 2000. Edition Hermes.

[Bachimont, 2001] B. Bachimont. Modélisation linguistique et modélisation logique des ontologies : l'apport de l'ontologie formelle. In *12th Journées Francophones d'Ingénierie des Connaissances (IC'01)*, pages 349–368, Grenoble, France, 25-28 Juin 2001. Presses Universitaires de Grenoble (PUG).

[Baget, 2003] J.F. Baget. Homomorphismes d'hypergraphes pour la subsomption en RDF. In *3ème Journées Nationales sur les Modèles de Raisonnement (JNMR'03)*, Paris, France, 27-28 Novembre 2003.

[Barry *et al.*, 2001] C. Barry, C. Cormier, G. Kassel, and J. Nobécourt. Évaluation de langages opérationnels de représentation d'ontologies. In *12th Journées Francophones d'Ingénierie des Connaissances (IC'01)*, pages 309–327, Grenoble, France, 25-28 Juin 2001. Presses Universitaires de Grenoble (PUG).

[Bechhofer *et al.*, 2001] S. Bechhofer, I. Horrocks, C. Goble, and R. Stevens. OilEd : a Reasonable Ontology Editor for the Semantic Web. In *Joint German/Austrian conference on Artificial Intelligence (KI'01)*, volume (2174) of *Lecture Notes in Artificial Intelligence*, pages 396–408, Vienne, Autriche, 2001. Springer Verlag.

[Bechhofer *et al.*, 2002] S. Bechhofer, L. Carr, C. Goble, S. Kampa, and T. Miles-Board. The Semantics of Semantic Annotation. In *First International Conference on Ontologies, Databases and Applications of SEmantics for Large Scale Information Systems (ODBASE'02)*, volume (2159) of *Lecture Notes in Computer Science*, pages 1151–1167, Irvine, Californie, USA, 29-31 Octobre 2002. Springer Verlag.

[Belaïd *et al.*, 2002] A. Belaïd, H. Emptoz, and G. Vignaux. Document : Numérisation Valorisation des Collections. Action Spécifique n°96, Département STIC du CNRS, 2002. `http://lcp.damesme.cnrs.fr/AS-2.pdf`.

[Benitez *et al.*, 2002] Ana B. Benitez, H. Rising, and C. Jörgensen. Semantics of Multimedia in MPEG-7. In *IEEE Conference on Image Processing (ICIP'02)*, Rochester, New York, USA, Septembre 2002.

[Benzaken, 1991] C. Benzaken. *Systèmes formels : introduction à la logique et à la théorie des langages*. Logique Mathématiques Informatiques. Masson, Paris, France, 1991.

[Berners-Lee *et al.*, 1998] T. Berners-Lee, R. Fielding, U.C. Irvine, and L. Masinter. Uniform Resource Identifiers (URI) : Generic Syntax. Request for Comments (RFC) 2396, IETF, 1998. `http://www.ietf.org/rfc/rfc2396.txt`.

[Berners-Lee *et al.*, 2001] T. Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web. *Scientific American*, 284(5) :34–43, Mai 2001.

[Berners-Lee, 2001] T. Berners-Lee. Conceptual Graphs and the Semantic Web. Design Issues, 2001. `http://www.w3.org/DesignIssues/CG.html`.

[Bes and Roisin, 2002] F. Bes and C. Roisin. Quels langages pour mieux contrôler le formatage des documents multimédias ? In *5ème Colloque International sur le Document Électronique (CIDE'5)*, pages 1–15, Hammamet, Tunisie, 20-23 Octobre 2002.

[Bimbo, 2000] A. del Bimbo. Issues and directions in visual information retrieval. In *International Conference on Pattern Recognition*, volume (4), pages 31–38, Barcelone, Espagne, Septembre 2000.

[Blázquez *et al.*, 1998] M. Blázquez, M. Fernández-López, J.M. García-Pinar, and A. Gómez-Pérez. Building Ontologies at the Knowledge Level using the Ontology Design Environment. In *11th Workshop on Knowledge Acquisition, Modeling and Management (KAW'98)*, Banff, Alberta, Canada, 18-23 Avril 1998. `http://ksi.cpsc.ucalgary.ca/KAW/KAW98/blazquez/`.

[Bourigault *et al.*, 2003] D. Bourigault, N. Aussenac-Gilles, and J. Charlet. Construction de ressources terminologiques ou ontologiques à partir de textes : un cadre unificateur pour trois études de cas. *Revue d'Intelligence Artificielle*, 2003. **à paraître**.

[Bourigault, 1994] D. Bourigault. *Extraction et structuration automatiques de terminologie pour l'aide à l'acquisition des connaissances à partir de textes*. PhD thesis, Ecole des Hautes Etudes en Sciences Sociales, Paris, 1994.

[Brachman and Schmolze, 1985] R.J. Brachman and J.G. Schmolze. An Overview of the KL-ONE Knowledge Representation System. *Cognitive Science*, 9(2) :171–216, 1985.

[Bringay, 2003] S. Bringay. Documents numériques médicaux : informatisation du dossier patient. Master's thesis, Laboratoire de Recherche en Informatique d'Amiens (LaRIA), Amiens, France, 17 Mai 2003.

