



HAL
open science

Contribution à la modélisation des métadonnées associées aux documents multimédias et à leur enrichissement par l'usage

Ana-Maria Manzat

► **To cite this version:**

Ana-Maria Manzat. Contribution à la modélisation des métadonnées associées aux documents multimédias et à leur enrichissement par l'usage. Recherche d'information [cs.IR]. Institut National Polytechnique de Toulouse - INPT, 2013. Français. NNT : 2013INPT0003 . tel-04225298

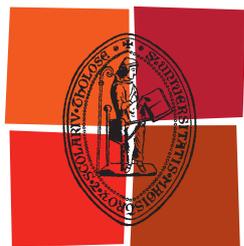
HAL Id: tel-04225298

<https://theses.hal.science/tel-04225298>

Submitted on 2 Oct 2023

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.



Université
de Toulouse

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :

Institut National Polytechnique de Toulouse (INP Toulouse)

Présentée et soutenue par :

Ana-Maria MANZAT

le mardi 5 février 2013

Titre :

Contribution à la Modélisation des Métadonnées Associées aux Documents
Multimédias et à leur Enrichissement par l'Usage

École doctorale et discipline ou spécialité :

ED MITT : Image, Information, Hypermedia

Unité de recherche :

Institut de Recherche en Informatique de Toulouse (IRIT - UMR 5505)

Directeur(s) de Thèse :

Florence SEDES, Université Paul Sabatier –Toulouse, Directrice de thèse
Romulus GRIGORAS, Institut National Polytechnique de Toulouse, Co-Directeur de thèse

Jury :

Richard CHBEIR, Université de Pau et des Pays de l'Adour, Rapporteur
Harald KOSCH, Université de Passau, Rapporteur
Vincent CHARVILLAT, Institut National Polytechnique de Toulouse, Président
Thierry DELOT, Université de Valenciennes, Examineur
Florence SEDES, Université Paul Sabatier –Toulouse III, Directrice de thèse
Romulus GRIGORAS, Institut National Polytechnique de Toulouse, Co-Directeur de thèse

Résumé

Les collections multimédias qui deviennent de plus en plus volumineuses, aidées en cela par le volume des métadonnées qui les décrivent. L'extraction des métadonnées devient de fait très coûteuse en consommation de ressources. Cela pose le problème de la gestion efficace de ces grands volumes de données, en minimisant cette consommation. Le fait que les utilisateurs sont en constante interaction avec les documents multimédias et les métadonnées complique encore plus cette gestion. Dans cette thèse, nous étudions le problème de la gestion de métadonnées en intégrant les interactions des utilisateurs à deux niveaux : dans le processus de création de métadonnées et dans leur enrichissement.

La grande variété de standards et normes de métadonnées existants ne sont quasiment pas interopérables (i.e., dans le domaine de la vidéosurveillance). Les solutions proposées à ce problème d'interopérabilité se sont focalisées sur la création d'ontologies qui décrivent les contenus multimédias du point de vue sémantique, sans forcément prendre en compte les standards de métadonnées et d'autres informations de plus bas niveau sur les documents. Pour aborder ce problème, nous proposons un format de métadonnées qui intègre les standards et normes les plus utilisés et qui soit flexible et extensible en structure et en vocabulaire.

Dans le cadre d'un système de gestion des contenus multimédias, le processus d'indexation est celui qui consomme le plus de ressources, à travers les algorithmes d'indexation qui extraient les métadonnées. Dans les systèmes classiques, cette indexation est accomplie avec un ensemble d'algorithmes d'indexation figé dans le temps, sans se soucier de la consommation des ressources ni de l'évolution des besoins de l'utilisateur. Pour prendre en compte les besoins que l'utilisateur spécifie dans sa requête, afin de n'extraire que les métadonnées nécessaires et ainsi limiter d'un côté le volume de métadonnées à gérer et de l'autre la consommation des ressources, nous proposons de répartir le processus d'indexation en deux phases: une fois à l'acquisition des contenus (indexation implicite), et une deuxième fois, si besoin, au moment de l'exécution de la requête de l'utilisateur (indexation explicite) en ayant recours à une liste d'algorithmes d'indexation déterminée en fonction de la requête de l'utilisateur.

L'utilisateur est de plus en plus pris en compte dans les systèmes multimédias, ses interactions avec le système et le document étant analysées pour identifier ses intérêts. Nous proposons d'aller plus loin dans la prise en compte de l'utilisateur, en considérant ses interactions avec les différentes parties du document mais aussi avec les métadonnées qui décrivent le document. Cela a été réalisé à travers l'introduction du concept de température pour chaque élément du format, qui varie dans le temps, calculée en fonction de la façon dont l'utilisateur interagit avec le document, mais aussi avec les métadonnées dans une période de temps donnée.

Nous avons validé nos propositions dans deux domaines différents : celui de la vidéosurveillance et celui du commerce électronique. Le projet LINDO nous a permis la validation du format des métadonnées, du résumé des métadonnées et de la sélection des algorithmes d'indexation dans le cadre de l'indexation explicite. Dans le domaine du commerce électronique, nous avons exploité les interactions des utilisateurs réels avec un site de vente en ligne pour calculer la température des métadonnées associées aux pages du site pendant une période de deux mois. Nous avons utilisé cette température pour réaliser le reclassement des résultats obtenus pour une requête de l'utilisateur. Les tests utilisateur montrent que pour certaines requêtes de l'utilisateur ce reclassement des résultats aide à localiser les informations recherchées plus vite.

Ce travail a permis de répondre au problème de la prise compte de l'utilisateur dans le processus de gestion des documents multimédias, en proposant : un modèle de métadonnées qui intègre les standards de métadonnées les plus utilisés ; l'indexation différée des contenus multimédias (indexation implicite et explicite); l'enrichissement des métadonnées en considérant les interactions des utilisateurs avec le système, les documents multimédias et les métadonnées.

Mots-clés:

Documents et métadonnées multimédias, processus d'indexation, usage, interactions des utilisateurs, température

Abstract

Nowadays, not only multimedia collections become larger, but also the metadata describing them. The metadata extraction is the most resource consuming process in the management of multimedia collections. This raises the problem of the efficient management of these large data volumes while minimizing resource consumption. Users' constant interactions with multimedia documents and metadata complicate this management process. In this thesis, we address this problem of metadata management by integrating users' interactions at two levels: in the process of metadata creation and in their enrichment.

The existing metadata standards are heterogeneous and not interoperable. The proposed solutions for this interoperability problem focused on creating ontologies that describe the multimedia contents from a semantic point of view, without necessarily taking into account metadata standards and other low level information. To solve this problem, we propose a metadata format that integrates the most widely used metadata standards and which is flexible and extensible in structure and vocabulary.

In a multimedia management system, the indexing process is the most resource consuming, through the indexing algorithms that extract metadata. In conventional systems, the indexing is accomplished with a fixed set of indexing algorithms, without considering the resource consumption and users' changing needs. To take into account the user's needs, specified in his query, in order to extract only the necessary metadata and thus, on one side, to limit the metadata volume and on the other to reduce the resource consumption, we propose to split the indexing process into two phases: first time, at the contents acquisition time (i.e., implicit indexation), and, a second time, if necessary, at the query execution time (i.e., explicit indexation), employing a list of indexing algorithms determined mainly according to the user's query.

The users are more and more taken into account in multimedia systems through their interactions with the system and the documents. We propose to go further in this consideration, by taking into account users' interactions with different parts of the document, and also with the document's metadata. This was achieved through the extension of the proposed metadata format, by associating a temperature to each metadata element. This temperature is calculated according to the users' interactions with the document and with the metadata, in a time period.

We have validated our proposals in two different domains: vidéosurveillance and e-commerce. The LINDO project has allowed us to validate the metadata format and indexing algorithms selection in the context of explicit indexation, for a video surveillance use case. For the e-commerce, we have used an online shopping site and the interactions of its real users, for a two months period, to calculate the temperature of the metadata associated to the web pages describing the site's products. We have used this temperature for reranking the results obtained for a user's query. We conducted a user study with twenty people, which shows that, for some users' queries, the results reranking helps the users to find faster the desired information.

This thesis has addressed the problem of taking into account the user in the multimedia documents management by: (1) proposing a model metadata that integrates the most used metadata standards; (2) splitting the multimedia indexing in two steps (implicit and explicit indexation); (3) enriching the metadata according to the users' interactions with the system, the multimedia documents and the metadata.

Key words:

Multimedia documents, multimedia metadata, indexing process, usage, users' interactions, temperature

Remerciements

Je remercie de tout mon cœur Madame Florence Sèdes, directrice de cette thèse, pour m'avoir prise sous son aile protectrice, pour sa patience, sa disponibilité et ses conseils toujours très pertinents et constructifs. Dans mes moments d'égarement, elle a toujours su comment me remettre sur la bonne voie et surtout me faire prendre du recul pour mener à bien mes travaux de recherche. Grâce à elle, j'ai réussi à avancer dans mes réflexions et à faire de cette thèse ce qu'elle est aujourd'hui.

Mes remerciements vont également à Monsieur. Romulus Grigoras, co-encadrant de cette thèse. Je le remercie pour son aide et ses conseils. Sans lui, cette thèse n'aurait pas été possible : dans un premier temps, il m'a permis de venir en France, à Toulouse, et dans un second temps, il m'a permis de rencontrer Mme Sèdes. Il a toujours trouvé du temps pour répondre à toutes mes questions d'ordre pratique et technique.

Je remercie également Mademoiselle Mihaela Brut et Monsieur Sébastien Laborie pour m'avoir fait partager leur expérience au fil de nos longues et nombreuses réunions. Leurs nombreux conseils m'ont aidée à progresser et à affiner mes contributions.

Je tiens à remercier Monsieur Richard Chbeir et Monsieur Harald Kosch d'avoir accepté d'être rapporteurs de ce mémoire. Leurs lectures précises de ce manuscrit et leurs remarques judicieuses m'ont permis d'améliorer la qualité de ce document.

Je tiens à remercier Monsieur Thierry Delot et Monsieur Vincent Charvillat pour l'intérêt qu'ils ont porté à mes travaux et pour l'honneur qu'ils me font en participant à ce jury.

Mes remerciements vont aussi à Monsieur Jean-François Sulzer, chef du projet LINDO. Travailler sous sa direction a été pour moi un très grand privilège. Je le remercie d'avoir accepté de faire partie de mon jury de thèse.

J'adresse mes sincères remerciements à Monsieur Claude Chrisment et Madame Josiane Mothe, responsables de l'équipe SIG, qui m'ont accueillie au sein de l'équipe. Je remercie particulièrement Madame Marie-Françoise Canut, Madame Nadine Jessel et Monsieur André Péninou pour leur disponibilité, leur grande patience, mais surtout pour leurs avis et conseils qui m'ont permis de faire évoluer et d'améliorer mon travail.

Je remercie également tous les membres du projet LINDO, Gregoire Goloubinow, Jean-Pierre Guignard, Michel Le Cam, Philippe Maurel, David Millan, Pedro Moreira, Olivier Pietquin, Jean-François Seignole,.... Travailler à leur côté a été une expérience très enrichissante tant d'un point de vue professionnel que personnel.

Je remercie à la société Devatics et Monsieur Jean-Paul Crenn, pour avoir mis à ma disposition les jeux de données qui m'ont permis de valider mes propositions.

Mes remerciements vont aussi à tous les membres de l'équipe SIG et à tout le personnel du laboratoire pour leur aide et leur gentillesse.

J'adresse également mes remerciements à tous les doctorants (Valentin Butoianu, Adrian Chifu, Dana Codreanu, Damien Dudognon, Philippe Ercolessi, Liana Ermakova, Imen Megdiche, Manel Mezghani, Laure Soulier, ...) avec qui j'ai partagé de bons moments au bureau, à la salle machine, aux pauses café, au RU, ...

Je n'oublie pas non plus les docteurs récents ou plus anciens (Dana Al-Kukhun, Benoit Baccot, Guillaume Cabanac, Gregory Claude, Sylvain Cussat-Blanc, Anass El Haddadi, Anthony Pajot, Viorica Patraucean, Bouchra Soukkarieh, Dieudonné Tchuenta,...).

Merci aussi aux stagiaires Andra Codreanu et Andrei Vucicu.

Je remercie également tous mes amis qui ont été à mes côtés tout au long de cette étape de ma vie : Alina, Cornelia, Valentina, Monia, Mathieu, Delphin, Eric, Rosie, ... Un grand merci à Sylvie pour tout son aide.

Je remercie du fond du cœur, mes parents, qui n'ont pas cessé de me soutenir et de croire en moi pendant toutes mes années d'études. Je remercie aussi tous les membres de ma famille (mes grands-mères, mon cousin et son épouse, mon neveu, mon oncle) qui ont toujours été proches de moi malgré la distance qui nous sépare.

Un très très grand merci à Ovidiu qui a su m'épauler et me soutenir tout au long de ces années, même dans les moments les plus difficiles.

Table des matières

RÉSUMÉ	I
ABSTRACT	III
REMERCIEMENTS	V
TABLE DES MATIÈRES	VII
LISTE DES FIGURES	XI
LISTE DES TABLEAUX	XIII
INTRODUCTION GÉNÉRALE	1
LE CONTEXTE	1
LA PROBLÉMATIQUE.....	3
LES CONTRIBUTIONS	4
L'ORGANISATION DU MÉMOIRE.....	7
PARTIE I : L'ÉTAT DE L'ART	9
INTRODUCTION.....	11
CHAPITRE 1. LES DOCUMENTS ET LES MÉTADONNÉES MULTIMÉDIAS	13
1.1. Introduction	13
1.2. Le document numérique.....	14
1.2.1. La définition du terme multimédia.....	14
1.2.2. Le texte	15
1.2.3. L'image.....	16
1.2.4. L'audio.....	16
1.2.5. La vidéo	16
1.2.6. Le document interactif.....	17
1.2.7. Le cycle de vie des documents multimédias	17
1.2.8. Bilan sur les documents multimédias.....	18
1.3. Les documents multimédias multistrués	19
1.3.1. La structure physique.....	19
1.3.2. La structure logique	19
1.3.3. La structure sémantique	19
1.3.4. La structure spatiale	20
1.3.5. La structure temporelle	20
1.3.6. La structure hypermédia	21
1.3.7. Bilan sur les documents multimédias multistrués	21
1.4. Les métadonnées multimédias.....	21
1.4.1. La définition.....	21
1.4.2. La classification des métadonnées	22
1.4.3. Le cycle de vie des métadonnées	24
1.4.4. Les standards et formats de métadonnées	25
1.4.4.1. Les standards de métadonnées pour l'image	25
1.4.4.2. Les standards de métadonnées pour le texte.....	26
1.4.4.3. Les standards de métadonnées pour les contenus audiovisuels	27
1.4.4.4. Les standards généraux de métadonnées	29
1.4.5. La représentation des métadonnées.....	30
1.4.6. Bilan sur les métadonnées multimédias	31
1.5. Bilan.....	33
CHAPITRE 2. LA GESTION DES DOCUMENTS ET DES MÉTADONNÉES MULTIMÉDIAS.....	37
2.1. Introduction	37
2.2. Le processus de gestion des documents multimédias.....	38
2.2.1. L'indexation des documents multimédias.....	38

2.2.2.	La recherche des documents multimédias.....	40
2.3.	<i>La gestion distribuée des documents multimédias</i>	41
2.3.1.	Les systèmes orientés services.....	41
2.3.1.	Les systèmes Pair-à-Pair.....	43
2.3.2.	Comparaison entre les systèmes distribués de gestion des documents multimédias.....	44
2.4.	<i>La gestion des métadonnées</i>	45
2.4.1.	L'interopérabilité des métadonnées.....	46
2.4.2.	L'hétérogénéité des formats de métadonnées et l'interopérabilité.....	46
2.4.3.	Les solutions existantes aux problèmes d'interopérabilité et d'hétérogénéité des métadonnées.....	49
2.5.	<i>Bilan</i>	51
CHAPITRE 3.	L'UTILISATEUR ET LES SYSTÈMES DE GESTION DES DOCUMENTS MULTIMÉDIAS.....	53
3.1.	<i>Introduction</i>	53
3.2.	<i>L'interaction de l'utilisateur avec un document</i>	53
3.3.	<i>L'analyse du comportement de l'utilisateur</i>	55
3.3.1.	Les intérêts de l'utilisateur.....	55
3.3.2.	Les intentions de l'utilisateur.....	57
3.4.	<i>La prise en compte de l'utilisateur dans les systèmes de gestion de documents multimédias</i>	58
3.4.1.	L'amélioration du processus de gestion des documents multimédias.....	58
3.4.2.	Les interfaces hommes-machine.....	60
3.4.3.	L'enrichissement de la requête.....	61
3.4.4.	Les systèmes de recommandations et de filtrage collaboratif.....	61
3.4.5.	L'adaptation de la présentation.....	62
3.5.	<i>L'utilisateur, générateur de métadonnées</i>	62
3.6.	<i>Bilan</i>	64
PARTIE 2 : LES CONTRIBUTIONS	67	
INTRODUCTION.....	69	
CHAPITRE 4.	LE FORMAT GÉNÉRIQUE DE MÉTADONNÉES.....	73
4.1.	<i>Introduction</i>	73
4.2.	<i>Les principes de l'approche proposée</i>	74
4.3.	<i>La modélisation proposée pour GMF</i>	74
4.4.	<i>Les métadonnées générales</i>	76
4.5.	<i>Les métadonnées de localisation</i>	78
4.6.	<i>Les métadonnées spécifiques à chaque type de média</i>	79
4.6.1.	Les métadonnées pour les documents textuels.....	79
4.6.2.	Les métadonnées pour les images.....	81
4.6.3.	Les métadonnées pour les vidéos.....	83
4.6.4.	Les métadonnées pour les documents audio.....	84
4.7.	<i>Les structures temporelle, spatiale et hypermédia</i>	86
4.8.	<i>La représentation d'une couche sémantique des métadonnées</i>	87
4.9.	<i>Bilan</i>	89
CHAPITRE 5.	LA PRISE EN COMPTE DE L'USAGE DANS LA GESTION DES MÉTADONNÉES.....	91
5.1.	<i>Introduction</i>	91
5.2.	<i>La prise en compte de l'usage dans la création des métadonnées</i>	93
5.2.1.	L'indexation implicite et explicite des documents multimédias.....	93
5.2.2.	Le résumé des métadonnées.....	97
5.2.3.	La description adoptée pour un serveur distant.....	99
5.2.4.	L'algorithme de sélection des algorithmes d'indexation.....	101
5.2.5.	La prise en compte des conditions d'acquisition des contenus multimédias pour la sélection des algorithmes d'indexation.....	105
5.2.6.	La prise en compte de l'historique des requêtes des utilisateurs pour la sélection des algorithmes d'indexation.....	107
5.3.	<i>L'enrichissement des métadonnées par l'usage</i>	108
5.3.1.	L'enrichissement du format de métadonnées pour inclure l'usage.....	108
5.3.1.1.	Le concept de température.....	109

5.3.1.2.	Les métadonnées pour définir la structure d'usage	111
5.3.2.	Les structures dynamiques des documents multimédias	113
5.3.3.	L'algorithme d'enrichissement	114
5.3.3.1.	L'augmentation de la température.....	114
5.3.3.2.	Le refroidissement de la température	117
5.3.4.	L'application de l'algorithme d'augmentation de la température	119
5.3.4.1.	Les requêtes de l'utilisateur.....	120
5.3.4.2.	L'interaction de l'utilisateur avec les métadonnées.....	121
5.3.4.3.	L'interaction de l'utilisateur avec la liste des résultats.....	123
5.3.4.4.	L'interaction de l'utilisateur avec le document multimédia	124
5.3.4.5.	L'algorithme général pour le calcul de la température	127
5.3.5.	Un exemple d'évolution de la température	130
5.3.6.	L'utilisation de la température et de la structure d'usage.....	133
5.4.	<i>Bilan</i>	135
PARTIE 3 : LES VALIDATIONS EXPÉRIMENTALES		137
INTRODUCTION.....		139
CHAPITRE 6. IMPLÉMENTATION D'UN SYSTÈME DISTRIBUÉ DE GESTION DES DOCUMENTS MULTIMÉDIAS.....		141
6.1.	<i>Introduction</i>	141
6.2.	<i>L'architecture générique du système proposée dans le cadre du projet</i>	141
6.2.1.	Les composants des serveurs distants	143
6.2.2.	Les composants du serveur central	144
6.3.	<i>L'architecture et l'implémentation du démonstrateur final du projet</i>	145
6.3.1.	L'architecture logicielle du système	146
6.3.2.	Les descriptions adoptées pour les ressources	148
6.3.2.1.	La description des serveurs distants	148
6.3.2.2.	La description des algorithmes d'indexation.....	148
6.3.2.3.	La description des contenus multimédias	149
6.3.3.	Les algorithmes d'indexation disponibles.....	150
6.3.3.1.	L'algorithme d'indexation audio.....	150
6.3.3.2.	Les algorithmes d'indexation vidéo	151
6.3.3.3.	L'algorithme speech to text.....	152
6.4.	<i>L'implémentation du Metadata Engine</i>	153
6.4.1.	L'intégration des métadonnées	153
6.4.2.	Le résumé des métadonnées.....	154
6.4.3.	L'exécution d'une requête	157
6.5.	<i>Bilan</i>	158
CHAPITRE 7. L'ENRICHISSEMENT DES MÉTADONNÉES PAR L'USAGE.....		159
7.1.	<i>Introduction</i>	159
7.2.	<i>L'architecture du prototype proposé</i>	160
7.3.	<i>L'analyse du site web et de la base de données d'interactions</i>	161
7.3.1.	Le choix des mots clés	161
7.4.	<i>La production des métadonnées</i>	167
7.5.	<i>La production des métadonnées d'usage et leur mise à jour continue</i>	169
7.6.	<i>L'exploitation des métadonnées</i>	171
7.7.	<i>Les expérimentations</i>	172
7.8.	<i>L'évaluation des résultats</i>	172
7.9.	<i>Bilan</i>	178
CONCLUSIONS GÉNÉRALES ET PERSPECTIVES		179
SYNTHÈSE DES PROPOSITIONS		179
PERSPECTIVES DE RECHERCHE		180
RÉFÉRENCES		183
ANNEXES		195

Liste des Figures

FIGURE 1 – LE PROCESSUS DE GESTION DES DOCUMENTS MULTIMÉDIAS	2
FIGURE 2 – LE POSITIONNEMENT DE NOS CONTRIBUTIONS DANS LE PROCESSUS DE GESTION DES DOCUMENTS MULTIMÉDIAS	6
FIGURE 3 – LE PLAN DU MÉMOIRE	7
FIGURE 4 – LE CYCLE DE VIE D'UN DOCUMENT MULTIMÉDIA (DE LA PRODUCTION À LA CONSOMMATION), SELON (HARDMAN ET AL., 2008)	18
FIGURE 5 – LE CYCLE DE VIE D'UN DOCUMENT MULTIMÉDIA ET LE RÔLE CENTRAL DES MÉTADONNÉES	24
FIGURE 6 – LA PILE DU WEB SÉMANTIQUE, SELON TIM BERNERS-LEE	31
FIGURE 7 – LE PROCESSUS DE RECHERCHE D'INFORMATION.....	38
FIGURE 8 – UNE COMPARAISON ENTRE STANDARDS DE MÉTADONNÉES POUR LES IMAGES (METADATA WORKING GROUP, 2009) ..	47
FIGURE 9 – LA DIMENSION IMPLICITE/EXPLICITE DE L'INTÉRÊT D'UN UTILISATEUR.....	56
FIGURE 10 – LE PROCESSUS ÉTENDU DE GESTION DES DOCUMENTS MULTIMÉDIAS	59
FIGURE 11 – LE SCHÉMA GÉNÉRAL DU TRAITEMENT DE L'INFORMATION DANS UN SYSTÈME ADAPTATIF	60
FIGURE 12 – LE FORMAT DE MÉTADONNÉES PROPOSÉ	75
FIGURE 13 – LE DIAGRAMME DES MÉTADONNÉES GÉNÉRALES	76
FIGURE 14 – LE DIAGRAMME DE LA CLASSE LOCALISATION.....	78
FIGURE 15 – LES MÉTADONNÉES ASSOCIÉES AUX DOCUMENTS TEXTUELS	80
FIGURE 16 – LES MÉTADONNÉES ASSOCIÉES AUX IMAGES	82
FIGURE 17 – LES MÉTADONNÉES ASSOCIÉES AUX VIDÉOS	83
FIGURE 18 - LES MÉTADONNÉES ASSOCIÉES AUX CONTENUS AUDIO	85
FIGURE 19 – LES MÉTADONNÉES DE STRUCTURE.....	86
FIGURE 20 – LES MÉTADONNÉES DÉCRIVANT LES OBJETS IDENTIFIÉS DANS UN DOCUMENT	88
FIGURE 21 – LE DIAGRAMME DU TRAITEMENT DE LA REQUÊTE.....	94
FIGURE 22 – LE DIAGRAMME DE SÉQUENCE POUR : A) L'INDEXATION IMPLICITE ET B) L'INDEXATION EXPLICITE.....	95
FIGURE 23 – EXTRAIT D'UNE DESCRIPTION RDF DES MÉTADONNÉES ASSOCIÉES À DES CONTENUS MULTIMÉDIAS STOCKÉS SUR UN SERVEUR (SERVEUR ₁)	98
FIGURE 24 – UNE DESCRIPTION RDF RÉSUMÉE DE LA FIGURE 23 AVEC $0 < \text{DEGRÉ} \leq 2$	99
FIGURE 25 – LE MODÈLE DE DESCRIPTION DES SERVEURS DISTANTS	99
FIGURE 26 - L'ARBRE D'EXÉCUTION DE L'ALGORITHME DE SÉLECTION DES ALGORITHMES D'INDEXATION	104
FIGURE 27 - LE MODÈLE DE MÉTADONNÉES ÉTENDU PAR LA TEMPÉRATURE	109
FIGURE 28 – LE MODÈLE DE MÉTADONNÉES ÉTENDU PAR LA STRUCTURE D'USAGE.....	112
FIGURE 29 – LES INTERACTIONS DE L'UTILISATEUR AVEC UN SYSTÈME DE GESTION DE DOCUMENTS MULTIMÉDIAS.....	119
FIGURE 30 – LA STRUCTURE TEMPORELLE D'UNE PRÉSENTATION SMIL	127
FIGURE 31 – L'ARBRE DE MÉTADONNÉES ET LA SUCCESSION D'INTERACTIONS PRISE EN COMPTE.....	130
FIGURE 32 – L'ÉTAT DE L'ARBRE APRÈS LE CALCUL DE LA TEMPÉRATURE À T=1	131
FIGURE 33 – L'ÉTAT FINAL DE L'ARBRE APRÈS LE CALCUL DE LA TEMPÉRATURE POUR T=2	132
FIGURE 34 – L'ÉTAT FINAL DE L'ARBRE APRÈS LE CALCUL DE LA TEMPÉRATURE POUR T=6	133
FIGURE 35 – L'ARCHITECTURE PROPOSÉE PAR LE PROJET LINDO (BRUT, CODREANU, DUMITRESCU, ET AL., 2011)	142
FIGURE 36 – L'ARCHITECTURE DU DÉMONSTRATEUR FINAL DU PROJET LINDO (BRUT, CODREANU, DUMITRESCU, ET AL., 2011) ..	146
FIGURE 37 – L'ARCHITECTURE LOGICIELLE DU SERVEUR DISTANT SITUÉ À PARIS	147
FIGURE 38 – GRAPHIQUE REPRÉSENTANT L'EXÉCUTION EN PARALLÈLE DE PLUSIEURS REQUÊTES	156
FIGURE 39 – LA SPÉCIFICATION D'UNE REQUÊTE PAR L'UTILISATEUR.....	157
FIGURE 40 – ARCHITECTURE DE L'APPLICATION	160
FIGURE 41 – LE NOMBRE D'UTILISATIONS DE CHAQUE REQUÊTE.....	166
FIGURE 42 – LA DISTRIBUTION DU NOMBRE D'ACCÈS AUX PAGES DU SITE.....	166
FIGURE 43 – LES ÉTAPES DE L'EXTRACTION DES MÉTADONNÉES.....	167

FIGURE 44 – LA PAGE DE PRÉSENTATION DU PRODUIT RHODIOLA 120 GÉLULES.....	168
FIGURE 45 – LA RÉPARTITION DES INTERACTIONS (CLICS ET MOUSE-OVER) DES UTILISATEURS DANS LE TEMPS	170
FIGURE 46 – L'ÉVOLUTION DE LA TEMPÉRATURE.....	170
FIGURE 47 – UN EXEMPLE DE LISTE DE RÉSULTATS RECLASSÉE.....	171
FIGURE 48 – LE NOMBRE DE FOIS QUE CHAQUE RÉSULTAT A ÉTÉ CLIQUÉ PAR LES UTILISATEURS DU SITE DE PRODUCTION.....	173
FIGURE 49 – LA RÉPARTITION DES CLICS DES UTILISATEURS PENDANT LES TESTS, SUR LA COPIE DU SITE.....	173
FIGURE 50 – LA RÉPARTITION DES PRODUITS CONSIDÉRÉS PERTINENTS	174
FIGURE 51 – LA COMPARAISON POUR LA REQUÊTE RHODIOLA	175
FIGURE 52 – COMPARAISON ENTRE LE P@K POUR LE SITE DE PRODUCTION ET LE P@K DES TESTS UTILISATEURS POUR LA REQUÊTE « RHODIOLA ».....	177
FIGURE 53 – LA STRUCTURE DE FICHER PROPOSÉE PAR DIG35	200
FIGURE 54 – LE MODÈLE ENTITÉ-RELATION DU FRBR	204
FIGURE 55 – LA STRUCTURE DE : A) ID3V1 ET B) ID3V2	205
FIGURE 56 – LA STRUCTURE DU FICHER QUICKTIME	206
FIGURE 57 – L'ORGANISATION DES SDS (MARTINEZ, 2004).....	210
FIGURE 58 – LE MODÈLE DE STOCKAGE UTILISÉ PAR DICOM	213
FIGURE 59 - LES CONCEPTS DU LSCOM LITE (NAPHADE ET AL., 2006)	214
FIGURE 60 – LE SCHEMA UML DES METADONNEES LOM	215

Liste des tableaux

TABLEAU 1 – ÉTUDE COMPARATIVE ENTRE DIFFÉRENTS STANDARDS DE MÉTADONNÉES	34
TABLEAU 2 – ÉTUDE COMPARATIVE ENTRE LES SYSTÈMES DISTRIBUÉS DE GESTION DE CONTENUS MULTIMÉDIAS	45
TABLEAU 3 – COMPARAISON ENTRE LES DIFFÉRENTS ÉLÉMENTS PRÉSENTS DANS DES STANDARDS DE MÉTADONNÉES	47
TABLEAU 4 – UN EXEMPLE DE MÉTADONNÉES MPEG-7	48
TABLEAU 5 – UN EXEMPLE DE MÉTADONNÉES DUBLIN CORE.....	48
TABLEAU 6 – UN EXEMPLE DE MÉTADONNÉES DIG35	48
TABLEAU 7 – LA CATÉGORISATION DES ACTIONS DE L'UTILISATEUR SELON (KELLY AND TEEVAN, 2003).....	54
TABLEAU 8 – UN EXEMPLE DE MÉTADONNÉES D'ATTENTION PROPOSÉ PAR (WOLPERS ET AL., 2007).....	63
TABLEAU 9 – EXTRAIT DE MÉTADONNÉES GÉNÉRALES POUR CE MANUSCRIT.....	77
TABLEAU 10 – EXTRAIT DE MÉTADONNÉES DE LOCALISATION ASSOCIÉES À CE MANUSCRIT.....	79
TABLEAU 11 – EXTRAIT DES MÉTADONNÉES ASSOCIÉES À CE MANUSCRIT	81
TABLEAU 12 – EXTRAIT DE MÉTADONNÉES ASSOCIÉES À UNE IMAGE	82
TABLEAU 13 – EXTRAIT DE MÉTADONNÉES ASSOCIÉES À UNE VIDÉO.....	84
TABLEAU 14 – EXTRAIT DES MÉTADONNÉES ASSOCIÉES À UN CONTENU AUDIO.....	86
TABLEAU 15 – EXTRAIT DES MÉTADONNÉES DE STRUCTURE ASSOCIÉES À UN DOCUMENT	87
TABLEAU 16 – EXTRAIT DES MÉTADONNÉES ASSOCIÉES À UNE IMAGE, QUI CONTIENT DEUX OBJETS IDENTIFIÉS	88
TABLEAU 17 – UN EXTRAIT DE LA DESCRIPTION D'UN SERVEUR DISTANT	101
TABLEAU 18 – L'ALGORITHME DE SÉLECTION DES ALGORITHMES D'INDEXATION	103
TABLEAU 19 – L'ALGORITHME COMPLET DE SÉLECTION DES ALGORITHMES D'INDEXATION.....	103
TABLEAU 20 – EXTRAIT DES MÉTADONNÉES ASSOCIÉES À CE MANUSCRIT, ENRICHIES PAR LA TEMPÉRATURE	110
TABLEAU 21 – EXTRAIT DE MÉTADONNÉES D'USAGE POUR LE DOCUMENT HUILE-VISAGE.HTML.....	112
TABLEAU 22 – L'ALGORITHME POUR L'AUGMENTATION DE LA TEMPÉRATURE.....	115
TABLEAU 23 – L'ALGORITHME DE PROPAGATION DE LA TEMPÉRATURE AU NIVEAU DES DESCENDANTS	115
TABLEAU 24 – L'ALGORITHME DE PROPAGATION DE LA TEMPÉRATURE AU NIVEAU DES ANCÊTRES.....	116
TABLEAU 25 – L'ALGORITHME POUR L'ATTÉNUATION DE LA TEMPÉRATURE PROPOSÉ DANS (MANZAT ET AL., 2010)	117
TABLEAU 26 – L'ALGORITHME POUR L'ATTÉNUATION DE LA TEMPÉRATURE	118
TABLEAU 27 – L'ALGORITHME DE CRÉATION DES LIENS INDIRECTS	121
TABLEAU 28 – L'ALGORITHME D'AUGMENTATION DE LA TEMPÉRATURE SUITE AUX INTERACTIONS DES UTILISATEURS AVEC LES MÉTADONNÉES AFFICHÉES DANS LA LISTE DES RÉSULTATS	122
TABLEAU 29 – L'ALGORITHME D'AUGMENTATION DE LA TEMPÉRATURE SUITE AUX CLICS DES UTILISATEURS SUR LA LISTE DES RÉSULTATS.....	123
TABLEAU 30 – L'ALGORITHME D'AUGMENTATION DE LA TEMPÉRATURE DES MÉTADONNÉES ASSOCIÉES AUX PARTIES D'UN DOCUMENT, SUITE AUX INTERACTIONS DES UTILISATEURS AVEC CES PARTIES	125
TABLEAU 31 – L'ALGORITHME GÉNÉRAL POUR LE CALCUL DE LA TEMPÉRATURE.....	128
TABLEAU 32 – UN EXTRAIT DE LA DESCRIPTION DU SERVEUR DE VIDÉOSURVEILLANCE SITUÉ À PARIS.....	148
TABLEAU 33 – UN EXTRAIT DE LA DESCRIPTION D'UN ALGORITHME D'INDEXATION.....	149
TABLEAU 34 – UN EXTRAIT DES MÉTADONNÉES ASSOCIÉES À UNE VIDÉO	149
TABLEAU 35 – EXEMPLE DE SORTIE FOURNIE PAR L'ALGORITHME DE TRAITEMENT DES CONTENUS AUDIO	150
TABLEAU 36 – LES ALGORITHMES D'INDEXATION DE CONTENUS VIDÉO	151
TABLEAU 37 – PREMIÈRE SORTIE XML POUR LA DÉTECTION D'UNE INTRUSION	151
TABLEAU 38 – SECONDE SORTIE XML POUR LA DÉTECTION D'UNE INTRUSION	151
TABLEAU 39 – UN EXTRAIT DES MÉTADONNÉES OBTENUES PAR L'ALGORITHME SPEECH-TO-TEXT.....	152
TABLEAU 40 – EXEMPLE DE SORTIE DU LOGICIEL SIMPLELPR ET SA TRANSFORMATION	153
TABLEAU 41 – LES REQUÊTES UTILISÉES POUR L'EXPÉRIMENTATION	155
TABLEAU 42 – RÉSULTATS DE L'ÉVALUATION DES REQUÊTES 1, 2 ET 3 EN MS.	155

TABLEAU 43 – RÉSULTATS DE L'ÉVALUATION DES REQUÊTES 4, 5 ET 6 EN MS.	156
TABLEAU 44 – EXTRAIT DES CORRESPONDANCES ENTRE LES ÉLÉMENTS DU MODÈLE ET LE TYPE DE MOT CONTENU.....	158
TABLEAU 45 – EXEMPLES DE LIENS OBTENUS POUR LA RECHERCHE DU MOT « RHODIOLA »	162
TABLEAU 46 – LES 100 MOTS LES PLUS RECHERCHÉS SUR LE SITE	162
TABLEAU 47 – LE NOMBRE DE RÉSULTATS PAR MOT RECHERCHÉ	164
TABLEAU 48 – MÉTADONNÉES EXTRAITES POUR UNE IMAGE	168
TABLEAU 49 – MÉTADONNÉES ASSOCIÉES À UN HYPERLIEN.....	169
TABLEAU 50 – LES RÉSULTATS DU CALCUL DU P@K POUR LE SITE DE PRODUCTION	176
TABLEAU 51 – LES RÉSULTATS DU CALCUL DU P@K POUR LES TESTS.....	176
TABLEAU 52 – LES PREMIERS RÉSULTATS PERTINENTS POUR LA REQUÊTE « RHODIOLA ».....	177
TABLEAU 53 – EXEMPLE DE SOUS-GROUPES DE MÉTADONNÉES DANS LE STANDARD SMPTE 335M-2001	208
TABLEAU 54 - DESCRIPTION DES ÉLÉMENTS DU STANDARD DUBLIN CORE	212

Introduction générale

Le contexte

L'évolution des technologies multimédias de ces dernières années, a rendu les dispositifs multimédias de plus en plus accessibles au grand public. Par exemple, dans les années 90 les téléphones portables n'étaient utilisés que pour communiquer avec d'autres personnes, mais progressivement on leur a ajouté d'autres fonctionnalités (e.g., la prise de photos, l'enregistrement de vidéos). Tout cela incite l'utilisateur à créer constamment du contenu multimédia et à le partager. Une fois les contenus multimédias créés, nous continuons à interagir avec eux, en effectuant différentes actions sur ces documents : recherche, annotation, modification, partage avec des tiers. Dans ce contexte, la recherche des documents pertinents pour l'utilisateur est un problème majeur, car les collections multimédias, de plus en plus volumineuses, sont, de fait, réparties sur plusieurs serveurs.

De manière générale, pour trouver une information pertinente dans une grande collection de documents, le plus facile serait de parcourir, non pas la collection elle-même, mais une version concise de celle-ci. Dans le cas des collections multimédias, cette information concise est constituée par les métadonnées, qui représentent « l'information structurée qui décrit, localise une ressource et en facilite la recherche, l'usage et la gestion » (National Information Standards Organization, 2004).

La création automatique des métadonnées n'est pas triviale à cause de « l'écart entre l'information qui peut être extraite automatiquement des données et l'interprétation qu'un utilisateur a des mêmes données. » (Smeulders et al., 2000). Ce fossé sémantique se situe entre les informations de bas niveau qui sont extraites à partir du contenu multimédia et leur signification réelle pour un utilisateur. Pour les documents numériques qui sont sauvegardés sous la forme de bitmap ou de signal l'extraction des informations sémantiques est plus difficile que pour les documents représentés sous forme textuelle (chaîne de caractères). En effet, l'extraction des métadonnées des contenus multimédias est tributaire de la complexité structurelle des contenus.

Les métadonnées sont situées dans le centre du cycle de vie du document multimédia (Hardman et al., 2008), (Smith and Schirling, 2006), et elles sont consommées pendant toutes les phases du cycle de vie du document auquel elles sont associées (Kosch et al., 2005). Tout cela mène à une constante interaction, directe ou indirecte, de l'utilisateur avec les métadonnées, et renforce l'idée de considérer l'utilisateur comme une source auxiliaire de métadonnées. D'une part, l'utilisateur génère des métadonnées de manière directe à travers ses tags et annotations (Cabanac et al., 2008), (Kahan et al., 2002). D'autre part, le comportement de l'utilisateur au moment de la consommation du document et des métadonnées permet d'en déduire d'autres informations. De cette façon, l'utilisateur crée implicitement des métadonnées (Brunie et al., 2002), (Pedersen and McDonald, 2008).

L'utilisateur n'est pas considéré uniquement comme une « source » de métadonnées par les systèmes de gestion de contenus multimédias. Il est intégré dans le processus de gestion à plusieurs niveaux (Figure 1), avec le but principal d'améliorer la liste des résultats obtenus pour sa requête (Manning et al., 2008). Cette prise en compte de l'utilisateur se fait dans le cadre du *Retour de Pertinence (RP)*. L'idée centrale du RP est de capturer, de manière itérative, le jugement de pertinence de la part de l'utilisateur pour une sous-collection des résultats obtenus pour sa requête. Ce jugement est exploité pour déterminer une meilleure représentation du besoin de l'utilisateur, afin de calculer et d'afficher une nouvelle liste affinée de résultats.

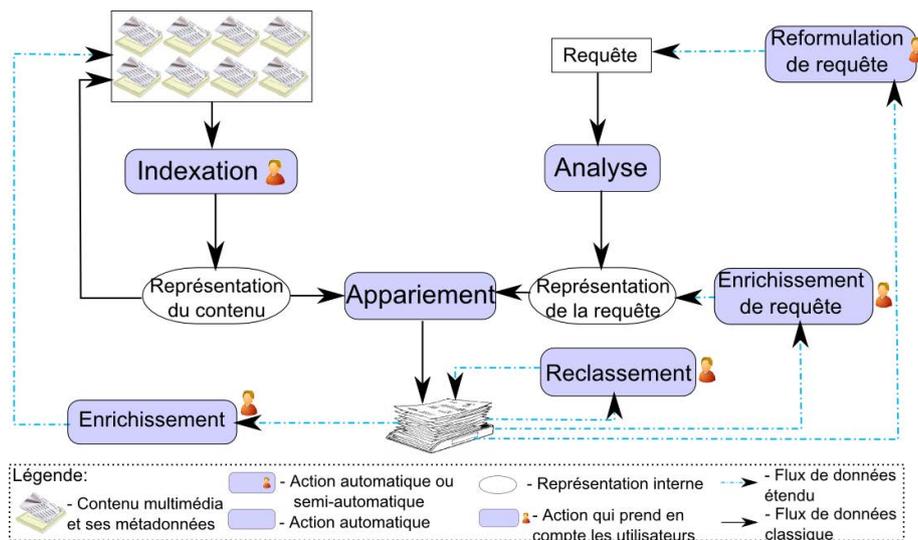


Figure 1 – Le processus de gestion des documents multimédias

Dans le cadre du RP, les utilisateurs sont considérés comme des personnes indépendantes et qui n'ont aucun lien entre elles. Mais avec l'avènement des réseaux sociaux, l'utilisateur n'est plus seul au monde : il est lié aux autres utilisateurs à travers des liens sociaux. Ce côté social de l'utilisateur et les interactions entre les utilisateurs, commencent à être pris en compte dans la détermination de ses intérêts (Cabanac, 2008), (Tchunte et al., 2011). En outre, dans l'ère du social, les utilisateurs ne tiennent plus à garder les informations pour eux, mais veulent partager ce qu'ils ont trouvé de potentiellement intéressant avec les autres.

La prise en compte de l'utilisateur et de son comportement devient une solution de plus en plus employée par les systèmes multimédias pour faire remonter les documents pertinents dans la liste des résultats : en général, les interfaces ne permettent pas d'afficher tous les résultats d'une requête simultanément et la grande majorité des utilisateurs n'a pas l'habitude de parcourir plusieurs pages de résultats pour retrouver l'information (Joachims et al., 2007). En effet, les utilisateurs considèrent avec prédilection les documents qui sont affichés au-dessus de la « ligne de flottaison »¹. Cette « ligne » est une ligne virtuelle, qui correspond à l'affichage de la première page des résultats sur un navigateur web : c'est la limite physique d'affichage en dessous de laquelle le contenu de la page n'apparaît plus à l'écran – le contenu situé sous cette ligne n'est donc visible que si l'on utilise la barre de défilement ou la molette de la souris. Elle est fortement liée au dispositif d'affichage. Cette ligne de flottaison est importante pour l'affichage des résultats d'une recherche, mais aussi pour l'affichage d'un document. Elle peut avoir une influence sur la structure du document, pour afficher prioritairement les informations les plus importantes. Cela montre l'importance de la structure d'un document dans la gestion et l'affichage de celui-ci.

La gestion des documents structurés est une problématique abordée depuis longtemps dans notre équipe (Sedes, 1998), (Jedidi, 2005), en intégrant les différentes structures d'un document dans un format de métadonnées. Plus récemment, la prise en compte de l'utilisateur dans la gestion des documents a été abordée, d'une part dans l'enrichissement de la requête de l'utilisateur en fonction de son profil et dans l'adaptation de la présentation des résultats en fonction du profil de l'utilisateur (Zayani, 2008). D'un part, dans l'enrichissement de la requête en fonction du contexte de l'utilisateur (Soukkarieh, 2010).

¹ <http://www.definitions-webmarketing.com/Definition-Ligne-de-flottaison>, consulté le 14 juin 2012

Notre travail s'est déroulé dans le cadre du projet européen LINDO (Large scale distributed INDEXation of multimedia Objects)². La problématique centrale du projet LINDO était d'indexer et de rechercher des segments de contenus multimédias stockés de façon distribuée qui sont pertinents. L'objectif principal du projet correspondait à établir une architecture distribuée d'indexation des contenus multimédias, dans laquelle les algorithmes d'indexation étaient déployés sur les différents sites distants où les contenus multimédias sont stockés, afin de réduire la quantité de données véhiculée sur le réseau.

C'est dans ce contexte de gestion distribuée de documents multimédias, où métadonnées et utilisateurs occupent une place importante, que se situe notre travail.

La problématique

Dans la continuité des travaux réalisés au sein de l'équipe, nous nous sommes intéressés à la gestion des métadonnées dans le cadre d'un système de gestion de contenus multimédias, plus précisément, à la génération des métadonnées, à leur évolution et à leur usage dans le temps en fonction des interactions de l'utilisateur avec le système, mais aussi avec les contenus multimédias et les métadonnées eux-mêmes.

Les métadonnées constituent l'élément central de notre travail. Beaucoup de standards et de formats de métadonnées ont été définis pour chaque type de média, qui ne sont pas forcément interoperables. L'interopérabilité est définie par (ALA/ALCTS/CCS Committee, 2004) comme « la capacité de deux ou plusieurs systèmes d'échanger des informations et des données et d'utiliser ces informations et ces données échangées sans effectuer de manipulations spéciales. ». Mais dans le cas des standards de métadonnées, cet échange d'information n'est pas évident à réaliser sans perte, car les éléments définis dans les différents standards et normes sont très hétérogènes et n'ont pas le même niveau de granularité. *Dans un premier temps nous nous intéressons à ce problème d'interopérabilité de métadonnées.*

Une fois le format de métadonnées établi, se pose le problème de la création des métadonnées associées au contenu multimédia. Les métadonnées peuvent être créées de plusieurs façons :

- directement par l'auteur du contenu multimédia ;
- automatiquement, à partir du contenu multimédia, en appliquant des algorithmes d'indexation ;
- par différents utilisateurs, en attachant des tags et des annotations au contenu.

De nombreux travaux se sont penchés sur l'extraction automatique des connaissances à partir des contenus multimédias. Ces travaux ont mené à la création d'une grande variété et à l'hétérogénéité d'algorithmes d'indexation, spécifiques à chaque type de media. Ces algorithmes sont toujours plus performants mais aussi, plus consommateurs de ressources (CPU, mémoire,...). L'utilisation de manière systématique de tous les algorithmes d'indexation disponibles mène à la surcharge du système et à la création de métadonnées qui peuvent ne jamais servir. Mais si le nombre d'algorithmes employés est trop restreint, la performance du système peut être altérée à cause du manque de métadonnées. En utilisant en permanence la même stratégie d'indexation, le système peut omettre des réponses à une requête de l'utilisateur, si son besoin évolue dans le temps. Le grand volume de contenus multimédias à gérer, fait du choix des algorithmes d'indexation un problème essentiel dans leur gestion. *Dans un second temps nous nous intéressons à ce problème de choix des algo-*

² <http://lindo-itea.eu/>

rithmes d'indexation, en favorisant la réduction de la consommation des ressources du système de gestion de contenus multimédias, tout en prenant en compte l'usage que l'utilisateur fait du système.

Ce problème de la gestion des ressources s'accroît dans le contexte actuel où les documents multimédias ne constituent plus uniquement de grands volumes, mais ils sont aussi répartis sur plusieurs serveurs hétérogènes en caractéristiques et capacités de stockage et traitement. Cette distribution des contenus multimédias sur des serveurs distants rend le choix des algorithmes d'indexation et le modèle de métadonnées encore plus importants.

Une étude approfondie de l'état de l'art nous montre que l'utilisateur est de plus en plus intégré dans le processus de gestion des contenus multimédias. La majorité des travaux dans ce domaine se focalise sur l'emploi du RP pour changer sa requête, l'enrichir ou modifier la liste de résultats obtenue suite à son exécution. Seuls quelques travaux utilisent le RP pour enrichir les métadonnées associées au document sur lequel l'utilisateur donne son avis de manière directe ou indirecte. Un autre objectif majeur de notre travail est *l'intégration de l'utilisateur dans le processus de gestion de métadonnées, en prenant en compte son comportement dans le temps, pour enrichir les métadonnées*. Nous considérons que l'utilisateur a une influence à deux niveaux : sur la structure du document, mais aussi sur les métadonnées, parce qu'il interagit d'une part avec les métadonnées pour choisir le document pertinent, et d'une autre part avec le contenu multimédia, en le consultant.

Pour synthétiser, nous souhaitons fournir des solutions aux verrous suivants :

1. Le problème de l'interopérabilité des métadonnées associées aux contenus multimédias ;
2. Quels algorithmes d'indexation utiliser pour n'extraire que les métadonnées susceptibles de servir pour la recherche des contenus multimédias, tout en prenant en compte les contraintes de consommation de ressources ?
3. Comment prendre en compte les interactions de l'utilisateur avec le système de gestion de contenus multimédias et avec les contenus multimédias, pour enrichir les métadonnées d'une manière implicite par l'usage ?

Les contributions

Nous nous sommes focalisés sur la gestion des métadonnées associées aux contenus multimédias et sur l'intégration de l'utilisateur dans ce processus de gestion. Dans notre travail nous partons du postulat que tout média est « élicitable », c'est-à-dire que l'on peut extraire des motifs, des régularités, des caractéristiques, des métadonnées tant de l'image, que de la vidéo ou du texte vu comme une chaîne de caractères.

Nous avons étudié les standards et les normes de métadonnées pour essayer de trouver une solution au problème de l'interopérabilité des métadonnées. En partant des solutions identifiées dans l'état de l'art par (Chan and Zeng, 2006), (Haslhofer and Klas, 2010), nous avons choisi *la proposition d'un format générique de métadonnées* (Brut, Laborie, et al., 2009a), (Laborie et al., 2010). Ce format intègre les standards et les normes les plus utilisés dans l'état de l'art, identifiés par le projet *LINDO*. Ce format est extensible en structure et en vocabulaire. Le format proposé contient deux grands types de métadonnées : des métadonnées générales, qui correspondent au document multimédia dans son ensemble, et des métadonnées spécifiques à chaque type de média inclu dans le document. En nous basant sur les règles de mappage définies par le W3C Media Annotations Working

Group³, nous avons établi les règles de mappage entre les standards sélectionnés dans le contexte du projet LINDO et notre format.

Dans le cadre d'un système multimédia distribué, tel que LINDO, il y a beaucoup de communications entre les différents serveurs : serveur central ou serveurs distants. Pour diminuer la quantité d'information véhiculée sur le réseau nous avons proposé *la création d'un résumé des métadonnées multimédias* (Laborie et al., 2009a) qui est envoyé au serveur central et utilisé pour sélectionner les serveurs distants susceptibles de contenir des documents pertinents pour la requête de l'utilisateur.

La réflexion sur l'optimisation de l'extraction des métadonnées, à travers les algorithmes d'indexation, tout en réduisant la consommation des ressources et en intégrant l'usage et les besoins de l'utilisateur, nous a amené à proposer *la réalisation de l'indexation des contenus multimédias en deux temps* : d'abord à l'acquisition des contenus, avec une liste fixe d'algorithmes d'indexation (nous qualifions cette indexation d'« *implicite* ») et ensuite, si nécessaire, au moment de la requête de l'utilisateur (nous qualifions cette indexation d'« *explicite* ») (Brut, Codreanu, Dumitrescu, et al., 2011). La sélection des algorithmes à utiliser sur chacun des serveurs distants du système, pour l'indexation implicite mais aussi pour l'indexation explicite, se réalise en fonction des caractéristiques des serveurs, des métadonnées déjà extraites (i.e., les résumés de métadonnées), des besoins de l'utilisateur (Brut, Laborie, et al., 2009b), et du contexte d'acquisition des contenus multimédias (Brut, Codreanu, Manzat, et al., 2011a).