[Broekstra, 2002] J. Broekstra. Sesame : a Generic Architecture for Storing and Querying RDF and RDF Schema. In Ian Horrocks and James Hendler, editors, *First International Semantic Web Conference (ISWC'02)*, volume (2342) of *Lecture Notes in Computer Science*, 9-12 Juin 2002.

[Brunie and Troncy, 2003] V. Brunie and R. Troncy. Essai d'utilisation de MPEG-7. Rapport de recherche, Institut National de l'Audiovisuel, Octobre 2003.

[Brunie, 1999] V. Brunie. *Désagrégation et reconstruction documentaire pour la lecture des hypertextes : problèmes et méthodes.* PhD thesis, Université de Technologie de Compiègne, 1999.

[Buckland, 1997] M. Buckland. What is a "document" ? *Journal of the American Society of Information Science*, 48(9) :804–809, 1997. `http://www.sims.berkeley.edu/~buckland/whatdoc.html`.

[Buckland, 1998] M. Buckland. What is a "digital document" ? *Document Numérique*, 2(2) :221–230, 1998. `http://www.sims.berkeley.edu/~buckland/digdoc.html`.

[Bush, 1945] V. Bush. As We May Think. *Atlantic Monthly*, 176(1) :101–108, Juillet 1945.

[Carrive, 2000] J. Carrive. *Classification de séquences audiovisuelles.* PhD thesis, Université Paris 6 (Pierre et Marie Curie), 2000.

[Charlet *et al.*, 2003] J. Charlet, B. Bachimont, and R. Troncy. Ontologies pour le Web Sémantique. Technical report, Action Spécifique Web Sémantique, Rapport final, Chapitre 3, 2003.

[Charlet, 2003] J. Charlet. L'ingénierie des connaissances : développements, résultat et perspectives pour la gestion des connaissances médicales. Habilitation à diriger des recherches, Université Pierre et Marie Curie, Paris, France, 2003.

[Chein and Mugnier, 1992] M. Chein and M.L. Mugnier. Conceptual graphs : fundamental notions. *Revue d'intelligence artificielle*, 6(4) :365–406, 1992.

[Chiaramella, 1997] Y. Chiaramella. Browsing and querying : two complementary approaches for multimedia information retrieval. In *Hypertext – Information Retrieval – Multimedia (HIM'97)*, Dortmund, Allemagne, Septembre 1997. Conférence invitée, `http://lrb.cs.uni-dortmund.de/HIM97/Konferenzband/Chiaramella/Paper0.ht%m`.

[Corby *et al.*, 2000] O. Corby, R. Dieng, and C. Hébert. A Conceptual Graph Model for W3C Resource Description Framework. In *8th International Conference on Conceptual Structures (ICCS'00)*, pages 468–482, Darmstadt, Allemagne, 14-18 Août 2000.

[Costa *et al.*, 2002] M. Costa, N. Correja, and N. Guimarães. Annotations as Multiple Perspectives of Video Content. In *10th International Conference on Multimedia, ACM Multimedia'2002*, Juan-les-Pins, France, 1-6 Décembre 2002.

[Dechilly and Bachimont, 2000] T. Dechilly and B. Bachimont. Une ontologie pour éditer des schémas de description audiovisuels, extension pour l'inférence sur les descriptions. In *11th Journées Francophones d'Ingénierie des Connaissances (IC'00)*, pages 59–70, Toulouse, France, 10-12 Mais 2000.

[Desmontils and Jacquin, 2002] E. Desmontils and C. Jacquin. Annotations sur le Web : notes de lecture. In *Journées Scientifiques Web Sémantique, (Action Spécifique STIC CNRS)*, Paris, France, 10-11 Octobre 2002. `http://www.lalic.paris4.sorbonne.fr/stic/octobre/programme0209.html`.

[Egyed-Zs., 2003] E. Egyed-Zs. *Modélisation des connaissances dans une base de documents multimédias.* PhD thesis, Institut National des Sciences Appliquées de Lyon, 2003.

[Euzenat *et al.*, 2000] J. Euzenat, A. Napoli, and R. Ducournau. Les représentations des connaissances par objets. *Technique et science informatiques*, 19(1), 2000.

[Euzenat *et al.*, 2003a] J. Euzenat, N. Layaïda, and V. Dias. A semantic framework for multimedia document adaptation. In *18th International Joint Conference on Artificial Intelligence (IJCAI'03)*, pages 31–36, Acapulco, Mexique, 9-15 Août 2003. Morgan Kauffman.

[Euzenat *et al.*, 2003b] J. Euzenat, A. Napoli, and J.F. Baget. XML et les objets (Objectif XML). *RSTI L'Objet*, 9(3) :11–37, 2003.

[Euzenat, 1999] J. Euzenat. Sémantique des représentations de connaissance. Notes de cours de DEA d'Informatique, Université Joseph Fourier, Grenoble, 1999.

[Euzenat, 2000] J. Euzenat. XML est-il le langage de représentatation de connaissances de l'an 2000 ? In *6th Journées Langages et Modèles à Objets (LMO)*, pages 59–74, Mont-Saint-Hilaire, Québec, Canada, 26-28 Janvier 2000. Hermes.

[Euzenat, 2002] J. Euzenat. Eight Questions about Semantic Web Annotations. *IEEE Intelligent Systems*, 17(2) :55–62, Mars-Avril 2002.

[Farquhar *et al.*, 1995] A. Farquhar, R. Fikes, W. Pratt, and J. Rice. Collaborative Ontology Construction for Information Integration. Rapport de recherche KSL-95-63, Knowledge Systems Laboratory, Department of Computer Science, 1995.

[Fatemi and Abou Khaled, 2001] N. Fatemi and O. Abou Khaled. COALA : Content-Oriented Audiovisual Library Access. In *8th International Conference on Multimedia Modeling (MMM'2001)*, pages 59–71, Amsterdam, Pays-Bas, Novembre 2001.

[Fatemi, 2003] N. Fatemi. *A Semantic Views Model for Audiovisual Indexing and Retrieval.* PhD thesis, Ecole Polytechnique Fédérale de Lausanne, Suisse, 2003.

[Fayet-Scribe, 2000] S. Fayet-Scribe. *Histoire de la documentation en France - Culture, Science et Technologie de l'information : 1895-1937.* Collection CNRS Histoire. CNRS, Paris, 2000.

[Fensel and Gómez-Pérez, 2002] D. Fensel and A. Gómez-Pérez. A survey on ontology tools. OntoWeb Deliverable 1.3, 31 Mai 2002. `http://ontoweb.aifb.uni-karlsruhe.de/About/Deliverables/`.

[Fensel *et al.*, 1998a] D. Fensel, S. Decker, M. Erdmann, and R. Studer. How to make the WWW Intelligent. In *11th Workshop on Knowledge Acquisition for Knowledge-Based Systems (KAW98)*, Banff, Alberta, Canada, 18-23 Avril 1998.

[Fensel *et al.*, 1998b] D. Fensel, M. Erdmann, and R. Studer. Ontobroker : The Very High Idea. In *11th International Flairs Conference (FLAIRS'98)*, pages 131–135, Sanibal Island, Floride, USA, Mai 1998.

[Fensel *et al.*, 2000] D. Fensel, I. Horocks, F. van Harmelen, S. Decker, M. Erdmann, and M. Klein. OIL in a Nutshell. In R. Dieng and O. Corby, editors, *12th International Conference on Knowledge Engineering and Knowledge Managment (EKAW'00)*, volume (1937) of *Lecture Notes in Artificial Intelligence*, pages 1–16, Juan-les-Pins, France, 2-6 Octobre 2000. Springer Verlag.

[Fensel *et al.*, 2001] D. Fensel, I. Horrocks, F. van Harmelen, D.L. McGuinness, and P.F. Patel-Schneider. OIL : An Ontology Infrastructure for the Semantic Web. *IEEE Intelligent Systems*, 16(2) :38–45, 2001.

[Fernández and Arroyo, 1982] S. Fernández and I. Arroyo. Reflexiones sobre "Documento" : Palabra/objecto. *Boletin Millares Carlo*, 3 :161–197, 1982.

[Fernández-López *et al.*, 1997] M. Fernández-López, A. Gómez-Pérez, and N. Juristo. METHONTOLOGY : From Ontological Art Towards Ontological Engineering. In *Workshop on Ontological Engineering*, AAAI Spring Symposium Series (American Association for Artifical Intelligence), pages 33–40, Stanford, Californie, USA, Mars 1997.

[Fernández-López, 1999] M. Fernández-López. Overview of Methodologies for building Ontologies. In V.R. Benjamins, editor, *16th International Joint Conference on Artificial Intelligence (IJCAI'99), Workshop on Ontologies and Problem-Solving Methods : Lessons Learned and Future Trends*, volume (18) of *CEUR-WS*, Stockholm, Suède, 2 Août 1999. http://CEUR-WS.org/Vol-18/.

[Furuta, 1997] R. Furuta. What can Digital Libraries teach us about hypertext ? *ACM SIGLINK Newsletter*, 6(3) :7–9, 1997.

[Gandon, 2002] F. Gandon. *Distributed Artificial Intelligence and Knowledge Management : Ontologies and Multi-Agent Systems for a Corporate Semantic Web*. PhD thesis, Université de Nice, 2002.

[Genest, 2001] D. Genest. *Extension du modèle des graphes conceptuels pour la recherche d'informations*. PhD thesis, Université Montpellier II, 2001.

[Gómez-Pérez *et al.*, 2002] A. Gómez-Pérez, M. Fernández-López, and Ó. Corcho. Technical roadmap. OntoWeb Deliverable 1.1.2, 25 Novembre 2002. `http://ontoweb.aifb.uni-karlsruhe.de/About/Deliverables/`.

[Gómez-Pérez, 2000] A. Gómez-Pérez. Développements récents en matière de conception, de maintenance et d'utilisation d'ontologies. *Terminologies Nouvelles*, 19 :9–20, 2000. Traduit de l'anglais par *S. Descotte*.

[Goldfarb, 1990] C. Goldfarb. *The SGML Handbook*. Oxford University Press, 1990.

[Gruber, 1993] T.R. Gruber. A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition*, 5(2) :199–220, 1993.