Pour prendre en compte l'utilisateur dans la gestion des documents multimédias, nous avons introduit la notion de « *structure d'usage* » qui caractérise un document. Cette structure d'usage est une structure dynamique, qui évolue du point de vue des liens existants entre les documents et/ou parties de documents en fonction du comportement de l'utilisateur tout au long du cycle de vie du document multimédia. Cette structure est composée de liens bidirectionnels. Chaque lien est déduit du comportement de l'utilisateur sur les deux documents qu'il lie. Par exemple, les parties d'un document qui ont été copiées dans le cadre de la même session vont être liées par le lien « copier ».

Mais l'intérêt des utilisateurs pour un document varie dans le temps. Un document peut être utilisé pendant une semaine puis être progressivement oublié. Dans nos travaux, nous voulons capturer cette évolution de l'usage du document dans le temps. Pour cela, nous avons enrichi le format de métadonnées proposé en introduisant *le concept de « température »*, qui varie dans le temps. Elle est calculée en fonction de la façon dont l'utilisateur interagit avec le document multimédia, mais aussi avec les métadonnées (Manzat et al., 2010). Le concept de température nous semble très approprié pour matérialiser cette évolution : plus un document est utilisé, plus « chaud » il devient. S'il n'est pas utilisé pendant une période il « refroidit » et dissipe sa température. Cette dernière peut être utilisée par le système de gestion des documents multimédias pour : améliorer le reclassement des résultats, modifier la présentation des documents, faire des recommandations.

Notre contribution se focalise ensuite sur la définition *d'un algorithme de mise à jour de la température en fonction des interactions des utilisateurs avec le document, ou avec ses métadonnées*. Nous avons choisi d'implémenter un processus de propagation amorti pour le calcul de la température.

Nous avons appliqué le concept de température au processus de gestion des documents multimédias, présenté dans la Figure 1, tout en essayant de ne pas modifier le système existant : par exemple nous ne voulons pas modifier la fonction d'appariement utilisée par le système. Pour cela, nous avons utilisé la température des documents gérés par le système pour faire un reclassement des

³ <http://www.w3.org/2008/WebVideo/Annotations/>

résultats obtenus, suite à l'exécution d'une requête, en remontant les documents les plus « chauds » vers les premières positions de la liste des résultats.

Le positionnement de nos contributions par rapport au processus de gestion des documents multimédias est présenté dans la Figure 2. Les réponses fournies aux questions posées dans la problématique, sont les suivantes :

- R1. La proposition d'un format générique de métadonnées, extensible en structure et vocabulaire, qui intègre les standards et normes les plus utilisés dans l'état de l'art.
- R2. La réalisation de l'indexation du contenu multimédia en deux temps : une première fois à l'acquisition du contenu, avec une liste fixe d'algorithmes d'indexation ; et une seconde fois, si nécessaire, au moment de la requête de l'utilisateur.
- R3. La notion de structure d'usage, avec l'introduction du concept de « température », qui varie dans le temps et qui est mise à jour en fonction des interactions des utilisateurs avec le document, ou avec ses métadonnées.

Notre travail s'est déroulé dans le cadre du projet LINDO, qui nous a permis de valider une partie de nos propositions : le format des métadonnées, le résumé des métadonnées, la sélection des algorithmes d'indexation. Le projet a abouti à la réalisation d'un système distribué pour un cas d'utilisation de vidéosurveillance. Dans le cadre de ce projet, nous avons subi des contraintes fortes liées surtout au choix de l'architecture et des technologies réalisés.

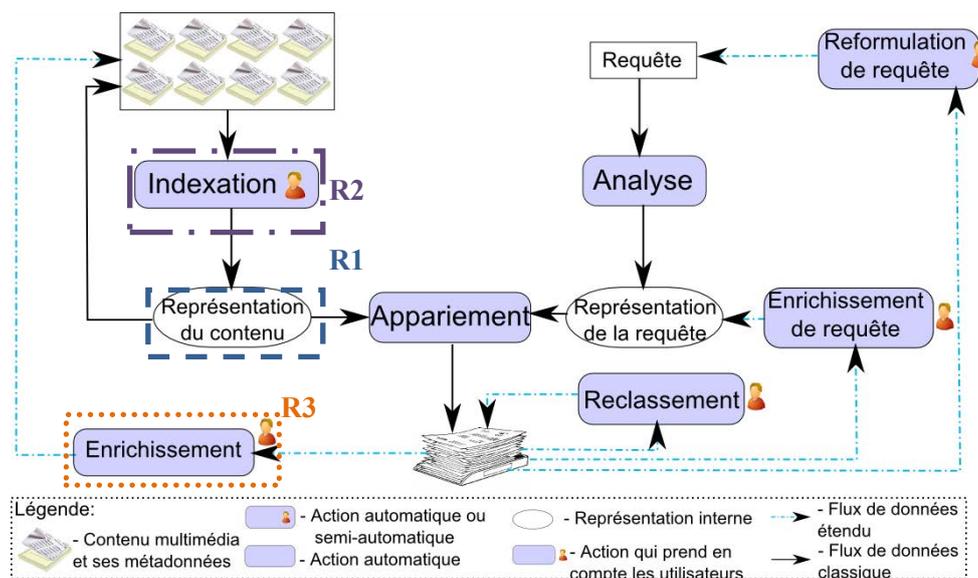


Figure 2 – Le positionnement de nos contributions dans le processus de gestion des documents multimédias

Le cadre industriel de LINDO ne prévoyait pas, ni d'étudier l'utilisateur, ni de prendre en compte son comportement dans la gestion de documents multimédias. Nous avons donc choisi un cas d'étude relevant d'un autre contexte industriel pour valider la prise en compte du comportement de l'utilisateur. Nous nous sommes intéressés plus précisément au commerce électronique comme domaine d'application pour prendre en compte l'utilisateur.

Pour réaliser l'autre partie de notre validation nous nous sommes focalisés sur le domaine du commerce électronique. Beaucoup de sites de vente en ligne ont été créés en utilisant des plateformes

de développement spécifiques qui mettent à disposition des entreprises, des fonctionnalités telles que : la gestion du site, le commerce mobile, des outils de marketing et de promotion, et la gestion du catalogue de produits. Pour la gestion du catalogue de produits, toute plateforme inclut un moteur de recherche basique, qui retourne tous les produits relatifs à une requête de l'utilisateur. Dans ce domaine le classement des résultats est très important car les utilisateurs ne regardent habituellement que les premiers résultats affichés (Joachims et al., 2007). Pour les sites de vente en ligne, afficher de mauvais résultats en début de liste peut signifier perdre un client. Cela est d'autant plus vrai si l'utilisateur consulte le site à partir de son Smartphone. La surface d'affichage est alors plus réduite et le nombre de résultats affichés est beaucoup plus faible que sur un écran classique.

La mise en œuvre et l'expérimentation ont été menées sur le site de vente en ligne de compléments alimentaires et produits naturels NaturaMundi.com. Les technologies développées par l'entreprise toulousaine qui le commercialise permettent d'enregistrer toutes les interactions d'un utilisateur avec le site web. Ainsi, nous avons accès à des données générées par les utilisateurs réels du site. Pour tester l'efficacité de l'utilisation de la température et de son évolution dans le temps nous avons réalisé un test utilisateur sur une vingtaine de personnes. Ce test nous a permis de dire que pour certaines requêtes de l'utilisateur, l'utilisation de la température dans le reclassement des résultats les aide à trouver les informations/produits recherchés plus vite.

Les expérimentations que nous avons menées dans le cadre de ces deux cas d'étude (le projet LINDO et le commerce électronique) nous permettent de démontrer l'extensibilité et l'adaptabilité du format de métadonnées à différents domaines d'application.

L'organisation du mémoire

Ce rapport est structuré en trois parties qui présentent respectivement, l'état de l'art, notre contribution, l'implémentation et l'évaluation effectuées. La figure suivante illustre la composition de ce mémoire, en présentant les correspondances entre les différents chapitres.

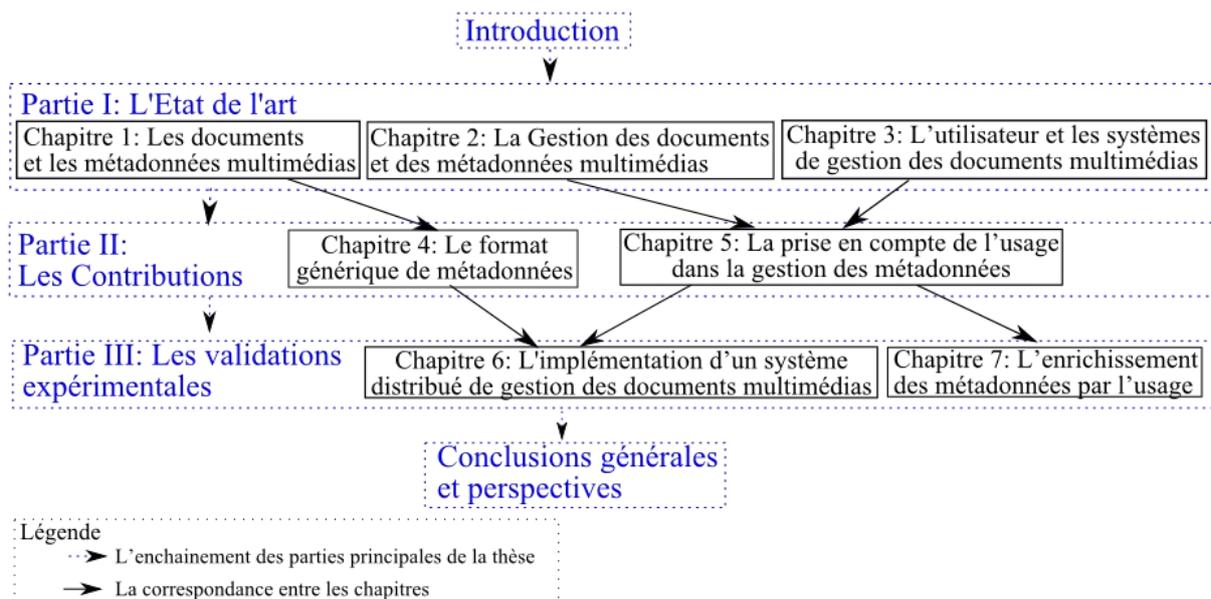


Figure 3 – Le plan du mémoire

Partie I : L'État de l'art

Introduction

Notre problématique porte sur la gestion des documents et métadonnées multimédias dans le cadre d'un système distribué, mais elle porte également sur l'intégration de l'utilisateur dans la gestion de ces contenus à travers l'enrichissement des métadonnées. Dans cette première partie, nous présentons un état de l'art décrivant les différentes approches qui abordent cette problématique.

Dans le premier chapitre, nous présentons d'abord les différentes définitions que le terme multimédia a eues au fil des ans. Nous nous attardons sur les différents types de contenus mono médias (audio, vidéo, image, texte) et multimédias (présentations, documents interactifs), tout en insistant sur leur différentes structures. Par la suite, nous détaillons pour chaque type de média les différents standards et normes de métadonnées qui sont définis. A la fin de ce chapitre une étude comparative de ces différents standards et normes est fournie.

Dans le deuxième chapitre, nous abordons les différentes stratégies envisageables pour la gestion des documents et métadonnées multimédias. Dans un premier temps nous présentons le processus de gestion de ces documents, en prenant en compte leur acquisition, leur indexation et leur recherche. La deuxième partie de ce chapitre est attribuée à la présentation des systèmes distribués de gestion de documents multimédias, en se focalisant surtout sur les projets. La dernière partie du chapitre est dédiée au problème d'interopérabilité des standards et normes de métadonnées multimédias et il présente quelques solutions à ce problème adoptées par les systèmes distribués.

Dans le troisième chapitre nous présentons en détail la prise en compte des interactions de l'utilisateur avec le système de gestion de contenus multimédias. Nous insistons sur les différents types de comportement qu'un utilisateur peut avoir pendant son interaction avec un document/système et sur la façon dont elles sont exploitées par les systèmes.

Chapitre 1. Les documents et les métadonnées multimédias

1.1.	<i>Introduction</i>	13
1.2.	<i>Le document numérique</i>	14
1.2.1.	La définition du terme multimédia.....	14
1.2.2.	Le texte	15
1.2.3.	L'image.....	16
1.2.4.	L'audio.....	16
1.2.5.	La vidéo	16
1.2.6.	Le document interactif.....	17
1.2.7.	Le cycle de vie des documents multimédias	17
1.2.8.	Bilan sur les documents multimédias.....	18
1.3.	<i>Les documents multimédias multistrués</i>	19
1.3.1.	La structure physique.....	19
1.3.2.	La structure logique	19
1.3.3.	La structure sémantique	19
1.3.4.	La structure spatiale.....	20
1.3.5.	La structure temporelle	20
1.3.6.	La structure hypermédia	21
1.3.7.	Bilan sur les documents multimédias multistrués	21
1.4.	<i>Les métadonnées multimédias</i>	21
1.4.1.	La définition.....	21
1.4.2.	La classification des métadonnées	22
1.4.3.	Le cycle de vie des métadonnées	24
1.4.4.	Les standards et formats de métadonnées	25
1.4.5.	La représentation des métadonnées.....	30
1.4.6.	Bilan sur les métadonnées multimédias	31
1.5.	<i>Bilan</i>	33

1.1. Introduction

L'objectif de ce chapitre est de présenter un état de l'art sur la notion de document multimédia et sur celle de métadonnées associées à ce document. Nous commençons par des considérations sur l'évolution du concept de multimédia dans le temps, et sur le cycle de vie d'un document multimédia. Un bilan des différentes structures d'un document est réalisé. Nous nous concentrons sur les différents standards et formats de métadonnées définis dans l'état de l'art. Mais avant de donner un aperçu de ces standards, nous détaillons le cycle de vie des métadonnées et le lien qui existe entre ce cycle de vie et celui du document multimédia. Avant de conclure, nous comparons les standards du point de vue : du type de ressource décrite, du domaine d'application, de la population cible, du niveau de description, du type des métadonnées contenues et de l'extensibilité, du vocabulaire contrôlé, de la sérialisation, du stockage et de la formalisation.

1.2. Le document numérique

Une définition générique du concept de « document » est donnée par le dictionnaire Larousse : « *tout renseignement écrit ou objet servant de preuve ou d'information* ». Cette définition est valable surtout pour les documents manuscrits traditionnels. L'évolution des technologies et la numérisation des documents ont eu comme conséquence l'apparition de nouvelles définitions. Une telle définition est fournie par l'ISO : « *Un document est l'ensemble d'un support d'information et des données enregistrées sur celui-ci sous forme en général permanente et lisible par l'homme et la machine* ».

L'évolution constante des documents numériques a mené à une distinction des documents en trois catégories (Sedes, 1998):

- *document non-structuré* : un document plat qui n'intègre aucune information, ni élément de structure et qui ne contient que du texte. Il peut donc être vu comme une « longue » suite de caractères (Tannier, 2006).
- *document structuré* : un document qui possède une structure, explicitement déclarée et a priori connue. La structure de document permet d'identifier les différents éléments et leur rôle. Des langages spécifiques ont été développés pour décrire ce type de document tels que XML, XML Schema, XSD.
- *document semi-structuré* : un pont entre les données structurées et non structurées (Trippe, 2001). Le langage XML, qui permet de produire des documents à la fois structurés et semi-structurés, est devenu un mode de représentation standard dans le domaine des documents numériques. Par un marquage inséré dans le texte, XML apporte des éléments sémantiques supplémentaires concernant la structure, le fond ou la forme du document. XML, par son caractère flexible, permet de produire des documents à structure irrégulière, non rigide, a priori inconnue (Abiteboul, 1997).

1.2.1. La définition du terme multimédia

Tout au long de l'évolution technologique qui a eu lieu ces dernières années, la structure des documents a beaucoup évolué. Les dispositifs électroniques sont devenus de plus en plus performants et ils nous permettent de créer des contenus numériques toujours plus complexes. C'est le cas des téléphones portables, qui, de nos jours, peuvent prendre des photos ou des vidéos et les joindre à un SMS. De cette façon, nous sommes passés de documents plats, purement textuels, aux documents multimédias, qui combinent au moins deux médias, via une structure (Chrisment and Sèdes, 2001).

Les premières références aux documents multimédias apparaissent bien avant les années 90. (Savage and Vogel, 2008) tracent la chronologie du multimédia, en partant des années 1914, avec l'apparition du premier film animé, *Gertie the Dinosaur*, jusqu'à nos jours, où l'apparition des dispositifs tactiles a eu une influence importante sur les documents multimédias. Cette chronologie nous permet de voir, comment le terme multimédia a évolué, en partant du texte simple, pour arriver aux documents interactifs.

Dans la littérature on trouve beaucoup de définitions du terme multimédia :

- « *c'est l'ouverture de l'informatique à d'autres formes d'informations que le texte et la graphique.* » (Forum Multimédia 92, 1992) ;
- « *technique de communication qui tend à rassembler sur un seul support l'ensemble des moyens audiovisuels (médium de base, dessins animés) et informatiques (données et pro-*

grammes) pour les diffuser simultanément et de manière interactive. » (Notaise et al., 1995) ;

- « *le terme multimédia désigne un système intégrant divers médias et autorisant leur utilisation interactive.* » (Martinez, 1996) ;
- « *Objet, nom commun, adjectif, expression, phénomène, système, langage, média, il est tout à la fois et rien ne l'arrête. On le transporte avec soi, on l'enseigne, on le crée, on le montre, on le pirate, on le développe, il nous transporte, il nous réjouit. Il peut être intime, compact, digital, didactique, virtuel, convivial, il séduit, on y pénètre, on y navigue et on s'y perd* » (Bouquillard, 2001) ;
- « *la création, l'intégration et la distribution de toute combinaison de texte, élément graphique, son, ou vidéo réalisées par l'ordinateur* » (Savage and Vogel, 2008) ;

Toutes ces définitions ont comme point commun l'intégration de plusieurs médias dans le même document. Au début des années 90, tout contenu, autre que le texte, était considéré comme un document multimédia. Mais au fil du temps une dimension interactive a été ajoutée au document, pour arriver aujourd'hui à « *un agencement interactif dans le temps et dans l'espace d'éléments distincts de natures différentes (issus de deux ou plusieurs types de média) afin d'enrichir le contenu de l'information diffusée* » (Mbarki, 2008).

Dans la littérature, un média est considéré aussi comme le support sur lequel est physiquement stocké, contenu ou transféré, une musique, un film, des photos ou plus généralement des données. En informatique, un média est défini comme « *un type de données abstrait c'est-à-dire une description d'une structure de données ainsi qu'un ensemble d'opérations définies sur ces données telles que les opérations pour la saisie et la présentation de données* » (Apers et al., 1997).

Les documents multimédias peuvent être divisés en deux grandes catégories : *documents non-interactifs* et *documents interactifs*, en fonction de l'interaction que l'utilisateur peut avoir avec le contenu du document. Dans les *documents multimédias non-interactifs*, *l'utilisateur n'a aucun contrôle sur le flux d'informations contenu dans le document*. C'est notamment le cas des documents textuels dans lesquels l'auteur a établi une structure du document qui est figée et l'utilisateur ne peut pas la changer, il est obligé de lire le document dans la façon dont il est conçu. Pour les *documents interactifs*, *l'utilisateur peut contrôler la façon dont il va consommer l'information contenue dans le document multimédia*. Un exemple de document interactif est un cours en ligne. Un étudiant peut lire les différentes parties du cours dans un ordre aléatoire. Il peut même sauter des cours, s'il le souhaite.

1.2.2. Le texte

Le texte est le principal média d'échange utilisé sur le Web, mais aussi dans le cadre de notre vie professionnelle et privée. Il est composé d'une succession organisée de chaînes de caractères. Les caractères forment des mots, des phrases et des paragraphes selon un langage. Selon (Marcoux, 1994), un texte désigne : « *un ensemble d'informations qui représente, dans les activités normales de manipulation ou d'acquisition des connaissances à l'échelle humaine, une unité que l'on peut raisonnablement considérer comme indivisible et complète* ».

Le texte est le plus ancien média étudié (Salton and McGill, 1986). Il est également celui qui rend le plus fidèlement la pensée humaine, en étant écrit dans notre langage courant. L'extraction de l'information des documents textuels est une étape très importante dans la gestion de ces documents. Elle n'est pas évidente à réaliser.

1.2.3. L'image

Selon le Larousse une image est « *une représentation d'un être ou d'une chose par les arts graphiques ou plastiques, la photographie, le film, etc.* ». Par image numérique, on entend toute image qui se présente sous la forme d'une matrice de pixels ou sous une forme vectorielle. Ainsi, elle est acquise, créée, traitée et stockée sous forme binaire.

L'image occupe une place de plus en plus importante dans notre vie. Nous créons constamment des photos pour nous souvenir des beaux moments passés. Mais, une image ne porte aucune sémantique en elle-même, contrairement au texte. A l'exception des informations concernant le contenu lui-même, une image ne contient aucune autre information supplémentaire, (i.e., le nom des personnes présentes sur la photo, l'événement immortalisé par la photo).

Les appareils photo ont beaucoup évolué en améliorant la qualité des images. Une autre évolution a été faite en ce qui concerne les informations supplémentaires, associées à l'image, qui sont sauvegardées dans le même fichier que l'image elle-même. Il s'agit des informations techniques sur les caractéristiques de l'appareil photo, l'auteur de l'image, mais aussi les coordonnées GPS de l'endroit où la photo a été prise.

1.2.4. L'audio

L'audio représente le support matériel qui permet de véhiculer un message sous la forme d'une information sonore, émise par une source et destinée à un point récepteur afin d'établir une communication entre les deux points. Cette information sonore est le résultat des phases d'échantillonnage et de quantification d'un signal continu, produit par la source sonore. Ainsi, l'étude de ce média revient à sa caractérisation par un ensemble de paramètres tels que la fréquence d'échantillonnage, la taille de l'échantillon, le pas de quantification, le nombre de canaux utilisés, etc. Par ailleurs, le son requiert de gros volumes.

Ce média peut jouer différents rôles dans le cadre d'un document multimédia : source d'information supplémentaire ou complémentaire à un document textuel, commentaire d'une image, agrément, canal de sortie supplémentaire lorsque les interfaces visuelles classiques sont saturées, aide aux handicapés visuels, etc.

1.2.5. La vidéo

Les documents audiovisuels, connus sous le nom de « *vidéos* », sont définis par le dictionnaire Larousse comme « *Œuvre, montage associant l'image, fixe ou animée, et le son enregistré* ». Ils sont formés d'une séquence d'images appelées « *frames* » et d'une bande sonore synchronisée. Cependant, une séquence vidéo n'est pas une simple succession d'images et de sons, mais un arrangement spatial et temporel de modules lui donnant un sens et une structure (Salazar and Valero, 1995). Les frames résultent soit de phases d'échantillonnage et de quantification d'un signal vidéo analogique, soit de programmes informatiques générant des images de synthèse.

La vidéo est le médium le plus gourmand en terme de volume et par conséquent il requiert encore plus de compression que les autres médias. La compression est réalisée à travers un codec qui désigne tout procédé capable de compresser ou de décompresser un signal analogique ou numérique. Ces procédés peuvent être matériels ou logiciels.

1.2.6. Le document interactif

Les médias présentés jusqu'à maintenant sont de médias non-interactifs. Cela signifie que l'utilisateur n'a aucun contrôle sur le flux d'informations contenues dans le document. Même si certains de ces médias évoluent dans le temps : c'est le cas des documents audio et des vidéos. Leur contenu est le même pour tous les utilisateurs, sans que ces derniers puissent choisir le contenu qu'ils désirent consulter. Tout cela a changé avec l'apparition du Hypertext (Conklin, 1987). Cette nouvelle technologie a révolutionné le monde des documents. Elle permet à un utilisateur de choisir quel contenu il veut voir et surtout dans quel ordre il veut le voir.

(Savage and Vogel, 2008) présentent plusieurs types de documents interactifs. Ils commencent par les documents qui offrent une *interactivité basique*, en terme de boutons, menus, objets cliquables. Une autre catégorie de documents interactifs est représentée par les *documents hypermédias*. Dans la cadre de ce type de documents, l'auteur établit une structure pour le contenu, mais il permet aussi à l'utilisateur de naviguer dans le contenu d'une manière assez simple, à travers des hyperliens vers d'autres contenus relatifs au document courant : textes, images, vidéos, animations.

Une autre catégorie de documents interactifs est formée par les *documents adaptatifs*. Le contenu de ces documents est affiché de manière différente, en fonction du contexte dans lequel ils sont utilisés. Par exemple, un document peut être personnalisé en fonction du profil de l'utilisateur, du type d'appareil d'affichage utilisé, ou en fonction de l'historique des activités de l'utilisateur.

De manière générale, les documents interactifs sont composés de plusieurs types de médias, comme ceux présentés dans les sous-sections précédentes. Mais, en plus de ces médias classiques, dans un document interactif peut être composé d'autres sortes de contenus, tels que les animations, les objets de synthèse en deux dimensions, mais également en trois dimensions.

1.2.7. Le cycle de vie des documents multimédias

Les principales étapes de ce cycle de vie, présenté par (Sellen and Harper, 2003), sont la création, l'archivage, la recherche et le partage.

Comme toutes les choses créées par l'homme, le cycle de vie des documents multimédias commence par une phase de conception, dans laquelle l'auteur établit le message qu'il veut transmettre à travers son document. Une fois cette étape obligatoire franchie, la phase de recherche d'informations commence. Dans cette phase, l'utilisateur cherche d'autres contenus qui peuvent être pertinents pour la création de son document. Cette recherche s'effectue parmi ses propres documents, mais aussi parmi ceux créés par d'autres personnes.

La phase suivante est celle de la création effective du document. Cette étape implique la réalisation de la structure du document et la rédaction du contenu proprement dit. Ainsi, le document final contient du contenu propre de l'auteur, mais également des parties des documents multimédias retrouvés à l'étape précédente.

Les trois étapes suivantes du cycle de vie du document multimédia sont très importantes dans le contexte actuel, quand le partage et l'échange de l'information sont primordiaux dans le monde du web sémantique et des réseaux sociaux. Pour que ce partage soit possible, le choix des formats utilisés pour le stockage des documents est primordial.

Le classement des documents est également très important pour leur réutilisation. Ce classement est à la base la phase de recherche du début du cycle de vie du document multimédia. Ce classement peut être réalisé directement par l'utilisateur – dans le système de fichier, en regroupant les

documents qui traitent d'un sujet identique dans le même répertoire –, ou par un système automatisé, qui accomplit l'indexation du nouveau document pour le classer avec ses semblables.

Ce cycle de vie est valable pour tout type de document, qu'il soit textuel ou multimédia. Pour la production des documents multimédias, (Hardman et al., 2008) ont défini un processus cano- nique, exemplifié dans la Figure 4.

Ce processus de production commence par une étape de planification, dans laquelle l'auteur du document établit ce qu'il veut transmettre à travers son document. Une fois cette phase réalisée, il commence à identifier et à créer les différents médias de base dont il a besoin. Par la suite, l'auteur ajoute des annotations à ces médias pour pouvoir les classer et les retrouver plus facilement. L'étape 5 de ce cycle de production est équivalente à la phase 1 (recherche d'information) du cycle de vie du document multimédia proposé par (Sellen and Harper, 2003). Dans cette étape l'auteur retrouve les médias nécessaires parmi ceux qu'il a créés à l'étape précédente mais également parmi les documents créés par d'autres personnes. Une fois cette phase réalisée, l'auteur peut commencer à construire son document en organisant ces médias. Ces deux étapes sont équivalentes avec la phase 3 du cycle de vie du document multimédia (finalisation des documents). Le document finalisé est publié, puis diffusé. Ainsi s'achève le cycle de production d'un document, qui peut être alors exploité.

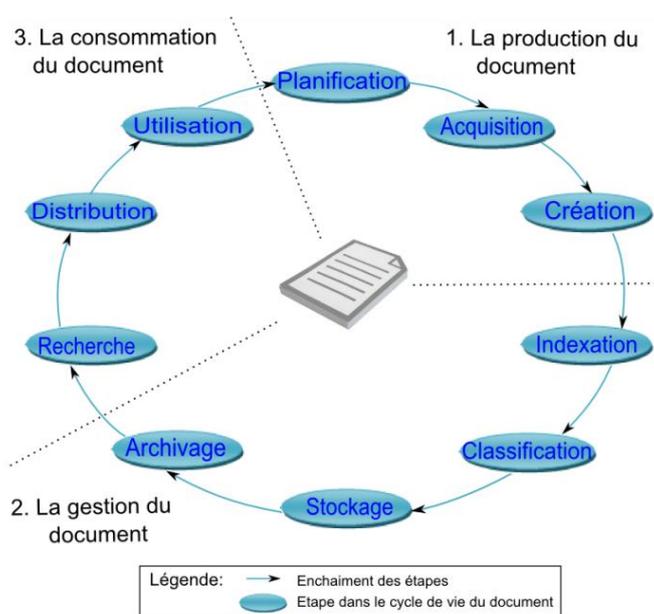


Figure 4 – Le cycle de vie d'un document multimédia (de la production à la consommation), selon (Hardman et al., 2008)

1.2.8. Bilan sur les documents multimédias

Suite à cette courte présentation des documents multimédias, une surcharge de langage peut être constatée concernant le terme multimédia. Pour certains auteurs, le terme multimédia est associé à tous les contenus numériques, à l'exception des documents purement textuels. Pour d'autres, le terme multimédia est associé aux dispositifs ou systèmes qui permettent la création de contenus numériques, autre que le texte, et qui en permettent aussi l'affichage de ces contenus.

La création de documents multimédias n'est pas aisée. Elle est réalisée en plusieurs étapes. Dans la littérature, de nombreux travaux ont défini le cycle de vie de ces contenus. Dans ce cycle de

vie l'accent est mis sur l'annotation des documents, leur distribution et leur partage suite à l'apparition du web social.

Dans notre étude, nous entendons par « multimédia » les contenus qui combinent au moins deux médias via une structure.

Le reste du chapitre présente un bref état de l'art sur les différentes structures qu'un document multimédia peut avoir.

1.3. Les documents multimédias multistructurés

La chronologie des documents multimédias réalisée par (Savage and Vogel, 2008) montre que ceux-ci ont des structures de plus en plus complexes : en passant des documents papiers aux documents numériques non-interactifs et, plus tard, aux documents multimédias interactifs.

Même sans prendre en compte l'interactivité, un document multimédia peut avoir plusieurs structures. De nombreux travaux sont axés sur ces différentes structures. Les auteurs (Poullet et al., 1997) identifient trois structures principales : la structure physique, la structure logique et la structure sémantique. A ces trois structures, s'ajoutent une structure spatiale et une temporelle (Soukkarieh and Sèdes, 2006). Ces structures sont valables pour tout type de documents multimédias, interactifs ou pas. L'apparition des documents interactifs a engendré l'apparition d'une autre structure du document : la structure hypermédia (Auffret et al., 1999), (Sedes, 1998).

1.3.1. La structure physique

La structure physique définit la présentation du document sur le support (papier, écran, etc.). Elle décrit la mise en page d'un document et en définit les différentes zones et caractéristiques. Cette structure se traduit par un ensemble de règles de présentation sous la forme d'une succession de lignes, de paragraphes, de colonnes, de pages, de caractéristiques typographiques, etc. Cette structure peut contenir des informations à différents niveaux de granularité. Cette structure peut être affichée sous forme d'arbre ; l'organisation est explicite et est définie *a priori*.

1.3.2. La structure logique

La structure logique définit une organisation hiérarchique de l'information contenue dans le document. (Fourel, 1998) présente cette structure comme « *l'organisation explicite d'abstractions logiques représentant des parties de document* ».

La structure logique permet un découpage de l'information contenue dans le document d'un point de vue logique. Elle est composée par d'entités d'informations qui existent indépendamment les unes des autres. Par exemple, la structure logique peut être composée de titres, chapitres, paragraphes, notes, schémas, etc. L'organisation de cette structure est explicite et elle est définie *a priori*.

1.3.3. La structure sémantique

La structure sémantique décrit les informations représentatives du document. Elle en représente la structure du discours. Elle permet de le manipuler selon des propriétés sémantiques particulières à l'information qui y est contenue. Les éléments de cette structure ajoutent un sens supplémentaire à un ou plusieurs éléments de la structure logique (Poullet et al., 1997).

Les éléments de cette structure peuvent être associés à des métadonnées, car ils offrent de l'information supplémentaire sur le contenu multimédia. Elle peut être créée de manière automatique, en utilisant des algorithmes d'indexation, ou, manuellement, par les utilisateurs. Cette structure n'est pas figée : elle peut évoluer dans le temps, en structure et en contenu, des nouvelles informations pouvant être ajoutées à cette structure pour mieux décrire le document.

1.3.4. La structure spatiale

La structure spatiale permet de présenter graphiquement les éléments du document multimédia. Elle spécifie également le positionnement des différents éléments qui y sont présents les uns par rapport aux autres. Elle définit la taille des différentes zones, les superpositions, les juxtapositions.

Le positionnement spatial peut se faire par rapport aux huit relations cardinales ou par rapport aux relations topologiques. Les relations cardinales sont fondées sur des notions géographiques (« Nord », « Sud », « Ouest », « Est »). Les relations topologiques expriment les notions de connectivité, d'orientation, d'englobement et des relations de contiguïté entre les objets spatiaux (Egenhofer, 1994), (Alboody, 2011).

(Charhad and Quenot, 2004) proposent d'autres types de relations spatiales. Ces relations sont basées sur la distance entre les entités décrites dans la dimension spatiale (e.g., « près », « loin »).

Comme pour les autres structures, la structure spatiale est établie par l'auteur du document multimédia *a priori* et, dans la majorité des cas, elle reste figée du moment où le document est finalisé jusqu'à la fin de son existence. Au moment de la création du document multimédia plusieurs structures spatiales peuvent être établies. Elles vont être utilisées pour la présentation du contenu en fonction du dispositif d'affichage employé.

1.3.5. La structure temporelle

Le temps occupe une place importante dans les documents multimédias, en favorisant la synchronisation et l'enchaînement des différentes parties des documents. La structure temporelle permet de traduire et de décrire ces synchronisations et ces enchaînements. Plusieurs études ont été menées afin de modéliser cette dimension (Allen, 1983), (Vazirgiannis et al., 1998).

Le temps peut être défini de plusieurs façons :

- Le temps d'existence du document multimédia, dans lequel le document peut évoluer et avoir plusieurs versions ;
- Le temps au sens du moment où l'utilisateur visionne le document ;
- Le temps dans le cadre du document multimédia, comme un élément de synchronisation et d'enchaînement des différents éléments de contenu.

Nous allons nous intéresser à la relation temporelle entre les séquences sonores et vidéos d'un contenu audiovisuel. Dans ce cas, le temps est relatif au début du document multimédia. Les relations d'Allen (Allen, 1991) permettent de structurer le contenu d'une séquence audiovisuelle en se basant sur les informations temporelles. Allen identifie un ensemble complet de relations temporelles qui peuvent exister entre deux intervalles. Il propose treize relations parmi lesquelles douze sont asymétriques (e.g., « Précède », « Rencontre », « Chevauche », « Débute », « Pendant », « Termine », « Équivalent »).

Comme pour les autres structures, la structure temporelle est établie par l'auteur du document multimédia *a priori* et, dans la majorité des cas, elle reste figée à partir du moment où le document est finalisé jusqu'à la fin de son existence.

1.3.6. La structure hypermédia

La structure hypermédia (Auffret et al., 1999), (Sedes, 1998) correspond à l'organisation inter et intra-média dans un ou plusieurs documents. Cette organisation est traduite par un ensemble de liens qui permettent la navigation inter ou intra-média en spécifiant de façon classique une ancre de départ, une ancre d'arrivée et le type de lien. Différents types de liens sont possibles : renvoi, référence, annotation, synchronisation, etc.

Certains de ces types de liens sont créés directement par l'auteur du document multimédia. En revanche d'autres peuvent être extraites automatiquement du contenu multimédia ou ajoutées ultérieurement par un tiers. Effectivement, les renvois vers d'autres parties du document ou vers d'autres documents multimédias sont spécifiés par l'auteur de celui-ci. Cependant, des annotations peuvent être créées par quiconque et être attachées au document multimédia ou à d'autres entités qu'il contient à travers les liens d'annotations.

1.3.7. Bilan sur les documents multimédias multistrukturés

Comme on a pu voir, un document peut avoir plusieurs structures différentes, qui se complètent les unes les autres et qui sont utilisées pour des buts différents. La majorité de ces structures sont construites *a priori* et elles sont figées pour toute l'existence du document. Les structures physiques, logiques, spatiales et temporelles du document multimédia sont utilisées surtout pour la présentation du document à l'utilisateur. La structure sémantique est employée pour ajouter de l'information supplémentaire au contenu. Cette structure est construite à partir du contenu du document, mais aussi sur les annotations supplémentaires ajoutées par les utilisateurs. La structure hypermédia est une structure complexe qui se trouve à la base des documents interactifs. Les liens de renvois contenus dans cette structure sont définis explicitement par l'auteur du document, mais les autres types de liens peuvent être extraits du contenu multimédia du document.

Ces structures coexistent dans le même document, et dépendent les unes des autres. Un modèle de coexistence de ces structures dans un même document a été présenté par (Djemal, 2010). Djemal considère les structures physiques et logiques comme des structures de base, parce qu'elles sont définies directement par l'auteur du document. En outre, le document est composé d'un ensemble de blocs de contenu organisés par les structures de base. La structure hypermédia est définie par les interactions qui peuvent avoir lieu entre les blocs du document. En outre, ces blocs de contenu sont décrits par les métadonnées qui sont regroupées dans une structure sémantique. L'auteur considère les structures spatiales et temporelles comme faisant partie de la structure sémantique.

Mais les documents multimédias peuvent être décrits en utilisant des métadonnées.

1.4. Les métadonnées multimédias

1.4.1. La définition

Avec la prolifération des contenus multimédias, les bases de données multimédias ont atteint une taille importante qui les rend difficiles à manipuler. Trouver une information pertinente devient

chronophage. L'utilisation d'une version concise des documents est apparue comme une solution pour la gestion de telles collections. Cette information concise représente les métadonnées.

La définition la plus connue du mot métadonnées est la suivante : « *données sur les données* ». Cette définition est floue et sujette à différentes interprétations. Une définition plus précise indique qu'une métadonnée est « *l'information structurée qui décrit, explique, localise la ressource et en facilite la recherche, l'usage et la gestion.* » (National Information Standards Organization, 2004).

(Greenberg, 2002) propose une autre façon de définir les métadonnées : « *des données structurées relatives à un objet, supportant des fonctions associées à cet objet.* ». En analysant cette définition on retrouve : le côté structuré des métadonnées, le fait qu'elles soient associées à toute ressource physique ou informatique, mais également les informations que les métadonnées peuvent contenir (e.g., sur le comportement de l'objet, qui facilite la recherche). Selon (Lamb, 2001), les deux principales fonctionnalités des métadonnées sont :

- De fournir de *l'information compréhensible par l'homme*, qui soit utilisée pour la recherche et l'interprétation des données électroniques ;
- De disposer de *données exploitables par les machines*, qui facilitent l'échange d'informations entre les systèmes, et qui facilitent le traitement des données par le système.

Les métadonnées en général, et les métadonnées associées aux documents multimédias plus précisément, décrivent un média, les relations intra-documents (e.g., entre les différentes parties du document), et les relations inter-documents (e.g., entre différents documents d'une collection).

La multiplicité des caractéristiques des métadonnées complexifie leur création et leur gestion. Il est donc nécessaire de les classer.

1.4.2. La classification des métadonnées

Plusieurs auteurs ont tenté de réaliser une classification des métadonnées en fonction de différentes caractéristiques (Böhm and Rakow, 1994), (De Sutter et al., 2006), (Lassila, 1998), (Pereira et al., 2008).

(Böhm and Rakow, 1994) proposent de classer les métadonnées en fonction de leur but principal, en obtenant la classification suivante :

- Métadonnées pour *la représentation des médias* : contenant des informations sur le format du média, les techniques de codage et de compression,
- Métadonnées qui *décrivent le contenu*,
- Métadonnées pour *la classification du contenu*,
- Métadonnées pour *la composition du document* : contenant des relations intra et inter-documents,
- Métadonnées pour *l'historique du document*,
- Métadonnées pour *la localisation du contenu*.

(Pereira et al., 2008) classent les métadonnées en deux grandes parties : métadonnées *associées au contenu* et métadonnées *associées à l'utilisateur*. Chacune de ces deux classes est divisée en sous-parties. D'une part, les métadonnées associées au contenu sont classifiées en :

- *Métadonnées de bas niveau* : extraites automatiquement du contenu (e.g., la couleur, la texture, la forme, la prosodie).

- *Métadonnées de haut niveau* : représentées, dans la majorité des cas, sous une forme textuelle et sont typiquement créées manuellement par l'homme. (e.g., les annotations, les mots-clés).
- *Métadonnées de structure* : utilisées pour représenter la structure des documents multimédias (e.g., segmentation spatiale et temporelle, structure des flux audio et vidéo).
- *Métadonnées sur le cycle de vie* : contiennent des informations sur l'acquisition, l'enregistrement, la modification, l'archivage et le codage du document.
- *Métadonnées d'identification et de localisation* : enferment des informations utilisées pour identifier et localiser le contenu multimédia.
- *Métadonnées de gestion* : contiennent des informations utiles pour la gestion des données en termes de droits et protections.

D'autre part, les métadonnées associées à l'utilisateur sont divisées en :

- *Métadonnées sur les interactions de l'utilisateur* : contiennent des informations sur l'expérience de l'utilisateur (e.g., ses préférences, l'historique de ses actions).
- *Métadonnées sur le contexte de l'utilisateur* : incluent des informations sur le contexte dans lequel l'utilisateur interagit avec le document multimédia (e.g., le réseau, le appareil d'affichage utilisé, la qualité de service demandée).

(De Sutter et al., 2006) proposent un classement différent axé sur trois critères :

- *Métadonnées internes ou d'échange* :
 - les métadonnées internes ne sont utilisées que dans le cadre d'un système de gestion de contenus multimédias.
 - les métadonnées d'échange décrivent la façon dont l'information est transmise de l'émetteur au récepteur.
- *Métadonnées plates ou hiérarchiques* :
 - les métadonnées plates décrivent le document en entier, et ne contiennent pas d'informations sur les composants du document.
 - les métadonnées hiérarchiques contiennent des informations sur chaque partie du document, en décrivant en détail les différents objets qui peuvent apparaître dans le document.
- *Le type des métadonnées* :
 - pour l'identification : permettent aux documents d'être identifiés de manière unique par l'homme et la machine.
 - pour la description et la classification : décrivent le contenu du document, et lui attribue un contexte de création et utilisation.
 - techniques : fournissent les caractéristiques techniques du document, e.g., le format, les codecs.
 - pour la sécurité et les droits : fournissent les informations nécessaires pour le transfert sécurisé du document, mais aussi les différents droits d'accès au document.
 - sur la publication du document.

Le domaine d'application des métadonnées est très vaste. (Lassila, 1998) présente une liste non exhaustive qui contient surtout des applications liées au Web et à la recherche des documents. Dans tous les domaines où les métadonnées sont utilisées, elles occupent une place importante dans la gestion des ressources, mais également dans le partage de l'information entre différentes applications. Nous allons nous intéresser au cycle de vie des métadonnées.

1.4.3. Le cycle de vie des métadonnées

Le cycle de vie des métadonnées est fortement lié à celui du document multimédia. Les métadonnées sont créées et mises à jour tout au long de l'existence du document multimédia. En outre, elles ne sont détruites qu'au moment où les documents multimédias, auxquels elles sont associées, sont détruits. En se basant sur ce constat, plusieurs travaux présentent les métadonnées au centre du cycle de vie du document multimédia (Hardman et al., 2008), (Smith and Schirling, 2006), ce qui les rend encore plus importantes dans la gestion des documents multimédias. De plus, les métadonnées sont consommées pendant toutes les phases du cycle de vie du document auquel elles sont associées (Kosch et al., 2005), (Smith and Schirling, 2006). Cette liaison entre les deux cycles de vie est représentée dans la Figure 5.

A chaque étape du cycle de vie d'un document multimédia l'utilisateur interagit avec les métadonnées d'une manière directe ou indirecte. Au moment de la création du contenu l'auteur crée des métadonnées à travers l'ajout des annotations sur les différentes parties du contenu. La première étape de la partie gestion des documents est l'indexation. Lors de cette étape le contenu multimédia est traité de manière automatique ou semi-automatique pour extraire d'autres informations qui peuvent être utiles ultérieurement dans les phases d'organisation et de recherche.

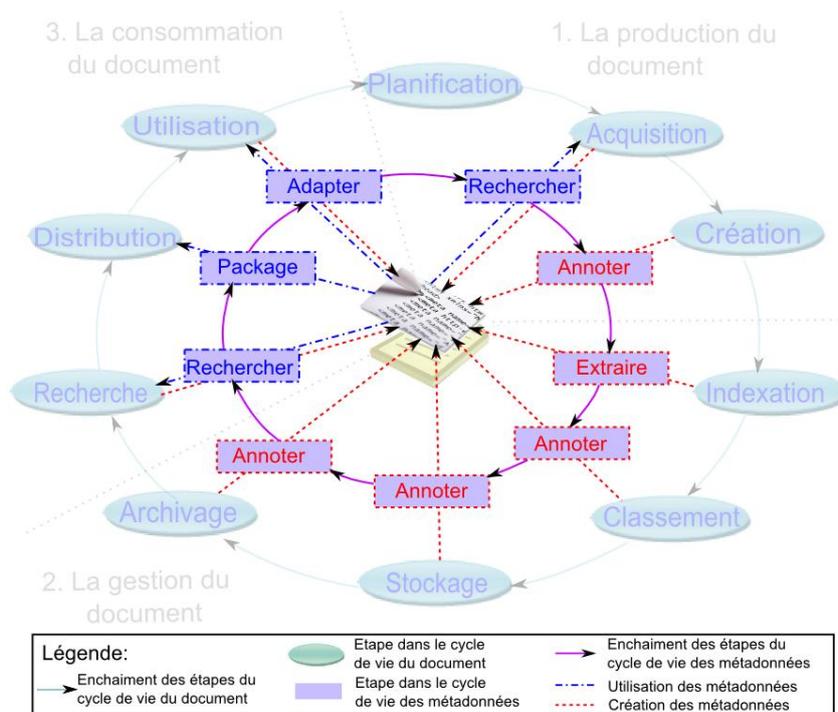


Figure 5 – Le cycle de vie d'un document multimédia et le rôle central des métadonnées

Au moment de l'organisation et du stockage du document multimédia, d'autres métadonnées peuvent être créées soit directement par l'utilisateur, soit déduites. Par exemple, la hiérarchie des dossiers au sein du disque de stockage permet de déduire certaines informations sur le document.

Avant l'utilisation du document multimédia, l'utilisateur passe par une phase de recherche, qui se réalise en interagissant directement avec les métadonnées, et par une phase d'adaptation qui est accomplie en fonction de différentes informations stockées dans les métadonnées. Pendant

l'utilisation du document, des métadonnées peuvent être créées en fonction des interactions de l'utilisateur avec le document (Mommel and Dengel, 2007; Wolpers et al., 2007; Manzat et al., 2010).

La Figure 5 montre que les métadonnées ont un rôle central dans la création et la gestion des documents multimédias. A chacune des étapes du cycle de vie du document un autre type de métadonnées est pris en compte. Par exemple, dans l'étape de création du document multimédia, des métadonnées administratives sont créées.

L'importance des métadonnées complexifie leur modélisation et leur standardisation. Il est donc essentiel de présenter une vue d'ensemble des différents standards de métadonnées existants.

1.4.4. Les standards et formats de métadonnées

Ces dernières années le nombre et l'hétérogénéité des formats de métadonnées ont fortement augmenté. Ces standards de métadonnées varient des plus simples aux plus complexes. La majorité de ces standards sont soit spécifiques à un type de média donné, soit généraux. Ceux derniers décrivent tout type de contenu multimédia ou sont spécifiques à chaque domaine.

1.4.4.1. Les standards de métadonnées pour l'image

Parmi les standards de métadonnées associées à l'image les plus connus et utilisés, citons : TIFF⁴, Exif⁵, XMP⁶, IPTC IIM⁷, DIG35⁸, MIX⁹.

La majorité de ces standards sont créés à l'initiative de compagnies privées dans le but de définir un format de métadonnées qui améliore l'échange des informations sur les images, que ces métadonnées soient stockées dans le même fichier que l'image brute ou dans un fichier spécifique.

TIFF (Tagged Image File Format) (Adobe Developers Association, 1992) est l'un des premiers formats de fichier à avoir essayé de rassembler sur le même support les données et les métadonnées correspondantes, en proposant une structure d'encapsulation de ces dernières. Mais, les informations qui peuvent être stockées dans le fichier TIFF sont assez limitées. Une trentaine de balises sont définies dans la norme de base, et une soixantaine dans l'extension. Toutes ces métadonnées sont de bas niveau.

Trois ans plus tard, la norme *Exif (Exchangeable image file format)* voit le jour (Japan Electronic Industry Development Association, 1995). Exif spécifie le format des images et des métadonnées associées. Cette norme ne s'applique qu'aux images créées par des appareils photos numériques. Les métadonnées sont créées par l'appareil photo au même moment que l'image. Elles sont de bas niveau, essentiellement liées aux caractéristiques de l'appareil photo au moment de la prise de vue (e.g., exposure time, F number, Aperture). Cette norme a introduit un élément nouveau : le stockage des coordonnées GPS de l'endroit où la photo a été prise.

Au début des années 1990, l'International Press Telecommunications Council (IPTC) et le Newspaper Association of America (NAA) ont commencé à travailler également sur un standard afin d'améliorer l'échange de l'information. Une première version de *IIM (Information Interchange Mo-*

⁴ <http://partners.adobe.com/public/developer/en/tiff/TIFF6.pdf>

⁵ http://www.digicamsoft.com/exif22/exif22/html/exif22_1.htm

⁶ <http://www.adobe.com/products/xmp/>

⁷ <http://www.iptc.org/std/IIM/4.1/specification/IIMV4.1.pdf>

⁸ <http://xml.coverpages.org/dig35.html>

⁹ <http://www.loc.gov/standards/mix/>

del) est proposée en 1991. Ce standard inclut la définition d'une structure pour l'intégration des métadonnées dans le même fichier que le contenu auquel elles sont associées, et un ensemble d'éléments de métadonnées. Les trente-trois métadonnées définies par cette norme sont de bas niveau également. Elles peuvent être appliquées aussi à des fichiers texte, à des images et à d'autres types de média.

Adobe a intégré une partie des éléments définis par IPTC-IIM dans les en-têtes de fichiers obtenus par son logiciel Photoshop, mais également dans la spécification du standard *XMP (Extensible Metadata Platform)* (Adobe Systems Incorporated, 2010a, 2010b, 2010c). La norme XMP est composée d'un modèle de données, d'un modèle de stockage et de schémas de métadonnées. Auparavant les métadonnées étaient intégrées dans le fichier de l'image sous forme d'une paire tag-valeur. Le XMP propose de représenter les métadonnées en utilisant du XML. En plus des métadonnées propres, il réutilise les métadonnées Dublin Core, TIFF et Exif.

Les producteurs d'appareils photos ont formé un groupe de travail en 1999. Leur principal objectif était de définir un ensemble de métadonnées standard pour les images numériques, qui permettent d'améliorer l'interopérabilité entre les appareils, les services et les logiciels qui les manipulent. Le standard *DIG35* a été ainsi proposé (Digital Imaging Group, 2000), (Digital Imaging Group, 2001). Il est instancié en utilisant du XML. En plus des métadonnées de bas niveau, DIG35 introduit des métadonnées sur l'historique et les différents droits intellectuels sur le document.

Le groupe NISO en collaboration avec l'ANSI, a proposé un schéma de métadonnées pour modéliser les propriétés des images, indépendamment de leur format de stockage. Il s'agit du *MIX*, qui se base sur le standard, *Data Dictionary - Technical Metadata for Digital Still Images* (ANSI/NISO Z39.87-2006) (MIX, 2006). Comme tous les autres standards présentés jusqu'ici, le MIX se focalise sur les métadonnées techniques qui accompagnent l'image matricielle, mais aussi sur l'historique des actions que l'image a subies tout au long de son cycle de vie. Les métadonnées MIX sont instanciées dans XML et elles sont sauvegardées dans un fichier externe à l'image.

Certains de ces standards sont utilisés par les plateformes de partage d'images. C'est le cas de Flickr¹⁰ et de Panoramio¹¹, qui exploitent les métadonnées EXIF qui se trouvent dans les images. En plus de ces métadonnées, les descriptions et les annotations des utilisateurs sont employées pour la gestion et la recherche d'image.

Les Annexes A à F fournissent d'avantage de précisions sur ces standards.

1.4.4.2. Les standards de métadonnées pour le texte

Parmi les standards de métadonnées associés au texte les plus connus et utilisés, citons : METS¹², textMD¹³, TEI¹⁴, FRBR¹⁵. La grande majorité de ces standards est créée à l'initiative des bibliothécaires, notamment ceux de la Librairie du Congrès Américain, qui est impliquée dans la création et le maintien de plusieurs standards, tels que METS et textMD.

Le standard *METS (Metadata Encoding and Transmission Standard)* représente une spécification pour l'encodage et la transmission des données, en exprimant les métadonnées nécessaires pour la gestion des objets numériques et leur partage entre les différents systèmes et institutions. Le stan-

¹⁰ <http://www.flickr.com/>

¹¹ <http://www.panoramio.com/>

¹² <http://www.loc.gov/standards/mets/>

¹³ <http://www.loc.gov/standards/textMD/>

¹⁴ <http://www.tei-c.org/>

¹⁵ <http://www.ifla.org/en/publications/functional-requirements-for-bibliographic-records>

dard a été réalisé à l'initiative des DLF (Digital Library Federation) et de la Librairie du Congrès Américain. METS a été conçu pour pouvoir décrire toute ressource bibliographique textuelle, numérique ou pas. On y retrouve des éléments de métadonnées qui décrivent le document du point de vue technique, administratif mais aussi structurel.

Le *TEI (Text Encoding Initiative)* est un consortium composé de chercheurs en sciences humaines et sociales, et de linguistes qui ont proposé des recommandations pour l'encodage des documents textuels, afin de faciliter leur partage et leur gestion. Ils ont réalisé une implémentation XML de ces recommandations. Le TEI définit une série d'éléments et d'attributs qui représentent différents concepts lexicaux et prosodiques du texte, mais aussi des métadonnées descriptives et structurelles. Les métadonnées se concentrent sur la description du contenu, mais elles contiennent également des informations techniques ainsi que d'autres données non-bibliographiques et l'historique des modifications antérieures du document.

Toujours en s'inspirant du domaine bibliographique, le *FRBR (Functional Requirements for Bibliographic Records)* contient des recommandations pour la modélisation des informations contenues dans les notices bibliographiques. C'est un standard qui n'est pas utilisé forcément pour la modélisation des documents textuels numériques. Toutefois, il offre une manière de représenter une information supplémentaire disponible sur une œuvre, qui peut contribuer à une recherche plus efficace dans les bases de données des bibliothèques. Le FRBR est conçu comme un modèle entité-relation. Les entités présentes dans le modèle sont regroupées en trois groupes, en fonction de leur rôle dans la fiche bibliographique. Les relations entre ces entités permettent une meilleure navigation et une compréhension accrue des notices. De plus, chaque entité a ses propres attributs qui la caractérisent et qui représentent les vraies métadonnées.

Les Annexes G à I fournissent d'avantage de précisions sur ces standards.

1.4.4.3. Les standards de métadonnées pour les contenus audiovisuels

Les contenus audiovisuels sont plus complexes parce qu'ils incorporent le paramètre temps. Parmi les standards de métadonnées pour les contenus audiovisuels les plus utilisés, citons : ID3¹⁶, QuickTime¹⁷, SMPTE¹⁸, MPEG-7¹⁹.

L'*ID3* est un acronyme utilisé pour représenter les métadonnées qui sont encapsulées dans les fichiers MP3. Les informations qui peuvent être sauvegardées par ce standard sont limitées, en nombre mais aussi en taille. La première version ne contient que six éléments (le titre de la chanson, l'interprète, le nom de l'album, la date de création, un commentaire et le genre). Dans la seconde version du standard un certain nombre d'éléments a été ajouté, comme par exemple les paroles, l'auteur, le compositeur. L'inconvénient majeur de cette norme est qu'elle n'offre aucun moyen de décrire plus en détails le contenu du fichier.

Le *QuickTime* est un framework multimédia défini en 1989 par Apple pour le stockage des contenus audio-visuels. Un fichier QuickTime est composé de deux parties : une description du contenu audiovisuel et le contenu lui-même. Pour la description du contenu, QuickTime utilise un nombre limité de métadonnées. Parmi celles-ci nous retrouvons : le titre, l'auteur, la date de création, le genre, les mots-clés et une description. Chaque élément de métadonnée a attribué un identifiant

¹⁶ http://www.id3.org/Developer_Information

¹⁷ <http://developer.apple.com/library/mac/#documentation/QuickTime/QTFF/Metadata/Metadata.html>

¹⁸ <http://www.smpete.org/standards>

¹⁹ <http://www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm>

unique utilisé pour sauvegarder sa valeur dans le même fichier que le contenu, sous la forme d'une paire clé-valeur.

Le SMPTE a défini un dictionnaire de métadonnées propre (SMPTE 335M-2001), qui contient des éléments décrivant le contenu audiovisuel. Ces métadonnées sont regroupées dans les catégories suivantes :

- identification et localisation du contenu.
- administration.
- description du contenu.
- paramétrage
- informations liées au processus de création et de traitement du contenu.
- relations entre les données.
- métadonnées spatio-temporelles.
- métadonnées réservées aux organisations, à usage public ou privé.
- métadonnées libres : qui peuvent être créées par tout autre utilisateur, en respectant la structure du standard.

Le dictionnaire des métadonnées a été publié sous forme d'un tableur. Pour pouvoir être utilisé plus facilement, le SMPTE a travaillé sur la représentation de ce dictionnaire en XML.

Le *standard MPEG-7* peut s'appliquer à tout type de contenu audiovisuel, dans n'importe quel domaine. Sa principale ambition est « de rendre les informations multimédias aussi faciles à trouver sur le Web que le texte l'est aujourd'hui » (Martinez, 2004). Son principal objectif est de fournir l'interopérabilité entre les systèmes et applications utilisées dans la production, la gestion, la distribution et la consommation du contenu audiovisuel. Les principaux éléments de la norme sont :

- **Les Descripteurs (D)** : définissent la syntaxe et la sémantique des caractéristiques du contenu multimédia.
- **Les Schémas de Description (SD)** : spécifient la structure et la sémantique des relations entre les composants du contenu (Ds ou SDs).
- **Le langage de Définition des Descripteurs (LDD)** : permet la création de nouveaux Ds et SDs. Il permet également l'extension et la modification des SDs existants ;
- **Les outils système** : permettent la gestion et la transmission efficace des descripteurs et des contenus multimédias, leur synchronisation, l'accès aux contenus tout en respectant les droits intellectuels.

Les Ds ont été conçus pour décrire des caractéristiques audiovisuelles de bas-niveau telles que : la couleur, la texture, la forme, le mouvement, la fréquence, etc.. Mais également, des caractéristiques générales telles que : la localisation, le temps, la qualité, etc.. Les SDs proposent des structures plus complexes qui décrivent des caractéristiques de haut-niveau, telles que : les régions, les segments ou encore les objets présents dans le contenu. Les SDs sont utilisés aussi pour modéliser des métadonnées liées à la création, la production et l'utilisation du contenu. Les SDs sont divisées en cinq classes :

- éléments de base.
- éléments pour la gestion du contenu : des métadonnées sur la création et la production du contenu, informations sur le codage, le stockage et le format de fichier utilisés.
- éléments pour la description du contenu : composés par des SDs de structure et des SDs sémantiques et des liaisons entre ces deux types de schémas.
- schémas pour l'organisation du contenu dans des collections.

- navigation et accès au contenu : réalisés à travers des SDs qui résument ou localisent une autre version du contenu.
- historique des actions de l'utilisateur et ses préférences sont aussi prises en compte dans le standard.

Le MPEG-7 est un standard très complexe, qui offre beaucoup de descripteurs et de types, mais qui le rendent difficile à comprendre et à utiliser. Pour la représentation du MPEG-7, le XML et le XSD sont utilisés. Toutefois, avec l'émergence du Web Sémantique, beaucoup de travaux ont été réalisés pour créer une ontologie en se basant sur MPEG-7 (Hunter, 2001; Troncy et al., 2007; García et al., 2008). On peut citer l'ontologie MPEG-7Onto²⁰.

Les Annexes J à M fournissent d'avantage de précisions sur ces standards.

1.4.4.4. Les standards généraux de métadonnées

Les standards présentés dans les parties précédentes sont dédiés à un type de contenu – soit du texte, soit de l'image, soit de l'audio, soit de la vidéo. Mais il existe aussi des standards généraux, qui s'appliquent à tout contenu multimédia, numérique ou pas. Parmi ces normes la plus connue et la plus utilisée est le *Dublin Core*²¹, (ANSI/NISO, 2007). Le DC est une norme ISO très simple, qui a un nombre limité de quinze éléments. Cette simplicité offre une facilité de création et de maintenance de ces métadonnées, mais aussi une sémantique couramment comprise. Les métadonnées DC sont facilement incorporées dans d'autres standards.

Dans leur grande majorité, les standards sont appliqués à tout contenu multimédia, dans n'importe quel domaine, mais d'autres normes ont été définies pour décrire les contenus multimédias dans un domaine précis. C'est le cas des standards suivants : DICOM²², LSCOM²³, LOM²⁴, NewsML, SportsML, EventsML²⁵.

Le *DICOM* (Digital imaging and communications in medicine) est un standard pour la gestion des données issues de l'imagerie médicale. Il a pour but de standardiser les données transmises entre les différents appareils de radiologie. En plus des données elles-mêmes, le fichier DICOM contient des informations supplémentaires sur le patient (e.g., son nom, sa date de naissance, ses allergies, son historique médical, etc.). Ce fichier contient aussi des informations dédiées aux caractéristiques de l'appareil de radiologie et des informations techniques relatives à l'image.

Le *LSCOM* (Large-Scale Concept Ontology for Multimedia) est une ontologie créée par un groupe de recherche composé de partenaires publics et privés. Il a pour but la création d'une taxonomie qui contienne uniquement des concepts qui respectent les critères : d'utilité, de couverture, de faisabilité et d'observabilité. Ceux-ci peuvent être extraits automatiquement à partir d'un contenu audiovisuel. Pour déterminer ces concepts, le groupe de travail s'est concentré sur le domaine du broadcast. La version complète de cette ontologie est composée de 834 concepts. Pour la version réduite de l'ontologie des algorithmes d'extraction automatique ont été déjà réalisés. Parmi les concepts de LSCOM-Lite on retrouve : des types de programmes (e.g., divertissement, sport, météo), des objets (e.g., avion, voiture, bus, animal), des personnes (e.g., personne, visage, police, dirigeant, foule), des endroits (e.g., dedans, dehors, montagne, ciel, bureau, ville).

²⁰ <http://rhizomik.net/html/ontologies/mpeg7ontos/>

²¹ <http://dublincore.org>

²² <http://medical.nema.org/Dicom/>

²³ <http://www.ee.columbia.edu/ln/dvmm/lscm/>

²⁴ http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf

²⁵ http://www.iptc.org/site/News_Exchange_Formats/

Le *LOM* (Learning Object Metadata) est un standard de description de n'importe quelle ressource pédagogique (i.e., toute entité – numérique ou pas – qui peut être utilisée pour l'apprentissage, l'éducation) (Learning Technology Standards Committee, 2002). Son but est de décrire le contenu des documents utilisés dans le domaine de l'E-Learning le mieux possible pour faciliter leur utilisation et leur recherche. Ce standard est composé de soixante-huit éléments de métadonnées, classés sous forme hiérarchique, et regroupés en plusieurs catégories :

- des métadonnées générales sur le contenu (e.g., le titre, les mots-clés).
- des métadonnées sur le cycle de vie du document.
- des éléments sur les droits d'utilisation du contenu.
- des informations pédagogiques : se focalisent sur le public cible du contenu.
- des informations utiles pour la classification du contenu.

Pour rendre les documents plus accessibles et pour faciliter l'échange de l'information des standards de représentation, des contenus multimédias ont été définis dans d'autres domaines. Par exemple, pour rendre les partitions musicales plus lisibles, le MusicXML²⁶ a été proposé. Outre le contenu de la partition, la norme contient aussi des informations administratives sur le document et sur les paroles de la chanson et des informations sur leur synchronisation avec le son. Dans le domaine des « news », les informations véhiculées sont très différentes. Les droits sur les contenus utilisés sont aussi très importants à prendre en compte. Pour répondre à tous ces besoins du domaine, l'IPTC a créé la News Architecture for G2-Standards (NAR). Cette architecture a pour but la description de tout contenu multimédia qui est utilisé dans le domaine des news, indépendamment de leur type. Le NAR se trouve à la base de tous les standards définis par le groupe, tel que NewsML-G2 – pour les news –, EventsML-G2 – pour les événements –, SportsML-G2 – pour des résultats sportifs –, ProgramGuideML-G2 et WeatherML. Toutes ces spécifications se focalisent sur la description des contenus, en définissant des métadonnées spécifiques au domaine. Des éléments de métadonnées définis dans le standard sont : le producteur, la date de création, la date de la dernière modification, les mots-clés, le sujet, la description, la localisation, le droit d'auteur, la période durant laquelle le document peut être utilisé, etc.. La valeur de certains de ces éléments est limitée à un vocabulaire défini par le l'IPTC.

Les Annexes N à R fournissent d'avantage de précisions sur ces standards.

1.4.5. La représentation des métadonnées

Ces dernières années la représentation des données en général a été très étudiée. Depuis la naissance du Web et jusqu'à maintenant, cette représentation a beaucoup évolué. On est passé de données brutes à une représentation beaucoup plus sémantique. Dans la littérature, la pile du web sémantique définie par Tim Berners-Lee (Figure 6) représente cette évolution a été réalisée en terme de recommandations W3C.

La base de cette pile est constituée par les URI, qui aident à identifier de manière unique les données, et sur l'UNICODE, qui est l'encodage le plus utilisé pour les données textuelles. Les URI font le lien entre les différentes couches de la pile, qui peuvent décrire la même donnée.

Une autre brique importante dans cette pile est représentée par le XML. Il offre une représentation syntaxique aux données, sous une forme hiérarchique. Le XML fournit une base de représentation flexible et extensible des données sous forme d'un arbre. Pour mettre un cadre à ces repré-

²⁶ <http://www.recordare.com/musicxml>

sentations, des recommandations telles que DTD et XSD ont été définies. Elles aident à définir un modèle de structure. Pour la manipulation des données représentées en XML des langages d'interrogation ont été proposés, tels que le XPath pour la localisation d'un élément dans l'arbre XML et le XQuery pour effectuer des requêtes sur des collections de documents XML. Le principal inconvénient du XML est sa focalisation sur la structure des données et non sur leur sémantique.

Pour dépasser cette limitation, le format RDF est utilisé (Manola and Miller, 2004). Il offre une représentation plus sémantique des données et les représente sous forme de graphe orienté. Le RDF se base sur des constructions de triplets : *sujet, prédicat, objet*, où le sujet représente la ressource à décrire. Le prédicat représente un type de propriété applicable à cette ressource. L'objet représente la valeur de la propriété. Le RDF peut être sérialisé en plusieurs formats tels que le RDF/XML ou le N3. Ces métadonnées peuvent être sauvegardées dans un fichier différent, mais elles peuvent être incluses dans le même, que le contenu qu'elles décrivent. On retrouve des métadonnées exprimées en RDF dans l'entête des fichiers SMIL, voire dans les fichiers HTML, en utilisant les propriétés du RDFa²⁷.

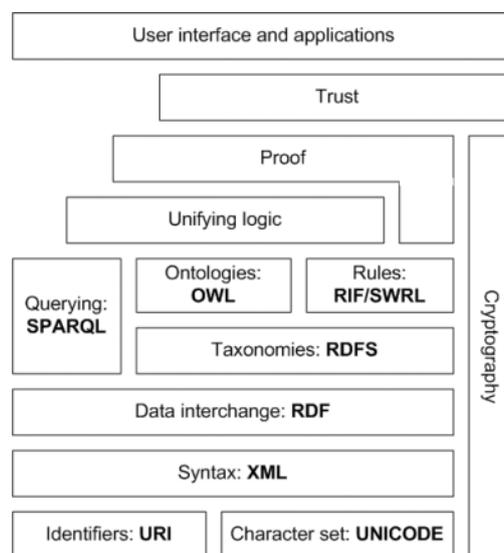


Figure 6 – La pile du web sémantique, selon Tim Berners-Lee

Le RDF est suffisamment générique pour servir de base à un certain nombre de schémas ou vocabulaires, tels que les ontologies RDFS ou OWL. En plus de la représentation sémantique des données, les ontologies offrent des règles d'inférence qui fournissent les outils pour raisonner et exploiter au maximum ces données.

Cette évolution des standards de représentation des métadonnées a été réalisée en prenant en compte l'utilisateur final. Le principal but de ces représentations est de pouvoir lui offrir une modalité d'exprimer le plus d'information possible pour faciliter la description, la gestion et la recherche des contenus.

1.4.6. Bilan sur les métadonnées multimédias

Au fil du temps, les métadonnées sont devenues de plus en plus importantes dans la gestion des contenus multimédias. Leur but principal est de rendre le contenu multimédia aussi accessible que

²⁷ <http://www.w3.org/TR/xhtml-rdfa-primer/>

le contenu textuel l'est aujourd'hui. Elles sont devenues un élément clé dans la gestion des contenus multimédias, en intervenant à chaque étape du cycle de vie de ces derniers.

Une grande variété de standards et de formats de métadonnées ont été définis pendant ces dernières années. Ils peuvent être comparés selon plusieurs critères :

- *L'initiative de création* : certains standards ont été créés à l'initiative des industriels, d'autres à l'initiative de groupes de recherche publics.
- *Le type de la ressource décrite* : les standards peuvent décrire : un contenu multimédia, un équipement d'acquisition ou d'affichage, une application, ou l'utilisateur.
- *Le domaine d'application* : les standards ont été conçus pour être utilisés dans un domaine précis ou dédiés à un usage général.
- *La population cible* : peut être différente d'un standard à un autre. Certains sont définis pour une utilisation purement professionnelle, d'autres sont dédiés au grand public.
- *Le niveau de description* : les métadonnées peuvent être extraites automatiquement du contenu multimédia (i.e., elles sont de bas-niveau), ou être représentées sous une forme textuelle et créées manuellement par l'homme (i.e., elles sont de haut-niveau).
- *Le type des métadonnées contenues* : chaque standard est différent par rapport aux métadonnées qu'il contient. On peut distinguer un certain nombre de catégories d'éléments :
 1. Identification, sous forme d'URI ou de titre, par exemple ;
 2. Localisation physique du contenu et de l'endroit de création du contenu ;
 3. Droits d'auteur et d'utilisation ;
 4. Description du contenu, en décrivant avec plus ou moins de détails des objets présents dans le document multimédia ;
 5. Description technique : des éléments de niveau bas, qui sont extraites automatiquement du document ;
 6. De structure : éléments qui décrivent sa structure spatiale ou temporelle ;
 7. Cycle de vie : métadonnées qui contiennent la suite des modifications souffertes par le document depuis sa création et jusqu'au moment présent ;
 8. Historique d'utilisation : toutes les opérations réalisées par les utilisateurs sur le document ;
 9. Informations externes : des informations supplémentaires sur le document ou sur une partie du contenu.
- *Extensibilité* : la possibilité d'inclure d'autres éléments est fournie par certains standards, tandis que d'autres ont une structure figée.
- *Vocabulaire contrôlé* : certaines normes fournissent la possibilité d'intégrer des vocabulaires contrôlés pour la définition des valeurs de leurs éléments.
- *Sérialisation* : pour la sauvegarde des métadonnées, le XML est le plus utilisé, parmi les standards présentés dans ce mémoire, certains définissent des formats binaires de stockage, efficaces en termes d'espace et de vitesse de transfert.
- *Stockage* : les métadonnées peuvent être stockées dans le même fichier que le contenu qu'elles décrivent ou dans un fichier externe.
- *Formalisation* : pour spécifier leurs éléments, les différentes normes de métadonnées utilisent soit des schémas XML et DTD, soit, plus rarement, des schémas RDF(S).

Une comparaison des standards de métadonnées est introduite dans le Tableau 1. Dans ce tableau, le symbole « x » a la signification que la norme, présente dans la colonne, respecte complètement ou presque le critère représenté dans la ligne. Le symbole « - » signifie que la norme, présente dans la colonne, respecte peu ou pas, le critère représenté dans la ligne.

En analysant ce tableau, nous pouvons voir qu'il y a un équilibre entre les standards créés à l'initiative des groupes de travail publics et industriels. Ils sont définis pour un certain type de media, et ils ne peuvent pas décrire tous les documents multimédias. Pour décrire exhaustivement un document multimédia, plusieurs standards de métadonnées sont nécessaires. Certains de ces standards sont définis pour un domaine d'application particulier et leur utilisation est restreinte à ce domaine. Tous les standards étudiés dans ce rapport sont assez compliqués à utiliser pour des utilisateurs profanes. Ils sont destinés à des usages professionnels.

Les standards de métadonnées contiennent soit des informations de bas niveau, soit de haut niveau, mais peu de formats contiennent les deux types d'informations (DIG35, METS, TEI, SMPTE, MPEP-7, DICOM). Ces standards ont pour la plupart d'entre eux une structure plate. Les éléments de métadonnées de ces standards sont de types variés. Mais aucun des standards ne couvrent l'ensemble ces types. Les plus complets sont METS, MPEG-7, SMPTE et TEI, mais ce sont des standards très complexes et trop difficiles à utiliser.

1.5. Bilan

A l'issue de l'étude présentée dans ce chapitre, on constate une « surcharge » de langage quand on parle de multimédia : pour certains, le terme multimédia est associé à tous les contenus numériques, excepté les documents purement textuels. Pour d'autres, le terme multimédia est associé aux dispositifs qui permettent la création de contenus numériques, autres que le texte, et qui permettent également l'affichage de ces contenus. Dans notre étude, nous entendons par « multimédia » les contenus qui combinent au moins deux médias via une structure.

En ce qui concerne la structure des documents, un autre problème se pose : en effet, un document peut avoir plusieurs structures différentes qui se complètent les unes aux autres et qui sont utilisées pour des buts distincts. La majorité de ces structures sont construites a priori et sont figées durant toute l'existence du document. Par exemple, les structures physiques, logiques, spatiales et temporelles du document multimédia sont essentiellement utilisées pour présenter le document à l'utilisateur. La structure sémantique, employée pour ajouter de l'information supplémentaire au document, est construite à partir du contenu du document, et des annotations supplémentaires ajoutées par les utilisateurs. Cette structure peut évoluer au fil du temps, par exemple, par l'ajout de nouvelles annotations.

La création des documents multimédias n'est pas toujours triviale. Dans la littérature, beaucoup de travaux ont défini le cycle de vie de ces contenus. Dans ce cycle de vie, l'accent est mis sur l'annotation des documents, la distribution et le partage, suite à l'apparition du web social.

La structure et la complexité des contenus multimédias, rendent leur gestion difficile. Pour rendre les documents multimédias aussi accessibles que les contenus textuels, les métadonnées semblent être une solution viable. Ces dernières années les métadonnées sont en constante évolution : beaucoup de standards et formats ont été définis à l'initiative des domaines publics ou privés. La grande diversité de standards de métadonnées rend difficile le choix du modèle de métadonnées à utiliser dans les systèmes de gestion multimédia. Dans le chapitre suivant nous présentons la manière dont les contenus multimédias sont gérés dans la littérature, en nous focalisant à la fois sur leur architecture et sur le modèle de métadonnées employés.

Tableau 1 – Étude comparative entre différents standards de métadonnées

Critères de comparaison	TIFF	EXIF	XMP	DIG35	MIX	IPTC-IIM	METS	TextMD	TEI	FRBR	ID3	QuickTime	SMPTE	MPEG-7	Dublin Core	LOM	LSCOM	DICOM	NewsML, SportsML, EventsML	MusicXML
Initiative de création																				
Publique	-	-	-	-	X	-	X	X	X	X	X	-	-	X	X	X	X	X	-	X
Privée	X	X	X	X	-	X	-	-	-	-	-	X	X	-	-	-	-	X	X	-
Descriptions associées																				
Au contenu	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Au Matériel	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
A l'application	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
A l'utilisateur	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Aux métadonnées générées par l'utilisateur	-	-	X	X	-	-	X	-	-	-	-	-	X	X	-	-	-	-	-	-
Type de contenu																				
Audio	-	-	X	-	-	-	-	-	-	X	X	X	X	X	X	X	X	-	X	X
Vidéo	-	-	X	-	-	-	-	-	-	X	-	X	X	X	X	X	X	-	X	-
Image	X	X	X	X	X	X	-	-	-	X	-	-	-	X	X	X	-	X	X	-
Texte	-	-	-	-	-	-	X	X	X	X	-	-	-		X	X	-	-	X	-
Multimédia	-	-	-	-	-	-	X	-	X	X	-	-	-	X	X	X	-	-	-	-
Domaine d'application																				
Général	X	X	X	X	X	X	-	-	X	-	X	X	-	X	X	-	-	-	-	-
Spécifique	-	-	-	-	-	-	X	X	-	X	-	-	X	-	-	X	X	X	X	X
Population cible																				
Professionnels	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Utilisateurs débutants	X	X	X	-	-	-	-	-	-	-	X	X	-	X	X	-	-	-	-	-
Niveau de description																				
Niveau bas	X	X	X	X	X	X	X	X	X	-	-	-	X	X	-	-	-	X	-	-
Niveau haut	-	-	-	X	-	-	X	-	X	X	X	X	X	X	X	X	X	X	X	X

Critères de comparaison	TIFF	EXIF	XMP	DIG35	MIX	IPTC-IIM	METS	TextMD	TEI	FRBR	ID3	QuickTime	SMPTE	MPEG-7	Dublin Core	LOM	LSCOM	DICOM	NewsML, SportsML, EventsML	MusicXML	
Type de métadonnées																					
Identification	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	-	X	X	X	
Localisation	X	X	X	X	X	X	X	-	X	X	X	X	X	X	X	X	-	X	X	-	
Droits	-	-	X	X	-	-	X	-	X	X	-	-	X	-	-	X	-	-	X	-	
Description du contenu	-	-	-	X	-	-	X	-	X	-	-	-	-	X	-	-	X	-	X	X	
Description technique	X	X	X	X	X	X	X	X	-	-	-	-	X	X	-	X	-	X	-	X	
De structure	-	-	-	-	-	-	X	-	X	-	-	-	X	X	-	-	-	-	X	-	
Cycle de vie	-	-	-	X	X	-	-	-	X	X	-	-	X	-	-	X	-	-	X	-	
Historique d'utilisation	-	-	-	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-	-	
Informations externes	-	-	-	-	-	-	X	-	X	X	-	-	-	-	-	X	-	X	X	-	
Extensibilité	-	-	X	X	-	-	X	-	-	-	-	-	X	X	-	-	-	-	-	-	
Vocabulaire contrôlé	-	-	-	-	-	-	X	-	-	X	-	-	X	X	-	-	X	-	X	-	
Sérialisation																					
XML	-	-	X	X	X	X	X	X	X	X	-	-	-	X	X	X	X	-	X	X	
Binaire	X	X	X	-	-	X	-	-	-	-	X	X	X	X	-	-	-	X	-	-	
Stockage																					
Interne	X	X	X	X	-	X	-	-	-	-	X	X	X	-	-	-	-	X	-	-	
Externe	-	-	X	X	X	-	X	X	X	X	-	-	-	X	X	X	X	-	X	X	
Formalisation																					
Texte	X	X	-	-	-	X	-	-	-	X	X	X	X	-	X	X	-	X	X	X	
DTD, XML schéma	-	-	X	X	X	X	X	X	X	X	-	-	X	X	X	X	-	-	X	X	
RDF(S)	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	-	-	-	

Chapitre 2. La Gestion des documents et des métadonnées multimédias

2.1.	<i>Introduction</i>	37
2.2.	<i>Le processus de gestion des documents multimédias</i>	38
2.2.1.	L'indexation des documents multimédias	38
2.2.2.	La recherche des documents multimédias	40
2.3.	<i>La gestion distribuée des documents multimédias</i>	41
2.3.1.	Les systèmes orientés services	41
2.3.1.	Les systèmes Pair-à-Pair	43
2.3.2.	Comparaison entre les systèmes distribués de gestion des documents multimédias	44
2.4.	<i>La gestion des métadonnées</i>	45
2.4.1.	L'interopérabilité des métadonnées	46
2.4.2.	L'hétérogénéité des formats de métadonnées et l'interopérabilité	46
2.4.3.	Les solutions existantes aux problèmes d'interopérabilité et d'hétérogénéité des métadonnées.....	49
2.5.	<i>Bilan</i>	51

2.1. Introduction

De nos jours, les contenus multimédias sont de plus en plus présents dans notre vie. Les avancées technologiques produites dans les domaines des appareils photos et des Smartphones incitent les utilisateurs à créer en permanence des contenus multimédias. La création de contenus multimédias n'est pas uniquement issue de nos activités privées, mais également de nos activités professionnelles. Dans beaucoup de domaines, les documents multimédias sont indispensables : le broadcast, la vidéosurveillance et plus récemment le domaine médical.

Ces contenus multimédias ne sont pas seulement nombreux, mais ils sont aussi très volumineux. Cela rend leur gestion efficace d'autant plus difficile à réaliser. Dans ce contexte, les métadonnées offrent une solution pertinente pour l'organisation et la recherche des contenus multimédias. L'apparition du Web Sémantique, qui a une influence directe sur les métadonnées et leurs formats, facilite aussi la description des contenus, en lui apportant de la sémantique supplémentaire. En même temps, le Web Sémantique complexifie en termes de modélisation et d'extraction des informations du contenu multimédia.

La gestion des contenus multimédias est encore plus difficile dans le contexte actuel, avec l'apparition du Web Social où tout tourne autour de l'utilisateur. Ce dernier est sans cesse en interaction avec le contenu et les métadonnées associées à ce contenu. En outre, une place importante est donnée au partage d'informations.

Ce chapitre expose la manière dont les documents multimédias sont gérés par des systèmes de l'état de l'art. Dans le contexte actuel où les documents sont distribués sur plusieurs serveurs, nous nous intéressons surtout aux solutions distribuées de gestion des documents multimédias.

Dans une première partie, nous nous intéressons au processus général de gestion des contenus multimédias et aux grandes étapes de traitement qui sont incluses dans ce processus. Nous présentons ensuite la façon dont ce processus peut être amélioré en prenant en compte l'utilisateur et ses interactions avec le système. La deuxième partie de ce chapitre contient une vue d'ensemble sur les architectures distribuées adoptées par des projets pour la gestion des contenus multimédias. La troisième partie présente le problème de l'interopérabilité des métadonnées associées aux contenus multimédias et quelques solutions développées pour le résoudre.

2.2. Le processus de gestion des documents multimédias

Par gestion de documents multimédias on entend le stockage intelligent des contenus afin d'obtenir une recherche d'information efficace, un accès rapide au contenu, et un partage facile des informations. Ce processus est divisé en deux grandes parties : l'indexation des documents et l'exécution des requêtes de l'utilisateur. Du point de vue conceptuel, le processus peut être vu comme un schéma en « U », présenté dans la Figure 7. D'un côté, se trouvent toutes les opérations qui sont réalisées par le système de gestion au moment de l'acquisition du contenu multimédia (i.e., l'indexation des documents, la partie gauche du schéma), de l'autre côté se situe le traitement de la requête de l'utilisateur (i.e., la partie droite du schéma). Entre ces deux parties, au milieu de l'« U », se trouve l'appariement, qui représente l'exécution de la requête sur la collection de documents multimédias pour obtenir une liste de résultats pertinents.

Ces deux parties du processus de gestion des contenus multimédias sont présentées dans les sections suivantes.

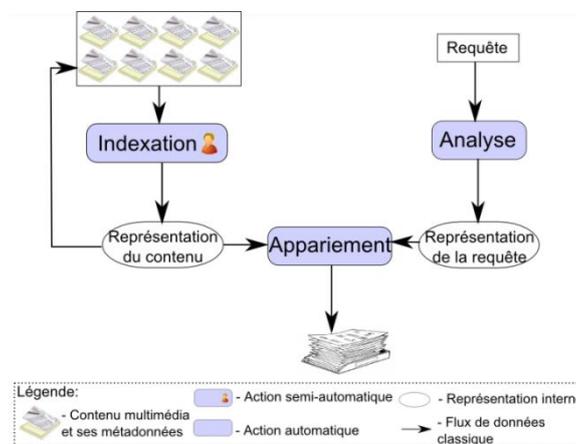


Figure 7 – Le processus de recherche d'information

2.2.1. L'indexation des documents multimédias

Selon le dictionnaire Larousse, l'indexation est « l'action d'indexer, d'établir l'index d'un document, d'un ouvrage ; placer un mot (auteur, matière, etc.) à sa place dans un index. »²⁸. Dans le cas des documents multimédias un seul mot ne suffit pas pour réussir à avoir une gestion efficace des contenus. Pour cela, dans la gestion des contenus multimédias une représentation plus complexe du contenu est utilisée. Cette représentation peut être constituée par les métadonnées elles-mêmes ou par une autre représentation purement numérique du document. Elle diffère d'un système de gestion à un

²⁸ <http://www.larousse.com/fr/dictionnaires/francais/indexer>, 2012

autre, en fonction du type de contenu et du domaine d'application du système qui sont pris en compte. Du point de vue conceptuel, la gestion des documents multimédias est équivalente à la gestion de leurs représentations.

Comme présenté dans le chapitre précédent, les métadonnées peuvent être sauvegardées soit dans le même fichier que le contenu soit dans un fichier différent. Pour être le plus générale possible, on peut supposer que nous pouvons avoir comme entrée dans un système de gestion de documents multimédias des contenus, mais aussi des métadonnées externes, fournies préalablement à l'entrée du contenu dans le système.

Dans le cas de documents multimédias, les métadonnées livrées avec les documents ne sont pas assez riches et donc elles ne sont pas suffisantes pour leur gestion. Une extraction supplémentaire de connaissances du contenu est donc nécessaire. De manière générale cette opération porte le nom d'« Indexation » (Figure 7). Elle consiste dans l'exécution de différents algorithmes d'indexation qui ont pour but d'extraire des informations qui peuvent être utiles pour la gestion des documents. Les algorithmes utilisés sont différents d'un domaine d'application à l'autre. De plus, les algorithmes employés sont fortement dépendants du type de contenu multimédia géré par le système. Certains algorithmes ne s'appliquent qu'à une certaine catégorie de document.

Par exemple, (Lambolez et al., 1995) proposent un algorithme d'extraction de l'information des documents textuels. (Ma and Manjunath, 1999) proposent l'utilisation de la couleur, de la forme et de la texture pour la gestion des images. Pour les documents audio, (Gauvain et al., 2001) réalisent de manière automatique la segmentation du contenu dans segments homogènes du point de vue acoustique et une transcription du discours en texte pour faciliter la gestion des contenus audiovisuels. D'autres types d'informations peuvent être extraits des contenus audio, tels que : la segmentation musique/non musique ou parole/non parole (Mauclair and Pinquier, 2004; Pinquier and André-Obrecht, 2006). La transcription du discours en texte (Barras et al., 2001) permet d'obtenir d'autres informations sur le contenu audio.

Pour les vidéos, la situation est plus complexe encore du fait de la complexité du contenu à gérer. (Ford et al., 2000) propose une segmentation de la vidéo en frames, en obtenant une structure plate et qui muette sur la sémantique de son contenu. (Kankanhalli and Chua, 2000) propose la segmentation du contenu en couches. Chaque couche correspond à un concept et elle modélise la présence ou l'absence de ce concept dans le contenu, tout au long de la vidéo. Beaucoup de travaux s'intéressent à l'extraction de la sémantique des contenus vidéos, (Snoek and Worring, 2005) présente un état de l'art sur ce qui se fait dans ce domaine. Le principal but de ces travaux est d'obtenir une segmentation hiérarchique de la vidéo. Dans cette segmentation nous distinguons des *Scènes* qui correspondent à des unités dramatiques de la vidéo. Une scène a toujours lieu dans une période de temps continue. Dans les scènes, nous distinguons les *Plans* qui correspondent à une unité de prise de vue, avec ou sans mouvements de caméra. Enfin il existe des entités *Événement*. Un événement est une sous-séquence d'images de la vidéo qui représente une action dans un plan. Il est rattaché au plan ou à une scène qui, le contient entièrement. Un niveau supplémentaire de cette structure est représenté par les *personnages* et les *objets* impliqués dans la réalisation d'un événement. Cette segmentation offre des informations supplémentaires très utiles pour le processus de recherche. L'attention des chercheurs s'est focalisée plus récemment sur l'extraction d'informations plus complexes du contenu, comme par exemple les concepts définis dans LSCOM (Jiang et al., 2010), ou les objets oubliés (Miezianko and Pokrajac, 2008).

La majorité des techniques d'indexation présentées dans l'état de l'art sont semi-automatiques, nécessitant une intervention de l'expert au moins dans la phase d'apprentissage, en fournissant à l'algorithme des exemples positives et négatives avec le concept qu'il doit extraire (e.g.,

des images avec des avions, et des images qui ne contiennent pas d'avions, pour un algorithme qui détecte la présence d'un avion dans une image). De plus en plus de travaux réalisent des algorithmes d'indexation de contenus multimédias, qui sont automatiques et qui réalisent des détections de plus en plus fiables.

Pour faciliter la gestion et notamment la recherche d'informations, une représentation unique des métadonnées doit être choisie. Elle doit être commune à tout type de contenus multimédias gérés par le système.

2.2.2. La recherche des documents multimédias

Une fois la phase d'indexation réalisée, les documents sont disponibles pour être utilisés et mis à la disposition des utilisateurs. Pour rendre plus accessible l'information, les systèmes de gestion des documents multimédias adoptent une interface de plus en plus performante pour la saisie de la requête par l'utilisateur et pour l'affichage des résultats obtenus, suite à l'exécution de la requête sur la collection des documents.

Les systèmes offrent la possibilité de saisir une requête de plusieurs façons : par mots clés, par contenu (Flickner et al., 1995), (Snoek and Worring, 2009), par croquis (Ghazal and Sedes, 2010) ou par combinaison de ces possibilités.

Pour pouvoir retrouver les documents pertinents à la requête de l'utilisateur, celle-ci doit être analysée pour obtenir la même représentation que la représentation utilisée pour modéliser les documents multimédias. L'utilisation de la même représentation des deux côtés de l'« U » aide à la réalisation de la fonction d'appariement.

La recherche par mots clés est utilisée surtout, pour la recherche de documents textuels (Salton and Mcgill, 1986). Ce type de recherche peut être employé pour la recherche de documents multimédias grâce au travail réalisé en amont, dans la phase d'indexation, pour l'extraction des connaissances des contenus multimédias et leur modélisation sous forme de métadonnées.

La phase d'appariement consiste à calculer une similarité entre les représentations des contenus multimédias obtenus suite à leur indexation et la représentation de la requête. Cette fonction de similarité peut être plus ou moins complexe, en fonction de la représentation des données choisie et du domaine d'application du système. Le résultat de cette fonction d'appariement fournit une liste de documents multimédias ordonnés selon le score obtenu par la fonction de similarité.

La recherche par le contenu est de plus en plus utilisée par les systèmes de gestion de documents multimédias. Le principe de ce type de recherche est le suivant : l'utilisateur envoie un exemple de contenu qu'il veut obtenir et il attend que le système lui renvoie tous les documents multimédias qui ressemblent au sien ou qui contiennent le contenu de la requête. Par exemple, l'utilisateur envoie comme requête une image avec un chat ; son but est de retrouver un maximum d'images contenant un ou plusieurs chats. Pour ce type de recherche, la phase d'analyse de la requête consiste dans l'exécution des mêmes algorithmes d'indexation sur la requête que ceux utilisés dans la phase d'indexation. Ainsi, il est certain qu'on va comparer la même chose dans la phase d'appariement.

Ce processus de gestion des contenus multimédias est indépendant de l'architecture physique ou logique du système de gestion. Il s'applique au système multimédia classique, où les contenus multimédias sont centralisés, mais également au système multimédia distribué, où les contenus multimédias sont stockés sur des serveurs distants. La gestion distribuée des documents multimédias est détaillé dans la section suivante.

2.3. La gestion distribuée des documents multimédias

La forte croissance de la quantité de contenus multimédias qui doivent être gérés a eu une importante influence sur l'architecture des systèmes de gestion. De plus, la rapidité de la recherche et de l'accès aux documents sont des facteurs clés du succès. Pour assurer cette rapidité, les systèmes multimédias distribués sont plus utilisés que les systèmes centralisés. Les documents multimédias, les métadonnées qui les décrivent, l'indexation et le traitement de la requête sont distribués également.

Nous allons présenter brièvement les différents types de systèmes distribués de gestion des contenus multimédias : les systèmes pair-à-pair et les systèmes orientés services.

2.3.1. Les systèmes orientés services

La nécessité de limiter le nombre de transferts sur le réseau et d'assurer l'interopérabilité entre les différents systèmes rend les systèmes adoptant une architecture orientée services de plus en plus populaires. Ces systèmes assurent l'accès aux contenus multimédias depuis n'importe quel lieu, à n'importe quel moment (Tanenbaum and Steen, 2001).

Nous présentons quelques projets qui gèrent des collections multimédias distribuées et qui adoptent une telle architecture.

Le projet MODEST (Multimedia Object Descriptors Extraction from Surveillance Tapes)²⁹ figure parmi les premiers projets qui proposent un système de vidéosurveillance basé sur une architecture multi-agent. L'idée de base de ce projet est de réaliser la détection d'événements inattendus, la détection du mouvement et le suivi d'objets dans la même vidéo ou dans des vidéos enregistrées par différentes caméras situées sur une autoroute. Dans cette architecture, les algorithmes d'indexation sont développés comme agents qui peuvent communiquer entre eux pour résoudre leur tâche (Abreu et al., 2000). Chaque serveur gère les contenus acquis par une seule caméra. Sur chaque serveur un agent s'occupe de la segmentation du contenu et de l'identification des différents objets présents dans le contenu. Cet agent produit en sortie des descripteurs MPEG-7 qui sont stockés sur le même serveur que le contenu. D'autres agents installés sur un serveur dédié, mais qui peuvent aussi se déplacer, utilisent ces descriptions pour réaliser des tâches plus complexes, comme la classification des objets détectés (e.g. véhicule, camion, personne) et le suivi de ces objets. Des informations statiques sur le contenu de chaque serveur sont utilisées pour détecter des comportements suspects et pour réaliser des statistiques.

Dans le domaine, (Roth et al., 2006) proposent l'utilisation des agents pour l'indexation distribuée des images. Ce sont alors les agents qui se déplacent sur le réseau et non les images. Les algorithmes d'indexation utilisés sont basiques. Les descripteurs obtenus sont représentés sous la forme de vecteurs. Deux stratégies sont étudiées pour le stockage de ces descripteurs : (1) de manière centralisée – la requête de l'utilisateur n'étant traitée que sur le serveur central qui contient tous les descripteurs ; (2) de manière distribuée, sur le même serveur que les images – la requête de l'utilisateur est envoyée à tous les serveurs du système pour retrouver l'information pertinente. (Roth et al., 2006) prônent la limitation des transferts sur le réseau pour diminuer la consommation des ressources.

Le MUSCLE network of excellence (Multimedia Understanding through Semantics, Computation and LEarning)³⁰ a développé un système de gestion des algorithmes d'indexation. Le but principal de ce projet fut la création d'algorithmes d'indexation et d'extraction d'informations du contenu multimédia (e.g., reconnaissance d'objets, détection d'un comportement suspect, détection de

²⁹ <http://www.tele.ucl.ac.be/PROJECTS/MODEST/index.html>

³⁰ <http://www.muscle-noe.org>

personnes, reconnaissance de la parole, résumés de vidéos), et pas la création d'une architecture pour la gestion et la recherche de contenus multimédias.

Le projet CANDELA (Content Analysis and Network DELivery Architectures)³¹ se focalise sur l'analyse et la recherche du contenu vidéo dans plusieurs domaines (e.g., vidéosurveillance (Jaspers et al., 2005), vidéos personnelles (Pietarila et al., 2005)). L'architecture utilisée par le projet est composée par un serveur central qui contient tous les services mis à disposition de l'utilisateur. Le contenu multimédia est téléchargé pour être indexé. Des serveurs spécifiques stockent les métadonnées générées par l'indexation (Petkovic and Jonker, 2003). Dans ce système les métadonnées sont représentées sous forme de descripteurs MPEG-7. Pour leur stockage une base de données distribuée est employée, les descripteurs MPEG-7 étant traduits en tables de base de données. Ces descripteurs sont extraits en appliquant un ensemble fixe d'algorithmes d'indexation développés dans le cadre du projet et qui génèrent les métadonnées sous le format MPEG-7.

Le projet WebLab³² propose une plateforme d'intégration pour les applications de gestion de contenus multimédias (Giroux et al., 2008). Du point de vue logique, l'architecture proposée est composée de trois couches : le WebLab Core, le WebLab Services et le WebLab Applications. Pour assurer l'intégration de tout algorithme d'indexation, la plateforme les considère comme étant des services web qui doivent respecter l'interface proposée. Dans une optique d'intégration, le projet a proposé un format de données XML, mais qui peut être étendu avec du RDF, pour exploiter d'avantage la sémantique qui se trouve dans les documents. Une palette assez importante de fonctionnalités est proposé par la plateforme : acquisition des données (depuis : le web, la radio, TV,...), annotation de documents, extraction d'entités des documents textuels, détection d'objets dans les images et les vidéos, transcription speech-to-text, recherche, classification. Cette plateforme est utilisée par plusieurs autres projets. Par exemple : WebContent³³, e-Wok Hub³⁴, VITALAS³⁵, AXES³⁶.

Le projet VITALAS (Video & image Indexing and retrieval in the Large Scale) utilise la plateforme WebLab pour intégrer des algorithmes d'indexation de contenus multimédias et les appliquer sur de grandes collections de documents (Viaud et al., 2008). Dans le prototype réalisé à la fin du projet, les documents se trouvent sur des serveurs distribués, et ils sont indexés offline, avec un set fixe d'algorithmes d'indexation. Pour la requête, le système utilise une recherche par mots-clés, et une recherche par l'exemple.

Le projet KLIMT (Knowledge InterMediation Technologies) propose également une architecture orientée services pour l'intégration des applications qui traitent des contenus multimédias dans un environnement distribué (Conan et al., 2003). Dans ce projet différents algorithmes d'indexation peuvent être installés sur chaque serveur. Les métadonnées obtenues sont gérées de manière centralisée et elles respectent un format XML défini dans le cadre du projet. Ce projet se concentre plutôt sur la création d'algorithmes d'indexation qu'à la recherche de contenu multimédia. Pour des scénarios d'utilisation identifiés dans le projet, un ensemble de règles d'enchaînement des algorithmes d'indexation a été déterminé.

³¹ <http://www.hitech-projects.com/euprojects/candela>

³² <http://weblab-project.org/>

³³ <http://www.webcontent.fr/>

³⁴ <http://www-sop.inria.fr/edelweiss/projects/ewok/>

³⁵ <http://vitalas.ercim.org/>

³⁶ <http://www.axes-project.eu/>

2.3.1. Les systèmes Pair-à-Pair

Les systèmes pair-à-pair (P2P) peuvent être divisés en deux grandes catégories : systèmes *pures* (i.e., il n'y a aucune différence entre les serveurs du système, tous étant pareils du point de vue physique et logique) ou *hybrides* (i.e., les serveurs sont différents entre eux, ayant des tâches différentes à réaliser ; certains serveurs – des supers-pairs – ont la charge de gérer d'autres serveurs qui se trouvent dans leur domaine).

Les systèmes P2P purs, sont à leur tour divisés en *systèmes structurés* et *systèmes non-structurés*. Les principales différences entre ces deux types de systèmes sont : la topologie du système et le protocole de communication entre les serveurs. Les systèmes structurés adoptent une topologie fixe et un protocole de routage des messages bien défini. Pour les systèmes non-structurés, la situation est différente : les serveurs peuvent connecter au système à n'importe où et chaque serveur peut communiquer avec tous ses voisins, dans n'importe quel ordre.

Dans les systèmes P2P structurés, la collection de documents est indexée en utilisant un mécanisme de hashing distribué, où chaque serveur est responsable d'un ou plusieurs termes, qui se retrouvent dans les documents de la collection. Plus précisément, chaque pair maintient, pour chaque terme dont il est responsable, une liste globale de tous les documents de la collection qui contiennent ce terme. La requête de l'utilisateur est reçue par n'importe quel serveur et est divisée en un ensemble de termes. Une fois cette phase réalisée, le serveur télécharge pour chaque terme, depuis le serveur qui le gère, la liste globale des documents. Une fois toutes les listes reçues, le serveur réalise une intersection entre elles et la requête est envoyée aux serveurs ainsi sélectionnés pour retrouver les documents. Chord (Stoica et al., 2003), CAN (Ratnasamy et al., 2001), Pastry (Rowstron and Druschel, 2001), Skipnets (Harvey et al., 2003), Tapestry (Zhao et al., 2004), MINERVA (Papapetrou et al., 2005), (Bender et al., 2005) sont de tels systèmes P2P structurés.

Dans les systèmes P2P non-structurés, chaque serveur indexe les contenus qu'il stocke et il construit un résumé de son contenu et le distribue à ses voisins. Dans ce genre de système il n'existe pas de vision globale sur les contenus gérés. Au moment de l'exécution d'une requête, le serveur qui la reçoit, utilise les résumés sur ses voisins pour décider à quel serveur l'envoyer afin d'y être exécutée. La requête est également envoyée aux autres serveurs, dans un souci d'exhaustivité des réponses. Ainsi elle peut être transmise à beaucoup de serveurs. Pour éviter la transmission d'une requête à l'infini dans ce genre de système, on y associe un TTL (Time to live) (i.e., une requête ne peut pas être envoyée à plus de n serveurs). PlanetP (Cuenca-Acuna et al., 2003), CORI (Callan et al., 1995), GLOSS (Gravano et al., 1999), Galanx (Wang et al., 2003) sont des systèmes P2P non-structurés.

Les systèmes P2P hybrides sont utilisés assez souvent pour la gestion des collections de documents textuels. De telles architectures sont adoptées par ODISEEA (Suel et al., 2003) pour la gestion et la recherche de documents textuels et (Lu and Callan, 2003) pour la recherche basée sur le contenu. Les deux approches utilisent une architecture P2P à deux niveaux : des supers-pairs qui contiennent un index global remonté des pairs feuilles qu'ils gèrent, et des pairs qui indexent et stockent les documents. Les supers-pairs ont pour but de réaliser l'appariement entre la requête de l'utilisateur qu'ils reçoivent tous et l'index stocké, pour cibler le maximum de serveurs qui contiennent l'information demandée.

Les systèmes que nous venons de présenter ne se focalisent que sur la gestion et la recherche de grandes collections de documents textuels. Dans le cadre de ces systèmes l'indexation des documents est réalisée en utilisant le classique TF/IDF. Pour la gestion des contenus multimédias de manière distribuée le choix de la représentation des contenus est très important. Parmi les projets qui se focalisent sur la gestion des contenus de manière distribuée, citons : SAPIR et DISCO.

Le projet SAPIR (Search on Audio-visual content using Peer-to-peer Information Retrieval) (Agosti et al., 2007), (Batko et al., 2010) utilise une architecture P2P structurée, où tous les serveurs ont les mêmes fonctionnalités. Les contenus multimédias sont indexés au niveau du serveur où ils sont stockés. Les mêmes algorithmes d'indexation sont utilisés sur chaque serveur. Des descripteurs MPEG-7 sont utilisés pour la représentation des contenus (e.g., pour les images, les descripteurs pris en compte sont : ColorStructure, ColorLayout, ScalableColor, EdgeHistogram, HomogeneousTexture). Ces descripteurs sont intégrés dans un format propre, défini dans le cadre du projet (Kaplan et al., 2007) et qui est instancié en XML. Les métadonnées obtenues suite à l'indexation sont stockées sur le même serveur que le contenu, sous la forme de M-Tree (Ciaccia et al., 1997). Chaque serveur envoie sur le réseau un résumé des métadonnées extraites de ces contenus, qui est concaténé dans un index global, qui à son tour est distribué aux serveurs du système en utilisant une structure de type M-Chord (Novak and Zezula, 2006)

Le projet DISCO³⁷ (Distributed Indexing and Search by Content) utilise aussi une architecture P2P structurée (Boisson et al., 2008). Dans ce système, les serveurs peuvent avoir leur propre manière de représentation des contenus. Comme dans tout système P2P, un index global est créé avec l'information représentative, envoyée par chaque serveur. Le système utilise une représentation pivot pour l'index global, et chaque serveur réalise le mapping entre sa représentation et l'index global.

2.3.2. Comparaison entre les systèmes distribués de gestion des documents multimédias

Pour synthétiser les différents avantages et inconvénients de chacun des projets présentés dans les deux sections précédentes, nous avons réalisé une comparaison en fonction de certains critères qui nous semblent importants :

1. *Le domaine d'application* du projet : si le projet est générique, s'il a été développé en partant d'un cas d'utilisation bien précis ou s'il n'est applicable que dans un domaine ;
2. *Le type de contenu* pris en compte : des textes, des images, des contenus audiovisuels ou des contenus multimédias ;
3. *Le type d'architecture* considéré : du pair-à-pair, architecture orientée services, multi-agents, ou une architecture distribuée qui détient un serveur central qui contrôle tout le système ;
4. La façon dont *l'indexation* est réalisée : les mêmes algorithmes d'indexation sont exécutés sur tous les serveurs du système (i.e., indexation uniforme) ; les algorithmes d'indexation exécutés ne changent pas au cours du fonctionnement du système (i.e., indexation fixe) ; pour être indexé, le contenu multimédia est transféré vers un serveur dédié ou il est indexé sur le serveur où il est stocké. Les métadonnées obtenues suite à l'indexation sont transférées vers un serveur central ou pas ;
5. *Les métadonnées* obtenues suite à l'indexation sont représentées dans un format propre au système ou dans un format standard. Toutes les métadonnées du système sont agrégées pour obtenir une version globale et centralisée du système.
6. *L'interrogation* du système se réalise de manière optimisée en n'envoyant la requête de l'utilisateur qu'à une partie des serveurs susceptibles de contenir des informations pertinentes.