[Guarino and Welty, 2000a] N. Guarino and C. Welty. A Formal Ontology of Properties. In R. Dieng and O. Corby, editors, *12th International Conference on Knowledge Engineering and Knowledge Managment (EKAW'00)*, volume (1937) of *Lecture Notes in Artificial Intelligence*, pages 97–112, Juan-les-Pins, France, 2-6 Octobre 2000. Springer Verlag.

[Guarino and Welty, 2000b] N. Guarino and C. Welty. Identity, Unity and Individuality : Towards a Formal Toolkit for Ontological Analysis. In W. Horn, editor, *14th The European Conference on Artificial Intelligence (ECAI'00)*, volume (54) of *Frontiers in Artificial Intelligence and Applications*, pages 219–223, Berlin, Allemagne, 20-25 Août 2000. IOS Press.

[Guarino, 1995] N. Guarino. Formal Ontology, Conceptual Analysis and Knowledge Representation. *International Journal of Human and Computer Studies*, 43(5/6) :625–640, 1995.

[Guarino, 1998] N. Guarino. Formal Ontology and Information Systems. In *First International Conference on Formal Ontology in Information Systems (FOIS'98)*, volume (46) of *Frontiers in Artificial Intelligence and Applications*, pages 3–15, Trento, Italie, 6-8 Juin 1998. IOS Press.

[Haarslev and Möller, 2001] V. Haarslev and R. Möller. Racer system description. In *International Joint Conference on Automated Reasoning (IJCAR'01)*, pages 701–705, Siena, Italie, 18-23 Juin 2001.

[Hendler and McGuinness, 2000] J. Hendler and D.L. McGuinness. The DARPA Agent Markup Language. *IEEE Intelligent Systems - Trends and Controversies*, 15(6) :67–73, Janvier 2000.

[Horrocks and Patel-Schneider, 2003] I. Horrocks and P.F. Patel-Schneider. Reducing OWL Entailment to Description Logic Satisfiability. In D. Fensel, K. Sycara, and J. Mylopoulos, editors, *2nd International Semantic Web Conference (ISWC'03)*, volume (2870) of *Lecture Notes in Computer Science*, pages 17–29, Sanibel Island, Floride, USA, 20-23 Octobre 2003. Springer Verlag.

[Horrocks, 1998] I. Horrocks. Using an Expressive Description Logic : FaCT or Fiction ? In *6th International Conference on Principles of Knowledge Representation and Reasoning (KR'98)*, pages 636–649, Trento, Italie, 2-5 Juin 1998. Morgan Kaufmann.

[Hunter and Iannella, 1998] J. Hunter and R. Iannella. The Application of Metadata Standards to Video Indexing. In *2nd European Conference on Research and Advanced Technology for Digital Libraries (ECDL'98)*, Crète, Grèce, 19-23 Septembre 1998. `http://archive.dstc.edu.au/RDU/staff/jane-hunter/ECDL2/final.html`.

[Hunter, 1999] J. Hunter. A Proposal for an MPEG-7 Description Definition Language (DDL). MPEG-7 AHG Test and Evaluation Meeting, Lancaster, Angleterre, 15-19 Février 1999. `http://archive.dstc.edu.au/RDU/staff/jane-hunter/mpeg7/p547.doc`.

[Hunter, 2001] J. Hunter. Adding Multimedia to the Semantic Web - Building an MPEG-7 Ontology. In *First International Semantic Web Working Symposium (SWWS'01)*, Stanford, Californie, USA, Août 2001.

[HyTime, 1997] HyTime. Information Technology - Hypermedia/Time based Structuring Language, seconde édition. Norme ISO/IEC n°10744, 1997. La première édition a été éditée en 1992.

[Isaac *et al.*, 2003] A. Isaac, R. Troncy, and V. Malaisé. Using XSLT for Interoperability : DOE and The Travelling Domain Experiment. In Y. Sure and O. Corcho, editors, *2nd International Workshop Evaluation of Ontology-based Tools (EON'03)*, volume (87) of *CEUR-WS*, pages 92–102, Sanibel Island, Floride, USA, 20 Octobre 2003. `http://CEUR-WS.org/Vol-87/`.

[Isaac, 2001] A. Isaac. Vers la mise en œuvre informatique d'une méthode de conception d'ontologies. Master's thesis, Institut des Sciences Humaines Appliquées, Université Paris IV, Paris, France, Septembre 2001.

[Karp *et al.*, 1999] P.D. Karp, V.K. Chaudhri, and J. Thomere. XOL : An XML-Based Ontology Exchange Language. Rapport de recherche, version 0.4, SRI International, 1999. `http://www.ai.sri.com/~pkarp/xol/`.

[Kayser, 1997] D. Kayser. *La représentation des connaissances.* Hermes, 1997.

[Kleene, 1971] S.C. Kleene. *Logique mathématique.* Armand Colin, 1971.

[Koenen, 1999] R. Koenen. MPEG-4 : Multimedia for our time. *IEEE Spectrum*, 36(2) :26–33, 1999.

[Landais *et al.*, 2004] R. Landais, C. Wolf, L. Vinet, and J.M. Jolion. Utilisation de connaissances a priori pour le paramétrage d'un algorithme de détection de textes dans les documents audiovisuels. Application à un corpus de journaux télévisés. In *14th Congrès Francophone de Reconnaissance des Formes et Intelligence Artificielle (RFIA'04)*, Toulouse, France, 28-30 Janvier 2004.