³⁷ <http://www.lamsade.dauphine.fr/disco/index>

Tableau 2 – Étude comparative entre les systèmes distribués de gestion de contenus multimédias

Critères de comparaison	SAPIR	DISCO	MINERVA	ODISEEA	MODEST	MUSCLE	CANDELA	WebLab	VITALAS	WebContent	KLIMT	CAM4HOME
Domaine												
Générique	X	X	X	X	-	X	X	X	-	-	-	X
Spécifique	-	-	-	-	X	-	-	-	X	X	X	-
Type de contenu												
Multimédia	-	X	-	-	-	X	X	X	-	-	X	X
Texte	X	-	X	X	-	-	-	-	-	X	-	-
Image	-	-	-	-	-	-	-	-	-	-	-	-
Audio, vidéo	X	-	-	-	X	-	-	-	X	-	X	-
Type architecture												
Pair-à-Pair	X	X	X	X	-	-	-	-	-	-	-	-
Orientée services	-	-	-	-	-	-	-	X	X	X	X	X
Contrôle central	-	-	-	-	-	X	X	-	-	-	-	-
Multi-agents	-	-	-	-	X	-	-	-	-	-	-	-
Indexation												
Uniforme	X	-	X	X	X	X	X	-	-	-	X	X
Fixe	X	X	X	X	X	X	X	X	X	X	X	X
Transfert de contenu	-	-	-	-	-	X	X	-	-	-	-	-
Transfert des métadonnées	X	X	X	X	-	X	X	-	-	-	X	X
Métadonnées												
Standard	-	-	X	X	X	-	X	-	-	-	-	-
Format propre	X	X	-	-	-	X	-	X	X	X	X	X
Vision globale et centralisée du système	-	-	-	-	-	X	X	-	-	-	X	X
Requête												
Sélection des serveurs	X	X	X	X	-	-	-	-	-	-	-	-

Le tableau ci-dessous montre que les systèmes pair-à-pair ont un domaine d'application générique, mais sont tributaires de leur structure en termes d'indexation et de construction de l'index global qui permet le routage d'une requête. Les autres systèmes sont développés en partant d'un domaine précis. Ils ne disposent pas de mécanismes de sélection des serveurs pertinents pendant le processus d'interrogation. L'indexation est réalisée pour tous les projets avec les mêmes algorithmes d'indexation tout au long de l'exécution de ce processus.

Comparaison montre que presque tous les projets ont défini leur propre format de métadonnées, ce qui ne rend pas facile la réutilisation de leurs résultats, en mettant en évidence un problème d'interopérabilité des systèmes.

Nous allons expliciter la manière dont les métadonnées sont gérées dans la littérature et comment est abordé le problème de l'interopérabilité.

2.4. La gestion des métadonnées

Dans le cadre des systèmes de gestion des documents multimédias, les métadonnées occupent une place importante. Elles permettent d'avoir une description sémantique des contenus. En

outre, elles permettent de mieux structurer la collection de documents et améliorent recherche. Mais se pose le problème du choix du format de métadonnées. Comment nous l'avons vu dans la section 1.4, beaucoup de formats de métadonnées hétérogènes ont été définis. Ces formats ne sont pas forcément interopérables.

2.4.1. L'interopérabilité des métadonnées

Le dictionnaire « Larousse » définit l'interopérabilité comme la « Compatibilité des équipements, des procédures ou des organisations permettant à plusieurs systèmes, forces armées ou organismes d'agir ensemble »³⁸.

Dans le domaine informatique, (National Information Standards Organization, 2004) propose une autre définition de l'interopérabilité : « la capacité de multiples systèmes, qui utilisent différentes plateformes matérielles et logicielles, différentes structures de données et interfaces, d'échanger et partager leurs données ». (ALA/ALCTS/CCS Committee, 2004) et (Taylor, 2004) enrichissent cette définition avec la notion d'effort : « la capacité de deux ou plusieurs systèmes d'échanger des informations et des données et d'utiliser ces informations et ces données échangées sans effectuer de manipulations spéciales. » Pour (Baker et al., 2002), l'interopérabilité des métadonnées représente « le potentiel des métadonnées à franchir les frontières entre les différents contextes de l'information. » Plus récemment, (Haslhofer and Klas, 2010) définissent cette interopérabilité des métadonnées comment étant « une propriété qualitative des objets d'information représentés par les métadonnées qui permet aux systèmes et aux applications de s'en servir ou de les utiliser au-delà des frontières du système. »

Ces définitions mettent en exergue des informations importantes : la compatibilité et l'échange facile d'informations entre les différents systèmes. Mais ce n'est pas forcément le cas des métadonnées.

2.4.2. L'hétérogénéité des formats de métadonnées et l'interopérabilité

La section 1.4 a montré qu'il existe une grande diversité de standards et de formats qui rendent le choix du format de métadonnées à utiliser, dans un contexte particulier, très difficile. En réalité, pour décrire complètement un document multimédia plusieurs standards de métadonnées doivent être employés, parce qu'une description complexe nécessite de multiples vocabulaires. (De Sutter et al., 2006) présentent quelques critères à prendre en compte au moment du choix du format des métadonnées à utiliser, mais une solution unique n'est pas fournie.

Le problème de l'interopérabilité se pose parce que deux vocabulaires différents peuvent partager certains éléments, mais chacun d'entre eux possède des éléments spécifiques. Ce problème est illustré pour des standards de métadonnées spécifiques aux images par (Metadata Working Group, 2009) (Figure 3). Cet exemple montre que les coordonnées GPS ne sont spécifiées que dans EXIF, mais pas dans les deux autres standards. De même EXIF ne permet pas d'introduire des mots-clés dans la description d'une image, alors que le XMP et l'IPTC IIM le permettent. Cette comparaison montre que ces trois standards de métadonnées n'ont que quelques éléments communs (e.g., le créateur, la description, la date de création).

³⁸ <http://www.larousse.com/en/dictionnaires/francais/interop%C3%A9rabilit%C3%A9/43787>

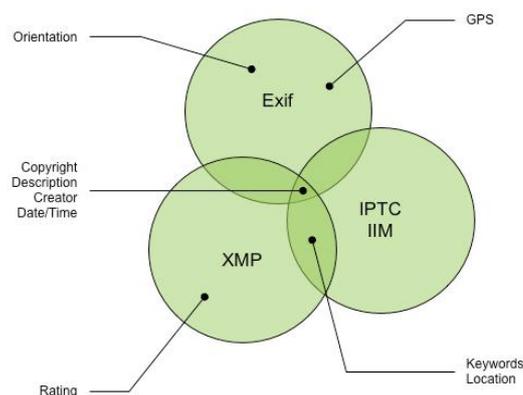


Figure 8 – Une comparaison entre standards de métadonnées pour les images (Metadata Working Group, 2009)

Ces trois standards ne sont donc pas interopérables parce qu'ils ne permettent pas l'échange aisé d'informations : d'une part, il n'existe pas de recouvrement total entre eux et d'autre part pour pouvoir échanger l'information entre ces standards une transformation avec perte d'information est nécessaire (Metadata Working Group, 2009). Cela est vrai pour tous les standards de métadonnées associés aux images.

Tableau 3 – Comparaison entre les différents éléments présents dans des standards de métadonnées

	DC	EXIF	DIG35	MIX	LOM	DICOM	FRBR	METS	ID3	MPEG-7	XMP	IPTC IIM
Auteur	X	X	X	X	X	X	X	-	X	X	X	X
Copyright	X	X	X	-	X	X	-	-	X	X	X	X
Date de création	X	X	X	X	X	X	-	X	X	X	X	X
Date de modification	-	X	X	X	-	X	-	X	-	X	X	X
Description	X	X	X	-	X	X	-	-	-	X	X	X
Événement	-	-	X	-	-	-	-	-	-	X	-	-
Genre	-	-	-	-	X	-	-	-	X	X	-	-
GPS Altitude	-	X	-	-	-	-	-	-	-	X	-	-
GPS Latitude	-	X	-	-	-	-	-	-	-	X	-	-
GPS Longitude	-	X	-	-	-	-	-	-	-	X	-	-
Langue	X	-	-	-	X	-	-	-	X	X	-	-
Mots-clés	-	-	-	-	X	-	-	-	-	X	X	X
Sujet	X	X	X	-	X	X	X	-	X	X	X	X
Titre	X	X	X	-	X	-	X	-	X	X	X	X

Nous avons réalisé une comparaison entre les différents standards de métadonnées présentés dans la Section 1.4.4 du point de vue des éléments fournis par ces standards. Un extrait de cette comparaison est présenté dans le Tableau 3. Nous avons voulu voir par exemple si le concept d'auteur est présent ou pas dans les standards, indépendamment du nom que cet élément peut avoir dans les diffé-

rents standards, que ce soit auteur, créateur ou contributeur. Cette comparaison démontre le fait qu'il y a un faible nombre d'éléments partagés entre les différents standards, ce qui rend difficile l'échange d'informations entre eux.

De plus, certaines caractéristiques peuvent être exprimées dans deux standards avec différents niveaux de détails, qui peuvent avoir de plus des structures et des types différents (Tzouvaras et al., 2007). Par exemple, dans le Dublin Core, l'auteur est spécifié en utilisant la balise creator (Tableau 5), où le nom et le prénom de l'auteur peuvent être écrits dans n'importe quel ordre. Dans MPEG-7, l'auteur est aussi spécifié en utilisant la balise creator (Tableau 4), mais dans ce cas il s'agit d'une balise composée qui différencie nom et prénom. Le DIG35 propose aussi une balise creator composée, du même type que MPEG-7 (Tableau 6). Le niveau de granularité est donc différent entre ces trois standards, ce qui complique l'échange d'informations entre eux.

Tableau 4 – Un exemple de métadonnées MPEG-7

```
<Mpeg7 xmlns="http://www.mpeg7.org/2001/MPEG-7_Schema" xml:lang="en">
  <ContentManagement xsi:type="CreationInformationType">
    <Creation>
      <Creator>
        <Role href="urn:mpeg:mpeg7:cs:RoleCS:AUTHOR"/>
        <Agent xsi:type="PersonType">
          <Name>
            <GivenName>Ana-Maria</GivenName>
            <FamilyName>Manzat</FamilyName>
          </Name>
        </Agent>
      </Creator>
    </Creation>
  </ContentManagement>
</Mpeg7>
```

Tableau 5 – Un exemple de métadonnées Dublin Core

```
<document xmlns:dc="http://purl.org/dc/elements/1.1/">
  <dc:creator>Ana-Maria Manzat</dc:creator>
</document>
```

Tableau 6 – Un exemple de métadonnées DIG35

```
<Metadata>
  <Image_creation>
    <General_creation_info>
      <Image_creator>
        <Person_name>
          <Name_comp type="given">ana-maria</Name_comp>
          <Name_comp type="family">manzat</Name_comp>
        </Person_name>
      </Image_creator>
    </General_creation_info>
  </Image_creation>
</Metadata>
```

Tout cela montre bien que les standards de métadonnées sont si hétérogènes qu'ils ne sont pas interopérables. Dans l'état de l'art beaucoup de chercheurs se sont penchés sur ce problème de l'interopérabilité de métadonnées.

2.4.3. Les solutions existantes aux problèmes d'interopérabilité et d'hétérogénéité des métadonnées

Pour résoudre l'épineux problème d'interopérabilité entre les standards de métadonnées plusieurs solutions possibles sont présentées dans (Haslhofer and Klas, 2010), (Chan and Zeng, 2006), (Zeng and Chan, 2006).

L'interopérabilité peut être envisagée à plusieurs niveaux : des schémas, des éléments et des collections. A chacun de ces niveaux, autres méthodes sont utilisées. Pour réaliser l'interopérabilité au niveau du schéma (Chan and Zeng, 2006) présentent plusieurs méthodes, parmi lesquelles nous citons :

- **La dérivation** : un nouveau schéma est obtenu en étendant un schéma existant. De cette façon on obtient une structure de base avec les éléments usuels, enrichie de composants différents en détails et en granularité.
- **Le profil d'application** : les éléments d'un ou plusieurs schémas sont combinés dans un nouveau schéma, optimisé pour une application particulière.
- **Le Crosswalks** : représente le mapping d'un schéma de métadonnées vers un autre, en réalisant une table de correspondances entre des éléments d'une source vers des éléments d'une destination. Cette correspondance est faite en tenant compte de la similarité ou de l'équivalence entre ces éléments, et elle est unidirectionnelle. Il y a deux types de crosswalk : absolue (i.e., seulement la correspondance exacte est prise en compte) et relative (i.e., tous les éléments de la source sont mis en correspondance avec au moins un élément de la destination, sans tenir compte d'une équivalence totale ou partielle entre ces éléments). Cette méthode a montré ses limites quant aux degrés que l'équivalence peut avoir : one-to-one, one-to-many, many-to-one, et one-to-none.
- **Le Switching-across** : un format de métadonnées – nouveau ou déjà existant – est utilisé comme format pivot et des correspondances sont réalisées entre les autres formats pris en compte et le pivot. Cette méthode s'utilise pour la prise en compte de plusieurs formats de métadonnées, car elle évite la réalisation de multiples correspondances entre chaque paire de standards considérés.
- **Le framework de métadonnées** : la réalisation d'un squelette, sur lequel d'autres objets peuvent être intégrés pour une certaine application ;
- **Le registre des métadonnées** : fournit des services de publication, d'enregistrement, de recherche et de gestion des schémas de métadonnées. Ce registre réalise également le crosswalk entre les différents formats de métadonnées gérés.

Une fois l'interopérabilité acquise au niveau des schémas, l'échange effectif d'informations peut être réalisé. Pour cela, l'interopérabilité au niveau des éléments de métadonnées doit être atteinte à travers une conversion des valeurs d'un schéma vers un autre, en tentant de ne perdre aucune information.

Ces stratégies peuvent être appliquées pour réaliser l'interopérabilité de tout format de métadonnées, dans n'importe quel domaine. Dans l'état de l'art, certaines approches se focalisent sur un domaine d'application bien précis ; d'autres ne considèrent que les métadonnées associées à un seul

type de média. C'est le cas de (Metadata Working Group, 2009), qui se focalise sur les métadonnées associées aux images. Ce groupe de travail propose des stratégies à adopter pour atteindre l'interopérabilité au niveau des éléments de métadonnées. Au niveau de schémas il propose d'utiliser un crosswalk entre les différents standards de métadonnées considérés (Exif, XMP, IPTC IIM).

Toujours dans le domaine des images, le projet JPSearch³⁹ a choisi une autre stratégie pour assurer d'interopérabilité des métadonnées : le switching-across, à travers la création d'un format pivot, qui est utilisé par le système pour assurer la représentation uniforme des métadonnées. Ce format est composé de vingt éléments principaux (e.g., Title, Description, Keyword, CreationDate) (Döller et al., 2010). Des règles de transformations entre les éléments des standards DIG35, DC, MPEG-7 et le format proposé sont également définies.

Le groupe de travail du W3C, Media Annotations Working Group, est allé encore plus loin que JPsearch en établissant un format pivot de métadonnées associées à tout document multimédia (Lee et al., 2012). Ce format est présenté sous la forme d'une ontologie. Pour la réalisation de ce format des tables de correspondances ont été réalisées entre ce format et chaque standard multimédia. En ce qui concerne les éléments représentés dans cette ontologie, il s'agit d'un nombre restreint d'éléments communs à la majorité des standards étudiés. Pour rendre leur travail accessible à tous, des services web ont été créés et partagés afin de transformer des éléments de métadonnées d'un standard quelconque vers leur ontologie⁴⁰.

En ce qui concerne les projets présentés dans la Section 2.3, la grande majorité de ces projets se satisfont d'utiliser des standards de métadonnées qui sont assez générales pour décrire les documents dans leur domaine d'application. Dans le cadre du projet WebLab un modèle d'échange d'information a été développé pour assurer l'interopérabilité entre les différents composants du système⁴¹. L'idée principale de ce modèle est d'assurer l'interopérabilité du système sans privilégier une description détaillée des documents multimédias gérés. L'élément central de ce modèle est la ressource : média, requête ou application.

Le projet CAM4HOME, (Amir, 2011) propose aussi un modèle de métadonnées qui regroupe plusieurs types de métadonnées nécessaires pour la création, la livraison et la consommation des contenus multimédias. Dans ce travail des règles de transformations sont définies également pour intégrer les différents formats de métadonnées dans le modèle proposé. Il se focalise sur le matching de schémas de métadonnées, en prenant en compte la structure et la sémantique des différents schémas.

Récemment l'accent a été mis sur la création d'ontologies pour la description des contenus multimédias (Brut and Sedes, 2010). Parmi les plus connues nous citons : COMM (Arndt et al., 2007), M3O (Saathoff and Scherp, 2010), AceMedia (Bloehdorn et al., 2005) et ABC (Lagoze and Hunter, 2001). Tous ces travaux se concentrent sur la transformation du standard MPEG-7 dans une ontologie, étendue ensuite à d'autres concepts et descripteurs pour offrir une description la plus appropriée possible pour n'importe quel contenu multimédia.

L'analyse de ces différentes propositions montre qu'il existe deux grandes catégories de solutions :

- Celles qui s'intéressent à un seul type de media ;
- Celles qui veulent intégrer les métadonnées de tout type de contenu multimédia.

³⁹ <https://www.dimis.fim.uni-passau.de/iris/index.php?view=jpsearch>

⁴⁰ <http://mawg.joanneum.at/web/mawg.html>

⁴¹ http://weblab-project.org/index.php?title=WebLab_data_exchange_model

En ce qui concerne les standards de métadonnées pris en compte, deux catégories sont identifiées :

- Celles qui s'attachent à assurer l'intégration du plus grand nombre de standards de métadonnées. Inconvénient : le format proposé est assez réduit en termes d'éléments ;
- Celles qui ont pour but de fournir la description la plus complète possible des documents multimédias. Inconvénient : le format proposé est trop détaillé pour être utilisé avec facilité dans des systèmes réels.

2.5. Bilan

Les documents multimédias occupent une place importante dans notre vie. Leur gestion est un domaine de recherche très actif ces dernières années.

La taille des collections multimédias augmente exponentiellement et pour contrecarrer cette hausse, l'attention des chercheurs s'est focalisée sur l'utilisation des métadonnées pour décrire ces contenus multimédias, et cela dans le cadre des systèmes distribués (Kosch, 2003). Mais la grande diversité et l'hétérogénéité des métadonnées ajoutent une difficulté supplémentaire à la gestion des contenus multimédias : quel format de métadonnées employer pour mieux décrire les contenus ? Les différents formats proposés dans l'état de l'art ne fournissent pas une solution générique et complète qui puisse décrire de manière exhaustive les contenus multimédias.

Les approches proposées pour une gestion efficace des contenus multimédias se focalisent sur plusieurs problèmes :

- L'architecture du système, du point de vue du matériel, mais aussi logiciel ;
- Le format des métadonnées à utiliser pour la description des contenus ;
- L'indexation des contenus multimédias.

Chacun de ces problèmes est traité de manière différente par les systèmes, mais une solution générique et généralement utilisée n'existe pas à ce jour. La grande majorité des projets se focalisent sur un cas d'utilisation précis et leur solution n'est applicable que dans le contexte choisi.

Les avancées dans le domaine du traitement du signal fournissent toujours plus d'algorithmes d'indexation qui extraient des informations du contenu multimédia. Mais ces algorithmes sont aussi très coûteux en termes de consommation de ressources (CPU et mémoire). La question qui se pose est de savoir quel algorithme d'indexation exécuter pour obtenir les métadonnées nécessaires et suffisantes à une recherche efficace, tout en minimisant la consommation des ressources.

Dans cette thèse, nous faisons un essai supplémentaire pour répondre à l'inconvénient de l'interopérabilité, en proposant des solutions aux deux derniers problèmes (le format des métadonnées et l'indexation des contenus multimédias), tout en prenant en compte l'utilisateur et son interaction avec le système. En ce qui concerne l'architecture du système et les standards de métadonnées à prendre en compte pour le format de métadonnées, ils nous ont été imposés par le projet LINDO, dans le cadre duquel nos travaux se sont déroulés.

Le chapitre suivant présente la manière dont l'utilisateur est pris en compte dans les systèmes de gestion des contenus multimédias.

Chapitre 3. L'utilisateur et les systèmes de gestion des documents multimédias

3.1.	<i>Introduction</i>	53
3.2.	<i>L'interaction de l'utilisateur avec un document</i>	53
3.3.	<i>L'analyse du comportement de l'utilisateur</i>	55
3.3.1.	Les intérêts de l'utilisateur	55
3.3.2.	Les intentions de l'utilisateur	57
3.4.	<i>La prise en compte de l'utilisateur dans les systèmes de gestion de documents multimédias</i>	58
3.4.1.	L'amélioration du processus de gestion des documents multimédias	58
3.4.2.	Les interfaces hommes-machine	60
3.4.3.	L'enrichissement de la requête	61
3.4.4.	Les systèmes de recommandations et de filtrage collaboratif	61
3.4.5.	L'adaptation de la présentation	62
3.5.	<i>L'utilisateur, générateur de métadonnées</i>	62
3.6.	<i>Bilan</i>	64

3.1. Introduction

L'utilisateur occupe une place centrale dans le processus de gestion des documents multimédias. Il est en interaction constante avec le système de gestion et avec les documents. L'utilisateur est de plus en plus pressé et souhaite trouver au plus vite l'information qui lui est nécessaire. Cela a un grand impact sur les systèmes, qui doivent intégrer l'utilisateur dans le processus de gestion. Celui-ci peut intervenir soit dans la partie indexation des contenus multimédias, à travers la création, implicite ou explicite, de métadonnées, soit dans la partie recherche, en modifiant la requête ou en donnant son avis sur la pertinence des résultats affichés, afin d'en obtenir de meilleurs.

Le plus souvent, la prise en compte de l'utilisateur passe par une analyse de son comportement. D'abord nous présentons plus en détails les différentes interactions qu'un utilisateur peut avoir avec un document en général. Ensuite nous présentons une analyse du comportement de l'utilisateur au moment de l'interaction avec les résultats affichés, pour une requête, par n'importe quel système de gestion de documents. Une fois cette analyse réalisée, nous nous intéressons à la façon dont l'utilisateur est réellement pris en compte dans le système de gestion de contenus multimédias, comme par exemple dans les systèmes hypermédias adaptatifs et les systèmes de recommandations. Enfin nous présentons l'utilisateur comme un créateur de métadonnées et les différentes modalités de création.

3.2. L'interaction de l'utilisateur avec un document

Comme on a pu voir dans la Section 1.4.3 (Figure 5), l'utilisateur a un rôle important dans le cycle de vie du document multimédia. Il interagit de manière continue avec le document, à chaque étape de son cycle de vie, mais surtout dans la phase de création du document, lorsque l'utilisateur recherche l'information dont il a besoin.

Il y a plusieurs façons de capturer le comportement de l'utilisateur. Une vue d'ensemble est fournie par (Kelly and Teevan, 2003). Des indicateurs peuvent être utilisés pour capturer le comportement des usagers : le nombre de clics en examinant le document, la succession de ses clics, le suivi du regard, le temps passé sur le document ou sur une partie du document.

(Kelly and Teevan, 2003) proposent une catégorisation du comportement de l'utilisateur. Parmi les catégories proposées on retrouve :

- *Examiner* le contenu pour pouvoir décider quelle partie du contenu est pertinente pour son besoin ;
- *Retenir* la partie du contenu pertinente ;
- *Référencer* le contenu à travers des citations par exemple ;
- *Annoter* le contenu en lui ajoutant des métadonnées pour pouvoir le retrouver plus vite ;
- *Créer* un contenu nouveau.

Tableau 7 – La catégorisation des actions de l'utilisateur selon (Kelly and Teevan, 2003)

		Granularité du document		
		Segment	Objet	Classe
Catégorie de comportement	Examiner	Visualiser, Écouter, Scroll, Trouver, Rechercher	Sélectionner	Naviguer
	Retenir	Imprimer	Marquer, Enregistrer, Effacer, Acheter, Email	S'abonner
	Référencer	Copier-coller Citer	Envoyer, Répondre, Lier, Citer	
	Annoter	Marquer	Publier, Noter	Organiser
	Créer	Écrire, Éditer	Créer	

Chacun de ces comportements a associé une ou plusieurs actions qui peuvent être réalisées par un utilisateur. Ces actions peuvent ne pas être effectuées sur la totalité du document, mais sur une partie seulement, ou sur des objets qui s'y trouvent. (Kelly and Teevan, 2003) considèrent que les actions peuvent être faites sur un segment, un objet ou une classe. Les différentes actions identifiées dans ce travail sont présentées dans le Tableau 7.

(McDonald et al., 2011) utilisent les activités des utilisateurs des pages de Wikipedia pour déterminer leur comportement de manière automatique. Dans ce travail, les auteurs ont identifié six types de comportement :

- Rédaction de contenu ;
- Gestion de la communauté, en relançant les discussions et les projets ;
- Contrôle du contenu, pour éviter le vandalisme et assurer le respect des droits d'auteur ;
- Actions administratives : la gestion des utilisateurs, analyse du contenu ;
- Actions collaboratives : la médiation des conflits ;
- Travail sur le méta-contenu : modification la structure et l'affichage du site ;
- Action non-déterminée.

Tous ces travaux se focalisent sur l'interaction de l'utilisateur avec des documents textuels. Au moment où on parle de documents multimédias, où le paramètre temps intervient, tout change. (Schoeffmann et al., 2010) présentent un état de l'art des systèmes de gestion des vidéos. Cette étude se focalise surtout sur les interfaces de ces systèmes et sur les interactions que l'utilisateur peut réali-

ser avec le document. Cette vue d'ensemble des systèmes existants a permis à (Haslhofer and Klas, 2010) d'identifier six types d'interactions de l'utilisateur avec la vidéo qui peuvent être exploitées dans le cadre du retour de pertinence implicite :

- *L'aperçu* : équivalent au passage de la souris sur une key frame pour obtenir de l'information supplémentaire sur la vidéo ;
- *Le clic sur un résultat* : pour visualiser le contenu ou pour effectuer d'autres actions sur le contenu ;
- *Le déroulement* : utiliser la barre de navigation pour dérouler la vidéo. Les utilisateurs usent de cette fonction jusqu'à ce qu'ils atteignent l'information souhaitée ;
- *L'exploration* des métadonnées associées au frame qui est affiché. Cette action montre l'intérêt de l'utilisateur à préserver, afin de décider de la pertinence du contenu ;
- *La navigation* : d'un key frame à un autre, en cliquant sur les frames qui sont semblables au frame courant ;
- *La visualisation* : complète de la vidéo, qui indique l'intérêt de l'utilisateur au contenu.

Le plus souvent les actions de l'utilisateur sont déterminées en analysant les clics qu'il réalise tout au long de son interaction avec le document.

3.3. L'analyse du comportement de l'utilisateur

3.3.1. Les intérêts de l'utilisateur

Un caractère important de l'intérêt est la manière dont il est obtenu. (Claypool et al., 2001) proposent de les considérer sous la dimension implicite/explicite (Figure 9), par rapport au moment quand l'utilisateur a réalisé l'action qui a permis de déterminer l'intérêt, mais aussi par rapport à la spécification directe de l'intérêt. L'intérêt explicite est spécifié directement par l'utilisateur à travers des notes ou des avis. L'intérêt implicite est déterminé, sans aucune action de la part de l'utilisateur, à travers une analyse des informations obtenues de manière non intrusive. En ce qui concerne les intérêts mixtes, l'utilisateur ne les spécifie pas de manière directe, mais il transmet des informations au système qui en les analysant peut déterminer les intérêts (e.g., des mots-clés).

Dans le domaine des intérêts implicites, le problème du choix des actions de l'utilisateur à exploiter pour obtenir ces intérêts se pose. Les différentes interactions qu'un utilisateur peut avoir avec un document, dans le contexte du web, ont été structurées en sept catégories (Claypool et al., 2001) :

- Actions *explicites* : la sélection directe d'un intérêt ;
- Actions de *marquage* : la création ou l'effacement d'un marque-page, la sauvegarde d'une page web, l'impression de la page;
- Actions de *manipulation* : le copier-coller, l'ouverture d'une nouvelle fenêtre de l'explorateur, la recherche d'un mot dans la page, le défilement ;
- Actions de *navigation* : le temps que l'utilisateur passe sur la page, suivre ou pas un lien ;
- Actions *externes* : réactions physiques de l'utilisateur qui pourraient indiquer l'intérêt de celui-ci (e.g., le regard, la mimique et les réactions de l'utilisateur, la fixation, la dilatation de la pupille, « saccades » et les trajectoires (Rayner, 1998)) ;
- Actions *répétitives* : consultation répétée de la même page, un grand temps de lecture ;
- Actions *négatives* : l'absence d'action peut être interprétée comme une action négative.

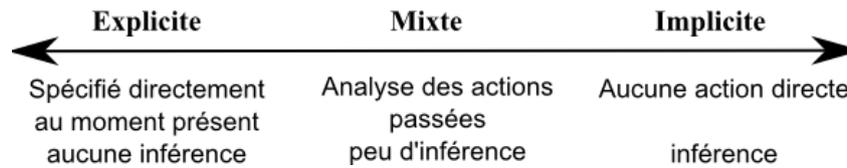


Figure 9 – La dimension implicite/explicite de l'intérêt d'un utilisateur

Les données les plus faciles à obtenir, en termes de coût d'acquisition et qui sont le moins intrusives, sont les clics. C'est essentiellement le cas des systèmes de gestion de documents textuels. Mais dans leurs cas, on doit faire attention comment les clics sont utilisés. (Joachims and Radlinski, 2007; Joachims et al., 2007) montrent que l'affichage des résultats introduit un biais de présentation et que l'utilisateur est influencé par le fait qu'il sait que les documents qui sont affichés au début de la liste des résultats sont considérés comme les plus pertinents, par le moteur de recherche. Par conséquent, l'utilisateur ne regarde pas tous les résultats avant de choisir de visualiser un document. Il est également influencé dans sa décision de cliquer sur un certain document, par la qualité générale de la liste des résultats affichée. Un autre constat qui doit être pris en compte au moment de l'analyse des clics de l'utilisateur est le fait que les utilisateurs parcourent de manière séquentielle la liste des résultats, du haut vers le bas et qu'ils ont tendance à regarder d'avantage les liens des résultats au-dessus du lien cliqué et pas ceux qui se trouvent plus loin dans la liste des résultats. Il en résulte qu'au moment où un utilisateur clique sur un document il n'a pas vu les autres documents suivants affichés dans la liste des résultats (Joachims et al., 2007).

Des travaux importants ont été menés dans l'analyse du comportement de l'utilisateur au moment de la recherche d'information. Cette analyse est réalisée surtout dans le domaine de la recherche sur le web, et elle se base sur tout ce qui peut être sauvegardé au moment de l'exécution d'une requête (e.g., le texte de la requête, le temps pendant elle a été saisie, les résultats obtenus, les clics que l'utilisateur a réalisés sur la liste des résultats) (Baeza-Yates and Navarro, 2011). Les études menées sur ces logs ont démontré que la dimension moyenne des requêtes que les utilisateurs réalisent est de trois mots, alors que la plus longue requête est composée de vingt-cinq mots (Jansen et al., 2007), (Skobeltsyn et al., 2008). En outre, les requêtes sont de plus en plus complexes. (Broder, 2002) a proposé une classification des requêtes des utilisateurs en :

- *requêtes de navigation* : recherche d'un site précis ;
- *requêtes informationnelles* : recherche d'informations supplémentaires sous forme de plusieurs pages web ;
- *requêtes transactionnelles* : par exemple, faire une réservation.

(Rose and Levinson, 2004) ont modifié cette taxonomie en insistant sur les requêtes informationnelles. Par exemple dans leur taxonomie on retrouve des *requêtes informationnelles directes* (apprendre quelque chose de précis sur un sujet), ou *indirectes* (apprendre quelque chose sur un sujet). En analysant les requêtes de l'utilisateur on peut obtenir d'un côté ses intentions et de l'autre ses intérêts. Pour l'extraction de ces intérêts des travaux récents se sont focalisés sur l'utilisation de la sémantique des mots des requêtes (Limam et al., 2010).

Pour les documents multimédias, qui évoluent dans le temps, en plus des clics de l'utilisateur, le temps que l'utilisateur passe sur le document est très important pour déterminer son intérêt à une partie du document par rapport à une autre partie. Cet indicateur est étudié depuis la fin des années 60, pour déterminer la corrélation entre le temps passé par l'utilisateur pour visualiser un document et l'intérêt de l'utilisateur pour ce document. Les résultats obtenus par (Faw and Nunnally, 1967) montrent qu'il y a une corrélation positive entre le temps de visualisation d'un document et leur

« caractère agréable ». (Day, 1966) a réalisé une étude sur le temps de visualisation des images qui démontre que de manière générale l'utilisateur a tendance à regarder d'avantage les images qu'il aime que les images qu'il n'aime pas. Cette étude a été généralisée par (Oostendorp and Berlyne, 1978) pour confirmer qu'il regarde plus longtemps les objets qui l'intéressent.

D'autres indicateurs peuvent être utilisés pour déduire l'intérêt de l'utilisateur pour un document : les actions réalisées au moment de la création d'un document. (Morita and Shinoda, 1994) ont étudié le comportement des utilisateurs qui lisent des articles de journaux, en se focalisant sur leur copie et leur sauvegarde, mais également sur le temps passé pour les lire. Ils ont montré que le fait qu'un utilisateur copie ou sauvegarde un document ne signifie pas forcément que celui-ci l'intéresse. Les résultats les plus marquants de cette étude sont liés au temps. L'utilisateur passe plus de temps à lire un document qui l'intéresse qu'un document qui ne l'intéresse pas. En outre, ce temps n'est pas influencé par la dimension du document ou par sa lisibilité, parce que l'utilisateur ne lit pas tout le document pour décider que celui-ci ne l'intéresse pas. Cette étude a été généralisée par (Claypool et al., 2001), en l'appliquant au domaine du web. Ils démontrent également que le temps passé sur une page est un bon indicateur de l'intérêt de l'utilisateur pour la page. Cette étude a également montré que les quantités de défilement et de clics sont aussi de bons indicateurs de l'intérêt des utilisateurs pour le document. Toutefois ces deux actions prises de manière individuelle (pour chaque utilisateur) ne sont pas pertinentes pour déterminer l'intérêt de l'utilisateur. (Hijikata, 2004) démontre que le suivi des mouvements de la souris que l'utilisateur réalise en consultant un document peut être utilisé pour déterminer son intérêt pour une partie du document plutôt qu'une autre.

Des études ont été réalisées pour savoir si le comportement de navigation est un bon indicateur pour l'intérêt de l'utilisateur. En employant cet indicateur, en le calculant sur une courte période de temps, dans le cadre d'un système de recherche d'informations. (Seo and Zhang, 2000) ont démontré que les performances du système s'en trouvent améliorées.

Un des nouveaux indicateurs d'intérêt qui a été étudié est le suivi du regard et de tous les mouvements des yeux pendant la lecture d'un document. Dans une étude réalisée sur un groupe de six utilisateurs, qui devaient lire des documents textuels pour identifier les parties pertinentes. (Loboda et al., 2011) ont démontré que le mouvement des yeux peut être employé pour identifier les phrases pertinentes d'un document textuel. Les données collectées leur ont permis d'identifier trois mesures qui permettent de réaliser cette sélection : le nombre de fixations, le nombre de fixations initiales et le temps total de visualisation pour une phrase.

Mais l'intérêt de l'utilisateur n'est pas figé dans le temps. Un utilisateur peut n'être intéressé que par la politique que pendant une période des élections donnée ; une fois les élections passées, la politique ne représente plus d'intérêt particulier pour lui. (Liu et al., 2010) se sont focalisés sur l'évolution dans le temps des intérêts d'un utilisateur de la plateforme Google News⁴². Les auteurs utilisent l'historique des clics que l'utilisateur réalise au fil de temps pour déterminer l'évolution de ses intérêts pour certaines catégories de sujets (e.g., sport, politique, divertissement, santé, nature). Les intérêts ainsi déduits sont utilisés pour donner des recommandations à l'utilisateur.

3.3.2. Les intentions de l'utilisateur

Les intentions des utilisateurs font partie des éléments importants à prendre en compte au moment où le comportement de celui-ci est interprété. (Lux et al., 2010) s'intéressent à la classifica-

⁴² <https://news.google.com/>

tion de l'intention de l'utilisateur dans le cadre d'un système de gestion d'images. Pour déterminer ces intentions, les auteurs analysent les logs du système en utilisant le nombre de photos regardées et la durée de temps nécessaires pour résoudre une tâche. (Lagger et al., 2012) ont réalisé une enquête sur plus de 200 utilisateurs pour déterminer quelles sont les intentions des utilisateurs au moment où ils cherchent une vidéo, mais aussi ce qui les détermine à regarder une vidéo plutôt qu'une autre. De manière générale, les utilisateurs recherchent des vidéos pour :

- le *divertissement* : au moment où ils ont du temps libre et ils s'ennuient ;
- *retrouver des informations* : sur des produits, sur un endroit qu'ils veulent visiter ;
- *des buts éducatifs* : tutoriels, présentations, cours de maths.

Les résultats de ce travail montrent que les intentions des utilisateurs sont bien différentes :

- pour les loisirs ;
- pour écouter de la musique ;
- pour apprendre quelque chose de nouveau ;
- pour retrouver des informations sur un certain sujet ;
- pour voir comment quelque chose fonctionne ;
- à cause de l'ennui ;
- pour voir à quoi un endroit ou un produit ressemble ;
- pour retrouver des avis sur un produit ;
- pour voir des programmes TV manqués.

Un élément important qui influence l'utilisateur lorsqu'il décide de regarder une vidéo est la qualité de celle-ci et la vitesse de téléchargement : l'utilisateur regarde surtout les vidéos de bonne qualité, qui se chargent vite et qui peuvent être regardées sans saccades. Mais en fonction de l'intention de l'utilisateur, il peut surpasser la contrainte de qualité (e.g., s'il veut apprendre quelque chose alors la vidéo doit être de très bonne qualité, mais s'il ne veut qu'écouter de la musique alors la qualité de la vidéo n'est pas très importante).

L'apparition des réseaux sociaux et leur utilisation intensive font émerger de nouveaux types d'interrelations à étudier et à utiliser. (Tchunte et al., 2011) utilisent le flux d'activités d'un utilisateur d'un réseau social, et de ses amis pour déterminer ses intérêts.

Nous allons voir comment cette analyse du comportement de l'utilisateur est utilisée dans l'état de l'art pour améliorer les performances des systèmes de gestion de contenus multimédias.

3.4. La prise en compte de l'utilisateur dans les systèmes de gestion de documents multimédias

Les résultats obtenus par (Morita and Shinoda, 1994; Claypool et al., 2001; Joachims et al., 2007; Hopfgartner, 2010; Teevan et al., 2010) démontrent que la prise en compte de l'utilisateur et de son comportement dans un système de gestion de documents améliore les performances du système. L'utilisateur est intégré à travers le retour de pertinence, à plusieurs niveaux que nous allons présenter par la suite.

3.4.1. L'amélioration du processus de gestion des documents multimédias

Cette amélioration du processus de gestion de documents multimédias peut se faire à plusieurs niveaux, à partir de l'interaction directe ou indirecte de l'utilisateur. La Figure 10 présente les

différents points où l'on peut intervenir dans le processus de gestion des documents multimédias, pour l'améliorer.

En effet, l'utilisateur est de plus en plus pris en compte dans le processus de recherche pour améliorer la liste des résultats (Manning et al., 2008). Cette prise en compte de l'utilisateur se fait dans le cadre du *Retour de Pertinence (RP)*. L'idée centrale du RP est de capturer, de manière itérative, que ce soit de manière implicite ou explicite, le jugement de pertinence de la part de l'utilisateur pour une sous-collection des résultats obtenus pour sa requête. Ce jugement est exploité pour déterminer une meilleure représentation du besoin de l'utilisateur, afin de calculer et d'afficher une nouvelle liste affinée de résultats, ou d'enrichir les représentations des documents.

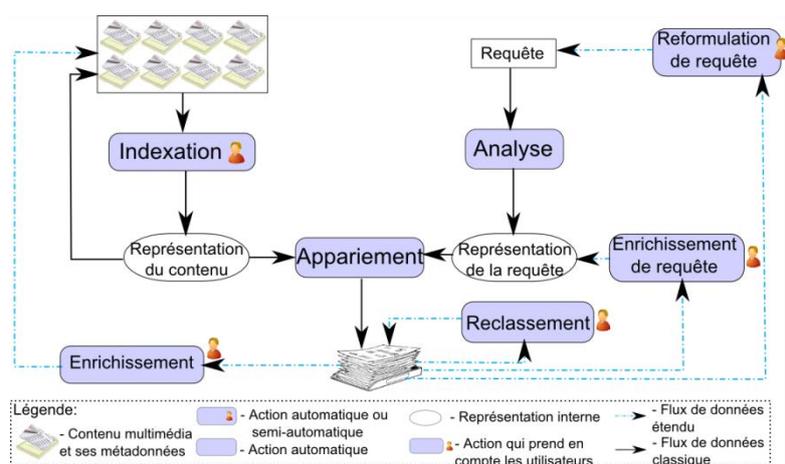


Figure 10 – Le processus étendu de gestion des documents multimédias

L'état de l'art contient plusieurs types de RP, selon le retour de pertinence obtenu : le *RP explicite* (on demande à l'utilisateur de donner son avis sur la pertinence des documents, de manière explicite) et le *RP implicite* (soit la requête initiale de l'utilisateur est exécutée et les premiers documents pertinents trouvés par le système sont utilisés pour l'enrichir, à travers une analyse locale ou globale de résultats, soit le comportement de l'utilisateur au moment de son interaction avec le système et les documents) est interprété (Baeza-Yates and Ribeiro-Neto, 2011). Selon (Nichols, 1998), les informations rassemblées à travers le RP implicite ne sont pas assez précises que celles obtenues à travers le RP explicite. Mais d'un autre côté (Agichtein et al., 2006) démontrent que les performances d'un système qui utilise le RP implicite sont améliorées de 31% par rapport à un système qui n'utilise aucun type de RP.

Les résultats obtenus par le RP peuvent être utilisés de plusieurs façons dans la gestion des contenus multimédias : le reclassement des résultats, l'enrichissement de la requête, la reformulation de la requête et l'enrichissement des métadonnées associées aux contenus multimédias. La Figure 10 illustre les différentes manières dont le RP peut être réalisé. Le reclassement des résultats est une action automatique du système, qui peut prendre en compte la pertinence de ceux-ci par rapport à une requête antérieure, calculée de manière explicite ou implicite. L'enrichissement de la requête peut se faire à travers l'ajout de nouveaux termes à la requête initiale de l'utilisateur ou à travers le changement du poids de chaque terme de la requête initiale. La reformulation de la requête est une opération qui peut être réalisée directement soit par l'utilisateur, soit par le système en modifiant les termes de la requête par des synonymes, par exemple. Une autre manière d'utiliser le RP est d'enrichir les repré-

sentations des documents à travers l'ajout de relations et de nouvelles informations qui peuvent être déduites à partir des requêtes des utilisateurs et de leur retour de pertinence.

Dans le cadre de la recherche d'informations, les seules données sur l'activité de l'utilisateur qui sont exploitées dans le cadre du retour de pertinence sont : les requêtes, les résultats obtenus pour ces requêtes et les documents que l'utilisateur a consultés. L'idée principale est d'attribuer une importance plus grande aux documents été cliqués par des utilisateurs semblables, pour des requêtes similaires.

Mise à part ces interactions, dans l'état de l'art des travaux prennent en compte les clics des utilisateurs et les interprètent comme des valeurs absolues (e.g., si un document est cliqué alors il est pertinent pour la requête) (Kemp and Ramamohanarao, 2002; Fox et al., 2005). Mais cette interprétation souffre du biais introduit par la présentation des résultats. Pour contourner ce biais, (Joachims et al., 2007) proposent de considérer les clics des utilisateurs comme des jugements relatifs entre les documents qui ont été évalués, cliqués ou pas.

3.4.2. Les interfaces homme-machine

Les interactions de l'utilisateur avec un système sont étudiées et exploitées surtout dans le domaine des interfaces homme-machine (Sears and Jacko, 2007). Dans ce domaine les actions de l'utilisateur suivent un processus de traitement particulier (Figure 11).

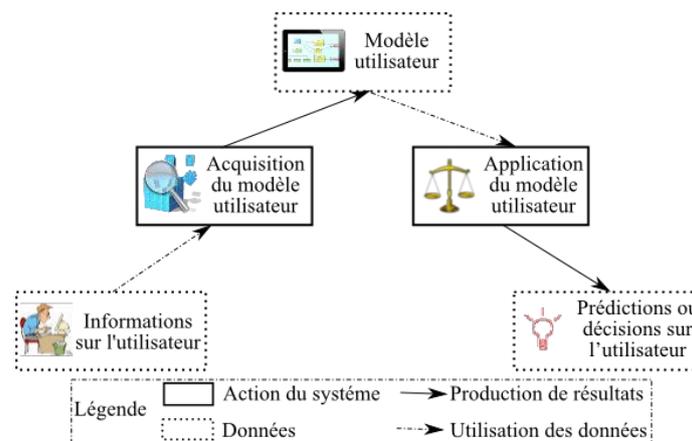


Figure 11 – Le schéma général du traitement de l'information dans un système adaptatif

Le système acquiert de l'information sur l'utilisateur en utilisant un ou plusieurs des indicateurs présentés dans la section 3.3. Ces données sont analysées pour modéliser de l'utilisateur en fonction du domaine d'application et des besoins identifiés. Une fois ce modèle défini, il est appliqué de manière récurrente pour réaliser des prévisions, sur ce que l'utilisateur pourrait aimer, ou prendre des décisions, sur la présentation d'un résultat.

Une vue d'ensemble sur les différentes manières dont l'utilisateur peut être intégré dans un système est présentée par (Jameson, 2008), tout en insistant sur l'interface homme-machine fournit dans chaque situation et comment elle s'adapte à l'utilisateur et à la tâche que celui-là effectue.

3.4.3. L'enrichissement de la requête

Hormis les interfaces homme-machine, la recherche d'informations constitue un des premiers domaines qui prend en compte l'utilisateur, à travers l'enrichissement de la requête (Rocchio, 1971). Depuis, beaucoup de travaux ont abordé ce problème en se focalisant sur l'utilisation des ontologies pour ajouter de nouveaux concepts à la requête initiale de l'utilisateur. (Järvelin et al., 2001) démontrent que l'enrichissement de la requête en y ajoutant des concepts, améliore les performances du système. Ce processus d'enrichissement peut être accompli automatiquement, en analysant le comportement de l'utilisateur et l'historique de ses requêtes, ou de manière interactive, en demandant explicitement à l'utilisateur d'ajouter lui-même des termes à la requête. Selon (Ruthven, 2003), les termes obtenus de manière automatique sont plus utiles que ceux obtenus à travers un processus interactif parce qu'ils apportent de l'information nouvelle à la requête de l'utilisateur. Un état de l'art dans ce domaine est présenté dans (Bhogal et al., 2007).

Dans le cadre d'un système de gestion des documents semi-structurés, (Zayani et al., 2006) proposent d'enrichir les requêtes XQuery de l'utilisateur avec des éléments de son profil mis à jour en permanence en fonction des requêtes qu'il exécute.

3.4.4. Les systèmes de recommandations et de filtrage collaboratif

Un autre domaine important où l'utilisateur est pris en compte est celui des systèmes de recommandations, qu'ils soient basés sur le contenu ou collaboratifs.

Dans le cadre des systèmes de recommandations basés sur le contenu, les documents qu'un utilisateur a consulté sont analysés et sont comparés avec d'autres pour déterminer la pertinence des ces derniers pour l'utilisateur. Les systèmes de recommandations basés sur le contenu s'appuient sur le constat suivant : si un utilisateur consulte constamment des documents qui abordent un certain sujet, il voudra en consulter encore d'autres documents qui parlent du même sujet toujours d'avantage. L'activité de l'utilisateur est analysée pour établir les intérêts de celui-là. Ces intérêts sont stockés dans le profil de l'utilisateur pour être utilisés dans la recommandation. Les études réalisées dans ce domaine, montrent que l'utilisation de la recommandation basée sur l'analyse du comportement de l'utilisateur améliore les performances du système (Candillier et al., 2009).

Les systèmes de filtrage collaboratif sont des systèmes de recommandations où les avis des utilisateurs semblables sont utilisés pour faire des recommandations. Plus précisément, le système recommande à un utilisateur des contenus consultés par des utilisateurs qui ont des intérêts similaires à lui. Dans ce type de système, le contenu n'est pas analysé. Le but principal de ces systèmes est de déterminer des groupes d'utilisateurs qui ont des intérêts communs. L'idée centrale est que si deux utilisateurs ont aimé systématiquement les mêmes documents, alors il est fort probable qu'ils aiment encore dans le futur le même type de document. La performance des systèmes de filtrage collaboratif est influencée par la précision et la qualité de l'analyse du comportement de l'utilisateur afin de déterminer ses intérêts. Une étude comparative entre les différents systèmes de filtrage collaboratif est présentée par (Schafer et al., 2007).

Dans le cadre du système Google News, (Liu et al., 2010) combinent les intérêts actuels de l'utilisateur et les principes du filtrage collaboratif classique pour améliorer le processus de recommandation. Ils utilisent la distribution des clics réalisés par un utilisateur pendant la période antérieure pour déduire l'intérêt présent de l'utilisateur, en utilisant des réseaux Bayesian. Les résultats obtenus en combinant ces deux techniques montrent une nette amélioration de la recommandation.

(Yang et al., 2007) utilisent le temps de visualisation des vidéos comment un indicateur de l'intérêt de l'utilisateur dans le cadre d'un système de recommandation des vidéos. Plus précisément, ils interprètent un temps court (moins de 15 secondes) de visualisation d'un contenu comme étant un indicateur du désintérêt de l'utilisateur, et un temps plus long comme étant un indicateur d'un intérêt accru envers le document. L'utilisation de cet indicateur montre une amélioration du système.

3.4.5. L'adaptation de la présentation

Un autre domaine où l'utilisateur a une place importante est celui de l'hypermédia adaptatif (Brusilovsky, 2001). L'élément essentiel d'un système hypermédia adaptatif est constitué par le modèle de l'utilisateur. Ce modèle est composé par les intérêts, les préférences et les connaissances de l'utilisateur. Ce modèle est utilisé et mis à jour tout au long de l'interaction de l'utilisateur avec le système. Le principal but des systèmes hypermédiés adaptatifs est la constante adaptation des contenus aux besoins de l'utilisateur.

L'adaptation de la présentation des documents multimédias peut se faire en fonction du contexte de l'utilisateur ou en fonction de ses intérêts. (Laborie and Euzenat, 2008) propose un algorithme d'adaptation qui prend en compte le profil de l'utilisateur (les auteurs s'intéressent surtout au dispositif d'affichage de l'utilisateur), mais aussi ses interactions avec le document.

(Pleşca et al., 2008) vont encore plus loin dans la prise en compte de l'utilisateur dans le processus d'adaptation du contenu. Ils intègrent dans le processus de prise de décision sur l'adaptation à réaliser, le degré d'intérêt de l'utilisateur pour le document visualisé, des paramètres de réseau (e.g., la bande passante) et ses préférences. Dans ce travail l'intérêt de l'utilisateur est calculé en fonction de la suite des interactions qu'il a avec le document.

(Kofler and Lux, 2009) réalisent une adaptation dynamique de la présentation des images obtenues comme résultats à une requête dans le cadre d'un système de recherche d'images en fonction de l'intention de l'utilisateur. Par exemple, pour des requêtes de navigation, les résultats sont affichés sous forme de vignettes et sont classés par intérêt, mais pour les requêtes informationnelles directes, les résultats sont affichés sous forme de liste et classés selon leur pertinence.

Mais le plus souvent l'utilisateur est considéré comme une source de métadonnées. Nous allons voir comment il en génère, de manière directe ou indirecte.

3.5. L'utilisateur, générateur de métadonnées

Le fait que les métadonnées se trouvent au milieu du cycle de vie du document multimédia (Smith and Schirling, 2006; Hardman et al., 2008), et qu'elles soient consommées pendant toutes les étapes du cycle de vie du document (Kosch et al., 2005) mène à une constante interaction, directe ou indirecte, de l'utilisateur avec les métadonnées. Ainsi, l'utilisateur peut être considéré comme une source auxiliaire de métadonnées. A travers ses actions et son comportement, il génère des métadonnées d'une manière implicite ou explicite.

L'utilisateur crée des métadonnées explicites en attachant des annotations et des tags (Kahan et al., 2002) aux documents multimédias. Dans l'état de l'art beaucoup de systèmes (e.g., Dailymotion, Youtube, Panoramio) comptent sur les utilisateurs pour ajouter des descriptions textuelles (i.e., des métadonnées) aux contenus audiovisuels gérés. De cette manière, ils évitent l'utilisation des algorithmes d'indexation qui sont coûteux en termes de consommation de ressources. Mais les métadonnées créées directement par les utilisateurs ne sont pas toujours fiables à 100%. En effet, les utilis-

teurs peuvent avoir des visions différentes sur le contenu et le vocabulaire utilisé. Cela engendre un problème de synonymie, de polysémie et d'homonymie. De plus, (Halvey and Keane, 2007) ont démontré que cette tâche est fastidieuse pour les utilisateurs et qu'ils sont réticents à annoter des contenus s'ils n'ont rien à gagner. Pour surpasser cette difficulté dans la réalisation des annotations (van Zwol et al., 2008) l'ont transposé dans un jeu en ligne, où les utilisateurs n'ont plus l'impression de travailler mais de s'amuser.

Les travaux réalisés sur l'analyse du comportement de l'utilisateur montrent qu'il peut être utilisé comme un indicateur de son intérêt pour un document. Donc, le comportement de l'utilisateur au moment de la consommation du document peut être utilisé pour déduire d'autres informations sur le document. De cette façon l'utilisateur crée des métadonnées implicites (Oard and Kim, 1998).

En plus des métadonnées implicites et explicites il existe aussi dans l'état de l'art les notions de métadonnées d'*attention* (Mommel and Dengel, 2007; Wolpers et al., 2007) et d'*usage* (Brunie et al., 2002). Les métadonnées d'attention sont associées à l'utilisateur, alors que les métadonnées d'usage sont liées aux contenus multimédias. Ces deux types de métadonnées sont déduits du comportement de l'utilisateur et de ses interactions avec les documents multimédias et leurs métadonnées.

Dans la vision de (Wolpers et al., 2007), les métadonnées d'attention représentent toutes les actions d'interaction de l'utilisateur avec un document. Un exemple de métadonnées d'attention fourni par (Wolpers et al., 2007) est présenté dans le Tableau 8. Ces métadonnées sont associées à un utilisateur et elles contiennent toutes les actions que l'utilisateur réalise pour créer un document qui a comme titre « J_Najjar: contextualized attention metadata streams ». Ces actions sont groupées en fonction du logiciel avec lequel il interagit. Pour chaque logiciel il y a une balise feed dans les métadonnées créées. Dans l'exemple nous avons un feed pour MS PowerPoint et un pour Firefox. Les différentes actions sont regroupées dans le cadre du feed en fonction de la partie du document concernée.

Par exemple, l'utilisateur cherche « java overering » dans le transparent qui a le titre « java ». Ces métadonnées d'attention peuvent être utilisées pour inférer le comportement de l'utilisateur dans un certain contexte (e.g., quand l'utilisateur travaille avec Microsoft PowerPoint, alors il écoute de la musique classique, mais quand il surfe sur le Web il écoute du Rock. L'utilisateur a consulté les sites A et B dans 80% des cas quand il a démarré une nouvelle présentation PowerPoint).

(Sun et al., 2005) analysent toutes les traces laissées par l'utilisateur pour déterminer des liens sous-jacents entre les utilisateurs, les requêtes et les documents. (Dou et al., 2007) vont encore plus loin dans cette analyse de l'historique de l'activité de l'utilisateur en rajoutant des liens entre différents sujets et les utilisateurs d'un côté, et entre les mêmes sujets et les documents de l'autre côté.

Tableau 8 – Un exemple de métadonnées d'attention proposé par (Wolpers et al., 2007)

```

<group>
  <title> J_Najjar: contextualized attention metadata streams</title>
  <feed>
    <title>PowerPoint</title>
    <item>
      <title>Java</title>
      <guid>http://ariadne...../ALOCOM/getLO.jsp?=alocom5.....</guid>
      <events>
        <event>
          <action>

```

```

    <actionType>search</actionType>
    <relatedData>
      <entry>
        <name>querytext</name>
        <content>java overering text fragment</content>
      </entry>
    </relatedData>
  </action>
  <dateTime>2006-12-22T09:54:34</dateTime>
  <session>.....</session>
</event>
</events>
</item>
</feed>
<feed>
  <title>Firefox</title>
  <item><title>Causing Trouble with Buttons</title><guid>..</guid>.... </item>
</feed>
</group>

```

Pour (Pedersen and McDonald, 2008) les activités d'un utilisateur sont utilisées pour déterminer des liens entre les documents qu'il utilise en même temps. Ces auteurs s'intéressent surtout aux systèmes fermés. En terme d'interactions de l'utilisateur prises en compte on retrouve : l'ouverture simultanée de deux ou plusieurs documents, l'alternance entre la consultation des documents, les actions de copier/coller entre différents documents. Des liens symétriques sont construits entre des paires de documents et ils ont une valeur entre 1 et 10. Plus grande est la valeur du lien, plus forte est le lien entre les documents. Chaque fois qu'un document est ouvert, un lien de valeur 1 est créé entre ce document et tous les autres documents ouverts. La valeur de chaque lien est mise à jour en fonction des actions qui sont accomplies par l'utilisateur sur les deux documents dans un intervalle de temps. Cette approche permet de retrouver des documents qui ont été pertinents à un moment donné pour l'utilisateur dans la réalisation d'une tâche.

Dans le cadre d'un système fermé de gestion de documents personnels, (Gyllstrom, 2009) propose de prendre en compte l'historique des actions et des interactions de l'utilisateur avec son ordinateur pour déterminer des relations entre les documents et applications utilisées dans un certain laps de temps. Les actions considérées vont de la création et de l'exécution du processus machine, jusqu'au suivi du regard. Ces relations entre les documents peuvent être vues comme des métadonnées implicites, même si les auteurs ne les considèrent pas comme telles.

3.6. Bilan

A travers l'étude présentée dans ce chapitre nous pouvons constater que l'utilisateur est, à juste titre, de plus en plus pris en compte, de manière implicite ou explicite, par les systèmes de gestion de documents, à plusieurs niveaux :

- Enrichissement de requête ;
- Reformulation de requête ;
- Enrichissement des métadonnées ;
- Adaptation de la présentation ;

- Recommandation et filtrage des résultats ;
- Reclassement des résultats.

Au début, les systèmes demandaient à l'utilisateur explicitement son intérêt pour les documents. Au fil du temps les systèmes de gestion de documents sont devenus de moins en moins intrusifs. Pour obtenir cette non-intrusion, toutes les actions de l'utilisateur sont stockées et analysées afin de déduire ses intérêts. Les indicateurs de l'intérêt de l'utilisateur qui peuvent être utilisés sont :

- Les clics ;
- Les requêtes ;
- Le temps passé en analysant un document ;
- Le déroulement d'une page ;
- Les actions d'édition (e.g., le copier/coller, la sauvegarde).

De plus en plus de travaux commencent à prendre en compte l'utilisateur comme source de métadonnées. En effet, l'utilisateur ajoute des informations supplémentaires sur un contenu multimédia, à travers ses tags par exemple. Mais cette spécification directe des métadonnées est très consommatrice de temps et elle devient très fastidieuse pour l'utilisateur. Beaucoup de travaux se focalisent donc sur l'exploitation des différents indicateurs (surtout les clics) de l'intérêt de l'utilisateur pour déduire, de manière indirecte, son intérêt pour un document.

Tous les travaux étudiés nous montrent que du comportement de l'utilisateur au moment de l'interaction avec des contenus multimédias peut être déduit de l'information supplémentaire sur les contenus.

Partie 2 : Les Contributions

Introduction

L'étude de l'état de l'art présentée dans la partie précédente, nous permet de constater que les métadonnées, utilisées par le système ou par l'utilisateur à chaque étape du processus de gestion des contenus multimédias, constituent l'élément central de ce processus. Dans la gestion des métadonnées nous pouvons identifier des insuffisances qui touchent leur modélisation, leur création et leur enrichissement.

1. *La modélisation des métadonnées :*

Il existe beaucoup de standards de métadonnées définis pour chaque type de média qui ne sont pas interopérables. Ces standards sont soit trop limités en terme de capacité de description des contenus (ils contiennent peu d'éléments ou seulement des éléments de bas niveau), soit trop vastes (ils contiennent beaucoup d'éléments et ils offrent la possibilité de décrire la même chose de plusieurs manières). Ils sont donc soit insuffisants, soit trop difficiles à employer.

Les travaux qui adressent le problème de l'interopérabilité des métadonnées se focalisent surtout sur le côté sémantique des métadonnées en tentant de créer une ontologie qui décrit le plus complètement possible un document. Or, la majorité des algorithmes d'indexation qui existent ont une sortie encodée en XML et nécessitant donc une transformation pour pouvoir utiliser ces ontologies. De plus, on peut noter que les technologies du web sémantique ont du mal à s'imposer dans le monde industriel.

2. *La création des métadonnées :*

Il y a de plus en plus d'algorithmes d'indexation qui traitent le contenu multimédia pour en extraire des informations (i.e., des métadonnées). Ces algorithmes sont de plus en plus performants mais aussi de plus en plus consommateurs de ressources.

Les travaux présentés dans l'état de l'art utilisent tous les algorithmes d'indexation dont ils disposent sur toute la collection de documents multimédias, pour extraire des informations. D'un côté, cette utilisation emploie beaucoup de ressources, en créant peut-être une surcharge du système, et d'un autre côté, beaucoup de métadonnées sont créées et certaines d'entre elles peuvent ne jamais être utilisées dans le processus de recherche.

En outre, pour les systèmes qui indexent les contenus multimédias en temps réel, les algorithmes utilisés pour indexer les contenus multimédias ne changent pas pendant le fonctionnement du système, et donc les systèmes ne s'adaptent pas au besoin de l'utilisateur. Dans le cadre des systèmes qui ne sont pas en temps réel, à tout moment de nouveaux algorithmes d'indexation peuvent être appliqués, mais de manière générale, toute la collection de documents est indexée par les mêmes algorithmes. Dans les deux cas, la création de métadonnées n'est pas optimale : dans le premier cas, des métadonnées qui intéressent l'utilisateur ne sont pas créées, et dans le deuxième cas trop de métadonnées sont créées.

3. *L'enrichissement des métadonnées :*

Les travaux de modélisation des métadonnées ne prennent pas en compte la notion d'usage. Dans ce cas, les métadonnées sont extraites à partir du contenu multimédia qu'elles décrivent, ou bien elles sont associées directement par l'utilisateur à travers des tags et des annotations.

Comme nous avons vu dans le Chapitre 3, le comportement de l'utilisateur est analysé afin de déterminer quels sont les meilleurs indicateurs de son intérêt envers un document multimédia avec lequel il interagit. Parmi ces indicateurs, les clics sont les plus utilisés pour prendre en compte son intérêt pour le document dans sa globalité. Il n'existe pas, à notre connaissance, de travaux qui regardent plus en détails les intérêts de l'utilisateur envers des parties du document.

Dans notre travail, nous essayons de prendre en considération et de donner des éléments de solution à ces insuffisances en proposant :

1. *Un format générique de métadonnées* qui intègre les standards et les normes les plus utilisés dans l'état de l'art et identifiés dans le projet LINDO. Ce format est extensible en structure et en vocabulaire. Des règles de mappage ont été définies pour les standards sélectionnés dans le contexte du projet LINDO et le format proposé.
2. *La création d'un résumé des métadonnées multimédias*, dans le cadre d'un système distribué. Ce résumé est envoyé au serveur central et utilisé d'une part, pour sélectionner les serveurs distants susceptibles de contenir des documents pertinents pour la requête de l'utilisateur, et d'autre part, pour diminuer la quantité d'information véhiculée sur le réseau. Ce résumé peut être appliqué au format de métadonnées que nous proposons ou à tout autre modèle et représentation de métadonnées.
3. *La réalisation de l'indexation des contenus multimédias en deux temps* : une première fois lors de l'acquisition des contenus, avec une liste fixe d'algorithmes d'indexation (nous qualifions cette indexation d'« *implicite* ») et une deuxième fois, si nécessaire, au moment de la requête de l'utilisateur (nous qualifions cette indexation d'« *explicite* »). La sélection des algorithmes à utiliser sur chacun des serveurs distants du système, pour l'indexation implicite mais aussi pour l'indexation explicite, se réalise en fonction des caractéristiques des serveurs, des métadonnées déjà extraites (i.e., les résumés de métadonnées), des besoins de l'utilisateur, et du contexte d'acquisition des contenus multimédias.
4. *L'extension du modèle de métadonnées proposé* pour prendre en compte l'utilisateur à deux niveaux : (1) l'ajout de liens entre les documents et/ou parties de documents en fonction du comportement de l'utilisateur tout au long du cycle de vie du document multimédia (i.e., une *structure d'usage*), (2) en capturant l'évolution de l'usage du document dans le temps à travers l'inclusion dans le modèle de métadonnées *du concept de « température »*, qui est calculé en fonction de la façon dont l'utilisateur interagit avec le document multimédia, mais aussi avec les métadonnées
5. *La définition d'un algorithme de mise à jour de la température en fonction des interactions des utilisateurs avec le document, ou avec ses métadonnées*. Nous avons choisi d'implémenter un processus de propagation amorti pour le calcul de la température.

Dans cette deuxième partie du manuscrit nous détaillons nos contributions.

Dans le premier chapitre nous nous focalisons sur la modélisation de métadonnées, en détaillant le format de métadonnées proposé. Nous présentons d'abord l'approche que nous avons abordée pour la réalisation de ce modèle. Ensuite chaque composant du modèle est décrit.

Le deuxième chapitre contient la plus grande partie de notre contribution, en incluant la prise en compte de l'usage dans la gestion de métadonnées. Le principe d'indexation en deux temps est présenté. Cette indexation différenciée peut être appliquée à n'importe quel système distribué de gestion de documents multimédias. En ce qui nous concerne, nous l'appliquons et l'adaptions à l'architecture proposée dans le cadre du projet LINDO. Cette indexation différenciée se base sur une sélection des algorithmes d'indexation les plus pertinents pour un besoin donné. Pour réaliser cette

sélection nous nous appuyons sur les descriptions des ressources du système et également sur un résumé des métadonnées. Nous présentons dans ce chapitre, trois versions de cet algorithme de sélection : (1) une version qui ne prend en compte que la requête de l'utilisateur ; (2) une version qui utilise les contextes d'acquisition et des caractéristiques techniques des contenus multimédias ; (3) une dernière version qui prend en compte l'historique des requêtes des utilisateurs pour mettre à jour les algorithmes utilisés dans la première phase de l'indexation des contenus multimédias (indexation implicite).

La dernière partie du deuxième chapitre est dédiée à la prise en compte des interactions de l'utilisateur avec le document multimédia mais aussi avec ses métadonnées. Dans un premier temps nous présentons l'enrichissement du modèle avec l'ajout du concept de température et de la structure d'usage. L'algorithme de calcul de la température est détaillé également et la création des liens entre les documents/parties des documents est présentée à la fin de ce chapitre.

Chapitre 4. Le format générique de métadonnées

4.1.	<i>Introduction</i>	73
4.2.	<i>Les principes de l'approche proposée</i>	74
4.3.	<i>La modélisation proposée pour GMF</i>	74
4.4.	<i>Les métadonnées générales</i>	76
4.5.	<i>Les métadonnées de localisation</i>	78
4.6.	<i>Les métadonnées spécifiques à chaque type de média</i>	79
4.6.1.	Les métadonnées pour les documents textuels	79
4.6.2.	Les métadonnées pour les images	81
4.6.3.	Les métadonnées pour les vidéos	83
4.6.4.	Les métadonnées pour les documents audio	84
4.7.	<i>Les structures temporelle, spatiale et hypermédia</i>	86
4.8.	<i>La représentation d'une couche sémantique des métadonnées</i>	87
4.9.	<i>Bilan</i>	89

4.1. Introduction

L'étude réalisée dans le Chapitre 1, sur les standards de métadonnées, montre que ceux-ci sont nombreux et hétérogènes. Ils ne sont pas interopérables. Cela les rend difficiles à utiliser ensemble dans le cadre d'un système. Les solutions au problème d'interopérabilité envisagées dans l'état de l'art se focalisent surtout sur la création d'une ontologie qui décrit le contenu multimédia du point de vue sémantique, en partant du standard MPEG-7. L'intégration des autres standards de métadonnées n'est prise en compte que par quelques travaux : (Metadata Working Group, 2009) pour les standards de métadonnées associées aux images, le W3C Media Annotations Working Group⁴³. Les membres de ce dernier groupe de travail ont étudié presque tous les standards de métadonnées pour trouver les éléments communs. En se basant sur ces éléments, une ontologie a été définie. Mais cette ontologie n'intègre que ces éléments communs.

Dans un premier temps nous nous intéressons au problème de l'interopérabilité des métadonnées associées aux contenus multimédias. En réponse à ce problème, nous proposons *un format générique de métadonnées, extensible en structure et vocabulaire, qui inclut les standards et normes les plus utilisés dans l'état de l'art*. En nous appuyant sur les règles de mappage définies par le W3C Media Annotations Working Group, nous avons établi les règles de mappage entre les standards pris en compte et le format proposé.

Parmi les solutions au problème de l'interopérabilité identifiées par (Chan and Zeng, 2006) nous avons adopté la construction d'un framework de métadonnées.

Notre travail a été guidé par le projet industriel LINDO qui nous a imposé le choix de certains standards et normes de métadonnées qui sont les plus utilisés dans le projet et le milieu industriel

⁴³ <http://www.w3.org/2008/WebVideo/Annotations/>

(e.g., DC ; DICOM ; XMP ; EXIF ; MXF ; des métadonnées stockées dans les fichiers pdf, office). Nous ne nous sommes pas limités aux standards et normes de métadonnées, nous avons aussi étudié les formats de métadonnées qui sont obtenus en sortie par différents algorithmes d'indexation utilisés dans le cadre du projet (e.g., le logiciel Transcriber⁴⁴). Nous avons effectué une étude de ces standards pour identifier les éléments et les structures communes. Cette comparaison nous a permis de déterminer les éléments et la structure du format proposé.

Dans ce chapitre nous présentons en détails le format proposé pour résoudre le problème de l'interopérabilité.

4.2. Les principes de l'approche proposée

L'annotation de documents multimédias est une problématique abordée depuis de nombreuses années dans notre équipe (Sedes, 1998), (Amous et al., 2004), (Amous et al., 2005), (Jedidi, 2005), (Zayani, 2008). Ces travaux ont abouti à la proposition d'un modèle de métadonnées (Jedidi, 2005). Dans ce modèle, des descripteurs assez simples sont associés à chaque type de média.

Nous avons apporté une première amélioration au modèle proposé par (Jedidi, 2005) par l'extension de chaque descripteur avec des éléments de métadonnées appartenant aux standards de métadonnées identifiés dans le cadre du projet LINDO (Brut, Laborie, et al., 2009a), expliquée ci-après. En outre, nous avons pris en compte différents types de métadonnées :

- Métadonnées qui décrivent un certain type de document (i.e., du texte, de l'audio, de la vidéo, de l'image ou multimédia)
- Métadonnées spécifiques à un domaine d'application ou générales ;
- Métadonnées administratives ;
- Métadonnées techniques ;
- Métadonnées de haut-niveau ou de bas-niveau.

Dans certains des standards étudiés dans le Chapitre 1, nous avons constaté que les métadonnées générales qui décrivent tout le document, du point de vue administratif, sont regroupées dans une seule structure. C'est notamment le cas de METS et de LOM. Nous avons voulu reprendre cette idée dans le format proposé.

Comme dans (Jedidi, 2005), nous avons introduit dans le format de métadonnées proposé, les structures qu'un document peut avoir. Nous avons privilégié les structures logique, spatiale, temporelle et hypermédia.

En plus de l'ajout des standards de métadonnées nous voulons prendre en compte des informations supplémentaires qui peuvent être extraites à partir du contenu multimédia. Pour cela nous avons étudié les sorties de certains algorithmes d'indexation qui traitent le contenu des documents pour identifier des objets ou qui segmentent le contenu en fonction de certains critères. Tout ce travail nous a permis de définir le format de métadonnées suivant.

4.3. La modélisation proposée pour GMF

La modélisation UML de notre proposition est présentée dans la Figure 12. Il s'agit d'une version simplifiée.

⁴⁴ <http://trans.sourceforge.net/en/presentation.php>

Nous avons associé à chaque document multimédia une métadonnée *MultimediaMetadata*, qui contient toutes les métadonnées liées à ce document. L'attribut *src* de cette classe fait le lien entre les métadonnées et le document multimédia.

Le *MultimediaMetadata* est composé d'une structure qui contient les informations générales (e.g., l'auteur, la date de création, le titre, le sujet, les mots-clés) sur le document (*GeneralInformation*), d'au moins un média et de 0 ou plusieurs *Link*.

Chaque classe *Media* regroupe tous les descripteurs de métadonnées correspondant à chaque type de média (i.e., texte, audio, vidéo, image). En plus des métadonnées spécifiques, un *Media* peut avoir ou non des informations générales (*GeneralInformation*).

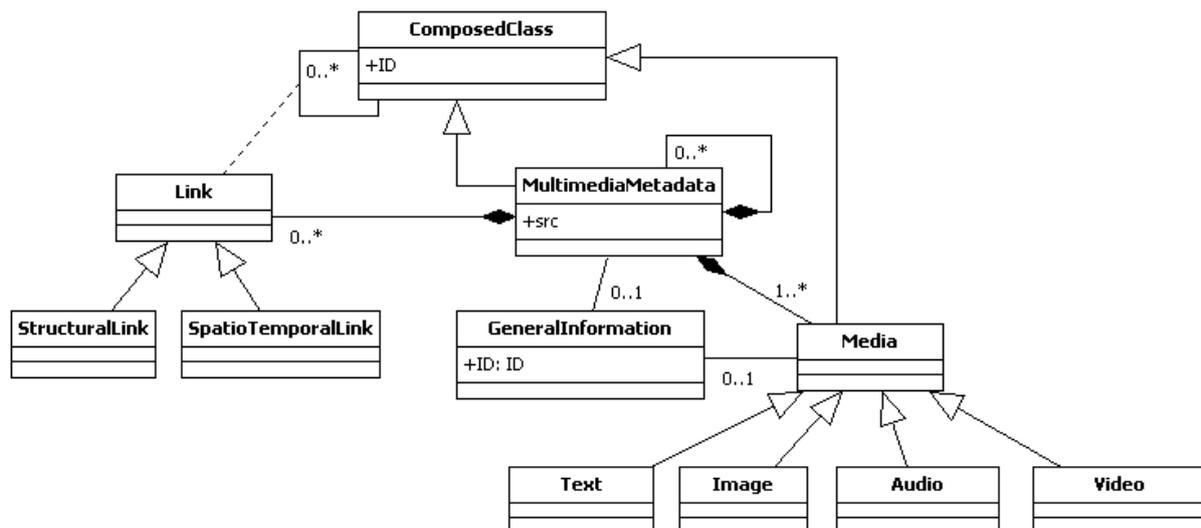


Figure 12 – Le format de métadonnées proposé

L'utilisation de la classe *GeneralInformation* pour décrire le document dans sa globalité, mais aussi pour chaque media qui le compose, a deux avantages majeurs :

- L'élimination de la redondance, en ce qui concerne les informations administratives (e.g., si les médias qui se trouvent dans le document ont le même auteur que le document, alors l'auteur peut n'être représenté que dans la classe *GeneralInformation* correspondant au document) ;
- La possibilité d'avoir des niveaux de granularité de métadonnées générales différents pour chaque document ou média.

Pour la prise en compte des différentes structures d'un document, nous avons introduit la classe *Link*. Elle réunit tous les différents liens qui peuvent être trouvés dans le document. Un lien peut être établi entre des médias, entre des documents ou entre des éléments des différents descripteurs des médias. Un lien peut être de deux types : lien spatio-temporel (i.e., utilisé pour décrire la structure spatiale ou temporelle du document) ou lien structurel (i.e., utilisé pour représenter la structure hypermédia). Plus de détails sur cette classe se trouvent dans la section 4.7.

Comme nous l'avons déjà vu dans la section 1.3, un document multimédia peut avoir plusieurs structures. Dans le modèle de métadonnées proposé, nous avons modélisé :

- La structure logique du document à travers la structure des métadonnées ;

- La structure spatiale à travers des liens spatiaux ;
- La structure temporelle à travers des liens temporels ;
- La structure hypermédia à travers des liens directs, ou hypermédia.

Chaque élément de métadonnée a un identifiant unique dans le cadre des métadonnées associées à un document multimédia. De cette façon on peut facilement les identifier de manière unique.

Pour l'implémentation de notre proposition, nous avons choisi le XML, parce qu'il est extensible en vocabulaire et en structure. En outre, dans l'état de l'art, beaucoup d'algorithmes d'indexation fournissent du XML en sortie et ainsi le traitement des métadonnées est homogène, sans la nécessité de transformation de langage de représentation.

Par la suite nous allons détailler chaque classe de notre modèle, en précisant sa structure et en fournissant un exemple XML de contenu.

4.4. Les métadonnées générales

Les métadonnées générales, correspondant à la classe GeneralInformation, peuvent être associées d'un côté au document multimédia et de l'autre côté à chaque média contenu par le document, conformément à la Figure 12. Le diagramme de ces métadonnées générales est présenté dans la Figure 13.

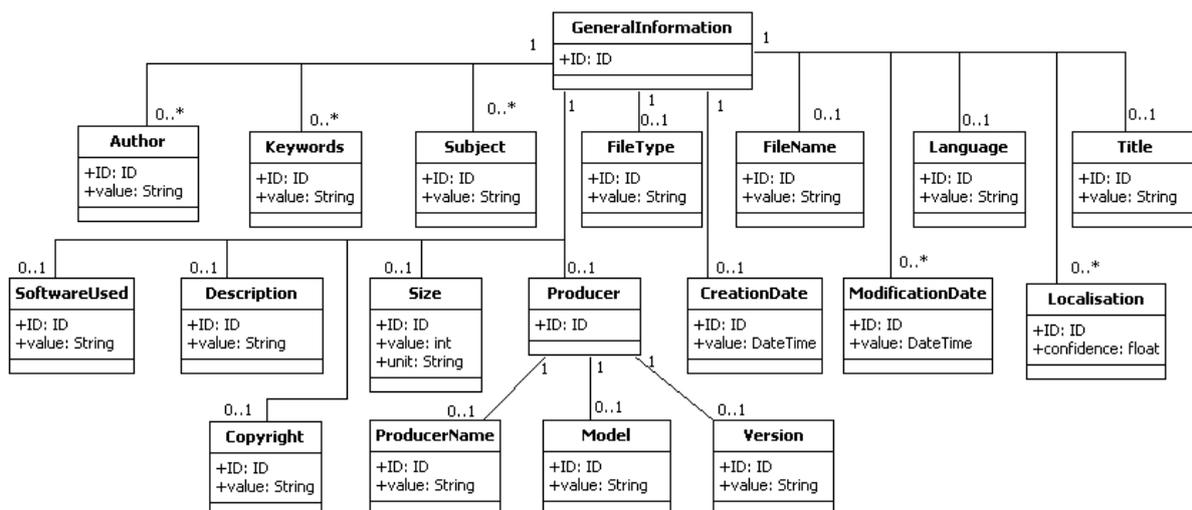


Figure 13 – Le diagramme des métadonnées générales

Ces métadonnées contiennent des éléments qui décrivent un document du point de vue administratif. Pour déterminer ces éléments, nous avons effectué une comparaison des standards et normes de métadonnées sélectionnés dans le cadre du projet (e.g., DC ; DICOM ; XMP ; EXIF ; MXF ; des métadonnées stockées dans les fichiers pdf, office). Un extrait de cette étude se trouve dans le Tableau 3. A partir de cette étude, nous avons choisi les éléments que nous avons considérés comme étant des informations générales. Nous avons choisi les éléments qui apparaissaient dans la majorité des standards considérés. On retrouve parmi les éléments de cette classe, ceux du standard Dublin Core, et d'autres éléments qui sont définis dans d'autres standards. Aucun de ces éléments n'étant pas obligatoire, l'utilisateur a la possibilité de saisir les métadonnées qu'il considère importantes.

Les métadonnées générales sont composées de :

- 0 ou plusieurs *auteurs*. Le nom et le prénom de l’auteur sont saisis dans une seule valeur, comme c’est le cas de l’auteur dans Dublin Core.
- 0 ou plusieurs *mots-clés*. Plusieurs mots-clés peuvent être saisis dans un même élément, ou bien chaque mot-clé peut être saisi dans un élément différent.
- 0 ou plusieurs *sujets*.
- Au plus un *type de fichier* (e.g., .pdf, .doc, .img).
- Au plus un *nom de fichier* (e.g., rapport.doc).
- Au plus une *langue*.
- Au plus un *titre*.
- Au plus une *description*.
- Au plus une *taille*.
- Au plus une *date de création*.
- 0 ou plusieurs *dates de modification*. Si l’utilisateur considère ces informations comme importantes dans la description du document, il peut garder l’historique des dates de modifications du document.
- Au plus un *logiciel utilisé* pour la création du document.
- Au plus un *producteur*. Cet élément est utilisé pour saisir le producteur du logiciel ou du dispositif utilisé pour la création du document.
- 0 ou *plusieurs localisations*, dans le temps et l’espace. Plus de détails sur cet élément se trouvent dans la section 4.5.

Le Tableau 9 présente un petit extrait des métadonnées générales associées à un document. Dans cet exemple, on retrouve le nom du fichier, le titre, la description et quelques mots-clés qui correspondent à ce manuscrit.

Tableau 9 – Extrait de métadonnées générales pour ce manuscrit

```

<document ID="U0" src=" rapport.docx ">
  <generalInformation ID="U1" >
    <fileName ID="U2">
      <value> rapport.docx </value>
    </fileName>
    <title ID="U3">
      <value> Contribution à la Modélisation des Métadonnées Associées aux Documents Multimé-
      dias: Enrichissement des Métadonnées par l’Usage </value>
    </title>
    <description ID="U4">
      <value>Le travail réalisé dans le cadre de cette thèse s’intéresse à la modélisation et gestion
      des métadonnées associées aux documents multimédias.... </value>
    </description>
    <keywords ID="U5">
      <value>multimédia, métadonnées, usage, indexation, reclassement</value>
    </keywords>
  </generalInformation>
</document>

```

4.5. Les métadonnées de localisation

A tout document multimédia peut être associée une position dans le temps et l'espace qui représente le positionnement du document au moment de sa création par exemple. Ce positionnement dans le monde réel se réalise en utilisant le temps absolu et les coordonnées GPS. Mais, dans certaines situations, il est important de localiser précisément une information (qui est extraite par exemple par un algorithme d'indexation) dans le document lui-même. A ce moment là, on parle d'une localisation intra document.

Ces informations ne sont prises en compte que par quelques standards de métadonnées (e.g., les coordonnées GPS se retrouvent dans MXF et EXIF). Nous pensons que cette localisation est aussi importante que les autres informations qui sont modélisées par les standards de métadonnées et nous l'avons introduite dans le modèle de métadonnées proposé sous la forme de la classe Localisation présentée dans la Figure 14.

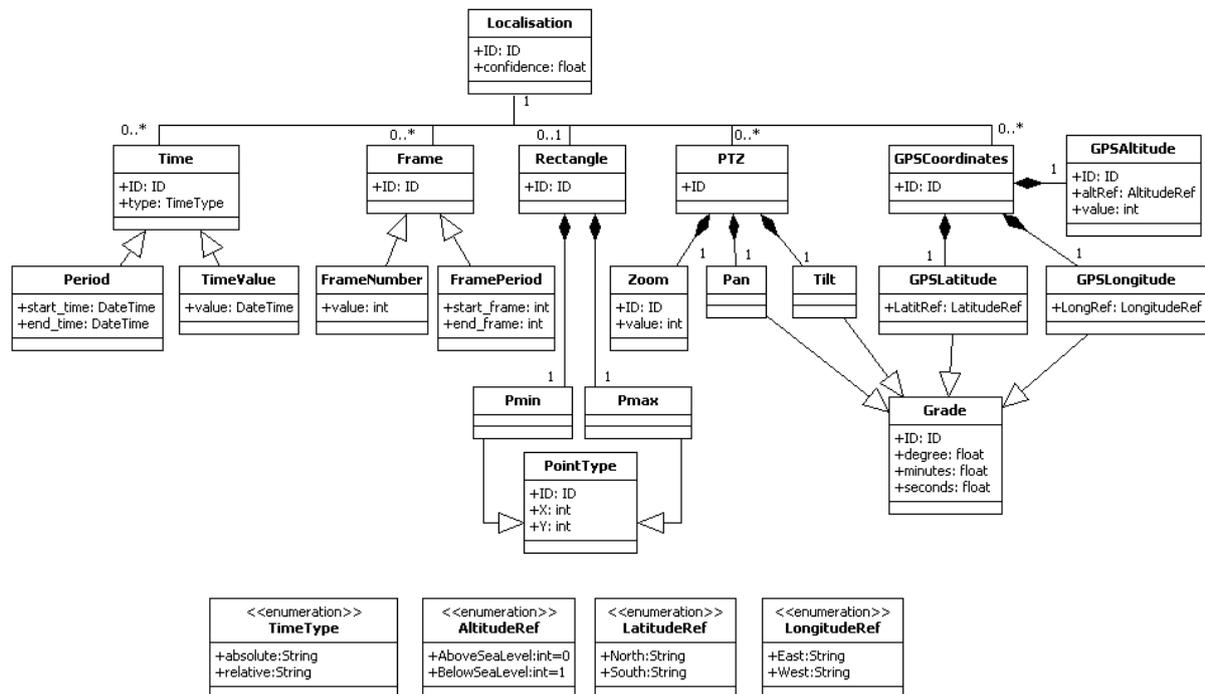


Figure 14 – Le diagramme de la classe Localisation

La localisation se fait de plusieurs manières, dans le temps et dans l'espace :

- Dans le temps, en utilisant l'élément *time*, relatif au début du document, ou absolu (cette propriété est spécifiée par l'attribut *type* de l'élément *time*). Le temps peut être saisi sous forme d'une valeur ponctuelle, ou sous forme d'une période. Cet élément peut être utilisé pour tout type de document multimédia.
- Pour la vidéo, le temps peut être exprimé sous la forme de *frames*, en valeur ponctuelle ou en tant que période.
- La localisation dans le monde réel, peut se faire en utilisant les *coordonnées GPS*, en longitude, latitude et altitude.
- Pour la vidéo, on peut avoir un positionnement plus particulier, en utilisant les propriétés *PTZ* de la camera qui a été utilisée pour son acquisition.
- Dans le cadre des images, par exemple, on peut avoir une région contenant une information intéressante. Cette *région* a été introduite dans la localisation, sous la forme d'un

rectangle, pour lequel on spécifie les coordonnées x et y du point du haut à gauche et celles du point du bas à droite.

Il y a des situations où un document ne peut pas être localisé avec une grande précision pour différentes raisons (e.g., la précision de la puce GPS, la qualité de l'algorithme d'indexation qui détermine les régions d'intérêt dans le document). Cette précision a été introduite dans le modèle proposé, en utilisant l'attribut *confidence*, qui a une valeur réelle entre 0 et 100. Cet attribut représente la précision de la localisation en pourcentage.

Cet élément est utilisé à plusieurs endroits dans le modèle, dans les métadonnées générales, mais aussi dans d'autres éléments.

Un exemple de métadonnées de localisation est fourni dans le Tableau 10. Ces métadonnées générales localisent le document en utilisant les coordonnées GPS. Ces coordonnées correspondent au centre de la ville de Toulouse, France.

Tableau 10 – Extrait de métadonnées de localisation associées à ce manuscrit

```
<document ID="U0" src="rapport.docx">
  <generalInformation ID="U1" >
    <localisation ID="U7" >
      <GPSCoordonates ID="U8" >
        <GPSLatitude ID="U9" LatitRef="North" >
          <degrees >43</degrees>
          <minutes >36</minutes>
          <seconds >43</seconds>
        </GPSLatitude>
        <GPSLongitude ID="U10" LongRef="East">
          <degrees >1</degrees>
          <minutes >27</minutes>
          <seconds >14</seconds>
        </GPSLongitude>
        <GPSAltitude ID="U11" AltiRef="1">
          <value>200</value>
        </GPSAltitude>
      </GPSCoordonates>
    </localisation>
  </generalInformation>
</document>
```

4.6. Les métadonnées spécifiques à chaque type de média

Dans cette partie nous allons présenter les éléments de métadonnées associées à chaque type de média dans le format de métadonnées proposé.

4.6.1. Les métadonnées pour les documents textuels

Dans la description des métadonnées associées aux documents textuels (Figure 15), on retrouve bien les mêmes informations générales que celles associées au document multimédia. Mais pour le texte, l'analyse des métadonnées qui sont sauvegardées dans un fichier textuel fait ressortir d'autres métadonnées telles que *le nombre de pages*, *le nombre de caractères* et *le nombre de mots*, qui ont été introduites dans le format de métadonnées.

Dans le cas où le texte fait partie d'un document plus complexe, le format de métadonnées permet de dire quelle est sa position dans le cadre du document, en utilisant l'élément *localisation*.

La structure du document textuel, et sa division en chapitres et sous chapitres est reprise dans les métadonnées sous la forme de *chapitres* qui sont composés par d'autres chapitres. Chaque chapitre peut être décomposé en une séquence de *paragraphes* et sous chapitres, la granularité de cette division est choisie en fonction du besoin. En plus du texte proprement dit, dans un document textuel, on retrouve aussi des références bibliographiques qui sont regroupées dans le cadre du format de métadonnées dans l'élément *bibliography*, et qui sont gérées au niveau du paragraphe à travers des références aux identifiants des *éléments bibliographiques*. Nous avons introduit les éléments bibliographiques qu'un document textuel peut contenir dans notre modèle parce qu'ils font ressortir des informations supplémentaires sur le contenu du document à travers les liens que le créateur du document a créé de manière directe avec les documents référencés.

Dans le cas où le document textuel ne peut pas être divisé de manière évidente en chapitres et sous chapitres, le format de métadonnées offre la possibilité de diviser le texte en *unités textuelles*.

Les métadonnées peuvent contenir aussi une partie du contenu textuel, si cette nécessité est identifiée. Cela se réalise à travers l'attribut *value* du *TextUnit* et de *Paragraphe*. L'attribut *RefID*, permet de garder un lien direct entre un paragraphe/unité textuelle et un autre élément qu'il contient (e.g., une image, un objet, un autre paragraphe,...).

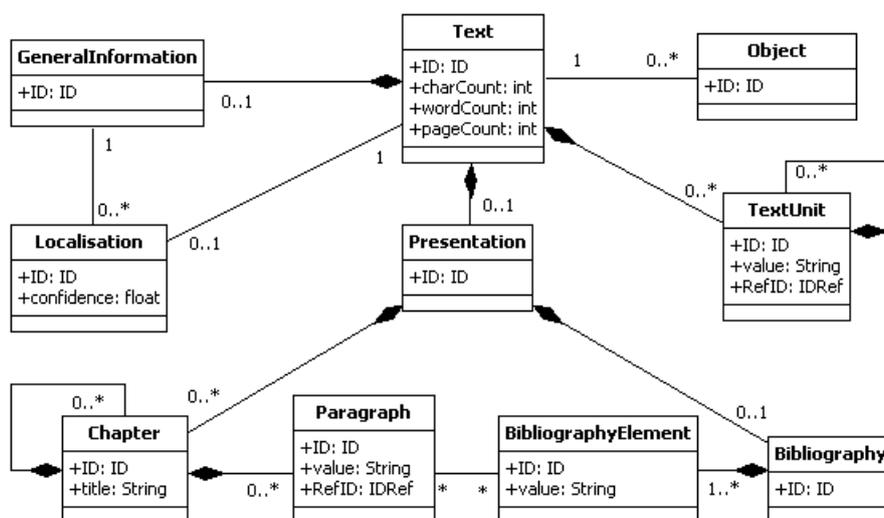


Figure 15 – Les métadonnées associées aux documents textuels

Dans le texte du document, on peut identifier aussi des concepts et des sujets qui sont plus importants. Dans le format de métadonnées, ces concepts sont décrits par les *objets*. Plus de détails sur le contenu de l'élément objet sont donnés dans la section 4.8.

Comme pour les autres éléments du modèle, toutes les métadonnées associées aux documents textuels ont un ID unique pour une meilleure identification et gestion.

Un exemple de métadonnées associées à ce manuscrit est présenté dans le Tableau 11. En plus des métadonnées générales, on retrouve une segmentation en chapitres/sous chapitres. Le deuxième paragraphe contient une référence bibliographique, et le troisième paragraphe contient une référence vers une image, qui est décrite dans un autre élément.

Tableau 11 – Extrait des métadonnées associées à ce manuscrit

```

<document ID="U0" src="rapport.docx">
  <generalInformation ID="U1" >
    <author ID="U12" >
      <value> Manzat Ana-Maria </value>
    </author>
  </generalInformation>
  <text ID="U14" >
    <generalInformation ID="U15" >
      <softwareUsed ID="U16">
        <value>Microsoft Word</value>
      </softwareUsed >
    </generalInformation>
    <presentation ID="U17">
      <chapter ID="U18" title="Introduction">
        <chapter ID="U19" title="Le contexte de notre travail" >
          <paragraph ID="U20">
            <value> L'évolution que les technologies multimédias </value>
          </paragraph>
          <paragraph ID="U21">
            <RefBibl refID="U24"/>
            <value> Ce fossé sémantique se trouve entre les informations de ....</value>
          </paragraph>
          <paragraph ID="U22" RefID="U180" >
            <value> Figure 1....</value>
          </paragraph>
        </chapter>
      </chapter>
      <bibliography ID="U23">
        <BibliographyElement ID="U24">
          <value>A.W.M. Smeulders, M. Worrying, S. Santini, A. Gupta, and R. Jain</value>
        </ BibliographyElement >
      </bibliography>
    </presentation>
  </text >
  <image ID="U180">...</image>
</document>

```

4.6.2. Les métadonnées pour les images

Pour les métadonnées associées aux images (Figure 16), on retrouve la même structure que pour le texte, dans le sens où on a associé des métadonnées générales et des métadonnées de localisation à une image. Les métadonnées générales contiennent des éléments que nous retrouvons aussi dans d'autres standards de métadonnées, tels que EXIF, DICOM, IPTC (c'est notamment le cas de l'auteur, la date de création).

En plus de ces informations générales, nous avons ajouté dans le format proposé, certains éléments des standards EXIF, IPTC et DICOM, en gardant la même hiérarchie que dans le format d'origine. Pour éviter la redondance, dans notre format nous avons regroupé les éléments communs des trois standards et nous ne les avons considérés qu'une fois. C'est notamment le cas des coordon-

nées GPS spécifiées dans le standard EXIF, que nous retrouvons dans la partie coordonnées GPS de l'élément localisation, et qui n'ont donc pas été incluses dans l'élément *Exif*.

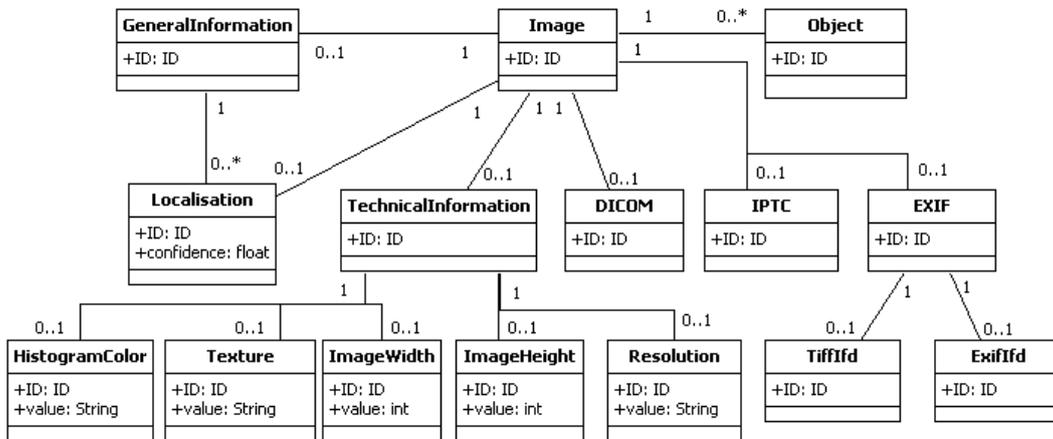


Figure 16 – Les métadonnées associées aux images

Les éléments communs des différents standards considérés ont été regroupés dans l'élément TechnicalInformation. Parmi les éléments qui se trouvent dans cette partie du modèle, nous mentionnons *la texture, l'histogramme de couleur, la hauteur et la largeur* de l'image, en pixels.

De plus en plus d'algorithmes d'indexation ont été définis pour extraire de l'information des images. Pour pouvoir décrire cette information et la relier à l'image, nous avons introduit dans le format de métadonnées les *objets*. Plus de détails sur le contenu de l'élément objet sont donnés dans la section 4.8.

Tableau 12 – Extrait de métadonnées associées à une image

```

<document ID="U0" src="image2.jpg">
  <generalInformation ID="U1">
    <fileName ID="U2">
      <value> image2.jpg </value>
    </fileName>
    <title ID="U3">
      <value> Reunion_de_Travail </value>
    </title>
    <size ID="U4" unit="bytes">
      <value> 5895680</value>
    </size>
  </generalInformation >
  <image ID="U5">
    <technicalInformation ID="U6" >
      <imageWidth ID="U7">
        <value>2448</value>
      </imageWidth>
      <imageHeight ID="U8">
        <value> 3264</value>
      </imageHeight>
    </technicalInformation>
  </image>
</document>
  
```

Dans le Tableau 12, un exemple de telles métadonnées est fourni, en se focalisant sur les informations techniques qui sont obtenues à partir des métadonnées EXIF stockées dans le même fichier que l'image.

4.6.3. Les métadonnées pour les vidéos

Les métadonnées associées aux vidéos (Figure 17) sont utilisées pour la description de la partie vidéo d'un contenu audiovisuel. Les métadonnées associées à la partie sonore du contenu sont incluses dans la classe audio, qui est détaillée dans la section 4.6.4.

Comme pour les autres médias, des informations générales et la localisation sont associées à la vidéo.

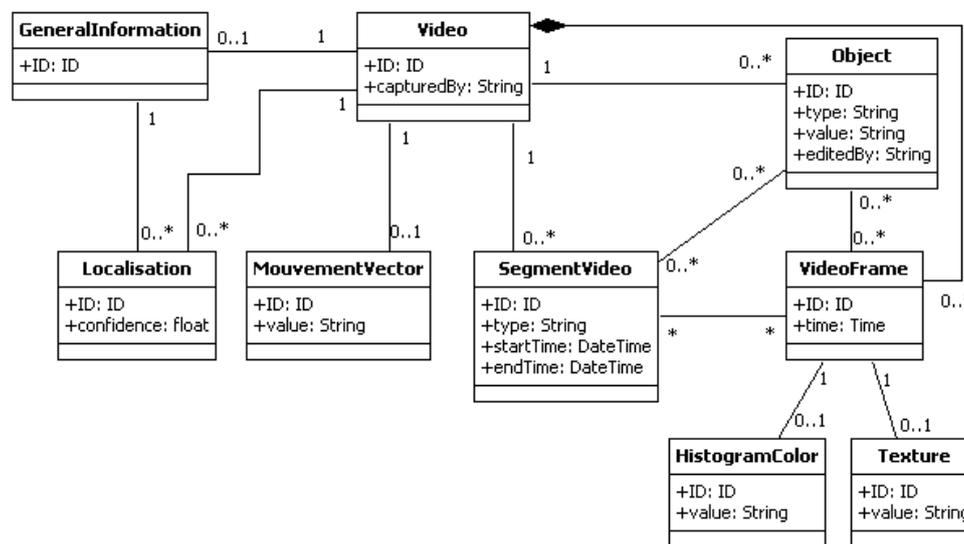


Figure 17 – Les métadonnées associées aux vidéos

Dans notre format de métadonnées, nous avons considéré certains éléments des standards MXF et MPEG-7. Par exemple, les coordonnées GPS du standard MXF ont été reprises dans la *localisation*. Du MPEG-7, nous avons repris *la texture et l'histogramme* qui sont associés à un frame. Le mouvement de la caméra qui enregistre la vidéo, et qui peut être décrit dans MXF et dans MPEG-7, est inclus dans la localisation de la vidéo, sous la forme des coordonnées PTZ. Pour maintenir un lien entre la vidéo et la caméra qui l'a capturée, nous avons introduit l'attribut *capturedBy*, qui stocke l'URI de la description de la caméra utilisée pour l'enregistrement de la vidéo.

Du point de vue de la structure de la vidéo, celle-ci peut être divisée en segments, selon différents critères. Cette segmentation est incluse dans le modèle de métadonnées, sous la forme d'une succession de segments. D'un autre côté, on peut segmenter la vidéo en frames représentatives, lesquelles sont caractérisées par différentes propriétés (e.g., l'histogramme). Dans le cas où on choisit de décrire les frames, mais aussi de représenter une segmentation de la vidéo, le format de métadonnées permet de dire qu'un frame est contenu dans un certain segment.

Dans le cas d'une vidéo, plus que pour le texte, on essaie d'extraire de l'information de son contenu, pour pouvoir retrouver plus facilement les parties intéressantes de la vidéo. Le format de

métadonnées proposé permet de décrire ces informations sous la forme *d'objets*. Plus de détails sur le contenu de l'élément objet se trouvent dans la section 4.8.

Le Tableau 13 présente un extrait des métadonnées obtenues suite à la segmentation d'une vidéo acquise par une camera de vidéosurveillance dans une station de métro. Cette vidéo est prise par la camera numéro 2, qui est orientée dans une certaine direction établie à travers ses paramètres PTZ. Ces paramètres sont ajoutés dans l'élément localisation de la vidéo.

Tableau 13 – Extrait de métadonnées associées à une vidéo

```
<document ID="U0" src="video2.avi">
  <generalInformation ID="U1">
    <description ID="U2">
      <value> vidéosurveillance de la station métro ... de Toulouse </value>
    </ description >
  </generalInformation>
  <video ID="U3" capturedBy="camera 2" >
    <localisation ID="U4">
      <PTZ ID="U5">
        <Pan ID="U6">
          <degrees>20</degrees>
          <minutes>0</minutes>
          <seconds>0</seconds>
        </Pan>
        <Tilt ID="U7">
          <degrees>30</degrees>
          <minutes>0</minutes>
          <seconds>9</seconds>
        </Tilt>
        <Zoom ID="U8">
          <value>2</value>
        </Zoom>
      </PTZ>
    </localisation>
    <segmentVideo ID="U9" startTime="2012-04-29T14:20:40" endTime="2012-04-29T14:21:17"/>
    <segmentVideo ID="U10" startTime="2012-04-29T14:21:17" endTime="2012-04-29T14:21:59"/>
  </video>
</document>
```

4.6.4. Les métadonnées pour les documents audio

Les métadonnées associées aux contenus audio sont présentées dans la Figure 18. Comme pour les autres médias, dans la description des documents audio, on retrouve les informations générales et la localisation dans le cadre du document.

En ce qui concerne les informations qui sont stockées dans les métadonnées ID3, elles sont incluses dans les métadonnées générales.

Les contenus audio peuvent être divisés en segments en fonction de certains paramètres (e.g., musique/non musique, parole/non parole). Ce type de segmentation est modélisé en utilisant *les segments audio*.

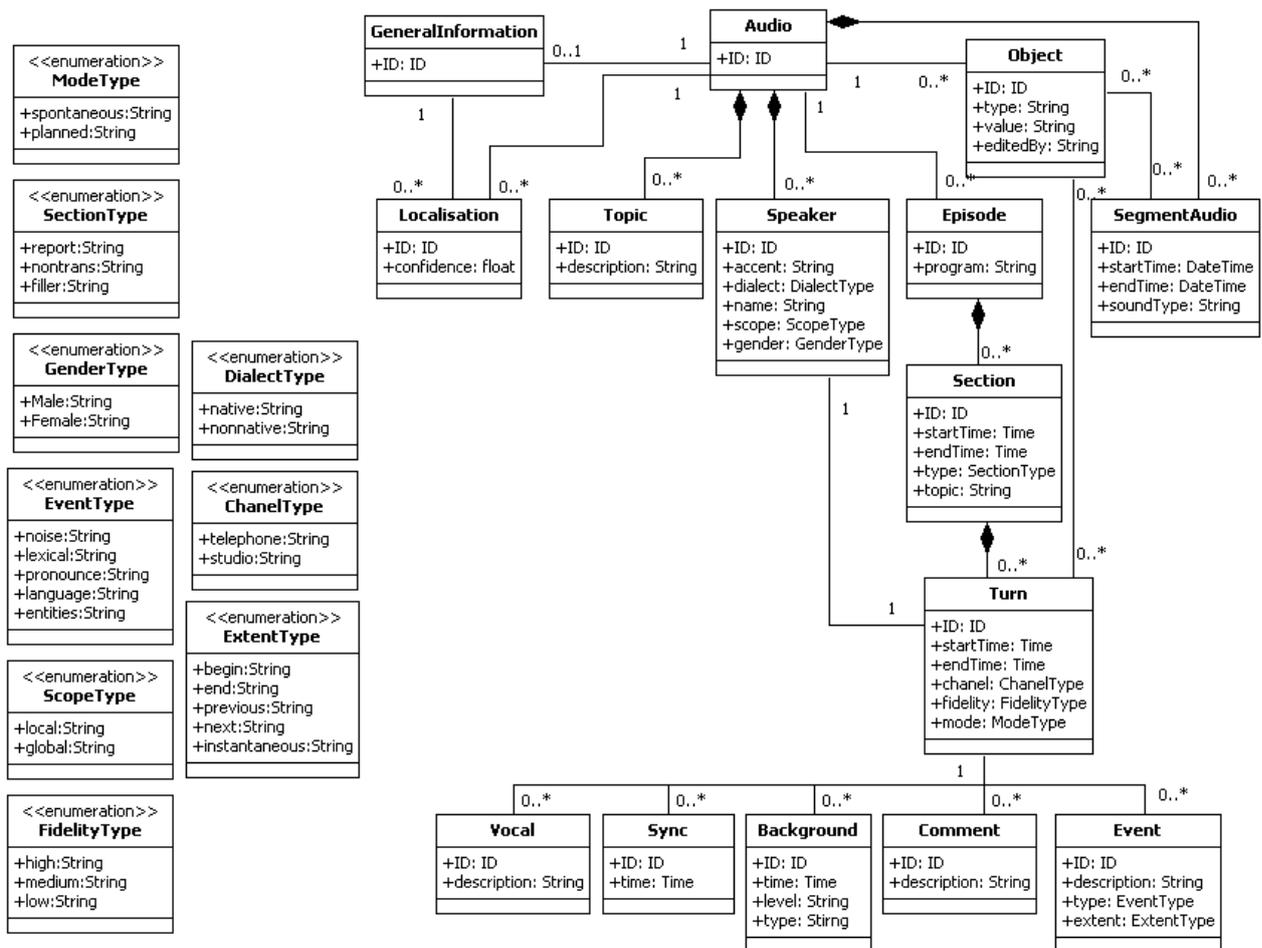


Figure 18 - Les métadonnées associées aux contenus audio

Dans le modèle de métadonnées proposé, nous avons introduit les descripteurs qui peuvent être générés en utilisant le logiciel Transcriber. La prise en compte de la sortie de ce logiciel était une autre demande imposée par le projet. En outre, les algorithmes d'indexation qui traitent de l'audio ont une sortie compatible avec ce logiciel pour pouvoir vérifier leurs résultats. Ce logiciel réalise une segmentation de l'audio en épisodes, sections et prises de parole. En plus de cette segmentation, il permet d'identifier aussi les *sujets* dont on parle dans l'audio et les *intervenants*. En utilisant les classes composantes de la classe Turn, on peut décrire plus en détails le contenu de l'audio, en termes de discours, mais aussi en termes de propriétés acoustiques de chaque intervenant.