[Layaïda, 1997] N. Layaïda. *Madeus : Système d'édition et de présentation de documents structurés multimédia.* PhD thesis, Université Joseph Fourier, Grenoble, 1997.

[Le Roux, 2003] E. Le Roux. *Extraction d'information dans des textes libres guidée par une ontologie.* PhD thesis, Université Parix X - Nanterre, 2003.

[Luke *et al.*, 1997] S. Luke, L. Spector, D. Rager, and J. Hendler. Ontology-based Web Agents. In *1st International Conference on Autonomous Agents (Agents'97)*, pages 59–66, Marina del Rey, Californie, USA, 1997. ACM Press.

[Manjunath *et al.*, 2002] B.S. Manjunath, P. Salembier, and T. Sikora. *Introduction to MPEG-7, Multimedia Content Description Interface.* Wiley, 2002.

[Marshall, 1998] C.C. Marshall. Toward an Ecology of Hypertext Annotation. In *9th ACM Conference on Hypertext and hypermedia*, pages 40–49, Pittsburgh, Pennsylvanie, USA, 20-24 Juin 1998. ACM SIGLINK/SIGWEB.

[Martínez *et al.*, 2002] J.M. Martínez, R. Koenen, and F. Pereira. MPEG-7 : The Generic Multimedia Content Description Standard, Part 1. *IEEE MultiMedia*, 9(2) :78–87, Avril-Juin 2002.

[Martínez, 2002] J.M. Martínez. MPEG-7 : Overview of MPEG-7 Description Tools, Part 2. *IEEE MultiMedia*, 9(3) :83–93, Juillet-Septembre 2002.

[Menelas, 1994] Consortium Menelas. MENELAS : an access system for medical records using natural language. *Computer Methods and Programs in Biomedicine*, (45), 1994.

[Metz, 1968] C. Metz. *Essais sur la signification au cinéma.* tome 1. Klincksieck, Paris, France, 1968.

[Michard, 1998] A. Michard. *XML : langage et applications.* Eyrolles, Paris, France, 1998.

[Miller *et al.*, 2002] L. Miller, A. Seaborne, and A. Reggiori. Three Implementations of SquishQL, a Simple RDF Query Language. In I. Horrocks and J. Hendler, editors, *First International Semantic Web Conference (ISWC'02)*, volume (2342) of *Lecture Notes in Computer Science*, pages 423–435, Chia, Sardaigne, Italie, 9-12 Juin 2002. Springer Verlag.

[Minsky, 1975] M. Minsky. A framework for representing knowledge. In P. Henry Winston, editor, *The Psychology of Computer Vision.* McGraw-Hill, New York, USA, 1975.

[Miquel *et al.*, 1990] P. Miquel, M. Boon, A. Ryst, and C. Vinay. *Lexique de l'audiovisuel.* Dalloz, Paris, France, 1990.

[Möller *et al.*, 2003] R. Möller, R. Cornert, and V. Haarslev. Graphical interfaces for Racer : querying DAML+OIL and RDF documents. In *16th International Workshop on Description Logics (DL'03)*, volume (81) of *CEUR-WS*, Rome, Italie, 5-7 Septembre 2003. `http://CEUR-WS.org/Vol-81/`.

[MPEG-7, 2001] MPEG-7. Information Technology - Multimedia Content Description Interface. Norme ISO/IEC n°15938, Décembre 2001.

[Mugnier and Chein, 1996] M.L. Mugnier and M. Chein. Représenter des connaissances et raisonner avec des graphes. *Revue d'intelligence artificielle*, 10(1) :7–56, 1996.

[Nanard *et al.*, 2003] M. Nanard, J. Nanard, and P. King. IUHM : a hypermedia-based model for integrating open services, data and metadata (**Prix Engelbart**). In *14th International ACM Conference on Hypertext and Hypermedia (HT'03)*, pages 128–137, Nottingham, Angleterre, 26-29 Août 2003. ACM.

[Napoli, 1997] A. Napoli. Une introduction aux logiques de description. Rapport de Recherche 3314, INRIA Lorraine, Nancy, France, 1997. `ftp://ftp.inria.fr//INRIA/publication/publi-ps-gz/RR/RR-3314.ps.gz`.

[Nebel, 1990] B. Nebel. *Reasoning and Revision in Hybrid Representation Systems*, volume (422). Springer-Verlag, lecture notes in artificial intelligence edition, Juin 1990. PhD thesis, Universitat des Saarlandes, Saarbrucken, West Germany.

[Noy and Hafner, 1997] N.F. Noy and C.D. Hafner. The State of the Art in Ontology Design : A Survey and Comparative Review. *AI Magazine*, 18(3) :53–74, 1997.

[Noy *et al.*, 2000] N.F. Noy, R.W. Fergerson, and M.A. Musen. The knowledge model of Protégé2000 : Combining interoperability and flexibility. In R. Dieng and O. Corby, editors, *12th International Conference on Knowledge Engineering and Knowledge Managment (EKAW'00)*, volume (1937) of *Lecture Notes in Artificial Intelligence*, pages 17–32, Juan-les-Pins, France, 2-6 Octobre 2000. Springer Verlag.