Pour intégrer plus d'informations sur les différents sujets identifiés dans le contenu, nous avons utilisé les *objets*. Plus de détails sur le contenu de cet élément se trouvent dans la section 4.8.

Dans le Tableau 14, on retrouve un exemple de métadonnées associées à un contenu audio. Cet exemple est obtenu suite à la segmentation manuelle d'une chanson du dessin animé Les Aristochats. Cette chanson est chantée par Thomas O'Malley. La segmentation nous permet d'établir à partir de quel moment à jusqu'à quel moment le personnage chante et de détailler ce qu'il dit.

Tableau 14 – Extrait des métadonnées associées à un contenu audio

```

<document ID="U0" src="aristocats_thomasomalleycat_clip.mp3">
  <generalInformation ID="U1">
    <title ID="U2">
      <value>Aristocats Thomas O'Malley cat</value>
    </title>
  </generalInformation>
  <audio ID="U3">
    <Speakers ID="U4">
      <Speaker ID="U5" name="Thomas Omalley " dialect="native" scope="local"/>
    </Speakers>
    <Topics ID="U6">
      <Topic ID="U7" desc="music"/>
    </Topics>
    <Episode ID="U8">
      <Section ID="U9" type="report" startTime="0" endTime="33.0187755102" topic="U7">
        <Turn ID="U10" startTime="0" endTime="33.0187755102" speaker="U5" fidelity="high">
          <Sync ID="U11" time="0"/> I'm like a chee-chee-chee-chee-rony like they make at home
          <Sync ID="U12" time="3.739"/> Or a healthy fish with a big backbone
        </Turn>
      </Section>
    </Episode>
  </audio>
</document>

```

4.7. Les structures temporelle, spatiale et hypermédia

Dans le modèle proposé, nous avons considéré les structures spatiale, temporelle et hypermédia qu'un document peut avoir. Nous les avons intégrées sous la forme de liens qui sont établis entre des documents multimédias (dans le cas de la structure hypermédia) et entre les différents médias qui font partie d'un même document (dans le cas des structures spatiale et temporelle). Ces liens ne sont pas bidirectionnels. Cette modélisation est présentée dans la Figure 19.

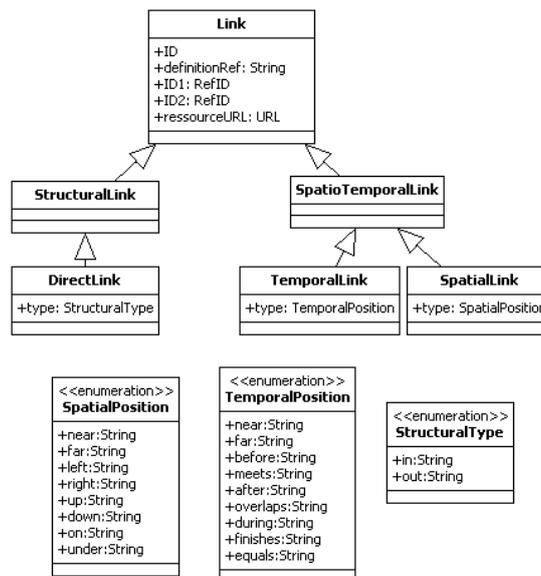


Figure 19 – Les métadonnées de structure

Pour identifier les ressources qui sont mises en correspondance, nous utilisons les identifiants de chaque élément. Dans notre modèle, chaque lien a un attribut nommé « type » :

- Pour le lien spatial, le type peut être une des relations topologiques (Clementini et al., 1994), (Alboody, 2011) ou cardinales (Papadias and Theodoridis, 1997), ou une combinaison de ces deux types de relations (Jedidi, 2005) ;
- Pour le lien temporel, le type peut être une des relations d'Allen (Allen, 1983) ;
- Pour le lien de structure, qui correspond aux liens classiques dans les documents hypermédia, le type peut être soit « entrant », pour les ancrs qui font référence au document, soit « sortant » pour les ancrs du document qui font référence à un autre document.

Tableau 15 – Extrait des métadonnées de structure associées à un document

```
<document ID="U0" src="naturaMundi/htmlFiles/huile-visage.html">
  <generalInformation ID="U1" >
    <fileName ID="U2">
      <value>naturaMundi/htmlFiles/huile-visage.html</value>
    </fileName>
  </generalInformation>
  <image ID="U3">...</image>
  <image ID="U25">...</image>
  <links ID="U67">
    <DirectLink resourceUrl="http://www.naturamundi.com/" type="out" ID="U68" />
    <DirectLink resourceUrl="http://www.naturamundi.com/creme-de-jour.html" type="in"
ID="U408"/>
    <SpatialLink ID="U409" ID1="U3" type="left" ID2="U25"/>
    <SpatialLink ID="U410" ID1="U25" type="right" ID2="U3"/>
  </links>
</document>
```

Le Tableau 15 fournit un extrait des structures spatiale et hypermédia correspondant à une page html. Cette page contient un lien vers la page «www.naturamundi.com », et la page « creme-de-jour.html » contient un renvoi vers la page courante. En ce qui concerne la structure spatiale, cette page html contient deux images côte à côte. Cette relation bidirectionnelle est traduite dans le format à travers les deux liens spatiaux avec les identifiants : U409 et U410.

4.8. La représentation d'une couche sémantique des métadonnées

Les métadonnées présentées jusqu'à maintenant contiennent surtout des informations techniques et administratives sur le contenu multimédia. Mais la nécessité de décrire, d'un point de vue sémantique le contenu multimédia, est devenue de plus en plus importante dans la gestion des contenus multimédias. Cette approche sémantique n'est pas forcément prise en compte par les standards de métadonnées. Pour pallier ce manque, nous avons introduit dans le format, la classe *objet*. Cet élément permet d'ajouter de l'information supplémentaire sur des sujets considérés comme étant importants dans le contenu multimédia. La modélisation de cette classe est fournie dans la Figure 20.

Chaque objet a un type qui identifie sa nature. Il peut être un texte (e.g., une personne, un véhicule, une construction, un arbre), ou bien une URL vers un concept d'une ontologie. Dans le cas

de l'objet, la valeur contient de l'information supplémentaire sur l'objet. L'attribut `editedBy` est utilisé pour identifier la personne ou l'algorithme d'indexation qui a extrait l'information sur l'objet.

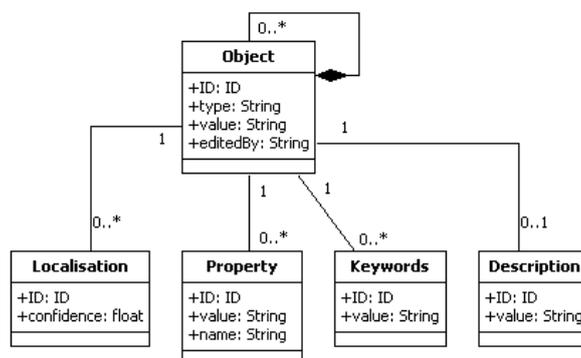


Figure 20 – Les métadonnées décrivant les objets identifiés dans un document

Un objet peut être positionné dans le temps et dans l'espace en utilisant la *localisation*. Cette localisation peut être réalisée dans le monde réel, mais aussi dans le cadre du document, comme décrit dans la section 4.5.

Pour décrire un objet, le modèle met à disposition plusieurs modalités :

- Une *description* : le niveau de détail dans cet élément n'est pas défini a priori ;
- Un ensemble de *mots-clés* ;
- Un ensemble de *propriétés* : nous n'avons pas introduit dans le format, des propriétés prédéfinies, mais nous avons choisi de laisser à l'utilisateur, la liberté d'ajouter dans la description d'un objet, les propriétés qu'il considère importantes. A travers l'attribut *name* de la propriété, l'utilisateur spécifie le nom de la propriété pour laquelle il donne la *valeur* dans l'attribut *value*.
- Si nécessaire, un objet peut être décomposé en sous-objets. Le niveau de granularité dans cette décomposition est libre.

Ces informations ne sont pas spécifiques à un type de media, mais on peut les retrouver dans n'importe quel media. C'est pour cela que nous avons ajouté l'objet dans la description de chaque media et que nous avons établi des liens entre l'objet et les différents segments ou parties du média correspondantes (conformément à la Figure 19).

Pour donner un exemple de ce type de métadonnées, le Tableau 16 présente un extrait des métadonnées associées à une image dans laquelle deux objets ont été identifiés : une personne et un sac-à-dos. Pour la personne (l'objet avec l'identifiant U11), on connaît sa position dans l'image, avec une précision de 75%, mais aussi le fait qu'il s'agit d'une personne qui rit. En ce qui concerne le deuxième objet, on sait qu'il est petit et qu'il est de couleur rouge. Cet objet est localisé dans l'image avec une précision plus grande, de 89%.

Tableau 16 – Extrait des métadonnées associées à une image, qui contient deux objets identifiés

```
<document ID="U0" src="image2.jpg">
  <generalInformation ID="U1"> </generalInformation>
  <image ID="U4">
    <object ID="U5" type="Person">
```

```

<localisation ID="U6" confidence="75">
  <rectangle ID="U7">
    <pMin ID="U8" x="150" y="50"/>
    <pMax ID="U9" x="200" y="100"/>
  </rectangle>
</localisation>
<description ID="U10">
  <value> souriante </value>
</description>
</object>
<object ID="U11" type="Backpack">
  <localisation ID="U12" confidence="89">
    <rectangle ID="U13">
      <pMin ID="U14" x="150" y="55"/>
      <pMax ID="U15" x="190" y="60"/>
    </rectangle>
  </localisation>
  <property ID="U16" name="color">
    <value>rouge</value>
  </property>
  <property ID="U17" name="size">
    <value>petit</value>
  </property>
</object>
</image>
</document>

```

4.9. Bilan

Dans ce chapitre nous avons abordé le premier volet de notre contribution : la proposition d'un format de métadonnées associées à tout type de contenu multimédia. Dans notre travail nous partons du postulat que tout média est « élicitable », c'est-à-dire qu'on peut extraire des motifs, des régularités, des caractéristiques, des métadonnées aussi bien de l'image, de la vidéo que du texte vu comme une chaîne de caractères, et que l'intégration de multiples standards et normes de métadonnées dans un format générique offre une description plus complète des contenus multimédias.

Pour la proposition de ce format de métadonnées, nous avons étudié des standards de métadonnées associées à chaque type de média, en essayant de les intégrer dans un même format extensible.

Un de nos principaux objectifs, (mis en évidence suite à un besoin explicite du projet LINDO) a été de créer un format de métadonnées pour la description de tout contenu multimédia, qui intègre les standards de métadonnées, mais qui prend aussi en compte la structure spatiale, temporelle et hypermédia du document.

Le format de métadonnées présenté dans ce chapitre a été proposé dans le cadre du projet LINDO, ce qui nous a imposé le choix des standards et des normes de métadonnées à prendre en compte. Pour l'intégration de ces standards, nous avons exploité les règles de mappage définies par le W3C Media Annotations Working Group, en établissant des règles de mappage entre les standards pris en compte et le format proposé.

Ce format n'est pas fermé, il peut évoluer à travers l'intégration d'autres standards et normes de métadonnées. Cette extension peut être réalisée par des mappages entre le nouveau standard considéré et le format proposé. Les éléments nouveaux qui ne sont pas présents dans notre format peuvent être ajoutés à l'élément media correspondant dans la Figure 12.

Par la suite, nous nous intéressons au deuxième volet de notre contribution : l'enrichissement des métadonnées à travers la prise en compte de l'usage.

Nous présentons la prise en compte de l'usage d'un système distribué de gestion de contenus multimédias dans le processus de création de métadonnées. Dans ce contexte distribué, nous utilisons le format de métadonnées présenté dans ce chapitre pour décrire les documents.

Dans la deuxième partie du chapitre suivant nous enrichissons le format de métadonnées pour prendre en compte les interactions de l'utilisateur avec : le système, avec les documents multimédias et les métadonnées. Pour exemplifier cette dernière contribution nous appliquons le format de métadonnées dans le contexte des ventes en ligne.

Cette application du modèle de métadonnées dans ces deux contextes différents (système distribué, commerce électronique) souligne l'extensibilité et l'adaptabilité du format.

Chapitre 5. La prise en compte de l'usage dans la gestion des métadonnées

5.1.	<i>Introduction</i>	91
5.2.	<i>La prise en compte de l'usage dans la création des métadonnées</i>	93
5.2.1.	L'indexation implicite et explicite des documents multimédias	93
5.2.2.	Le résumé des métadonnées	97
5.2.3.	La description adoptée pour un serveur distant	99
5.2.4.	L'algorithme de sélection des algorithmes d'indexation	101
5.2.5.	La prise en compte des conditions d'acquisition des contenus multimédias pour la sélection des algorithmes d'indexation	105
5.2.6.	La prise en compte de l'historique des requêtes des utilisateurs pour la sélection des algorithmes d'indexation	107
5.3.	<i>L'enrichissement des métadonnées par l'usage</i>	108
5.3.1.	L'enrichissement du format de métadonnées pour inclure l'usage	108
5.3.1.1.	Le concept de température	109
5.3.1.2.	Les métadonnées pour définir la structure d'usage	111
5.3.2.	Les structures dynamiques des documents multimédias	113
5.3.3.	L'algorithme d'enrichissement	114
5.3.3.1.	L'augmentation de la température	114
5.3.3.2.	Le refroidissement de la température	117
5.3.4.	L'application de l'algorithme d'augmentation de la température	119
5.3.4.1.	Les requêtes de l'utilisateur	120
5.3.4.2.	L'interaction de l'utilisateur avec les métadonnées	121
5.3.4.3.	L'interaction de l'utilisateur avec la liste des résultats	123
5.3.4.4.	L'interaction de l'utilisateur avec le document multimédia	124
5.3.4.5.	L'algorithme général pour le calcul de la température	127
5.3.5.	Un exemple d'évolution de la température	130
5.3.6.	L'utilisation de la température et de la structure d'usage	133
5.4.	<i>Bilan</i>	135

5.1. Introduction

La gestion efficace des collections de documents multimédias est un problème récurrent qui est traité par beaucoup de systèmes dans l'état de l'art, comme nous l'avons vu dans le Chapitre 2. La majorité de ces systèmes se focalisent principalement sur la gestion distribuée des contenus et on peut noter que l'optimisation de la consommation des ressources du système n'est pas un de leurs objectifs. Nous considérons que la réduction de la consommation des ressources est une problématique très importante surtout dans un contexte de mise à l'échelle des systèmes de gestion de documents multimédias. Dans ce contexte de mise à l'échelle des systèmes, plusieurs problèmes peuvent être identifiés :

- *La lenteur de traitement d'une requête* : le traitement d'une requête sur la totalité des métadonnées du système a de fortes chances de surcharger le système, surtout lors du traitement de requêtes complexes et lorsque plusieurs requêtes sont traitées simultanément.

- *La surcharge de la bande passante* : tous les contenus multimédias ou bien toutes les métadonnées doivent être transférés par le réseau pour être traités.
- *La surcharge du système* : l'exécution de tous les algorithmes d'indexation disponibles dans le système sur tous les contenus multimédias génère d'un côté, beaucoup de métadonnées qui peuvent ne jamais être utilisées et qui peuvent aussi ralentir l'exécution des requêtes, et d'un autre côté, la consommation de beaucoup de ressources (CPU, mémoire RAM, stockage).
- *Le non-respect des droits sur les données* : certains documents multimédias et métadonnées ne doivent pas être stockés sur le serveur central pour des raisons, par exemple, de vie privée.

Tous ces problèmes sont directement liés à la stratégie d'indexation des contenus multimédias choisie par le système. Effectivement, ce processus est celui qui consomme le plus de ressources. Pour trouver une réponse à tous ces problèmes, un compromis doit être trouvé entre la consommation des ressources, le niveau de détails des métadonnées extraites des contenus multimédias et la précision du système.

En étudiant plus en détails le Chapitre 3 de l'état de l'art, on se rend compte que l'utilisateur occupe une place importante dans le système de gestion des documents multimédias, en interagissant constamment avec le système, avec les documents et avec les métadonnées associées à ces documents. Son comportement est analysé pour extraire de l'information supplémentaire sur le document et sur son intérêt pour le document. Beaucoup de travaux se focalisent sur la prise en compte de l'utilisateur dans le processus de gestion des documents multimédias, à différents niveaux, dans le cadre du processus du Retour de Pertinence.

Nous considérons qu'il est important d'intégrer l'utilisateur dans le cadre du processus d'indexation, et d'adapter l'indexation à ses besoins spécifiques, tout en limitant la consommation des ressources.

Dans un premier temps, nous nous intéressons à la création des métadonnées dans le cadre des systèmes distribués de gestion de contenus multimédias, tout en ayant à l'esprit la limitation de la consommation des ressources. Pour atteindre ce dernier but, nous proposons la réalisation de l'indexation en deux temps, au moment de l'intégration des contenus dans le système, avec un ensemble d'algorithmes d'indexation différents sur chaque serveur du système établis en fonction du domaine et de l'utilisation de chaque serveur ; une deuxième fois au moment de la requête de l'utilisateur si besoin (cf. section 5.2.1). Cette deuxième indexation ne s'exécute pas sur toute la collection de documents et seulement quelques algorithmes d'indexation sont employés. Une étape de sélection des algorithmes est réalisée en fonction de la requête de l'utilisateur et des caractéristiques de chaque serveur.

Dans la section 5.2.3 , nous présentons les descriptions des serveurs sur lesquels se base la sélection des algorithmes. Cette sélection des serveurs et des algorithmes s'appuie également sur les métadonnées qui ont déjà été extraites sur les serveurs distants. Pour éviter la transmission de toutes ces métadonnées sur le réseau, nous proposons dans la section 5.2.2 la création d'un résumé de métadonnées. Puis nous présentons l'algorithme de sélection des algorithmes d'indexation en fonction des requêtes des utilisateurs.

Dans la section 5.2.5 nous présentons une amélioration de cet algorithme en prenant en compte les conditions d'acquisition des contenus multimédias qui doivent être indexés. Tout au long de l'utilisation du système, des besoins spécifiques qui émergent. Dans la section 5.2.6 nous présentons la prise en compte de l'historique des requêtes des utilisateurs pour mettre à jour la liste des algorithmes implicites.

D'une manière plus classique, dans l'état de l'art, l'usage du système et les interactions de l'utilisateur avec les documents sont utilisés dans le cadre du RP. Dans notre vision, le comportement de l'utilisateur au moment de l'usage du système, des documents et des métadonnées peut être employé pour enrichir les métadonnées.

Comme nous avons vu dans le Chapitre 3, les indicateurs de l'intérêt de l'utilisateur pour le document le plus utilisé est constitué par les clics et les requêtes que l'utilisateur exécute pour retrouver l'information qui répond à son besoin. Ces indicateurs sont employés pour faire surtout du reclassement des résultats, de l'enrichissement et de la reformulation de requête, ainsi que du filtrage et de la recommandation de ressources. Mais il n'y a pas de travaux qui utilisent ces indicateurs pour enrichir les métadonnées associées au document multimédia.

Nous nous intéressons à cette prise en compte du modèle d'interaction de l'utilisateur avec le système, le document et les métadonnées dans l'enrichissement des métadonnées.

Le format de métadonnées tel qu'il a été défini dans le Chapitre 4, ne prend pas en compte l'utilisateur et ses interactions avec le système de gestion des documents multimédias, avec le document et avec les métadonnées associées à ce document. Parmi les différentes façons possibles de prise en compte de l'usage, nous proposons et étudions l'utilisation de la température, par analogie à la mesure physique. Pour intégrer l'utilisateur, nous avons étendu, dans la section 5.3.1, le modèle de métadonnées proposé de deux façons :

- En introduisant le concept de *température* à chaque élément de métadonnées ;
- En intégrant une *structure d'usage* en tant que nouveau descripteur.

Dans la section 5.3.3 de ce chapitre, nous présentons en détail les algorithmes de calcul de la température, dans le temps, en fonction des interactions des utilisateurs avec le document et ses métadonnées. L'application de ces algorithmes est détaillée dans la section 5.3.4. La dernière partie de ce chapitre est dédiée à l'utilisation de la température pour réaliser le reclassement des résultats d'une recherche.

5.2. La prise en compte de l'usage dans la création des métadonnées

L'étude des systèmes de gestion des documents multimédias (Chapitre 2) et de la prise en compte des interactions des utilisateurs (Chapitre 3) montre que les métadonnées sont créées de manière indépendante de l'usage qu'on peut faire du système. Les mêmes types de métadonnées sont produits tout au long de l'existence du système, sans aucune adaptation aux besoins de l'utilisateur. Les systèmes existants se focalisent surtout sur les performances du système en termes de rappel/précision. Mais, en ayant toujours la même stratégie d'indexation, à un moment donné, l'utilisateur peut ne pas retrouver l'information dont il a besoin si celle-ci ne fait pas partie des cas envisagés par le système. Dans notre vision, nous proposons d'adapter l'indexation des documents multimédias au besoin de l'utilisateur, tout en ayant à l'esprit la réduction de la consommation des ressources du système.

5.2.1. L'indexation implicite et explicite des documents multimédias

En analysant tous les processus qui ont lieu dans un système de gestion de contenus multimédias, qu'il soit distribué ou non, on observe que le processus d'indexation des contenus multimédias est celui qui consomme le plus de ressources (e.g., CPU et RAM). Donc, si nous voulons réduire

cette consommation, nous devons agir sur celui-ci, tout en prenant en compte l'utilisateur et ses besoins.

Afin de limiter cette consommation des ressources, nous avons proposé de réaliser l'indexation en deux temps : à l'acquisition du contenu – i.e., *l'indexation implicite* – et à la demande, au moment de l'exécution d'une requête de l'utilisateur – i.e., *l'indexation explicite*, (Brut, Codreanu, Manzat, et al., 2011b). En effet, cela évite l'exécution de tous les algorithmes d'indexation possibles à la fois, et offre la possibilité de réaliser une indexation supplémentaire adaptée au besoin courant de l'utilisateur. Nous proposons d'exécuter les algorithmes d'indexation qui extraient des informations générales des contenus multimédias au moment de l'acquisition des contenus et les algorithmes qui extraient des informations spécifiques au moment de l'indexation explicite. Nous avons proposé un algorithme pour réaliser la sélection des algorithmes d'indexation explicites, que nous présentons plus tard, dans la section 5.2.4.

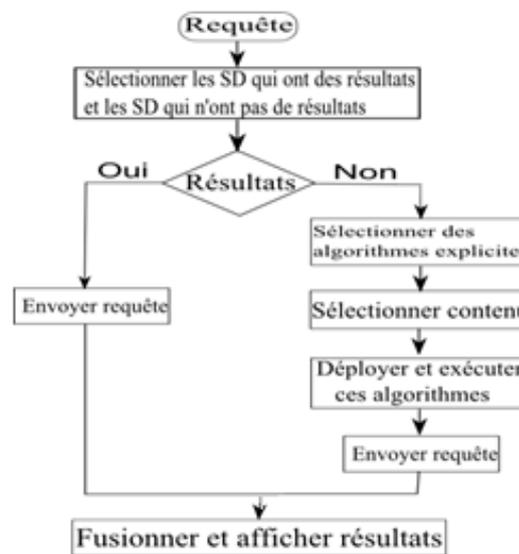


Figure 21 – Le diagramme du traitement de la requête

Cette technique d'indexation marche bien pour les systèmes distribués qui ont une indexation identique sur chaque serveur, mais aussi et surtout pour des systèmes qui adoptent une indexation différenciée sur chaque serveur.

Nous avons appliqué cette approche à l'architecture proposée dans le cadre du projet LINDO présentée dans la section 6.2 (Brut, Codreanu, Dumitrescu, et al., 2011). Le diagramme du processus d'indexation adoptée dans le cadre de ce projet est présenté dans la Figure 21. La Figure 22 présente le diagramme de séquence pour les processus d'indexation et de recherche, en mettant en avant les deux types d'indexation.

Quand un nouveau contenu multimédia est ingéré par un serveur distant, il est stocké par le module Storage Manager (SM) du serveur. Ensuite, le module Feature Extractors Manager du serveur distant (FEMrs) démarre le processus d'indexation implicite (étapes 2 à 19). Ce procédé consiste en l'exécution d'un ensemble prédéfini d'algorithmes d'indexation, c'est à dire, des algorithmes d'indexation implicites (iIA), sur le contenu multimédia acquis dans le but d'extraire des métadonnées. A cet effet, le FEMrs récupère le contenu multimédia du module SM à travers le module Access Module (AM). Principalement en fonction des types de médias et du contexte d'acquisition, le FEMrs sélectionne et exécute les algorithmes d'indexation implicites (étapes 8 et 9). Une fois l'exécution d'un algorithme d'indexation achevée, les métadonnées obtenues sont transmises au module de filtrage et

sont stockées par le Metadata Engine du serveur distant (MDers) dans la collection de métadonnées. Chaque fois que la collection de métadonnées est modifiée, le MDers envoie une version concise des métadonnées ajoutées au module de traduction sur le serveur central. Cette version concise des métadonnées, que nous appelons *résumé*, peut être obtenue en utilisant l'algorithme que nous avons proposé dans la section 5.2.2 (Laborie et al., 2009a). Une fois la traduction réalisée, les métadonnées sont finalement envoyées au Metadata Engine du serveur central (MDEcs) afin d'être stockées et utilisées pour le processus d'interrogation.

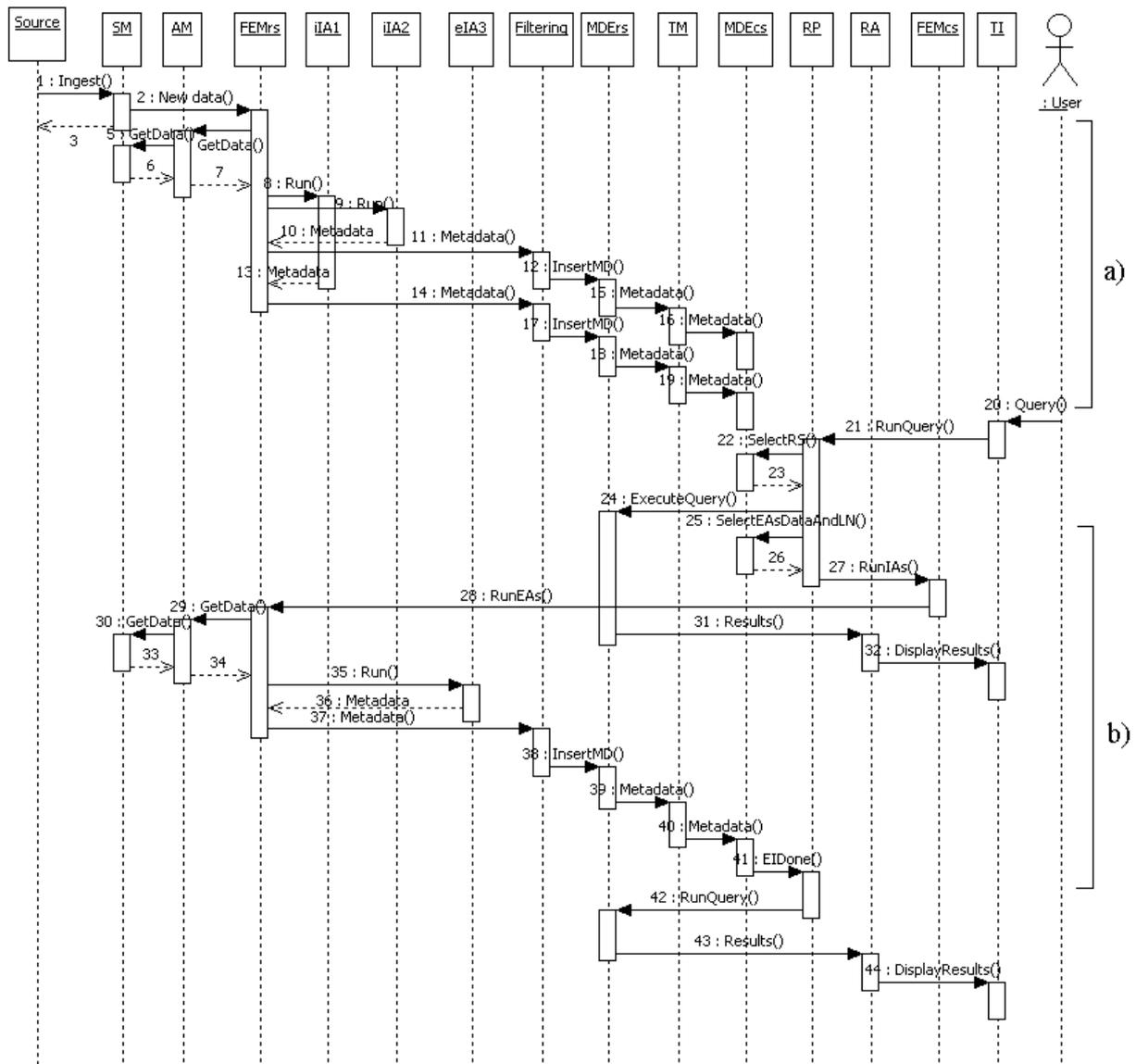


Figure 22 – Le diagramme de séquence pour : a) l'indexation implicite et b) l'indexation explicite

Le processus d'interrogation commence par la spécification de la requête par l'utilisateur en utilisant le Terminal Interface (TI). L'utilisateur formule la requête par le biais d'une interface graphique qui lui permet de préciser cinq éléments de la requête :

- la requête elle-même : texte libre ;
- l'emplacement : texte libre ;

- une période de temps : calendrier ;
- le domaine : liste case à cocher ;
- le type du média recherché : vidéo, image, audio ou texte.

Le texte de la requête est analysé afin d'obtenir un ensemble de caractéristiques (Brut, Co-dreanu, Dumitrescu, et al., 2011). Le traitement de la requête est réalisé en utilisant la similarité JCN (Jiang and Conrath, 1997) pour obtenir les synonymes les plus appropriés des mots de la requête et WordNet⁴⁵ pour traiter les adverbes et les adjectifs. En se basant sur l'information stockée dans le MDEcs (les résumés des métadonnées, les descriptions des serveurs distants et les descriptions des algorithmes d'indexation installés sur chaque serveur distant) le Request Processor (RP) sélectionne les serveurs distants qui peuvent avoir des résultats pour la requête. Le RP envoie la requête aux MDErs des serveurs sélectionnés (étapes 22 à 24), où la requête est exécutée et les résultats sont transférés au module Results Aggregator (RA) pour être classés et, enfin, envoyés à la TI pour l'affichage (31 étapes et 32).

À cette étape du processus d'interrogation, il est possible que seulement une partie des serveurs distants soit sélectionnée pour exécuter la requête. Cela ne signifie pas nécessairement qu'ils n'ont pas de résultat pour la requête de l'utilisateur. Il est possible que leurs contenus multimédias ne soient pas indexés avec les bons algorithmes d'indexation. Pour cette raison, le module RP sélectionne :

- les serveurs distants qui doivent être réindexés,
- les algorithmes d'indexation supplémentaires, i.e., des algorithmes d'indexation explicites (eIA) qui peuvent être déployés sur ces serveurs distants afin de produire des métadonnées supplémentaires qui pourraient fournir des réponses à la requête de l'utilisateur (étape 25)
- les contenus multimédias sur lesquels les algorithmes d'indexation doivent être exécutés.

L'utilisateur est informé que l'indexation supplémentaire est effectuée en vue de récupérer d'autres résultats à sa requête et qu'il sera averti lorsque les résultats sont disponibles.

Toutes ces informations sont fournies au FEMcs qui est en charge du déploiement de ces algorithmes explicites sur les serveurs distants concernés, si nécessaire (étape 28). Sur chaque serveur distant concerné par ce processus d'indexation explicite, le FEMrs prend le contrôle de l'indexation (étapes 29, 30, et 33 à 40). Pour pouvoir à tout moment savoir avec précision quel algorithme d'indexation a été exécuté sur les différents serveurs distants, le FEMrs indique au Service Description Controller du serveur distant (SDCr), les algorithmes explicites qu'il est en train d'exécuter. Le SDCr, à son tour, indique au Service Description Controller du serveur central (SDCc), ce changement dans sa description (plus précisément la liste des algorithmes d'indexation exécutés). Lorsque l'indexation explicite est terminée, le MDEcs informe le RP que d'autres résultats sont disponibles (étape 41). A ce moment là, le module RP envoie la requête de l'utilisateur aux MDErs des serveurs distants, où l'indexation explicite a été accomplie. Les résultats obtenus sont envoyés au module RA qui informe l'utilisateur que les résultats complémentaires sont disponibles en réponse à sa requête (étapes 43, 44).

A tout moment, l'utilisateur a le contrôle sur le système, dans le sens où il est informé du fait qu'une indexation supplémentaire est nécessaire et que cela risque de prendre du temps. L'utilisateur peut donc décider s'il attend les nouveaux résultats ou s'il se contente des réponses qu'il a déjà reçues.

⁴⁵ <http://wordnet.princeton.edu/>

Dans le cadre de ce processus d'indexation, plusieurs sélections sont réalisées : la sélection des serveurs distants sur lesquels l'indexation s'effectue ; la sélection des algorithmes d'indexation explicites ; la sélection des contenus multimédias à indexer. Ces sélections s'appuient sur la description des serveurs distants et sur le résumé de métadonnées remonté par les serveurs distants vers le serveur central. Par la suite, nous présentons nos propositions pour la description des serveurs et le résumé des métadonnées.

5.2.2. Le résumé des métadonnées

Pour pouvoir réduire les communications et les transferts de données sur le réseau, une étape de sélection de serveurs distants dans le processus d'interrogation est nécessaire. Cette sélection accomplie au niveau du serveur central se base sur les métadonnées extraites à partir des documents multimédias. Mais il n'est pas envisageable d'envoyer toutes les métadonnées vers le serveur central de par la capacité de stockage, le nombre de communications sur le réseau pour l'envoi de ces métadonnées et les restrictions d'accès aux contenus multimédias. Pour pallier tous ces problèmes, nous proposons d'envoyer une partie des métadonnées vers le serveur central que nous avons nommé « résumé »(Laborie et al., 2009a) (Laborie et al., 2009b).

Toutefois, à partir d'une collection de métadonnées, il est possible d'engendrer de multiples résumés. De plus, ces résumés sont des parties de collection qui peuvent varier en fonction de la précision que l'on souhaite lui donner. La définition que nous avons adoptée pour le résumé de métadonnées est la suivante.

Définition 1 : Le *résumé de métadonnées* est une extraction d'une collection de métadonnées dont le degré de granularité peut varier en fonction de la précision/du rappel que l'on souhaite lui donner.

En vue de sélectionner des parties de collection de métadonnées, nous proposons de définir la notion de degré de résumé, ou bien de granularité d'un résumé, telle que :

- degré = 0 signifie qu'aucune information n'est sélectionnée.
- $0 < \text{degré} < n$ signifie qu'une partie de l'information contenue dans une collection de métadonnées est sélectionnée. L'importance de cette sélection dépend de la valeur du degré. Dans notre cas, plus le degré sera faible, plus le résumé sera concis.
- degré = n signifie que l'ensemble de la collection de métadonnées est sélectionné, c'est-à-dire que l'on préserve l'ensemble des informations de la collection.

Pour illustrer cette définition du résumé, nous prenons l'exemple suivant. Sur un serveur quelconque de l'architecture, nous avons sauvegardé des contenus multimédias qui sont indexés avec un certain nombre d'algorithmes d'indexation. Les métadonnées obtenues suite à cette indexation (qu'elle soit implicite ou explicite) sont concaténées dans le module MDErs sous la forme d'un graphe RDF. Un extrait de ce graphe est présenté dans la Figure 23. Cet extrait contient des informations sur deux contenus (un audio et une vidéo). Dans le contenu vidéo, les algorithmes d'indexation ont identifié une personne blonde et un véhicule bleu qui a la plaque minéralogique « 1234 AB 56 ». En utilisant cette information, depuis d'autres sources d'informations extérieures, nous pouvons obtenir le nom du propriétaire de la voiture. En outre, depuis la description du serveur nous savons que celui-ci est localisé à Paris.

Pour résumer cette description, nous devons nous focaliser sur l'URI du Serveur₁ et sélectionner les informations environnantes qui lui sont liées. En effet, l'URI du Serveur₁ est le principal sujet de notre résumé, sélectionner ses informations environnantes ne fera qu'accroître sa description. De plus, à l'aide de la représentation sous forme de graphe d'une description RDF, il est possible de constater que plus l'information est éloignée de l'URI du serveur, plus on dispose de détails, pas seulement sur les données du serveur lui-même, mais aussi sur les descriptions qui lui sont associées.

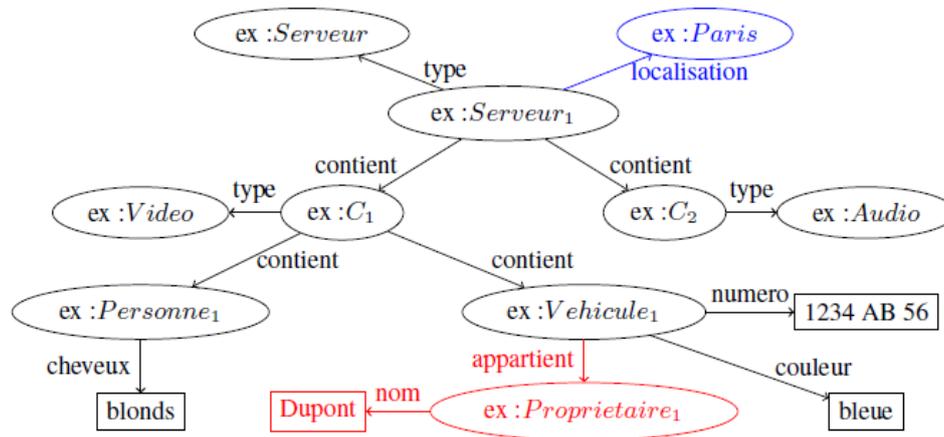


Figure 23 – Extrait d'une description RDF des métadonnées associées à des contenus multimédias stockés sur un serveur (Serveur₁)

À partir de cette constatation, nous proposons de sélectionner, à partir de l'URI du serveur toutes les métadonnées présentes sur un chemin d'une certaine longueur. Dans ce cadre, la longueur de ce chemin correspondra à la notion de degré du résumé. Par conséquent, lorsque l'on applique cette méthode sur la description RDF de la Figure 23 :

- Si degré = 0, le résumé contient seulement l'URI du Serveur₁, et aucune information sur ce serveur.
- Si $0 < \text{degré} < 4$, le résumé contient l'URI du Serveur₁ ainsi qu'un sous graphe de la Figure 23. Par exemple, si degré = 2, le résumé spécifiera que le Serveur₁ contient deux données multimédias : un contenu sonore et une vidéo dans laquelle des personnes ainsi que des véhicules ont été identifiés (cf., Figure 24).
- Si degré = 4, le résumé correspond strictement à la description de la Figure 23, c'est-à-dire qu'elle contient l'ensemble des informations de la collection de métadonnées.

Ce résumé peut être créé en utilisant des stratégies différentes, en fonction du domaine d'application (e.g., pour la vidéosurveillance, le résumé peut consister en des statistiques basées sur les métadonnées obtenues dans la dernière heure de l'enregistrement, pour le broadcast, le résumé peut être composé des titres et des participants pour chaque contenu (article, exposition, etc.)).

Dans le cadre de l'architecture LINDO, les résumés de métadonnées des serveurs distants aident le serveur central à avoir une vision globale du système, sans pour autant en détenir toutes les informations. Cette vision globale est utilisée dans l'exécution de la requête de l'utilisateur pour sélectionner les serveurs distants qui pourraient fournir des réponses mais aussi dans la sélection des algorithmes d'indexation explicites.

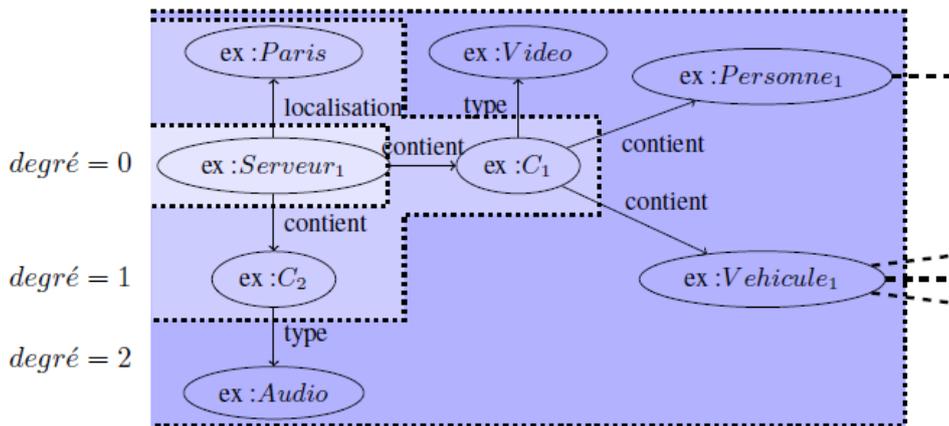


Figure 24 – Une description RDF résumée de la Figure 23 avec $0 < \text{degré} \leq 2$

5.2.3. La description adoptée pour un serveur distant

Pour la description des serveurs distants, nous avons voulu fournir une description générique, qui puisse décrire tout serveur distant du système du point de vue de sa configuration physique, mais aussi logique. Chaque serveur étant capable de stocker des contenus multimédias acquis en temps réel, une autre caractéristique importante qui doit être incluse dans la description du serveur est la liste des capteurs (qu'ils soient microphones, cameras ou capteurs de température, présence, etc.) utilisés pour acquérir les contenus.

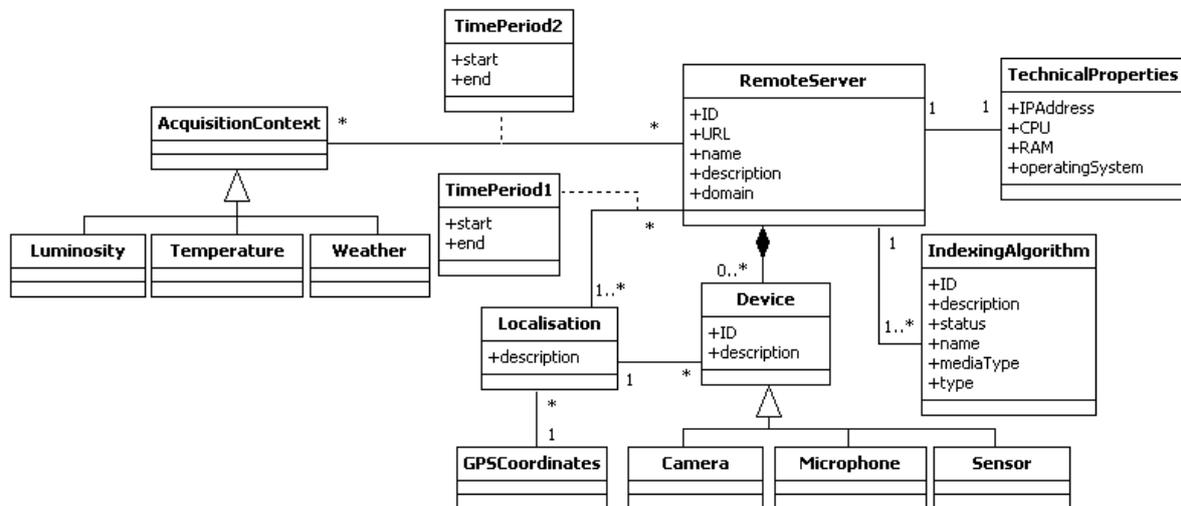


Figure 25 – Le modèle de description des serveurs distants

Chaque serveur distant a associé une description qui capture ses caractéristiques physiques, une description de son contexte (par exemple, son domaine, son emplacement) ainsi que les algorithmes d'indexation installés. Une partie importante de cette description est dédiée à la modélisation du contexte d'acquisition des données multimédias. Ainsi, la description du serveur distant présentée dans la Figure 25, contient les classes suivantes :

- La classe **RemoteServer** : contient des informations générales sur le serveur (e.g., son nom, sa description, son domaine)

- La classe **TechnicalProperties** : contient des éléments qui décrivent le serveur du point de vue technique (e.g., le système d'exploitation, le CPU). Ces éléments sont importants au moment de la sélection des algorithmes d'indexation à installer sur le serveur, car certains algorithmes ont besoin de certaines conditions d'exécution particulières (e.g., il y a des algorithmes qui ne sont conçus que pour une certaine version de Linux) ;
- La classe **IndexingAlgorithm** : décrit un algorithme d'indexation qui a été exécuté sur le serveur. Cette description contient certaines informations qui se retrouvent également dans la description générale de l'algorithme (plus de détails sur cette description se trouvent dans (Codreanu et al., 2011)) : son nom, une courte description textuelle, le type de media qu'il traite, son statut (s'il est démarré ou pas), son type (implicite ou explicite). Un serveur distant peut avoir plusieurs algorithmes d'indexation installés, c'est pour cela que sa description contient plusieurs classes `IndexingAlgorithm` ;
- La classe **Device** : décrit tout type de dispositif qui peut être utilisé pour acquérir du contenu multimédia stocké et traité sur le serveur distant. Dans notre cas nous avons considéré des cameras, des microphones et des capteurs (e.g., de température, d'humidité). Chaque dispositif a une description textuelle associée. En outre, un dispositif peut être fixe ou mobile, et pour modéliser cette propriété, nous avons associé au Device une localisation. Cet élément n'est pas obligatoire pour la description du serveur distant, car il est possible d'avoir un serveur distant qui gère des contenus multimédias générés par un logiciel de montage ;
- La classe **Localisation** : contient les informations sur la localisation à un certain moment. Cette localisation contient une description textuelle et les coordonnées GPS de l'endroit qui est décrit. Un serveur peut être situé à des endroits différents au fil du temps, s'il se trouve dans un bus, dans le contexte d'une application de vidéosurveillance, mais il peut également être situé à un emplacement fixe. La localisation est également associée à un dispositif, parce que nous pouvons considérer que l'appareil est localisé à un certain lieu, et il est orienté vers une certaine direction. La localisation d'un dispositif peut aussi varier dans le temps ;
- La classe **AcquisitionContext** : décrit le contexte d'acquisition du contenu multimédia stocké sur le serveur, à un certain moment. Un lien peut être créé entre le contexte d'acquisition et le dispositif pour mettre en évidence le fait que seulement le contenu généré par ce dispositif est influencé par ces conditions environnementales. Les principales caractéristiques qui sont prises en compte sont les conditions météorologiques, les conditions de luminosité et la température, mais d'autres conditions environnementales peuvent être prises en compte en fonction des capteurs disponibles et du domaine d'application.

Le module SDCrs de chaque serveur distant gère la description du serveur en la mettant à jour chaque fois qu'il y a un changement dans la configuration du serveur (e.g., un nouveau dispositif d'acquisition est ajouté au serveur, un nouvel algorithme d'indexation est installé). Le module SDCcs situé sur le serveur central centralise tous les descriptions des serveurs distants.

Un exemple d'une description est fourni dans le Tableau 17. Cette description est associée à un serveur distant situé dans la gare d'Austerlitz à Paris qui gère les cameras de vidéosurveillance situées dans le parking, en face de la gare. Ce serveur gère les vidéos acquises par une camera qui se trouve dans le coin nord du parking. Dans cet exemple, il s'agit d'un serveur fixe, et c'est pour cela que les éléments de localisation n'ont pas de période de temps associée. En ce qui concerne les conditions d'acquisition du contenu, nous avons des informations sur le temps qu'il fait, si le ciel est nuageux ou ensoleillé pour certaines périodes de temps. Un autre élément de contexte pris en compte est

la luminosité ambiante, qui peut être différente à certains moments de la journée. Sur ce serveur, un algorithme d'indexation implicite est installé et il est démarré.

Tableau 17 – Un extrait de la description d'un serveur distant

```

<RemoteServer id="rs1" name="RServer-1">
  <localisation>
    <description>parking de la gare d'Austerlitz, Paris, France</description>
  </localisation>
  <description>Gère le contenu acquis par les caméras situées dans le parking en face de la gare.
</description>
  <devices>
    <camera id="c1Paris">
      <description>localisée dans le coin nord du parking</description>
    </camera>
  </devices>
  <acquisitionContext>
    <weather>
      <timePeriod start="2011-07-14T08:07:00" end="2011-07-14T11:33:00">nuageux</timePeriod>
      <timePeriod start="2011-07-14T11:34:00" end="2011-07-14T19:14:00">ensoleillé</timePeriod>
    </weather>
    <luminosity>
      <timePeriod start="2011-07-14T08:07:00" end="2011-07-14T11:33:00">75</timePeriod>
      <timePeriod start="2011-07-14T11:34:00" end="2011-07-14T19:14:00">100</timePeriod>
    </luminosity>
  </acquisitionContext>
  <indexingAlgorithms>
    <indexingAlgorithm id="ia2rs1" mediaType="video" name="détection de personne"
type="implicit" status="started">
      <description>Détection des personnes à l'extérieur et leur couleur prédominante. </description>
    </indexingAlgorithm>
  </indexingAlgorithms>
</RemoteServer>

```

5.2.4. L'algorithme de sélection des algorithmes d'indexation

Nous avons défini un algorithme de sélection des algorithmes d'indexation à utiliser en fonction d'une requête de l'utilisateur.

Cet algorithme de sélection est basé sur une modélisation uniforme de la requête, des métadonnées multimédias et des descriptions des algorithmes d'indexation. Cette uniformité est acquise principalement en considérant les caractéristiques multimédias présentes dans tous ces modèles. Plus précisément :

- une requête est considérée comme une liste de caractéristiques qui doivent être récupérées. Ces caractéristiques sont extraites à partir du texte de la requête. Par exemple, la requête « *Trouve les vidéos contenant une personne avec un sac à dos* » est représentée par les caractéristiques : *vidéo, personne, sac à dos*.
- les métadonnées contiennent une liste des caractéristiques présentes dans un contenu multimédia. Par exemple, une image qui contient une personne avec un sac à dos rouge, est représentée par les caractéristiques : *image, personne, sac à dos*. Dans le modèle de

métadonnées que nous avons proposé dans le Chapitre 4, ces caractéristiques peuvent être identifiées à travers les noms des classes du modèle, mais aussi dans l'attribut type de la classe Object.

- un algorithme d'indexation extrait une liste de caractéristiques. Ces caractéristiques sont extraites à partir de la description textuelle qui se trouve dans l'élément MetadataObjectDescription de la description de l'algorithme proposé par (Codreanu et al., 2011).

Répondre à une requête de l'utilisateur consiste à retrouver les contenus multimédias en interrogeant la collection de métadonnées pour retrouver les caractéristiques demandées. En sachant que ces métadonnées/caractéristiques ont été extraites à partir du contenu multimédia par des algorithmes d'indexation, nous proposons de déterminer l'ensemble des algorithmes d'indexation nécessaires pour répondre à une requête en faisant la correspondance entre les caractéristiques demandées par la requête et celles extraites par les algorithmes d'indexation.

Considérons une requête Q , une liste d'algorithmes d'indexation L_A et une caractéristique f , telle que $f \in Q$. L'algorithme défini dans le Tableau 18 sélectionne, à partir de la liste L_A , un ensemble pertinent d'algorithmes d'indexation qui extrait la caractéristique f et autant d'autres caractéristiques que possible parmi celles requises par la requête Q . Cet algorithme est exécuté pour chaque serveur sur lequel l'indexation doit être effectuée. De cette façon, pour chaque serveur, la liste d'algorithmes d'indexation obtenue peut être différente en fonction des caractéristiques physiques et logicielles de chaque serveur.

Au début, la liste des algorithmes d'indexation L_A contient tous les algorithmes disponibles au sein du système. La caractéristique f est sélectionnée dans Q en fonction du nombre maximum d'algorithmes d'indexation contenus dans L_A qui extraient f . Lors de l'appel récursif de l'algorithme 1 la liste d'algorithmes L_A sera transformée dans une nouvelle liste L .

Pendant l'exécution de cet algorithme, la liste L_A est affinée pour ne garder que les algorithmes qui extraient la caractéristique f . Par la suite, cette liste est affinée encore plus, pour ne garder que les algorithmes qui extraient le plus de caractéristiques supplémentaires de la requête. Les caractéristiques qui pourraient être identifiées par tous algorithmes d'indexation figurant dans la liste L_A sont marquées. Ainsi, à un moment donné, seules les caractéristiques qui n'ont pas déjà été identifiées par des algorithmes d'indexation sont prises en considération.

Au moment de la sélection des algorithmes qui extraient la caractéristique f , si plusieurs algorithmes d'indexation peuvent être appliqués pour l'extraire, la méthode *selectOneAlgorithm* est utilisée pour sélectionner un seul algorithme, évitant ainsi l'application d'algorithmes d'indexation multiples pour déterminer la même caractéristique. Cette sélection est réalisée en fonction des caractéristiques physiques et logicielles du serveur. Dans la description de chaque serveur nous retrouvons les conditions dans lesquelles l'algorithme peut être exécuté (e.g., le système d'exploitation, le CPU, la mémoire). En faisant une correspondance entre ces contraintes et les propriétés du serveur (qui se retrouvent dans la classe TechnicalProperties de sa description) les algorithmes qui respectent ces conditions sont les seuls sélectionnés. Du fait que plusieurs algorithmes d'indexation pourraient être exécutés sur un serveur, pour n'en sélectionner qu'un, d'autres conditions peuvent être envisagées, par exemple sur le temps d'exécution des algorithmes.

Après l'exécution de cet algorithme, une liste d'algorithmes d'indexation qui extraient un certain nombre de caractéristiques de la requête Q est obtenue. Néanmoins, cet algorithme se révèle incomplet, car la liste obtenue est connexe à la caractéristique f . Ainsi, si certaines caractéristiques ne sont pas marquées, il est possible que d'autres algorithmes d'indexation les identifient, en particulier des algorithmes qui ne considèrent pas la caractéristique f . Pour pallier cet inconvénient nous avons

proposé l'algorithme 2, présenté dans le Tableau 19, qui comprend de multiples appels de l'algorithme 1 de manière à couvrir le plus grand nombre possible des caractéristiques non marquées de la requête.

Tableau 18 – L'algorithme de sélection des algorithmes d'indexation

Algorithme 1 : indexingAlgorithmsSelection

Entrée : Une requête de l'utilisateur Q , une liste d'algorithmes d'indexation L_A , une caractéristique f , tel que $f \in Q$, un serveur distant S
Sortie : Une liste d'algorithmes d'indexation

$L_A \leftarrow$ the indexing algorithms of L_A which extract f ;
if $L_A = \emptyset$ **then**
 | mark f in Q ;
 | $L \leftarrow \emptyset$;
 | **foreach** f_i unmarked in Q **do**
 | | $L \leftarrow L \cup \text{indexingAlgorithmsSelection}(Q, L_A, f_i, S)$;
 | | **if** $L = \emptyset$ **then**
 | | | $e \leftarrow \text{selectOneAlgorithm}(L_A, S)$;
 | | | **return** $\{e\}$;
 | | **else return** L ;
else return \emptyset ;

Tableau 19 – L'algorithme complet de sélection des algorithmes d'indexation

Algorithme 2 : getAllIndexingAlgorithms

Entrée : Une requête de l'utilisateur Q , une liste d'algorithmes d'indexation L_A , un serveur distant S
Sortie : Une liste d'algorithmes d'indexation

$L \leftarrow \emptyset$;
foreach f_i unmarked in Q **do**
 | $L \leftarrow L \cup \text{indexingAlgorithmsSelection}(Q, L_A, f_i, S)$;
return L ;

A la fin de l'exécution de cet algorithme, on peut analyser la liste de caractéristiques multimédias de la requête pour voir si toutes sont marquées ou pas :

- si toutes les caractéristiques sont marquées, cela signifie que les algorithmes sélectionnés extraient tous, ces caractéristiques, et donc ils peuvent produire des métadonnées qui pourraient répondre à la requête ;
- si quelques caractéristiques ne sont pas marquées, cela signifie que la liste d'algorithmes ne peut pas extraire toutes les caractéristiques multimédias demandées par la requête, et donc que le système ne dispose pas des algorithmes suffisants pour répondre à la requête.

L'algorithme 2 détermine non seulement les algorithmes nécessaires pour extraire les caractéristiques demandées par une requête mais aussi, de manière indirecte, les caractéristiques qui ne sont pas couvertes par les algorithmes d'indexation disponibles dans le système. Il permet donc d'identifier des besoins émergents de l'utilisateur et d'adapter le système à ces besoins en ajoutant de nouveaux algorithmes d'indexation dans le système.

Pour illustrer le fonctionnement de cette sélection des algorithmes d'indexation, supposons que nous avons la liste d'algorithmes L_A qui extraient certaines caractéristiques :

- $A_1 = \{\text{vidéo, personne, couleur}\}$
- $A_2 = \{\text{vidéo, véhicule}\}$
- $A_3 = \{\text{vidéo, personne}\}$
- $A_4 = \{\text{vidéo, véhicule, plaque minéralogique}\}$
- $A_5 = \{\text{image, personne}\}$

Un utilisateur exécute la requête suivante : « recherche les vidéos qui contiennent une personne habillée en rouge et une voiture ». Suite au traitement de la requête, elle est représentée par les caractéristiques : $\{\text{vidéo, personne, couleur, voiture}\}$. L'arbre d'exécution de l'algorithme de sélection des algorithmes d'indexation est présenté dans la Figure 26.

La première étape dans l'exécution de l'algorithme est la sélection de la caractéristique de la requête qui est extraite par le plus grand nombre d'algorithmes de la liste L_A . C'est le cas de la caractéristique « vidéo », qui est sélectionnée et marquée. Tous les algorithmes qui l'extraient sont sélectionnés également. Dans notre cas, il s'agit des algorithmes $\{A_1, A_2, A_3, A_4\}$.

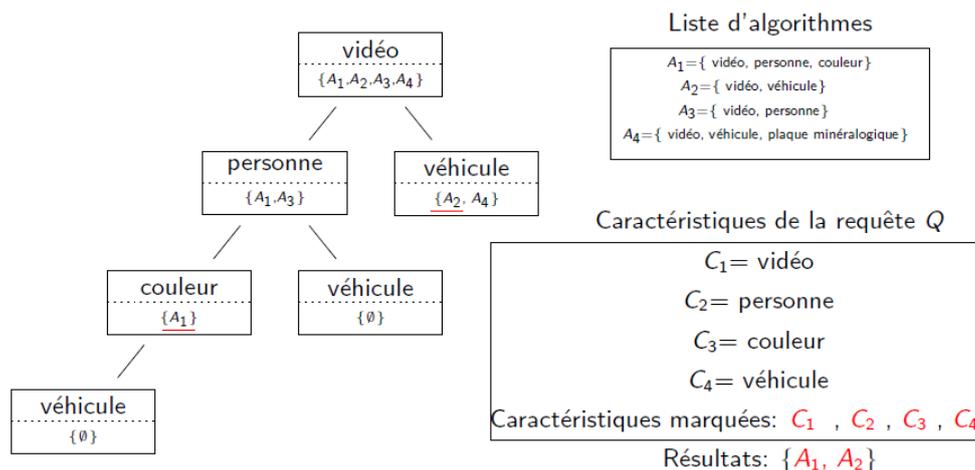


Figure 26 - L'arbre d'exécution de l'algorithme de sélection des algorithmes d'indexation

Ensuite, nous regardons parmi ces quatre algorithmes s'il y a un algorithme qui extrait aussi la caractéristique « personne ». C'est le cas des algorithmes $\{A_1, A_3\}$. Parce que la caractéristique « personne » est extraite par au moins un algorithme, elle est marquée.

Nous continuons l'exécution de l'algorithme par la sélection de la caractéristique « couleur ». Parmi les deux algorithmes sélectionnés à l'étape précédente, seul l'algorithme A_1 l'extrait aussi, donc l'algorithme est sélectionné et la caractéristique est marquée. Dans la liste de caractéristiques demandées par la requête, il reste seulement « véhicule ». Nous regardons si l'algorithme A_1 extrait aussi cette caractéristique. Ce n'est pas le cas, donc nous intégrons l'algorithme A_1 à la liste de résultats et nous revenons à la liste d'algorithmes sélectionnés à l'étape précédente pour voir si, dans cette liste, il y a un algorithme qui extrait cette caractéristique. Ce n'est pas le cas, donc on revient à la liste précédente. Dans cette liste, $\{A_1, A_2, A_3, A_4\}$, les algorithmes $\{A_2, A_4\}$ qui extraient cette caractéristique sont sélectionnés et la caractéristique est marquée. La liste de caractéristiques demandées par la requête ne contient plus de caractéristiques qui ne sont pas marquées, donc, pour finir l'exécution de l'algorithme, nous devons choisir un algorithme d'indexation. Parce que notre but prin-

cial est de réduire la consommation des ressources, nous ne voulons exécuter qu'un algorithme d'indexation. Plusieurs stratégies peuvent être mises en place. Dans cet exemple, nous avons choisi l'algorithme qui extrait le moins de caractéristiques supplémentaires par rapport à la requête : l'algorithme A_2 .

A la fin de l'exécution de l'algorithme nous obtenons la liste d'algorithmes d'indexation $\{A_1, A_2\}$ qui extraient toutes les caractéristiques de la requête.

Cet algorithme, tel qu'il a été décrit jusqu'à maintenant, ne prend en compte que les caractéristiques multimédias de la requête et du serveur sur lequel les algorithmes d'indexation vont être exécutés, mais il ne s'intéresse pas aux conditions d'acquisition des documents multimédias sur lesquels les algorithmes vont être exécutés. Les algorithmes d'indexation ont aussi des contraintes d'exécution liées au contenu multimédia, qui ont une grande influence sur la qualité de la détection qu'ils réalisent phrase pas claire. Dans la section suivante, nous présentons comment ces conditions sont prises en compte dans notre proposition.

5.2.5. La prise en compte des conditions d'acquisition des contenus multimédias pour la sélection des algorithmes d'indexation

Les algorithmes d'indexation ont des conditions d'exécution dans lesquelles les résultats obtenus sont optimaux. Ces conditions sont liées notamment à la qualité du contenu multimédia qui est indexé, aux conditions météorologiques (e.g., ensoleillé, nuageux), aux conditions de luminosité (e.g., la nuit, le jour, sombre, clair), au lieu (à l'intérieur ou à l'extérieur), à la langue du contenu multimédia (e.g., pour la détection de la parole).

Si le système dispose de plusieurs algorithmes d'indexation qui extraient les mêmes caractéristiques du contenu multimédia mais qui ont des conditions d'exécution différentes, pour obtenir des détections de qualité, le système doit pouvoir utiliser l'algorithme d'indexation qui obtient les meilleurs résultats en fonction de la qualité et des conditions d'acquisition du contenu multimédia. Ce choix des algorithmes d'indexation correspond à une adaptation de l'indexation au type du contenu multimédia et à ses caractéristiques. Cette adaptation doit se faire tout au long du processus d'indexation, qu'il s'agisse de l'indexation implicite ou explicite.

Notre proposition consiste dans l'application d'un algorithme qui détecte le changement des conditions d'acquisition ou d'autres caractéristiques d'un contenu multimédia, avant l'exécution des autres algorithmes d'indexation. La segmentation obtenue suite à l'application de cet algorithme, est utilisée par la suite, dans le choix des algorithmes d'indexation employés pour indexer chaque segment. L'application de cet algorithme de segmentation peut être assez coûteuse en terme de consommation de ressources et son exécution peut prendre beaucoup de temps. Dans le cas des systèmes dont l'acquisition du contenu multimédia s'effectue en temps réel, cette segmentation peut poser des problèmes. Nous proposons alors, l'utilisation d'autres moyens de détection du changement des conditions environnementales, par exemple à travers des capteurs. Les informations acquises par ces capteurs sont stockées sur le serveur avec une estampille temporelle qui va être utilisée pour faire la correspondance entre ces informations et le contenu multimédia. De cette façon, la segmentation du contenu est réalisée de manière indirecte. Cette approche ne peut être utilisée que si le système dispose pour chaque caractéristique multimédia, d'au moins deux algorithmes différents qui l'extraient, mais qui ont des conditions d'exécution différentes (e.g., pour la détection de personne à l'extérieur, le système doit détenir au moins un algorithme qui donne des détections optimales pendant la journée, et un autre qui donne des détections optimales pendant la nuit).

Nous avons appliqué notre proposition dans le cadre du projet LINDO. Dans le cadre de ce projet, l'adaptation de l'indexation implicite réalisée par chaque serveur distant et l'adaptation de l'indexation explicite réalisée par le serveur central sont basées sur les descriptions des serveurs distants.

Dans la description des serveurs adoptée par le système (présentée dans la section 5.2.3.), les changements des conditions environnementales, acquises par des capteurs associés à chaque serveur, sont sauvegardées dans l'élément `AcquisitionContext`. Au niveau du serveur distant, la description complète du serveur est gérée par le module `Service Description Controller`. Pour ne pas surcharger le réseau, seulement un résumé de la description de chaque serveur est envoyé au serveur central pour l'exploiter dans la sélection des algorithmes explicites.

Au moment de l'indexation implicite, le FEMrs regarde dans la description du serveur pour voir s'il y a eu un changement des conditions d'acquisitions. Si c'est le cas, le FEMrs remplace chaque algorithme d'indexation implicite qui détecte une certaine caractéristique (e.g., la présence d'une personne) avec un autre algorithme qui détecte la même caractéristique, mais qui a une meilleure performance dans les conditions actuelles.

Au moment de l'indexation explicite, suite à l'analyse de la requête, on obtient les caractéristiques multimédias supplémentaires à détecter, ainsi que les documents multimédias qui doivent être indexés (e.g., correspondant à une certaine période de temps et à un sous-ensemble de serveurs distants). Pour prendre en compte ces changements de conditions environnementales, nous étendons l'algorithme de sélection des algorithmes d'indexation présenté dans le Tableau 19 en modifiant la fonction *selectOneAlgorithm*. Cette nouvelle fonction, pour un serveur distant, recoupe le résumé de ses conditions environnementales avec les documents multimédias qui doivent être indexés pour obtenir pour chaque document, les conditions environnementales correspondantes (pour un document, on peut avoir plusieurs conditions d'acquisition si, par exemple, un changement de conditions météorologiques se produit pendant l'enregistrement du document). En prenant en compte cette correspondance document – conditions environnementales, la fonction *selectOneAlgorithm* sélectionne l'algorithme d'indexation qui correspond au mieux aux valeurs de ces conditions environnementales. Donc, pour un document multimédia, plusieurs algorithmes d'indexation peuvent être sélectionnés et exécutés par la suite.

Pour illustrer nos propositions, considérons l'exemple d'un système de vidéosurveillance d'une gare. Sur le serveur qui gère les contenus multimédias correspondant aux cameras vidéo situées dans le parking de la gare, nous avons installé les algorithmes d'indexation implicites qui détectent les caractéristiques suivantes : des personnes ; des voitures ; des plaques minéralogiques. Pour chacune de ces caractéristiques deux algorithmes sont installés dans le système : un qui obtient des performances meilleures pendant le jour et l'autre pendant la nuit. Cette différence entre le jour et la nuit est déterminée par un capteur de luminosité. Les informations sur la luminosité sont stockées dans la description du serveur. Dès que le FEMrs constate que la luminosité extérieure baisse en dessous d'un certain seuil, il modifie les algorithmes d'indexation pour prendre en compte ce changement.

Cette prise en compte des changements des conditions d'acquisition des contenus multimédias permet d'améliorer les performances du système quant à la précision des détections. Il est à noter que les besoins émergents des utilisateurs ne sont pas pris en compte. Pour pallier ce manque, nous proposons une solution pour prendre en compte ces besoins pour améliorer le temps de réponse pour une requête de l'utilisateur.

5.2.6. La prise en compte de l'historique des requêtes des utilisateurs pour la sélection des algorithmes d'indexation

Le fait de réaliser une partie de l'indexation au moment de l'exécution de la requête augmente le temps de réponse du système. Ce temps est d'autant plus important qu'il y a beaucoup de caractéristiques à extraire au moment de l'indexation explicite. Pour limiter ce temps d'indexation, nous proposons d'analyser l'historique des requêtes exécutées par les utilisateurs du système afin de déterminer quelles sont les caractéristiques multimédias les plus demandées pendant une période de temps Δt dans l'indexation explicite. Suite à cette analyse nous ajoutons, de manière temporaire, les algorithmes qui extraient ces caractéristiques à la liste des algorithmes implicites.

Dans ce cadre, nous associons un poids à chaque caractéristique multimédia qui est extraite dans le système, qu'elle soit extraite au moment de l'indexation implicite ou explicite. En analysant les requêtes des utilisateurs correspondant à un serveur distant pendant une certaine période de temps, il est possible que certaines caractéristiques apparaissent de manière récurrente dans ces requêtes. Le poids de chaque caractéristique augmente avec chaque requête qui la demande. Dans le cadre d'un système distribué, cette analyse est réalisée au niveau du serveur central pour chaque serveur distant. Ainsi, les poids des caractéristiques peuvent être différents sur chaque serveur parce que les serveurs sont différents en termes de contenus multimédias et donc les requêtes qui leur sont envoyées par le serveur central peuvent être différentes.

En analysant les résultats de ce calcul, on peut vérifier, pour chaque serveur distant :

- Si les algorithmes d'indexation utilisés pour l'indexation implicite ont été bien choisis : si le poids des caractéristiques multimédias extraites par ces algorithmes est petit, cela signifie que ces caractéristiques n'ont pas été trop demandées par les requêtes des utilisateurs. En analysant l'évolution de ces poids dans le temps, l'administrateur du système peut décider si une modification permanente de ces algorithmes est nécessaire ou non, pour mieux répondre aux besoins des utilisateurs.
- S'il y a besoin de nouveaux algorithmes d'indexation dans le système : si parmi les caractéristiques qui ont un poids important, il y a certaines qui ne sont extraites par aucun algorithme d'indexation, on peut penser qu'il s'agit d'un nouveau besoin des utilisateurs auquel le système ne peut pas répondre. Cette analyse permet d'améliorer le système en fonction des besoins des utilisateurs si l'administrateur considère que c'est nécessaire.

En attendant une action permanente de la part de l'administrateur, nous proposons de considérer, de manière temporaire, une caractéristique multimédia donnée dans le processus d'indexation implicite (plus précisément, l'algorithme qui la détecte) à partir du moment où son poids atteint un certain seuil. Dans le cas où plusieurs algorithmes peuvent détecter cette caractéristique multimédia, le choix de l'algorithme plus adapté au contexte d'exécution du moment est réalisé comme décrit dans la sous-section précédente. Au début de chaque période de temps (e.g., heure, jour, semaine), les poids de toutes les caractéristiques multimédias supplémentaires sont remis à zéro. Ceci évite que certaines caractéristiques temporaires deviennent permanentes.

Pour illustrer nos propositions, considérons l'exemple d'un système de vidéosurveillance d'une gare. Il est possible que pendant une semaine, les requêtes de plusieurs utilisateurs concernant un « sac arraché » soient effectuées et en conséquence, sur un serveur distant, l'algorithme qui détecte la caractéristique multimédia « sac arraché » soit exécuté de manière explicite. Par conséquent, le poids de cette caractéristique « sac arraché » augmente. Lorsque son poids atteint le seuil considéré, l'algorithme qui détecte cette caractéristique « sac arraché » devient implicite. Au début d'une nouvelle semaine, le poids de cette caractéristique est remis à zéro, tandis que l'algorithme reste implicite.

Si, pendant la semaine en cours, il n'y a que quelques requêtes concernant la caractéristique « sac arraché » qui sont exécutées, alors le poids de cette caractéristique restera bas et l'algorithme ne sera pas conservé parmi les algorithmes implicites pour la semaine suivante.

Jusqu'à maintenant nous n'avons considéré que les interactions que les utilisateurs ont avec le système de gestion de contenus multimédias. Nous avons utilisé les requêtes des utilisateurs pour adapter l'indexation des contenus multimédias – i.e., la création des métadonnées – aux besoins des utilisateurs. Par la suite, nous allons encore plus loin dans l'intégration de l'utilisateur dans la gestion de métadonnées à travers l'enrichissement des métadonnées en fonction des interactions de l'utilisateur avec le système, les documents multimédias et les métadonnées.

5.3. L'enrichissement des métadonnées par l'usage

5.3.1. L'enrichissement du format de métadonnées pour inclure l'usage

Nous considérons l'utilisateur comme une source importante de métadonnées implicites, car il peut produire des métadonnées en interagissant avec les documents multimédias.

Dans un système de gestion de documents multimédias, de nombreux algorithmes d'indexation différents sont appliqués. Ainsi, les métadonnées multimédias obtenues sont hétérogènes, des caractéristiques les plus simples, de bas niveau, aux caractéristiques sémantiques plus complexes, de haut niveau. Habituellement, toutes les métadonnées générées sont utilisées dans le processus de recherche. Or, il y a des métadonnées qui sont utilisées plus souvent que d'autres.

Pour faire ressortir cette utilisation différente des métadonnées, nous enrichissons les métadonnées obtenues après le processus d'indexation avec le concept de la *température*. Nous avons choisi ce concept, pour capturer l'évolution de l'intérêt que le document suscite chez les utilisateurs dans le temps, parce qu'il nous semble qu'il y a une vraie analogie avec le concept de température dans le domaine de la physique. Ainsi, plus les documents ou leurs métadonnées sont utilisés, plus ils sont chauds. Au moment où le document n'est plus utilisé il se refroidit, en dissipant sa température dans l'environnement.

L'usage que les utilisateurs font des documents fait aussi ressortir des liens entre ces documents qui ne peuvent pas être identifiés autrement. Par exemple, si un utilisateur utilise deux parties du même document pour la création d'un autre, en le copiant, alors on peut déduire qu'il y a un lien entre ces deux parties. Nous avons étendu le modèle de métadonnées présenté dans le Chapitre 4, en ajoutant ces liens d'usage (e.g. liens indirects) entre les différents éléments du modèle.

Dans le cadre de cet enrichissement des métadonnées, nous ne prenons pas comme hypothèse que tous les utilisateurs sont semblables, mais nous prenons en compte le regroupement des utilisateurs selon certains critères.

Dans notre travail, nous ne nous concentrons pas dans la mise en œuvre d'un algorithme de classification pour réaliser les groupes d'utilisateurs, ni sur la proposition d'un profil utilisateur. Dans notre équipe, des travaux ont été menés dans ce domaine (Zayani, 2008; Tchuente et al., 2012).

En se basant sur ce profil, les groupes d'utilisateurs peuvent être réalisés en fonction de plusieurs critères : l'âge, la position géographique ou encore en fonction des préférences et des intérêts. Le choix du meilleur critère se fait en fonction du domaine d'application. Dans notre approche, nous ne faisons aucune hypothèse sur les groupes d'utilisateurs. Les utilisateurs peuvent appartenir à un seul groupe, ou bien ils peuvent appartenir à plusieurs groupes.

Pour pouvoir mettre en place cet enrichissement des métadonnées, le système de gestion de documents multimédias doit enregistrer le plus d'information possible sur les activités de l'utilisateur et sur ses interactions avec les documents multimédias. Pour mettre en place la structure d'usage, l'historique des requêtes des utilisateurs est nécessaire. Un autre type d'indicateur qui est impératif est la succession des clics des utilisateurs sur la liste des résultats et sur les documents multimédias. En plus de ses indicateurs, d'autres éléments peuvent être employés parmi ceux présentés dans le Chapitre 3, mais leur utilisation n'est pas impérative, elle est juste souhaitable.

5.3.1.1. Le concept de température

Une première amélioration du modèle de métadonnées proposé est l'ajout de la température à chaque élément du modèle. La Figure 27 présente le modèle résultant de cet ajout.

La température correspond à une métadonnée d'usage, qui reflète la popularité d'un document ou d'un élément de métadonnée à un moment donné.

La température est définie comme une valeur réelle entre 0 et 100 (i.e., $t \in \mathbb{R}, 0 \leq t \leq 100$). La valeur initiale de toute température est 0.

Nous avons associé à chaque métadonnée deux caractéristiques de température :

- une température par groupe d'utilisateurs ;
- une température moyenne.

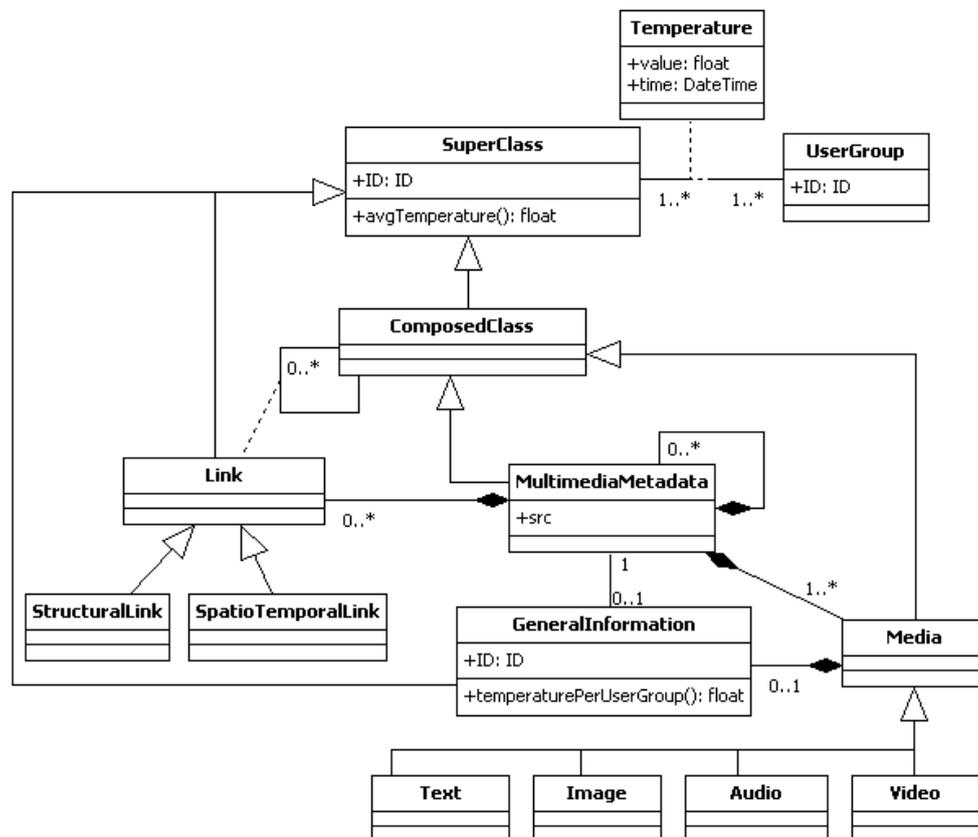


Figure 27 - Le modèle de métadonnées étendu par la température

La température d'une métadonnée pour un certain groupe d'utilisateurs, à un certain moment, traduit l'intérêt de ce groupe pour la partie du document que la métadonnée décrit. De cette

façon, en analysant les températures des métadonnées d'un document pour les différents groupes d'utilisateurs, on peut déduire quel groupe d'utilisateurs s'intéresse plus à ce document.

La température moyenne, comme son nom l'indique, représente d'intérêt global des utilisateurs pour la métadonnée. L'équation (1) est utilisée pour calculer cette température moyenne.

$$avgTemperature^i(\Delta t) = \frac{\sum_{j=0}^n nrUsers(gr_j) * temp_{gr_j}^i(\Delta t)}{\sum_{j=0}^n nrUsers(gr_j)} \quad (1)$$

Dans cette équation nous avons :

- i = l'identifiant d'un élément de métadonnée,
- Δt = une période de temps,
- n = le nombre de groupes d'utilisateurs pris en compte,
- gr_j = le groupe j ;
- $nrUsers(gr_j)$ = le nombre d'utilisateurs du groupe gr_j ,
- $temp_{gr_j}^i(\Delta t)$ = la température que l'élément de métadonnée i a pour le groupe gr_j pour la période Δt .

Tableau 20 – Extrait des métadonnées associées à ce manuscrit, enrichies par la température

```
<document ID="U0" src="rapport.docx" avgTemperature="0.0">
  <generalInformation ID="U1" avgTemperature="0.0">
    <Temperature userGroupID="UGR1" time="2012-04-29T00:00:00">
      <value>0.0 </value>
    </Temperature>
  </generalInformation>
  <text ID="U14" avgTemperature="0.0">
    <presentation ID="U17" avgTemperature="0.0">
      <chapter ID="U18" title="Introduction" avgTemperature="0.0">
        <chapter ID="U19" title="Le contexte de notre travail" avgTemperature="0.0">
          <paragraphe ID="U21" avgTemperature="0.0">
            <value> Ce fossé sémantique se trouve entre les informations de ....</value>
            <Temperature userGroupID="UGR1" time="2012-04-29T00:00:00">
              <value>0.0 </value>
            </Temperature>
          </paragraphe>
          <Temperature userGroupID="UGR1" time="2012-04-29T00:00:00">
            <value>0.0 </value>
          </Temperature>
        </chapter>
        <Temperature userGroupID="UGR1" time="2012-04-29T00:00:00">
          <value>0.0 </value>
        </Temperature>
      </chapter>
      <Temperature userGroupID="UGR1" time="2012-04-29T00:00:00">
        <value>0.0 </value>
      </Temperature>
    </presentation>
  </text>
  <Temperature userGroupID="UGR1" time="2012-04-29T00:00:00">
    <value>0.0 </value>
  </Temperature>
```

```

</presentation>
<Temperature userGroupID="UGR1" time="2012-04-29T00:00:00">
  <value>0.0 </value>
</Temperature>
</text>
<image ID="U150" avgTemperature="0.0">...</image>
<Temperature userGroupID="UGR1" time="2012-04-29T00:00:00">
  <value>0.0 </value>
</Temperature>
</document>

```

Pour illustrer le résultat obtenu en appliquant cette extension, nous reprenons l'exemple de métadonnées associées à ce manuscrit (présenté dans le Tableau 11) et nous lui rajoutons la température. Par soucis de lisibilité, dans l'exemple fourni dans le Tableau 20, nous ne considérons qu'un seul groupe d'utilisateurs.

5.3.1.2. Les métadonnées pour définir la structure d'usage

L'usage que les utilisateurs font des documents ne reflète pas seulement la popularité des documents, mais crée naturellement des liens logiques entre ces documents. Par exemple, si deux paragraphes d'un document sont copiés par l'utilisateur pour les utiliser dans un autre document, alors un lien d'usage (qui ne pourrait pas être identifié autrement) peut être établi entre ces deux paragraphes. Pour cela nous avons inclus dans notre format la structure d'usage.

Définition 1: La *structure d'usage* est une structure dynamique qui évolue du point de vue des liens existants entre les différentes parties du document, tout au long du cycle de vie du document multimedia et qui est composée de liens indirects établis entre les documents et/ou les parties de documents en fonction du comportement de l'utilisateur au moment de son interaction avec le(s) document(s).

La modélisation de cette structure d'usage est présentée dans la Figure 28. Les liens de cette structure d'usage sont traduits dans le modèle par *les liens indirects*. Le lien indirect a un attribut *type*, dont la valeur n'est pas restreinte à un nombre limité de valeurs.

Tout comme pour les autres types de liens, les liens indirects peuvent être établis entre le document décrit par les métadonnées et n'importe quel autre document identifié par un URL. Si un lien indirect est utilisé pour lier deux documents différents, alors le lien est présenté dans les métadonnées associées à chacun des documents.

Le lien indirect peut avoir plusieurs significations liées à l'usage commun des deux parties mises en correspondance par le lien. Par exemple, deux paragraphes d'un document qui sont copiés par l'utilisateur pour les utiliser dans un autre document vont être liés à travers un lien indirect de type « copié ». Un autre exemple de lien indirect peut être considéré comme étant les mots d'une requête. En supposant que deux documents doc1 et doc2 ont été considérés pertinents pour une requête q1 de l'utilisateur, un lien indirect est créé entre les deux documents qui a comme type la requête q1. Dans la section 5.3.4.1, nous présentons un algorithme qui crée des liens indirects entre des documents pertinents pour une requête, au fur et à mesure que des requêtes sont exécutées par les utilisateurs.

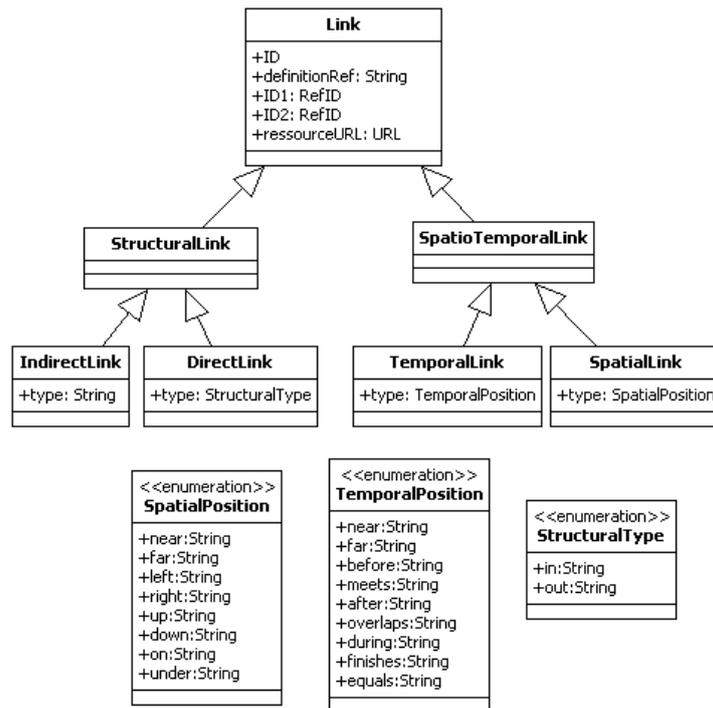


Figure 28 – Le modèle de métadonnées étendu par la structure d'usage

Par analogie au reste des métadonnées, à chaque lien indirect est aussi associée une température. Cette température varie aussi dans le temps, en fonction de l'usage des utilisateurs. Si la même action est réalisée par plusieurs utilisateurs, sur les mêmes documents, alors la température du lien indirect établi entre les documents va augmenter, le lien devient plus chaud et donc plus populaire.

Tableau 21 – Extrait de métadonnées d'usage pour le document huile-visage.html

```

<document ID="U0" src="naturaMundi/htmlFiles/huile-visage.html" avgTemperature="0.0">
<generalInformation ID="U1" avgTemperature="0.0">...</generalInformation>
<image ID="U59" avgTemperature="0.0"> ...</image>
<links ID="U67" avgTemperature="0.0">
  <DirectLink resourceUrl="http://www.naturamundi.com/" type="out" avgTemperature="0.0"
  ID="U68" >
    <Temperature userGroupID="UGR1" time="2012-04-29T00:00:00">
      <value>0.0 </value>
    </Temperature>
  </DirectLink>
  <DirectLink resourceUrl="http://www.naturamundi.com/huile-corps.html" type="in" avgTempera-
  ture="0.0" ID="U408">
    <Temperature userGroupID="UGR1" time="2012-04-29T00:00:00">
      <value>0.0 </value>
    </Temperature>
  </DirectLink>
  <IndirectLink resourceUrl="http://www.naturamundi.com/huile-corps.html" type="huile"
  avgTemperature="0.0" ID="U461">
    <Temperature userGroupID="UGR1" time="2012-04-29T00:00:00">
      <value>0.0 </value>
    </Temperature>
  </IndirectLink>
</links>

```

```

</IndirectLink>
<Temperature userGroupID="UGR1" time="2012-04-29T00:00:00">
  <value>0.0 </value>
</Temperature>
</links>
<Temperature userGroupID="UGR1" time="2012-04-29T00:00:00">
  <value>0.0 </value>
</Temperature>
</document>

```

Le Tableau 21 présente un extrait de cette structure d'usage, associée à une page HTML. Les liens indirects pris en compte dans cet exemple sont constitués par des requêtes de l'utilisateur. Le document huile-visage.html et le document huile-corps.html sont liés par le lien indirect avec l'identifiant U461, qui a comme type « huile », qui est une requête pour laquelle les deux documents sont pertinents (le mot clé de la requête associée est « huile »).

5.3.2. Les structures dynamiques des documents multimédias

Les structures spatiale, temporelle et hypermédia (cf., section 1.3) sont des structures *statiques*. Elles sont définies à la création du document multimédia par l'auteur du document et elles ne changent pas au cours de l'utilisation du document. Nous proposons d'ajouter à ces structures, une dimension d'usage qui évolue dans le temps, qui va les rendre *dynamiques*. Les structures dynamiques peuvent être utilisées pour une meilleure adaptation du document à l'usage, aux variations de l'intérêt, de l'environnement, etc..

Nous appliquons le concept de température à la structure du document. De cette manière nous obtenons une structure dynamique qui varie en fonction du nombre et des interactions des utilisateurs qui consomment le document multimédia ou ses métadonnées.

Définition 2 : La *structure dynamique* d'un document multimédia est une structure renforcée avec le concept de température, qui évolue toute au long du cycle de vie et de l'utilisation du document.

Une telle structure peut être dérivée de n'importe quelle autre structure (e.g., logique, physique, sémantique, spatiale, temporelle, d'usage). Donc, si on renforce la structure d'usage avec le concept de température, on obtient une structure hyper-dynamique, qui évolue dans le temps selon deux axes :

- *en structure* : des liens peuvent apparaître à tout moment ;
- *en température* : plus un document est utilisé, plus il va devenir chaud, et à l'inverse, moins un document est utilisé, plus il refroidira.

Ces structures d'usage évoluent dans le temps, des liens étant créés tout au long de l'utilisation du document. La valeur de la température de chaque élément de métadonnées est modifiée en fonction des actions des utilisateurs. Ces métadonnées sont des métadonnées implicites, parce qu'elles sont générées en analysant le comportement de l'utilisateur, sans demander une action explicite de sa part.

5.3.3. L'algorithme d'enrichissement

Le format de métadonnées proposé est un format arborescent. Dans notre modèle, les liens définis pour représenter les différentes structures d'un document, sont considérés comme des simples nœuds dans l'arbre des métadonnées.

Dans le monde physique, la température d'un corps se propage aux autres éléments qui l'entourent. Pour prendre en compte cette propriété physique de la température, nous proposons de propager tout changement de température, survenu à n'importe quel niveau dans l'arborescence des métadonnées, à ses descendants et à ses ancêtres. En outre, si dans le monde réel, il n'y a plus d'intervention de l'extérieur sur un corps, alors il se refroidit, en perdant de la température. Par conséquent, nous avons mis en place un algorithme d'atténuation de la température.

Dans le cadre de notre proposition, nous considérons les changements de température pour chaque élément, pour une période de temps Δt (minute, heure, jour, semaine,...). Le choix de la durée pour laquelle le changement de température est calculé peut être déterminé en fonction des besoins du domaine d'application.

Une première version de cet algorithme d'enrichissement a été publiée dans (Manzat et al., 2010). Dans cette version, nous retrouvons les mêmes principes pour l'augmentation et le refroidissement de la température. Une amélioration ultérieure a été effectuée en ce qui concerne l'application du refroidissement de la température.

Dans cette partie, nous présentons la façon dont la propagation de la température a lieu, sans prendre en compte la manière dont le changement de température est survenu. Dans un premier temps nous nous intéressons à l'augmentation de la température et sa propagation dans l'arbre des métadonnées. Dans un second temps, nous présentons son atténuation.

5.3.3.1. L'augmentation de la température

Le changement de température peut être réalisé pour chaque période de temps Δt et il dépend du nombre d'utilisateurs appartenant au même groupe qui ont consommé les métadonnées dans la période de temps Δt . L'algorithme utilisé pour l'augmentation de la température est présenté dans le Tableau 22. Cet algorithme s'applique pour chaque indicateur de l'intérêt de l'utilisateur considéré comme pertinent dans le domaine d'application.

Les paramètres de cet algorithme sont : l'élément de métadonnée pour lequel la température doit être recalculée, le nombre d'utilisateurs qui ont interagi avec la métadonnée et l'identifiant du groupe auquel ils appartiennent.

La première étape consiste à calculer la température du groupe pour l'élément reçu comme paramètre. Ce calcul est réalisé en deux étapes :

- le calcul de la valeur de température ($\Delta temp$) à ajouter à la température actuelle de la métadonnée. Cette température est directement proportionnelle au nombre d'utilisateurs n qui ont consommé la métadonnée. Un paramètre α est employé pour réaliser ce calcul.
- Cette valeur $\Delta temp$ est additionnée à la température actuelle de la métadonnée MD.

Pour pouvoir utiliser cette valeur de la température plus tard, nous la sauvegardons dans un historique. Pour chaque période Δt , l'historique contient une seule valeur de la température pour chaque groupe d'utilisateurs. Ensuite, la température moyenne de cet élément est déterminée, en utilisant l'équation (1).

Tableau 22 – L’algorithme pour l’augmentation de la température

Algorithme 3 : `increaseTemperature`

Entrée : La métadonnée, MD , pour laquelle la température est recalculée ; le nombre n d’utilisateurs du même groupe qui ont consommé la métadonnée ; l’identifiant du groupe auquel les utilisateurs appartiennent, gID ; le paramètre α , utilisé pour le calcul de la température pour un élément de métadonnées.

Sortie : la métadonnée md , avec la température augmentée pour tous ses descendants et ses ancêtres.

```
 $\Delta temp \leftarrow computeGroupTemperature(n, \alpha);$ 
 $md \leftarrow setGroupTemperature(MD, gID, \Delta temp);$ 
 $setHistory(\Delta t, MD.ID, gID, \Delta temp);$ 
 $avgTemp \leftarrow computeAvgTemperature(MD);$ 
 $md \leftarrow setAvgTemperature(md, avgTemp);$ 
if  $MD$  has children then
   $\perp md \leftarrow propagateTemperatureDown(md, gID, \Delta temp, 1);$ 
if  $MD$  has parent then
   $\perp md \leftarrow propagateTemperatureUp(md.parent, gID, n, \Delta temp, 1);$ 
return  $md$ ;
```

Chaque fois que la température d’un élément est modifiée en utilisant la fonction *increaseTemperature*, ce changement est propagé à tous ses descendants, en utilisant l’algorithme présenté dans le Tableau 23. Cet algorithme suit les mêmes étapes que le premier algorithme. La température pour chaque descendant est changée avec une valeur directement proportionnelle à la variation de la température de son parent et à son niveau dans l’arbre. La propagation peut être limitée à un certain niveau, établi à travers la constante *MAXLevel*. On applique le même raisonnement pour la propagation de la température à tous les ancêtres de l’élément qui a initialisé le processus d’augmentation. L’algorithme défini pour cette opération est présenté dans le Tableau 24, il suit les mêmes règles que l’algorithme qui propage la température aux descendants.

Tableau 23 – L’algorithme de propagation de la température au niveau des descendants

Algorithme 4 : `propagateTemperatureDown`

Entrée : La métadonnée, MD , pour laquelle la température des enfants doit être modifiée; l’identifiant du groupe, gID , pour lequel la température doit être modifiée; la différence de température du parent $\Delta temp$, qui est utilisée pour modifier la température de chaque fils; le niveau, $level$, dans la procédure récursive.

Sortie : La métadonnée, md , qui a la température de tous ses fils augmentée.

```
 $inc \leftarrow computeTemperatureDown(\Delta temp);$ 
foreach child of  $MD$  do
   $md \leftarrow md \cup setGroupTemperature(child, gID, inc);$ 
   $avgTemp \leftarrow computeAvgTemperature(child);$ 
   $md \leftarrow md \cup setAvgTemperature(child, avgTemp);$ 
  if  $level < MAXLevel$  then
     $\perp md \leftarrow md \cup propagateTemperatureDown(child, gID, inc, level+1);$ 
return  $md$ ;
```

La différence majeure entre l'algorithme d'augmentation et ceux de propagation est que, pour le premier algorithme, seule la valeur de la température est sauvegardée dans l'historique. Nous considérons que le changement de la température produit directement par l'interaction des utilisateurs avec la métadonnée est plus important que celui obtenu suite à la propagation.

Au moment du calcul de la température de toute métadonnée, une attention particulière est portée à la limite supérieure que la température peut avoir. Dans notre cas, la température d'une métadonnée ne peut pas dépasser 100. Pour cela, avant de réaliser toute affectation de la température à une métadonnée, une comparaison est réalisée dans le cadre de la fonction *setGroupTemperature*, afin de respecter cette contrainte.

Tableau 24 – L'algorithme de propagation de la température au niveau des ancêtres

Algorithme 5 : *propagateTemperatureUp*

Entrée : La métadonnée, *MD*, pour laquelle la température des ancêtres doit être modifiée; l'identifiant du groupe, *gID*, pour lequel la température doit être modifiée; la différence de température du parent $\Delta temp$, qui est utilisée pour modifier la température de chaque fils; le niveau, *level*, dans la procédure récursive.

Sortie : La métadonnée, *md*, qui a la température de son parent augmentée

```

if MD ≠ null then
    inc ← computeTemperatureUp( $\Delta temp$ );
    md ← md U setGroupTemperature(MD, gID, inc);
    avgTemp ← computeAvgTemperature(MD);
    md ← md U setAvgTemperature(MD, avgTemp);
    if level < MAXLevel then
        md ← md U propagateTemperatureUp(MD.parent, gID, inc, level+1);
return md;

```

L'équation (2) est utilisée pour le calcul de la température d'un élément de métadonnée, en fonction du nombre d'utilisateurs qui l'ont visualisé (*nrUsers*).

$$computeGroupTemperature(nrUsers, \alpha) = \alpha * nrUsers \quad (2)$$

Pour le calcul de la température qui est propagée aux fils, l'équation (3) est employée. Dans cette formule, la température acquise par un élément de métadonnée est propagée à ses fils, de manière égale à chaque nœud, en étant divisée par le nombre de fils de l'élément de métadonnée dans l'arborescence.

$$computeTemperatureDown(\Delta temp^{parent}) = \beta * \frac{\Delta temp^{parent}}{NbChildren(parent)} \quad (3)$$

Pour la propagation de la température au niveau du parent, l'équation (4) est employée. Seulement une partie de la température du fils est propagée au niveau du parent. Ce pourcentage est établi par le paramètre δ .

$$\text{computeTemperatureUp}(\Delta\text{temp}^{\text{child}}) = \delta * \Delta\text{temp}^{\text{child}} \quad (4)$$

Au niveau de l'augmentation de la température, la différence entre cet algorithme et la version publiée dans (Manzat et al., 2010) se retrouve au niveau de la gestion de l'historique de l'évolution de la température, qui a été modifiée suite au changement réalisé dans la gestion du refroidissement. Dans la version publiée, nous mettons à jour l'historique chaque fois que la température est modifiée, soit parce que l'élément de métadonnée a été consommé de manière directe par les utilisateurs, soit parce qu'il y a eu lieu une propagation du changement de la température, suite à une augmentation au niveau d'un de ses descendants ou de ses ancêtres. Une autre différence se situe au niveau des formules utilisées pour le calcul de la température. Par exemple, dans (Manzat et al., 2010), nous utilisons la moyenne arithmétique, dans ce manuscrit de thèse, pour prendre en compte la dimension différente des groupes des utilisateurs, dans la version de l'algorithme présenté nous avons utilisé une moyenne pondérée, où le poids est constitué par les nombre d'utilisateurs de chaque groupe d'utilisateurs.

5.3.3.2. Le refroidissement de la température

La première version de cet algorithme de refroidissement publiée dans (Manzat et al., 2010) est présentée dans le Tableau 25.

Ici la température est diminuée pour tous les éléments de métadonnées qui n'ont pas eu de changement de température, ni par les interactions directes des utilisateurs, ni à travers le processus de propagation. En outre, le refroidissement d'un élément de métadonnées n'a pas d'influence sur ses descendants ni sur ses ancêtres dans l'arborescence des métadonnées. Si la température d'un élément de métadonnée n'est pas modifiée pendant la dernière période de temps Δt , alors la température de l'élément revient à la dernière valeur de la température obtenue suite à une interaction de l'utilisateur ou suite à une propagation de la température. Le processus de réduction de la température peut être appliqué après chaque période Δt .

Tableau 25 – L'algorithme pour l'atténuation de la température proposé dans (Manzat et al., 2010)

Algorithme 6 : decreaseTemperature
<p>Entrée : La métadonnée, <i>MD</i>, pour laquelle la température doit être diminuée; l'historique, <i>history</i>, du changement de la température pour toutes les métadonnées de l'arborescence ; le nombre <i>N</i> d'intervalles Δt pour lesquels la température doit être diminuée.</p> <p>Sortie : les métadonnées, <i>md</i>, qui ont la température diminuée.</p> <pre> foreach <i>child</i> of <i>MD</i> do for <i>i=1</i> to <i>N</i> do <i>dec</i> = <i>getLastHistoryValue</i>(<i>child</i>, <i>history</i>, <i>gID</i>); <i>history</i> ← <i>removeLastHistoryValue</i>(<i>child</i>, <i>history</i>, <i>gID</i>); <i>md</i> ← <i>md</i> U <i>setGroupTemperature</i>(<i>child</i>, <i>gID</i>, <i>dec</i>); <i>avgTemp</i> ← <i>computeAvgTemperature</i>(<i>child</i>); <i>md</i> ← <i>md</i> U <i>setAvgTemperature</i>(<i>child</i>, <i>avgTemp</i>); <i>md</i> ← <i>md</i> U <i>decreaseTemperature</i>(<i>child</i>, <i>history</i>, <i>N</i>); return <i>md</i>; </pre>

En analysant plus en détails l'évolution de la température en appliquant cette stratégie de refroidissement, nous nous sommes rendus compte que cette évolution ne reproduit pas une évolution réelle de la température. Pour cela, nous avons amélioré l'algorithme de refroidissement en changeant d'une part, la façon dont l'historique du changement de la température est géré, et d'autre part, en essayant d'avoir un refroidissement plus lent des éléments de métadonnées qui ne sont pas utilisés. En outre, vu que l'augmentation de la température a été propagée au niveau des fils et des parents, la diminution de la température doit elle aussi suivre la même propagation. Le nouvel algorithme de diminution de la température est présenté dans le Tableau 26.

Tableau 26 – L'algorithme pour l'atténuation de la température

Algorithme 7 : decreaseTemperature

Entrée : La métadonnée, *MD*, qui se trouve à la racine de l'arborescence pour laquelle les températures doivent être diminuées ; l'identifiant du groupe, *gID*, pour lequel la température doit être modifiée; les dates de début, *startTime*, et fin, *endTime*, de la période pour laquelle la température doit être diminuée;

Sortie : les métadonnées, *md*, qui ont la température diminuée pour les métadonnées non utilisées pendant la dernière période de temps Δt .

```

lastItemsUsed ← getLastItemsUsed(MD, history, gID, startTime, endTime);
foreach item in lastItemsUsed do
    lastUsagePeriod ← getLastUsagePeriod(item.ID, history, gID);
    usedTimes ← getDiminuationUsedTimes(item.ID, history, gID, lastUsagePeriod);
    if usedTimes = NBPERIODS then
        dec ← getHistoryValue(item.ID, history, gID, lastUsagePeriod);
        history ← removeLastHistoryValue(item.ID, history, gID, lastUsagePeriod);
    else
        dec = getLastHistoryValue(item.ID, history, gID, lastUsagePeriod)* $\gamma$ ;
        history ← replaceHistoryValue(item.ID, history, gID, lastUsagePeriod, dec);
        history ← incrementUsedTimes(item.ID, history, gID, lastUsagePeriod);
    MD ← MD U setGroupTemperature(item, gID, -dec);
    avgTemp ← computeAvgTemperature(item);
    MD ← MD U setAvgTemperature(item, avgTemp);
    MD ← MD U propagateTemperatureDown(MD, gID, -dec, 1);
    md ← md U propagateTemperatureUp(MD.parent, gID, -dec, 1);
return md;

```

L'algorithme commence par la sélection des métadonnées qui n'ont pas été utilisées dans la dernière période. La liste résultante est parcourue et la température du chaque élément est calculée comme suit :

- La période de la dernière utilisation, différente de la période courante, est déterminée ;
- Pour cette dernière utilisation, on récupère le nombre de fois que la valeur de la température qui se trouve dans l'historique, a été utilisée dans le processus d'atténuation ;
- Si, pour cet élément de métadonnée, la dernière température de l'historique a été utilisée plus de *NBPERIODS* fois, alors la valeur de sa dernière température est soustraite à la valeur courante de la température et la température de l'élément de métadonnée afférente à cette dernière période d'utilisation est éliminée de l'historique ;
- Sinon, seul un pourcentage γ de la dernière valeur de la température qui figure dans l'historique est enlevé de la valeur courante de la température. Dans l'historique, cette

dernière valeur est remplacée par le pourcentage qui est soustrait et le nombre d'utilisations de cette température est incrémenté ;

- La température moyenne est recalculée ;
- Le changement de température est propagé à tous les descendants et ancêtres de l'élément. Pour la réalisation de cette propagation, les algorithmes utilisés sont les mêmes que ceux utilisés pour la propagation de l'augmentation de la température.

L'équation (5) résume le calcul de l'atténuation de la température.

$$temp_{gr_j}^i(\Delta t + 1) = \begin{cases} temp_{gr_j}^i(\Delta t) - last\Delta Temp(lastUsed), lastUsed \text{ a été utilisé} \\ \text{plus de } NBPERIODES \text{ fois} \\ temp_{gr_j}^i(\Delta t) - \gamma * last\Delta Temp(lastUsed), lastUsed \text{ a été utilisé} \\ \text{moins de } NBPERIODES \text{ fois} \end{cases} \quad (5)$$

Dans ce qui suit, nous présentons une façon d'appliquer l'algorithme de calcul de la température et de création des liens indirects, en se focalisant sur les interactions d'un utilisateur avec un système de gestion de documents multimédias.

5.3.4. L'application de l'algorithme d'augmentation de la température

Pour pouvoir établir l'algorithme de mise à jour de la température d'un élément de métadonnées, nous avons regardé le moment où l'utilisateur interagit avec le document ou avec ses métadonnées. La Figure 29 présente les différentes interactions qu'un utilisateur peut avoir avec un système de gestion de documents multimédias (présenté dans la Figure 10).