[OWL, 2003] OWL. Web Ontology Language XML Presentation Syntax. W3C Note (11 Juin), 2003. `http://www.w3.org/TR/owl-xmlsyntax/`.

[OWL, 2004] OWL. Web Ontology Language Reference. W3C Recommendation (10 Février), 2004. `http://www.w3.org/TR/owl-ref/`.

[Pan and Horrocks, 2003] J.Z. Pan and I. Horrocks. Web Ontology Reasoning with Datatype Groups. In D. Fensel, K. Sycara, and J. Mylopoulos, editors, *2nd International Semantic Web Conference (ISWC'03)*, volume (2870) of *Lecture Notes in Computer Science*, pages 47–63, Sanibel Island, Floride, USA, 20-23 Octobre 2003. Springer Verlag.

[Patel-Schneider and Siméon, 2002] P.F. Patel-Schneider and J. Siméon. Building the Semantic Web on XML. In I. Horrocks and J. Hendler, editors, *First International Semantic Web Conference (ISWC'02)*, volume (2342) of *Lecture Notes in Computer Science*, pages 147–161, Chia, Sardaigne, Italie, 9-12 Juin 2002. Springer Verlag.

[Pédauque, 2003] Roger T. Pédauque. Document : forme, signe et relation, les re-formulations du numérique. Réseau Thématique Pluridisciplinaire "Document et contenu : création, indexation, navigation" (RTP-33), Département STIC du CNRS, Document de travail (8 juillet), 2003. `http://archivesic.ccsd.cnrs.fr/documents/archives0/00/00/05/11/index_fr%.html`.

[Pfeiffer and Srinivasan, 2000] S. Pfeiffer and U. Srinivasan. TV Anytime as an application scenario for MPEG-7. In *8th International Conference on Multimedia, ACM Multimedia 2000, Workshop on Standards, Interoperability and Practice*, Los Angeles, Californie, USA, Novembre 2000.

[Pichon, 1996] J. Pichon. Le traitement documentaire des programmes de radio et de télévision. Technical report, Inathèque, 1996.

[Prié and Garlatti, 2003] Y. Prié and S. Garlatti. Méta-données et annotations dans le Web sémantique. Technical report, Action Spécifique Web Sémantique, Rapport final, Chapitre 4, 2003.

[Prié, 1999] Y. Prié. *Modélisation de documents audiovisuels en Strates Interconnectées par les Annotations pour l'exploitation contextuelle.* PhD thesis, Institut National des Sciences Appliquées de Lyon, 1999.

[Quillian, 1968] M.R. Quillian. Semantic memory. In *Semantic Information Processing*, pages 227–270. MIT Press, 1968.

[Rastier *et al.*, 1994] F. Rastier, M. Cavazza, and A. Abeillé. *Sémantique pour l'analyse.* Masson, Paris, France, 1994.

[RDF-MT, 2004] RDF-MT. RDF Semantics, Model Theory. W3C Recommendation (10 Février), 2004. `http://www.w3.org/TR/rdf-mt/`.

[RDF, 2004] RDF. Ressource Description Framework Primer. W3C Recommendation (10 Février), 2004. `http://www.w3.org/TR/rdf-primer/`.

[RDFS, 2004] RDFS. RDF Vocabulary Description Language 1.0 : RDF Schema. W3C Recommendation (10 Février), 2004. `http://www.w3.org/TR/rdf-schema/`.

[SGML, 1986] SGML. Information Processing - Text and Office Systems - Standard Generalized Markup Language. Norme ISO/IEC n°8879, 1986.

[Simov and Jordanov, 2002] K. Simov and S. Jordanov. BOR : a Pragmatic DAML+OIL Reasoner. Deliverable 40, On-To-Knowledge Project, 2002.

[Sintek and Decker, 2002] M. Sintek and S. Decker. TRIPLE - A Query, Inference, and Transformation Language for the Semantic Web. In I. Horrocks and J. Hendler, editors, *First International Semantic Web Conference (ISWC'02)*, volume (2342) of *Lecture Notes in Computer Science*, pages 364–378, Chia, Sardaigne, Italie, 9-12 Juin 2002. Springer Verlag.

[SMIL, 1998] SMIL. Synchronized Multimedia Integration Language (SMIL 1.0) Specification. W3C Recommendation, 1998. `http://www.w3.org/TR/REC-smil`.

[SMIL2, 2001] SMIL2. Synchronized Multimedia Integration Language (SMIL 2.0). W3C Recommendation, 2001. `http://www.w3.org/TR/smil20`.

[Smith and Welty, 2001] B. Smith and C. Welty. Ontology : Towards a New Synthesis. In N. Guarino, editor, *Formal Ontology in Information Systems, Proceedings of the 2nd International Conference (FOIS-01)*, Ogunquit, Maine, USA, 17-19 Octobre 2001. ACM.

[Sowa, 1984] J. Sowa. *Conceptual structures : information processing in mind and machine.* Addisson-Wesley, Massachusset, USA, 1984.

[Staab and Maedche, 2000] S. Staab and A. Maedche. Ontology Engineering beyond the Modeling of Concepts and Relations. In V.R. Benjamins, A. Gomez-Perez, and N. Guarino, editors, *14th European Conference on Artificial Intelligence (ECAI'00), Workshop on Applications of Ontologies and Problem-Solving Methods*, Berlin, Allemagne, 21-22 Août 2000.

[Staab *et al.*, 2001] S. Staab, H.-P. Schnurr, R. Studer, and Y. Sure. Knowledge Processes and Ontologies. *IEEE Intelligent Systems, Special Issue on Knowledge Management*, 16(1) :26–34, 2001.

[Studer and Volz, 2003] A. Maedche B. Motik L. Stojanovic R. Studer and R. Volz. Ontologies for Enterprise Knowledge Management. *IEEE Intelligent Systems*, 18(2) :26–33, Mars-Avril 2003.

[Sure *et al.*, 2002] Y. Sure, M. Erdmann, J. Angele, S. Staab, R. Studer, and D. Wenke. OntoEdit : Collaborative Ontology Engineering for the Semantic Web. In I. Horrocks and J. Hendler, editors, *First International Semantic Web Conference (ISWC'02)*, volume (2342) of *Lecture Notes in Computer Science*, pages 221–235, Chia, Sardaigne, Italie, 9-12 Juin 2002. Springer Verlag.

[Tardif, 2000] L. Tardif. *Kaomi : réalisation d'une boite à outils pour la construction d'environnements d'édition de documents multimédia.* PhD thesis, Institut National Polytechnique de Grenoble, 2000.

[Terzi *et al.*, 2001] E. Terzi, A. Vakali, J. Fan, and M.-S. Hacid. The MPEG-7 Multimedia Content Description Standard and the XML Schema Language. In *7th International Conference on Distributed Multimedia Systems (DMS'01)*, Taipei, Taiwan, Septembre 2001.

[Tran Thuong, 2003] T. Tran Thuong. *Modélisation et traitement du contenu des médias pour l'édition et la présentation de documents multimédias.* PhD thesis, Institut National Polytechnique de Grenoble, 2003.

[Troncy and Isaac, 2002] R. Troncy and A. Isaac. DOE : une mise en oeuvre d'une méthode de structuration différentielle pour les ontologies. In *13th Journées Francophones d'Ingénierie des Connaissances (IC'02)*, pages 63–74, Rouen, France, 28-30 Mai 2002.

[Troncy, 2001] R. Troncy. Etude du Manuel d'Indexation commun à tous les documentalistes. Rapport de recherche, Institut National de l'Audiovisuel, Avril 2001.

[Troncy, 2003a] R. Troncy. Integrating Structure and Semantics into Audio-visual Documents. In D. Fensel, K. Sycara, and J. Mylopoulos, editors, *2nd International Semantic Web Conference (ISWC'03)*, volume (2870) of *Lecture Notes in Computer Science*, pages 566–581, Sanibel Island, Floride, USA, 20-23 Octobre 2003. Springer Verlag.

[Troncy, 2003b] R. Troncy. Le raisonnement dans les descriptions documentaires : l'apport de la représentation des connaissances (**Prix du meilleur article**). In *14th Journées Francophones d'Ingénierie des Connaissances (IC'03)*, pages 161–176, Laval, France, 1-4 Juillet 2003. Presses Universitaires de Grenoble (PUG).

[Ullman, 1996] J.D. Ullman. The Database Approach to Knowledge Representation. In *13th National Conference of the American Association for Artificial Intelligence (AAAI'96).* MIT Press, 1996.

[Uschold and Gruninger, 1996] M. Uschold and M. Gruninger. Ontologies : Principles, Methods, and Applications. *Knowledge Engineering Review*, 2 :93–155, 1996.

[van Ossenbruggen *et al.*, 2001] J. van Ossenbruggen, J. Geurts, F. Cornelissen, L. Hardman, and L. Rutledge. Towards Second and Third Generation Web-based Multimedia. In *10th International World Wide Web Conference (WWW'01)*, pages 479–488, Hong Kong, 1-5 Mai 2001. ACM Press.

[van Slype, 1987] G. van Slype. *Les langages d'indexation : conception, construction et utilisation dans les systèmes documentaires.* Editions d'organisation, Paris, France, 1987.

[Vanoye and Goliot-Lété, 2001] F. Vanoye and A. Goliot-Lété. *Précis d'analyse filmique*, volume 17 of *Cinéma 128.* Nathan, France, 2001.

[Veneau, 2002] E. Veneau. *Macro-segmentation multi-critère et classification des séquences par le contenu dynamique pour l'indexation vidéo.* PhD thesis, Université de Rennes I, 2002.

[Vercoustre, 2002] A.-M. Vercoustre. Éléments de métadonnées du Dublin Core, Version 1.1 : Description de Référence. Traduction en français, 2002. `http://www-rocq.inria.fr/~vercoust/METADATA/DC-fr.1.1.html`.

[Villard, 2002] L. Villard. *Modèles de documents pour l'édition et l'adaptation de présentations multimédias.* PhD thesis, Institut National Polytechnique de Grenoble, 2002.

[Winston *et al.*, 1987] M. Winston, R. Chaffin, and D. Herrmann. A taxonomy of part-whole relationships. *Cognitive Science*, 11(4) :417–444, 1987.

[XML NS, 1999] XML NS. Namespaces in XML. W3C Recommendation, 1999. `http://www.w3.org/TR/REC-xml-names/`.

[XML Schema, 2001] XML Schema. W3C Recommendation, 2001. `http://www.w3.org/XML/Schema`.

[XML, 2000] XML. Extensible Markup Language 1.0, Second Edition. W3C Recommendation, 2000. `http://www.w3.org/TR/REC-xml`.

[XPath, 1999] XPath. XML Path Language Version 1.0. W3C Recommendation, 1999. `http://www.w3.org/TR/xpath`.

[XPointer, 2001] XPointer. XML Pointer Language Version 1.0. W3C Recommendation, 2001. `http://www.w3.org/TR/xpointer`.

[XSLT, 1999] XSLT. XSL Transformations. W3C Recommendation, 1999. `http://www.w3.org/TR/xslt`.

[Zettl, 1999] H. Zettl. *Sight, Sound, Motion : Applied Media Aesthetics.* Wadsworth, third edition, 1999.

Résumé

La manipulation de contenus audiovisuels est une tâche à la fois complexe et spécifique, qui nécessite le plus souvent de recourir à des représentations médiatrices. La nature temporelle de l'audiovisuel impose de passer par le biais de la description pour enrichir les documents et donc les exploiter. La numérisation des documents audiovisuels permet d'envisager de nouvelles exploitations des contenus telles que leur recherche « intelligente », leur recomposition dynamique ou la personnalisation de leur accès. Le système technique qui rend ces services doit alors être intégré et faire le lien entre le contenu et sa description. Nous soutenons qu'une représentation de la *structure* et du *contenu* des documents est nécessaire. Par structure, nous entendons la structure documentaire c'est-à-dire l'organisation méréologique des éléments qui composent le document, tandis que le contenu est une structure conceptuelle, c'est-à-dire une caractérisation de ces éléments. Cette double représentation fait ressortir le besoin d'un format de description homogène et exploitable par la machine, à la fois expressif et optimal en terme de manipulations. Après une revue des propositions actuelles de modélisation des documents audiovisuels, issues de l'ingénierie documentaire et de l'ingénierie des connaissances, nous montrons qu'aucun des langages étudiés ne permet de traiter ces deux aspects de manière satisfaisante. Nous proposons alors une architecture générale permettant la représentation formelle de la structure et du contenu des documents audiovisuels, qui engendrera une base de connaissances sur laquelle il est possible d'effectuer des raisonnements. Cette architecture se compose d'une ontologie de l'audiovisuel, dont on traduit une partie dans un langage documentaire pour contrôler la structure logique des documents, et d'une ontologie de domaine pour décrire formellement leur contenu. Deux ontologies ont donc été modélisées : l'ontologie générique de l'audiovisuel et une ontologie du cyclisme qui est le domaine d'application de notre architecture. Nous avons développé pour cela l'outil DOE (*Differential Ontology Editor*), qui implémente la méthodologie de construction d'ontologies utilisée. Nous montrons finalement la pertinence de l'approche à l'aide de deux expérimentations utilisant un corpus de vidéos annoté et pour lesquelles une implémentation de la base de connaissances est proposée, illustrant ainsi les types d'inférences possibles.

Mots-clés: Modélisation de documents et de contenus audiovisuels, ingénierie documentaire, ingénierie des connaissances, construction d'ontologies, raisonnement, OWL, RDF et MPEG-7.

Abstract

The manipulation of audio-visual documents is both a difficult and a specific task that requires to deal with intermediary representations. The temporal nature of audio-visual force to use a description to enrich the documents with metadata and then exploit them. The digitalizing of audio-visual documents allows new use of the content such as "intelligent" search, dynamic recomposition, or personalized access. The technical system that produces these services has to be integrated and must link the content with its description. We defend that a representation of both the *structure* and the *content* of the documents is necessary. By structure, we mean the documentary structure, that is the mereological organization of the elements that compose the document, whereas the content refers to a conceptual structure, that is a characterization of these elements. This double representation highlights the need for a homogeneous description format that is usable, very expressive and efficient for the machine. After a review of the current propositions for modeling audio-visual documents, coming from the document engineering side and the knowledge representation side, we demonstrate that none of the studied languages can deal with these two aspects in a satisfactory way. We then propose a general architecture that allows to describe formally the structure and the semantics of audio-visual documents and that will generate a knowledge base on which we will perform some reasoning. This architecture is composed of an audio-visual ontology, whose a part is translated into a documentary language in order to control the logical structure of the document, and a domain ontology for describing the semantics of the content. Consequently, two ontologies have been built: the generic audio-visual ontology and a cycling ontology which is the application domain of our architecture. We have developed the DOE tool (*Differential Ontology Editor*) that implements the chosen methodology for building the ontologies. Finally, we demonstrate the relevance of our approach with two experiments on an annotated corpus of video and for which an implementation of knowledge bases is proposed, showing hence the kind of inference we can make.

Keywords: Audio-visual documents and contents modeling, document engineering, knowledge representation, ontologies building, reasoning, OWL, RDF and MPEG-7.