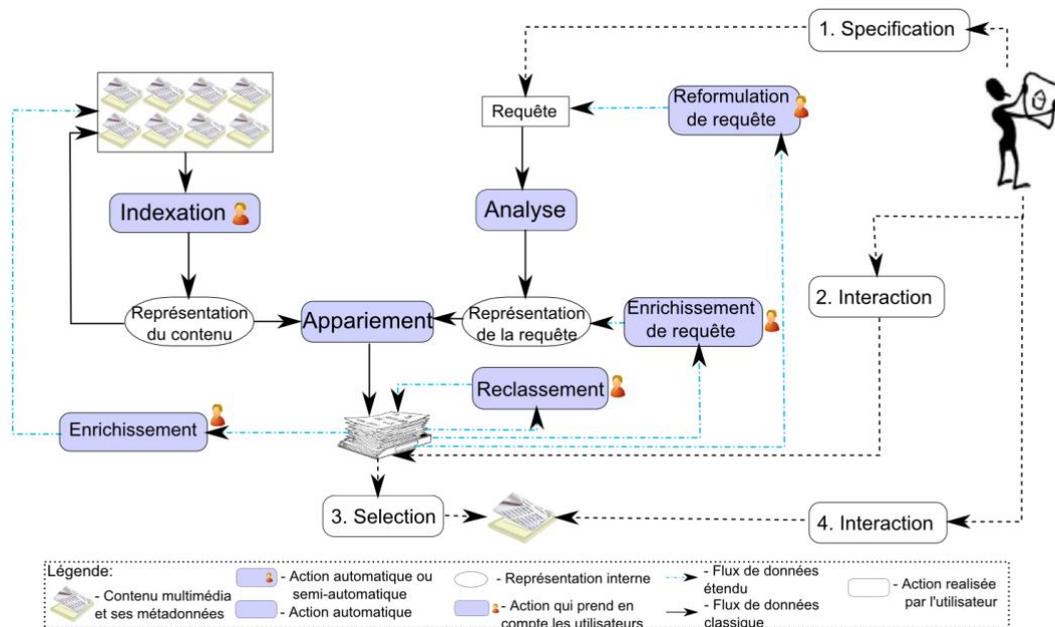


Figure 29 – Les interactions de l'utilisateur avec un système de gestion de documents multimédias

La première interaction d'un utilisateur avec le système de gestion de documents multimédias est sa requête (étape 1 dans la Figure 29). A cette étape, la requête est exécutée sur un sous-

ensemble des métadonnées, on peut donc dire qu'à ce moment là, il y a une interaction indirecte de la part de l'utilisateur avec les métadonnées.

Une fois la requête exécutée, le système retourne à l'utilisateur une liste de résultats pertinents pour sa requête. La liste de résultats affichée à l'utilisateur est composée par des liens vers les documents et une courte présentation de chaque document. L'utilisateur consulte cette description pour voir si le document répond vraiment à son besoin, sans parcourir le document. Cette description peut être composée de certaines métadonnées associées au document. Donc à cette étape, l'utilisateur interagit avec les métadonnées (étape 2 dans la Figure 29). D'un autre côté, les requêtes des utilisateurs ajoutent de l'information supplémentaire sur les documents qui sont retournés par le système de gestion.

Une fois le document choisi, l'utilisateur interagit avec lui (étape 4 de la Figure 29). Cette interaction est celle qui est la plus étudiée dans l'état de l'art, et plusieurs indicateurs de l'intérêt de l'utilisateur ont été identifiés.

Sur la base de ces considérations, nous proposons de réaliser l'enrichissement des métadonnées en trois étapes :

- en prenant en considération les requêtes de l'utilisateur (étape 2 dans la Figure 29) ;
- en prenant en considération l'interaction de l'utilisateur avec les métadonnées associées aux résultats de la requête (étape 2 de la Figure 29) ;
- en prenant en considération l'interaction de l'utilisateur avec le document multimédia lui-même (étape 4 de la Figure 29).

5.3.4.1. Les requêtes de l'utilisateur

Les requêtes qu'un utilisateur soumet au système de gestion de contenus multimédias reflètent un intérêt de l'utilisateur envers un sujet donné, car une requête est la transcription du besoin de l'utilisateur. A travers sa requête, l'utilisateur peut identifier des relations entre les documents et les termes de la requête, qui ne peuvent pas être extraites du contenu des documents. Pour cela, nous créons des liens indirects entre les documents qui sont considérés comme étant pertinents pour la requête par le système. Ces liens peuvent être utilisés dans l'exécution de la requête.

Chaque fois qu'une requête est exécutée par un utilisateur, des liens indirects sont créés (indépendamment du groupe auquel l'utilisateur appartient) entre tous les documents pertinents pour la requête et pour lesquels la pertinence est supérieure à un certain seuil. Une température est associée à chaque lien. Une fois le lien créé, sa température évolue, pour chaque groupe, en fonction du nombre de fois que la requête est exécutée par les utilisateurs de chaque groupe.

L'algorithme utilisé pour la création des liens indirects est présenté dans le Tableau 27. Comme nous l'avons déjà dit, les liens indirects sont créés de manière répétée pour chaque période de temps Δt . Dans cet algorithme, la période de temps est spécifiée à travers les dates de début et de fin.

Pour cette période de temps, toutes les requêtes différentes sont récupérées. Pour la création des liens indirects, le nombre de fois que chaque requête a été exécutée n'est pas important. Pour chaque requête, les résultats retournés par le système sont récupérés. Dans le processus de création des liens, seulement les résultats qui sont classés dans les premiers rangs sont considérés, ceux qui ont un rang supérieur à *linkThreshold*. Pour pouvoir créer les liens indirects, nous avons besoin des métadonnées associées à chacun des résultats pris en compte. Pour assurer l'identification unique des éléments de métadonnées associées à un document, nous considérons que les identifiants des éléments

sont créés de manière séquentielle et donc en identifiant le dernier ID attribué et en l'utilisant dans le processus de création des liens, nous obtenons des identifiants uniques. Par la suite, pour chaque résultat considéré, un lien indirect est créé entre le résultat et tous les autres documents pertinents pour la requête, indifféremment de leur rang dans la liste des résultats. Le type du lien créé est constitué par la requête, *query*, pour laquelle les deux documents liés sont pertinents. Pour chaque triplet (result1, result2, query) seulement un lien indirect est créé.

Tableau 27 – L'algorithme de création des liens indirects

Algorithme 8 : createIndirectLinks

Entrée : Les dates de début, *startTime*, et fin, *endTime*, de la période pour laquelle les liens doivent être créés.

Sortie : Les métadonnées des premiers *linkThreshold* résultats pour chaque requête modifiées par l'ajout des liens indirects

```

queries ← getExecutedQueries(startTime, endTime);
foreach query in queries do
  results ← execute(query);
  for i=1 to linkThreshold do
    result ← results(i);
    md ← getAssociatedMetadata(result);
    IDMax ← getIDMax(md);
    foreach result2 in results do
      if result2 ≠ result then
        indirectLink ← getIndirectLink(md, result, result2, query);
        if indirectLink = null then
          md ← md U createLink(md, result, result2, query, IDMax);
    save(md);

```

Une fois les liens indirects créés, ils ne sont pas jamais détruits, car il nous semble important de garder cette structure d'usage, pour pouvoir les exploiter à tout moment. Les liens ayant une température, nous pouvons voir comment cette structure d'usage évolue dans le temps.

5.3.4.2. L'interaction de l'utilisateur avec les métadonnées

L'utilisateur interagit de manière indirecte avec les métadonnées associées aux documents multimédias à travers ses requêtes. Dans le processus d'exécution d'une requête, qui correspond à la partie droite du schéma en « U » présentée dans le Chapitre 2, Figure 7, la requête de l'utilisateur est transformée pour pouvoir réaliser un appariement avec les métadonnées contenues dans le système qui décrit les documents multimédias. Cet appariement se réalise entre la requête et certains éléments de métadonnées. En fonction des requêtes, certains éléments de métadonnées sont plus utilisés que d'autres. La température des métadonnées qui sont utilisées dans l'exécution des requêtes est augmentée. Par exemple, si un utilisateur recherche des documents créés par « Ana-Maria Manzat », alors l'appariement est effectué entre la requête de l'utilisateur et les éléments auteur des métadonnées générales. Donc la température des éléments auteur, de tous les documents, qui ont la valeur « Ana-Maria Manzat » est augmentée en utilisant l'algorithme présenté dans le Tableau 22.

L'utilisateur interagit de manière directe avec les métadonnées qui sont affichées à côté du lien vers le document, dans la liste des résultats. De manière générale, à côté du document une petite

description du document est affichée, des mots-clés et d'autres informations qui sont considérées comme étant importantes afin d'inciter l'utilisateur à regarder le document.

Pour décider si un document présent dans la liste de résultats affichée pour une requête l'utilisateur parcourt cette liste en analysant ces métadonnées. Selon l'étude de (Joachims et al., 2007), les utilisateurs parcourent cette liste de manière séquentielle, de haut en bas, sans forcément lire toutes les informations affichées. Ce parcours des métadonnées est utilisé pour augmenter la température des éléments de métadonnées consommés de cette manière.

Pour capturer cet intérêt de l'utilisateur envers les métadonnées, le suivi du regard est utilisé. D'autres indicateurs qui peuvent être employés sont : le temps que l'utilisateur reste sur les métadonnées, la fixation, la dilatation de la pupille, « saccades » (Rayner, 1998). Pour chaque indicateur d'intérêt sélectionné, l'algorithme du Tableau 22 est utilisé. Pour chaque type d'interaction pris en compte dans l'augmentation de la température, un paramètre différent peut être utilisé pour le calcul de la température. De cette façon, les fixations sur un élément de métadonnée peuvent être plus importantes que les mouseovers. Dans cet algorithme, nous ne considérons pas les clics des utilisateurs sur les liens de la liste des résultats comme étant des interactions avec les métadonnées.

Tableau 28 – L'algorithme d'augmentation de la température suite aux interactions des utilisateurs avec les métadonnées affichées dans la liste des résultats

Algorithme 9 : *increaseTemperatureFromResultsList*

Entrée : Les dates de début, *startTime*, et fin, *endTime*, de la période pour laquelle les liens doivent être créés.

Sortie : Les métadonnées des documents avec lesquels les utilisateurs ont interagi ont une température qui augmente

```

interactions ← getUsersInteractionsWithResultsList(startTime, endTime);
docsGroup ← groupeInteractionsPerDocuments(interactions);
foreach doc in docsGroup do
    MD ← getAssociatedMetadata(doc);
    docInteractions ← groupeInteractionsForDoc(interactions, doc);
    foreach interGr in docInteractions do
        usedMD ← getUsedMetadata(interGr, doc);
        α ← getAssociatedParameter(interGr);
        foreach md in usedMD do
            usersGroups ← groupeInteractionsPerUserGroup(md, interGr);
            foreach userGr in usersGroups do
                grID ← getUserGroupID(userGr);
                nrUtilisations ← getNrUsers(userGr, md);
                MD ← MD U increaseTemperature(md, nrUtilisations, grID, α);
    save(MD);

```

L'algorithme complet défini pour l'augmentation de la température dans ce cas est présenté dans le Tableau 28. Comme nous l'avons déjà dit, la température est recalculée de manière répétée pour chaque période de temps Δt . Dans cet algorithme, la période de temps est spécifiée à travers les dates de début et de fin. Nous ne favorisons pas l'utilisation d'un indicateur particulier de l'intérêt de l'utilisateur. Nous considérons toutes les interactions des utilisateurs, mais chaque type d'interaction pour avoir une autre influence sur la température.

L'algorithme commence par la sélection des interactions qui ont été accomplies par les utilisateurs pendant la période de temps Δt , et nous groupons ces interactions par document. Pour chaque document, nous prenons toutes les interactions effectuées sur le document et les métadonnées associées au document. Pour chaque type d'interaction considérée, nous réalisons une sélection des éléments de métadonnées sur lesquelles des interactions ont été enregistrées. A chaque type d'interaction est associé un paramètre α , qui est aussi sélectionné à cette étape de l'algorithme. Pour chaque élément de métadonnée faisant partie de la liste déterminée à l'étape précédente, un calcul des nombre d'interactions pour chaque groupe d'utilisateurs est réalisé. La température de l'élément de métadonnée est augmentée, pour chacun des groupes d'utilisateurs, en utilisant l'algorithme présenté dans le Tableau 22.

5.3.4.3. L'interaction de l'utilisateur avec la liste des résultats

L'interaction de l'utilisateur avec la liste des résultats ne se résume pas seulement à l'interaction avec les métadonnées. Il y a aussi les clics de l'utilisateur sur certains des résultats. Comme nous avons vu dans le Chapitre 3, les clics de l'utilisateur sur les résultats d'une recherche sont considérés comme un bon indicateur de l'intérêt de l'utilisateur pour les documents cliqués. En nous appuyant sur ces constats, nous considérons les clics de l'utilisateur sur un document donné dans la liste des résultats comme un indicateur de son intérêt pour ce document et par conséquent nous augmentons la température du document cliqué. En plus de cet indicateur, nous considérons que les clics des utilisateurs sur deux documents pour la même requête traduisent un lien entre les deux documents, et donc suite à ce type d'interaction, nous augmentons également la température du lien indirect qui existe entre les deux documents.

Les clics des utilisateurs réalisés sur les liens de la liste des résultats sont interprétés de deux manières :

- la température des documents afférents aux liens est augmentée ;
- la température des liens indirects créés entre des documents qui ont été cliqués ensemble dans la période de temps considérée est augmentée aussi.

L'algorithme défini pour l'augmentation de la température des documents et des liens indirects est présenté dans le Tableau 29.

Tableau 29 – L'algorithme d'augmentation de la température suite aux clics des utilisateurs sur la liste des résultats

Algorithme 10 : `increaseTemperatureFromClickedLinks`

Entrée : Les dates de début, *startTime*, et fin, *endTime*, de la période pour laquelle les liens doivent être créés.

Sortie : Les métadonnées des documents avec lesquels les utilisateurs ont interagi ont la température augmentée

```

queries ← getExecutedQueries(startTime, endTime);
foreach query in queries do
    results ← execute(query);
    clickedResults ← getClickedResults(results, query, startTime, endTime);
    α ← getAssociatedParameter("click");
    foreach different resultClicked in clickedResults do
        | doc ← getDocument(resultClicked);

```

```

MD ← getAssociatedMetadata(doc);
clicksPerUserGroupes ← groupeClicksPerUserGroupes(clickedResults,
                                                    resultClicked);

foreach userGr in clicksPerUserGroupes do
  grID ← getUserGroupID(userGr);
  nrUtilisations ← getNrUsers(clicksPerUserGroupes,userGr);
  MD ← MD U increaseTemperature(MD, nrUtilisations, grID,  $\alpha$ );
 $\beta$  ← getAssociatedParameter("click");
foreach different result in clickedResults do
  if resultClicked2  $\neq$  resultClicked then
    indirectLink ← getIndirectLink(MD, resultClicked, result, query);
    if indirectLink = null then
      IDMax ← getIDMax(MD);
      indirectLink ← createLink(md, resultClicked, result, query, IDMax);
      MD ← MD U indirectLink;
    clicksPerUserGroupes2 ← groupeClicksPerUserGroupes(
                                                    clickedResults, result);
    foreach userGr in clicksPerUserGroupes2 do
      grID ← getUserGroupID(userGr);
      nrUtilisations2 ← getNrUsers(clicksPerUserGroupes2,userGr);
      if nrUtilisations2 > nrUtilisations then
        MD ← MD U increaseTemperature(indirectLink, nrUtilisations,
                                      grID,  $\beta$ );
      else
        MD ← MD U increaseTemperature(indirectLink, nrUtilisations2,
                                      grID,  $\beta$ );
  save(MD);

```

La première étape consiste dans la sélection de toutes les requêtes qui ont été exécutées par les utilisateurs dans la période de temps considérée. Pour chacune de ces requêtes, nous prenons tous les liens qui ont été cliqués. Pour chaque lien cliqué, nous augmentons la température des métadonnées, plus précisément il s'agit de la racine, pour chaque groupe d'utilisateurs qui a ouvert le résultat. Puis nous parcourons les autres liens cliqués afin d'augmenter la température des liens indirects entre ces résultats et le résultat courant. Vu que nous ne créons pas des liens indirects entre tous les résultats d'une requête, il est possible que le lien pour lequel nous voulons augmenter la température n'existe pas, nous testons donc son existence et s'il n'a pas été créé, nous le créons à ce moment-là. Le fait de parcourir entièrement la liste des résultats cliqués deux fois assure une augmentation complète des liens indirects, dans les deux sens.

5.3.4.4. L'interaction de l'utilisateur avec le document multimédia

Cette interaction est réalisée sur n'importe quel type de document, qu'il soit mono-média ou composé de plusieurs médias. En ce qui concerne les interactions prises en compte, nous ne privilégions pas un type donné, mais comme pour les interactions avec les métadonnées, nous considérons tout type d'indicateur de l'intérêt de l'utilisateur. Au moment de l'implémentation de cet algorithme, nous pouvons utiliser un ou plusieurs des indicateurs présentés dans la Chapitre 3.

L'idée principale de cette partie est de capturer toutes les interactions que les utilisateurs ont avec le contenu d'un document multimédia, en prenant soin d'identifier les parties du document qui font l'objet de ces interactions. Nous exploitons ces interactions en augmentant la température des

métadonnées associées à ces parties du document. L'algorithme défini pour l'augmentation de la température dans ce cas est présenté dans le Tableau 30. Dans l'application de cet algorithme, la manière dont l'utilisateur est arrivé sur le document n'est pas importante, mais seulement ses interactions avec le document.

Cet algorithme est appliqué pour chaque document géré par le système. L'algorithme commence avec la sélection de toutes les interactions qui ont eu lieu sur le document pendant la période de temps considérée. Ces interactions sont ensuite regroupées en fonction de la partie du document avec laquelle les utilisateurs ont interagi. Pour chaque partie du document, tous les types d'interactions sont étudiés afin d'augmenter la température de l'élément de métadonnée associé à la partie du document, pour chaque groupe d'utilisateurs. Toutes les parties du document ont le même traitement, sauf les liens hypermédias parce que suite à un clic sur une partie du document, un autre document est ouvert et donc cela a un impact sur l'autre document aussi. Donc dans le cas d'un clic sur un lien hypermédia, la métadonnée sélectionnée est de type `DirectLink`. En utilisant les informations détenues sur cet élément (i.e., l'attribut `resourceURL`), nous retrouvons le document qui a été ouvert (`docOut`). Pour ce document, nous récupérons les métadonnées et nous augmentons leur température. Mais, dans le cadre de ces métadonnées, nous retrouvons aussi le lien entrant qui fait la liaison entre ce document et le document qui contient le lien qui a déterminé l'ouverture de celui-là. Pour mettre en évidence, dans le document ouvert suite au clic, les liens avec d'autres documents qui sont les plus utilisés, nous augmentons la température de ce lien également (i.e., le `DirectLink` entre le `docOut` et le `src`, qui est de type entrant).

Dans cet algorithme, différentes stratégies d'augmentation de la température peuvent être mises en place pour chaque type d'interaction. Ces stratégies sont dépendantes du type de media (i.e., statique – e.g., un document purement textuel qui ne change pas pendant sa présentation – ou dynamique – e.g., une vidéo qui évolue dans le temps, une présentation SMIL composée de plusieurs media), mais également du temps d'affichage du contenu. Par exemple nous pouvons considérer les stratégies d'augmentation de la température suivantes :

- Pour les documents statiques, nous pouvons considérer le nombre de clics, de mouseovers, ou d'autres types d'interaction, comme étant un indicateur de l'intérêt des utilisateurs envers la partie du document prise en compte ;
- Pour les documents dynamiques, nous pouvons considérer les mêmes types d'indicateur de l'intérêt de l'utilisateur que pour les documents statiques, mais ces indicateurs doivent être rapportés au temps d'affichage du document prévu dans sa structure temporelle. Une différence doit être réalisée entre les parties du document qui sont affichées plus longtemps que d'autres.

Tableau 30 – L'algorithme d'augmentation de la température des métadonnées associées aux parties d'un document, suite aux interactions des utilisateurs avec ces parties

Algorithme 11 : `increaseTemperatureFromDocumentInteractions`

Entrée : L'identifiant, `src`, du document pour lequel la température doit être mise à jour ;
les dates de début, `startTime`, et fin, `endTime`, de la période pour laquelle les liens doivent être créés.

Sortie : Les métadonnées du document avec lequel les utilisateurs ont interagi ont la température augmentée

`MD` ← `getAssociatedMetadata(src)`;

`interactions` ← `getUsersInteractionsWithDocument(src, startTime, endTime)`;

```

partGroups ← groupeInteractionsPerDocumentParts(interactions);
foreach part in partGroups do
  md ← getAssociatedMetadata(part, MD);
  interactionsTypes ← getInteractionsPerType(interactions, part);
  foreach interGr in interactionsTypes do
    α ← getAssociatedParameter(interGr);
    usersGroups ← groupeInteractionsPerUserGroup(md, interGr);
    foreach userGr in usersGroups do
      grID ← getUserGroupID(userGr);
      nrUtilisations ← getNrUsers(userGr, md);
      MD ← MD U increaseTemperature(md, nrUtilisations, grID, α);
      if md = DirectLink and interGr = "click" then
        docOut ← getRessourceDocument(md);
        MDOut ← getAssociatedMetadata(docOut);
        linkIn ← getDirectLink(MDOut, src, "in");
        foreach userGr in usersGroups do
          grID ← getUserGroupID(userGr);
          nrUtilisations ← getNrUsers(userGr, md);
          MDOut ← MDOut U increaseTemperature(linkIn, nrUtilisations,
                                                    grID, α);
          MDOut ← MDOut U increaseTemperature(MDOut, nrUtilisations,
                                                    grID, α);
      save(MDOut);
save(MD);

```

Afin d'illustrer cet enrichissement des métadonnées, nous pouvons considérer la structure temporelle d'une présentation SMIL (Figure 30). Cette présentation est composée d'une vidéo, un contenu audio et les diapositives de la présentation sous la forme d'images. Les images sont affichées de manière séquentielle, chaque image étant affichée pour une certaine période de temps (e.g., la première image est affichée pendant la première minute de la présentation, pendant que la dernière image est affichée pendant les 3 dernières minutes). En utilisant un algorithme de segmentation, le contenu audio est segmenté en 4 segments différents qui ne se chevauchent pas. C'est le même cas pour la partie vidéo de la présentation, qui est décomposée en 3 segments. Chacun de ces segments, qu'ils soient audio ou vidéo, a des métadonnées associées, qui décrivent leur contenu.

Nous supposons que plusieurs utilisateurs appartenant au même groupe ont consommé cette présentation. Comme nous l'avons déjà dit, pour l'application de cet algorithme d'augmentation de la température, il n'est pas important de savoir comment les utilisateurs sont arrivés sur le document, mais il est important de savoir qu'ils sont là et qu'ils interagissent avec le document. Dans cet exemple, le seul indicateur de l'intérêt de l'utilisateur que nous prenons en compte, est le temps que l'utilisateur passe sur le document. Mais d'autres éléments d'interaction peuvent être utilisés. Nous supposons que les utilisateurs ont regardé une présentation SMIL de 1'20" à 5'20". Dans ce cas, la température des métadonnées associées à tous les contenus multimédias affichés dans cette période de temps doit être modifiée. De la chronologie présentée dans la Figure 30, nous pouvons en déduire que les segments vidéo *seg_Video₁* et *seg_Video₂*, les segments audio *seg_Audio₁*, *seg_Audio₂* et *seg_Audio₃* et les images *img₂.jpg*, *img₃.jpg* et *img₄.jpg* sont candidats pour l'augmentation de la température.

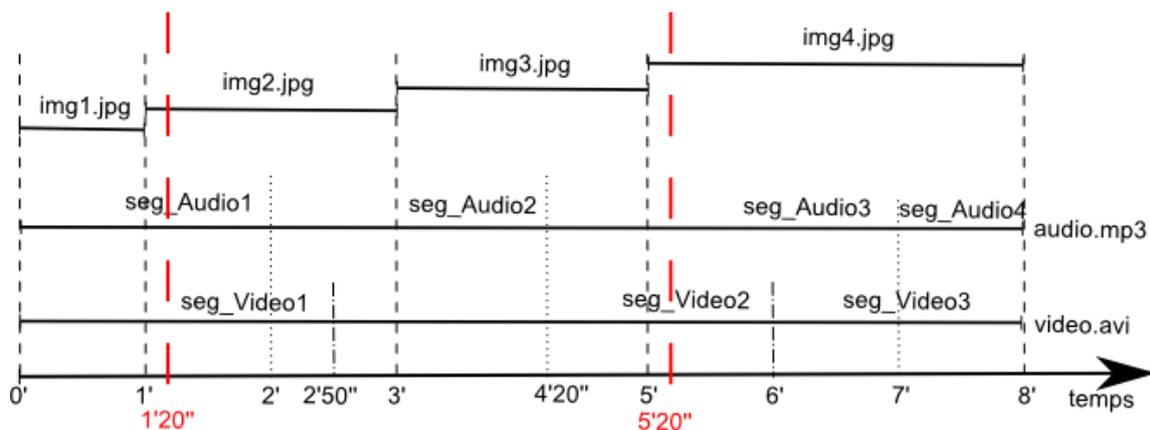


Figure 30 – La structure temporelle d'une présentation SMIL

À ce stade, plusieurs stratégies peuvent être établies pour le choix des segments à utiliser pour l'augmentation de la température de leurs métadonnées. Par exemple, si la partie du document a été regardée pendant au moins la moitié de sa longueur, alors sa température est modifiée. Dans ce cas, la température des segments audio `seg_Audio1` `seg_Audio3` ne sera pas modifiée, car ils ont été écoutés pendant moins de leur longueur (e.g., pendant moins d'une minute pour `seg_Audio1` qui a une longueur de 2 minutes). Dans le cas des segments vidéos, la température des deux segments considérés est augmentée parce qu'ils respectent la condition prise en compte. En ce qui concerne la vidéo, nous devons aussi augmenter la température de tous les frames qui sont affichés à l'utilisateur pendant la période de temps où il regarde la présentation. En ce qui concerne les images, vu que leur contenu ne change pas pendant le temps d'affichage, leur température est augmentée indépendamment de la durée du temps pendant laquelle elles ont été regardées. Bien sûr, l'augmentation de la température des images peut se faire avec une valeur proportionnelle à leur durée d'affichage.

Jusqu'à maintenant nous avons présenté les différents algorithmes définis pour le calcul de la température en fonction des interactions que les utilisateurs peuvent avoir avec les contenus multimédias. Par la suite nous présentons l'algorithme général qui calcule cette température, tout en prenant en compte le refroidissement des documents qui ne sont pas utilisés.

5.3.4.5. L'algorithme général pour le calcul de la température

Pour calculer la température, en prenant en compte les différentes façons dont l'utilisateur interagit avec les documents et leurs métadonnées, tous les algorithmes décrits dans les sections précédentes sont utilisés dans un algorithme général, qui est présenté dans la Tableau 31. Cet algorithme a comme entrée les dates de début et de fin de la période pour laquelle la température doit être mise à jour. De cette manière, il peut être exécuté très facilement à toutes les périodes Δt en le mettant comme « démon », par exemple.

Pour la période considérée, l'algorithme commence par la création des liens indirects et par la prise en compte des interactions des utilisateurs avec les métadonnées affichées dans la liste des résultats. L'étape suivante est l'augmentation de la température en fonction des interactions des utilisateurs avec la liste des résultats. Pour chaque document de la collection de documents multimédias gérés par le système, la température des métadonnées est augmentée en fonction des interactions qui ont lieu avec les différentes parties du document. Pour tous les documents du système, l'opération de refroidissement est effectuée, même pour les documents qui ont eu leur température augmentée.

Tableau 31 – L’algorithme général pour le calcul de la température

Algorithme 12 : UpdateTemperature

Entrée : Les dates de début, *startTime*, et fin, *endTime*, de la période pour laquelle la température doit être mise-à-jour.

Sortie : Les métadonnées de tous les documents gérés par le système qui ont la température mise-à-jour.

```

createIndirectLinks(startTime, endTime);
increaseTemperatureFromResultsList(startTime, endTime);
increaseTemperatureFromClickedLinks(startTime, endTime);
foreach doc in DocumentCollection do
    MD ← getAssociatedMetadata(doc);
    increaseTemperatureFromDocumentInteractions(doc, startTime, endTime);
    foreach userGr in usersGroups do
        grID ← getUserGroupID(userGr);
        decreaseTemperature(MD, grID, startTime, endTime);
    save(MD);

```

En appliquant les algorithmes présentés dans ce chapitre et les différentes fonctions utilisées pour le calcul de la température, nous obtenons des formules différentes en fonction du type de l’élément de métadonnées. Pour le calcul de la température de la racine des métadonnées, l’équation (6) est obtenue. Dans cette équation « i » représente l’identifiant de la racine pour laquelle la température est calculée. Cette formule s’applique pour chacun des groupes des utilisateurs (gr_j) considérés par le système.

$$\begin{aligned}
 & temp_{gr_j}^i(\Delta t + 1) \\
 = & \begin{cases}
 temp_{gr_j}^i(\Delta t) + \alpha_{visualisation} * NBVisualisations + \delta * \sum_{k=1}^{NBChildren} \Delta temp_{gr_j}^k(\Delta t + 1), & \text{si } i \text{ est utilisé} \\
 temp_{gr_j}^i(\Delta t) - \gamma * last\Delta temp(lastUsed) + \delta * \sum_{k=1}^{NBChildren} \Delta temp_{gr_j}^k(\Delta t + 1), & \text{si } i \text{ n'est pas utilisé, et} \\
 & lastUsed \text{ a été employé moins de } NBPERIODES \text{ fois} \\
 temp_{gr_j}^i(\Delta t) - last\Delta temp(lastUsed) + \delta * \sum_{k=1}^{NBChildren} \Delta temp_{gr_j}^k(\Delta t + 1), & \text{si } i \text{ n'est pas utilisé, et} \\
 & lastUsed \text{ a été employé plus de } NBPERIODES \text{ fois}
 \end{cases} \quad (6)
 \end{aligned}$$

$$\begin{aligned}
 & temp_{gr_j}^i(\Delta t + 1) \\
 = & \begin{cases}
 temp_{gr_j}^i(\Delta t) + \alpha_{utilisation} * NbUtilisations + \beta * \frac{\Delta temp_{gr_j}^{parent(i)}(\Delta t + 1)}{NbSiblings(i)}, & \text{si } i \text{ est utilisé} \\
 temp_{gr_j}^i(\Delta t) - \gamma * last\Delta temp(lastUsed) + \beta * \frac{\Delta temp_{gr_j}^{parent(i)}(\Delta t + 1)}{NbSiblings(i)}, & \text{si } i \text{ n'est pas utilisé, et} \\
 & lastUsed \text{ a été employé moins de } NBPERIODES \text{ fois} \\
 temp_{gr_j}^i(\Delta t) - last\Delta temp(lastUsed) + \beta * \frac{\Delta temp_{gr_j}^{parent(i)}(\Delta t + 1)}{NbSiblings(i)}, & \text{si } i \text{ n'est pas utilisé, et} \\
 & lastUsed \text{ a été employé plus de } NBPERIODES \text{ fois}
 \end{cases} \quad (7)
 \end{aligned}$$

Ces équations assurent une évolution continue de la température pour chaque élément de métadonnée. Elles sont utilisées pour le calcul de la température pour toutes les périodes de temps Δt . Le choix de cette durée peut être fait en fonction du besoin du système et aussi en fonction du nombre d'utilisations des documents.

5.3.5. Un exemple d'évolution de la température

Pour mieux comprendre l'évolution et le calcul de la température, nous présentons dans cette section un exemple de l'application des formules et des algorithmes présentés précédemment sur les métadonnées associées à un document. Pour des soucis d'espace et de lisibilité, nous nous concentrons seulement sur une partie de ces métadonnées.

La Figure 31 présente le contexte de cet exemple, à travers les métadonnées associées à un document multimédia et les interactions des utilisateurs avec ces métadonnées au fil du temps. Dans la Figure 31 a) chaque cercle représente un élément de métadonnée. Pour chaque élément, nous n'avons représenté que son identifiant et la valeur de la température pour un seul groupe d'utilisateurs. Par exemple, le cercle 1, représente la racine des métadonnées – multimediaMetadata –, le cercle 2 est l'élément generalInformation, le cercle 3 représente les métadonnées image. La structure du document et des métadonnées est assez simple pour faciliter la compréhension de l'exemple.

Au début, la température de tous les éléments de métadonnées est initialisée à 0 (Figure 31 a)). Pour illustrer l'algorithme de propagation, considérons la succession d'utilisations des éléments de métadonnées présentée dans Figure 31 b). Dans cet exemple, l'élément de métadonnée qui a l'identifiant U3 est cliqué par 100 utilisateurs dans la période de temps t_1 mais aussi dans la période de temps t_2 . Pour cet exemple, nous considérons un seul type d'interaction, les clics, pour simplifier les calculs et les illustrations.

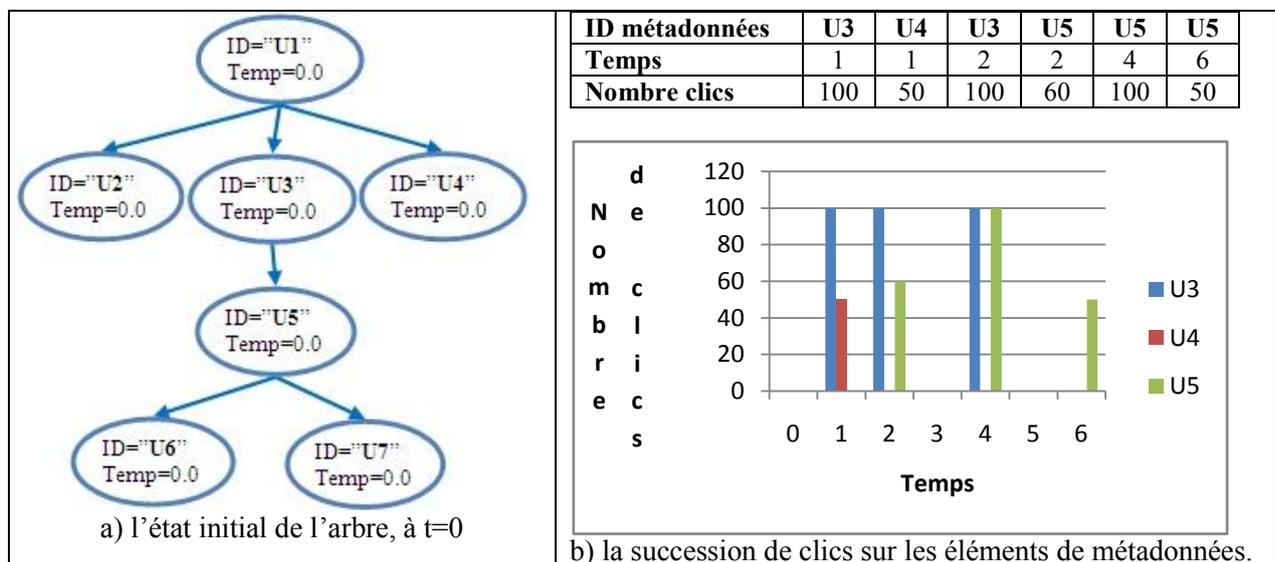


Figure 31 – L'arbre de métadonnées et la succession d'interactions prise en compte

Dans cet exemple, nous avons considéré les valeurs suivantes pour les paramètres :

- $\alpha = 0.1$ – Il y a un seul paramètre α , parce que nous considérons un seul type d'interaction.

- $\beta = 0.5$
- $\delta = 0.5$
- NBPERIODES = 4 ;
- $\gamma = 0.5$.

Dans un premier temps la température de l'élément U3 est calculée. En appliquant l'équation (2) on obtient la valeur de $\Delta temp = \alpha * 100 = 0.1 * 100 = 10$, qui est additionnée à l'ancienne valeur de la température de cet élément, et on obtient la valeur finale de 10. En suivant l'algorithme, la valeur du $\Delta temp$ est inséré dans l'historique de l'évolution de la température (Figure 32, c)).

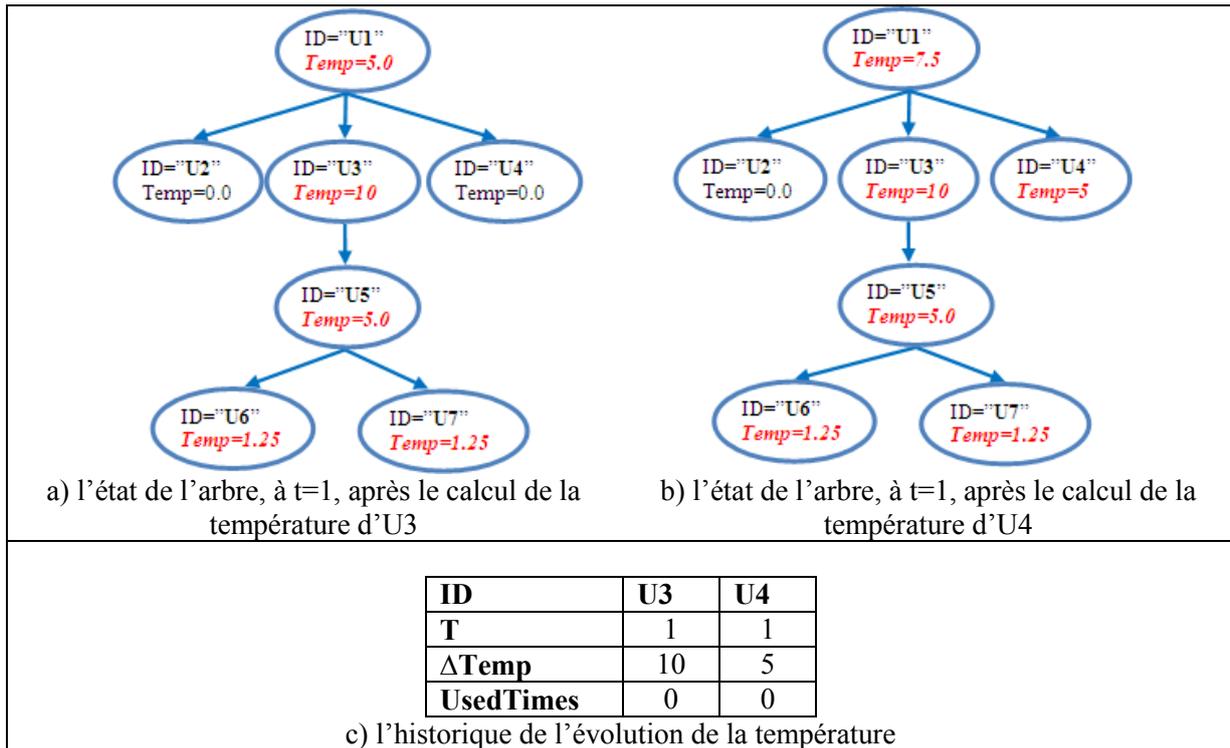


Figure 32 – L'état de l'arbre après le calcul de la température à t=1

Ensuite, la température est propagée aux fils de cet élément. Dans notre exemple le nœud U3 n'a qu'un fils, donc le réchauffement de la température va être appliqué seulement à celui-ci. En calculant la température du nœud U5 avec l'équation (3), on obtient $\Delta temp = \beta * 1/1 * 10 = 0.5 * 1 * 10 = 5$. Le nœud U5 a deux fils, et donc la température va encore être propagée aux deux. Pour calculer la température pour ces deux nœuds (U6 et U7), l'équation (3) est utilisée. On obtient : $\Delta temp = \beta * 1/2 * 5 = 0.5 * 0.5 * 5 = 1.25$. Parce que ces deux nœuds n'ont plus de fils, la propagation de la température s'arrête.

L'étape suivante dans la propagation de la température est la propagation vers les ancêtres du nœud U3. La température du nœud U1 est modifiée avec $\Delta temp = \delta * 10 = 0.5 * 10 = 5$. Parce que U1 est la racine de l'arborescence l'algorithme de propagation se termine. Le résultat obtenu après la propagation de la température est représenté dans la Figure 32 a).

Dans cette période de temps, l'élément avec l'identifiant U4 a été utilisé, donc sa température va être aussi augmentée en suivant le même algorithme. La première étape est le calcul de la température du nœud U4 : $\Delta temp = \alpha * 50 = 0.1 * 50 = 5$, qui lui est attribuée et est également insérée dans

l'historique. Parce que ce nœud n'a pas de fils, on ne réalise que la propagation de la température vers son parent. Le résultat de cette augmentation de la température est présenté dans la Figure 32 b).

La Figure 33 a) présente le résultat obtenu après l'exécution de l'algorithme d'augmentation de la température pour deux périodes de temps (t=1 et t=2).

Au moment du calcul de la température à t = 2, en analysant les interactions, on se rend compte que le nœud U4 a été utilisé dans la période t=1, mais pas dans la période t=2. Donc le processus d'atténuation de sa température doit être déclenché. Dans cet exemple nous détaillons seulement la deuxième version de l'algorithme de refroidissement, présentée dans le Tableau 26.

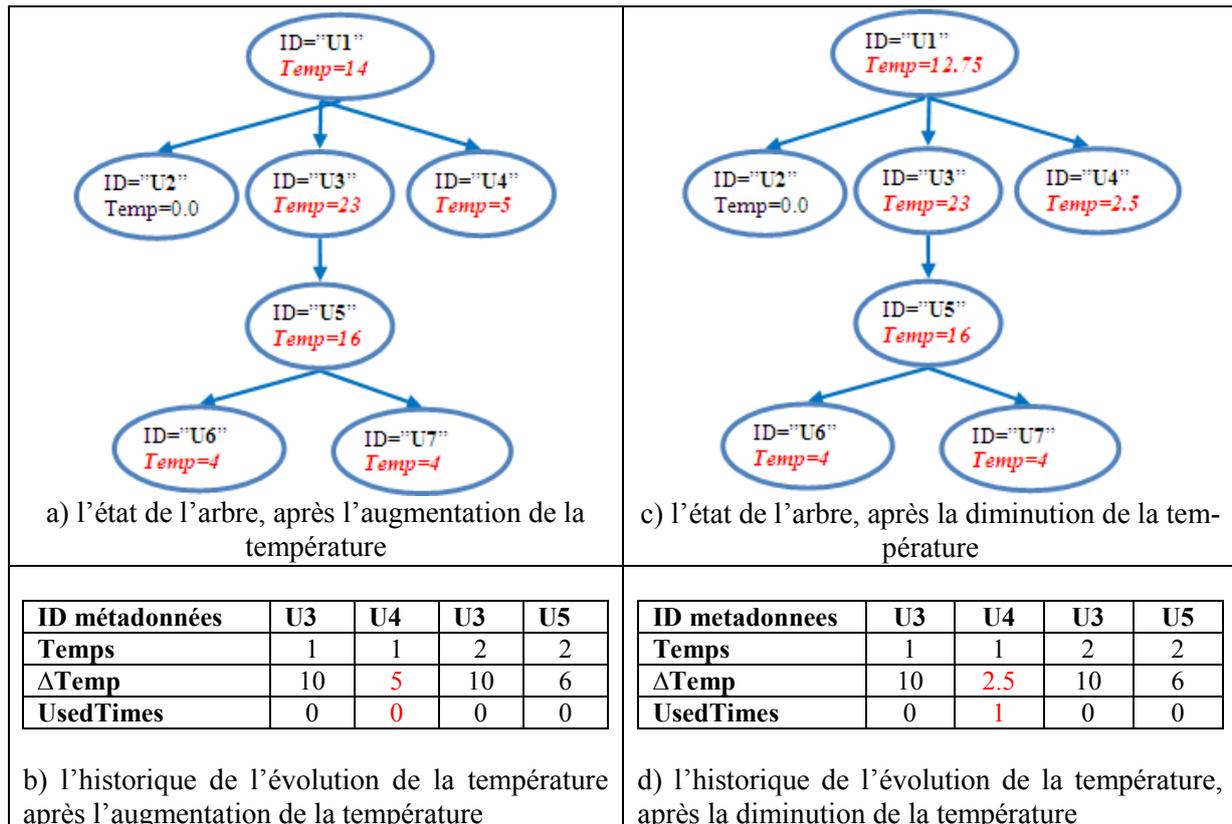


Figure 33 – L'état final de l'arbre après le calcul de la température pour t=2

La sélection du nœud U4 est la première étape de l'algorithme de diminution. Ensuite on regarde dans l'historique pour voir quelle est la dernière période où cet élément a été utilisé. Il s'agit de la période t=1. On regarde dans l'historique combien de fois cette valeur a été employée dans le processus de diminution. Pour U4, cette valeur n'a jamais été utilisée, donc elle peut être utilisée cette fois et le paramètre nombre de fois est incrémenté dans l'historique. La valeur qui va être soustraite de la température actuelle est $dec = \gamma * 5 = 0.5 * 5 = 2.5$. Cette valeur est mise à jour dans l'historique et n'est propagée qu'au niveau du parent de U4, car ce nœud n'a pas de fils. La Figure 33 c) présente le résultat obtenu suite à l'exécution de l'algorithme de diminution de la température, en présentant l'état de la température de chaque élément mais aussi celui de l'historique.

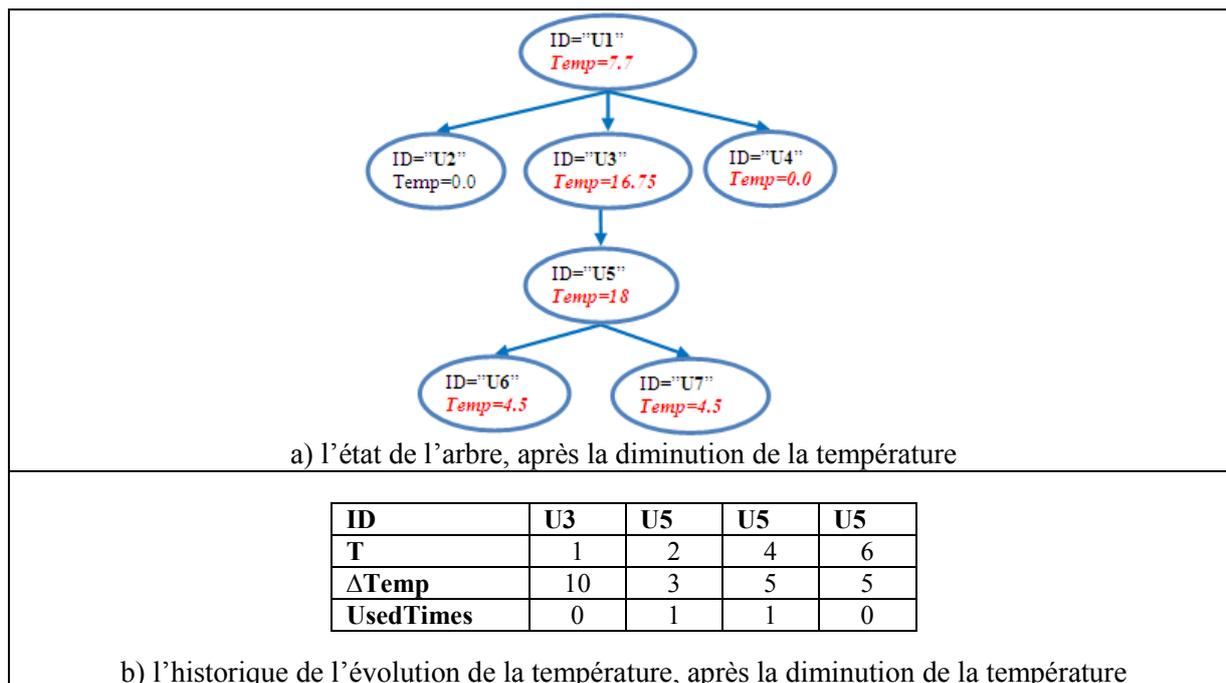


Figure 34 – L'état final de l'arbre après le calcul de la température pour $t=6$

Le résultat final de l'application de l'algorithme de calcul de la température pour les 6 périodes prises en compte dans notre exemple, est affiché dans la Figure 34 a). Cette figure présente aussi l'état de l'historique de changement de la température. Dans cet historique on observe que les identifiants des nœud U4 et U3, pour la période $t=2$, ont disparu parce qu'ils ont été utilisés de manière progressive dans le refroidissement de ces éléments.

Par la suite nous allons voir comment cette température peut être exploitée dans le cadre du système de gestion de contenus multimédias.

5.3.6. L'utilisation de la température et de la structure d'usage

Une fois la température calculée, elle peut être utilisée de différentes façons, dans plusieurs étapes du processus en « U » de gestion des documents multimédias pour : l'exécution de la requête, l'enrichissement de la requête, le reclassement des résultats mais aussi dans la présentation des résultats et des documents multimédias.

Pour l'exécution de la requête, les liens indirects peuvent être utilisés pour obtenir des réponses plus rapidement. En effet, cela prend moins de temps de rechercher de l'information dans une partie de la collection des métadonnées que dans la collection entière de métadonnées.

Les liens indirects peuvent être également utilisés pour proposer à l'utilisateur des mots pour sa requête. Pour faire cette proposition, nous utilisons la température des liens indirects et nous réalisons une classification des requêtes les plus utilisées par le groupe auquel l'utilisateur appartient et les mots de ces requêtes sont proposés à l'utilisateur.

Pour l'enrichissement de la requête, la température des liens indirects peut être utilisée. Plus précisément, au moment où un utilisateur exécute une requête, celle-ci est exécutée sur la collection des métadonnées et une liste de résultats est obtenue. En analysant cette liste, et plus précisément les

liens indirects des documents qui sont cliqués par l'utilisateur, nous obtenons d'autres mots qui ont été utilisés par les autres membres du même groupe que l'utilisateur courant.

Dans le reclassement des résultats, la température de la racine des métadonnées des documents est utilisée en favorisant les documents qui ont une température plus élevée. Si l'utilisateur qui exécute la requête appartient à un groupe d'utilisateurs déjà identifié dans le système, alors la température des documents sélectionnés correspondant à ce groupe est utilisée, sinon, c'est la température moyenne des documents qui est utilisée. En classant les documents en fonction de la température, nous obtenons un classement des documents en partant du plus populaire au moins populaire. Ce classement s'ajoute au classement des documents pour la requête de l'utilisateur. Pour la combinaison de ces deux classements, plusieurs stratégies bien connues dans la recherche d'information et la fusion des données peuvent être utilisées. Si la fonction d'appariement du système fournit la liste des résultats avec les scores de pertinence des documents pour la requête, alors ce score peut être combiné avec le score obtenu pour la température (obtenu suite à une normalisation de celle-ci), en utilisant l'équation (10), ou des fonctions classiques telles-que CombSUM, CombANZ, CombMNZ, CombMIN, CombMAX ou CombMED. Si la fonction d'appariement ne renvoie que le classement des documents sans aucune information sur le score de pertinence, un autre type de méthode de combinaison des ces deux classements peut être employé, tel que la méthode Borda Count, le comptage de Condorcet ou la méthode rank position. Ces fonctions peuvent être utilisées en ne privilégiant aucun des deux classements (en considérant que la pertinence d'un document pour une requête a la même importance que sa popularité). Mais un compromis peut être réalisé entre les deux, pour favoriser, soit la pertinence (en utilisant un paramètre $\varphi > \omega$), soit la popularité (en utilisant un paramètre $\varphi < \omega$).

$$reclassementScore(doc, query, user) = \varphi * score(doc, query) + \omega * \frac{temp_{group(user)}^{doc}(\Delta t)}{\sum_{i=0}^n temp_{group(user)}^i(\Delta t)} \quad (10)$$

D'un autre côté la température peut être utilisée dans la personnalisation de la présentation de la liste de résultats et des documents multimédias également. De manière habituelle, une liste de résultats est composée par des liens vers les documents pertinents pour la requête qui sont accompagnés par des descriptions qui aident l'utilisateur dans sa décision sur la pertinence d'un document pour sa requête. Cette description est composée par des métadonnées mais aussi par une image (une vignette plus précisément). Pour le choix de cette description, la température des métadonnées peut être utilisée. Il n'y a que certaines métadonnées qui sont significatives pour la description du document et qui apportent de l'information supplémentaire sur le document (e.g., il ne sert à rien d'afficher des informations techniques des métadonnées Exif à côté des images). Parmi ces métadonnées, le choix peut se faire de manière automatique en fonction de leur température correspondant à un groupe d'utilisateurs. Donc pour chaque groupe d'utilisateurs, les métadonnées affichées à côté du document peuvent être différentes si leurs températures sont différentes par groupe. En ce qui concerne la vignette, elle peut être choisie parmi les images ou les frames qui ont la température la plus élevée dans le document.

La température peut être aussi prise en compte dans la présentation du document. A tout moment les médias qui composent le document peuvent être mis en valeur différemment, en fonction de leur température. Les médias qui ont une température plus élevée sont mis en avant.

5.4. Bilan

Dans ce chapitre nous avons présenté nos propositions pour la prise en compte de l'usage dans un système de gestion des documents multimédias et des interactions des utilisateurs avec le document multimédia et avec les métadonnées, dans le cadre de la gestion de métadonnées multimédias.

Dans un premier temps, nous nous sommes intéressés à la façon dont les métadonnées sont générées et nous avons proposé la prise en compte de l'usage dans l'indexation des contenus multimédias à travers la réalisation de l'indexation en deux temps : dans un premier temps à l'acquisition des contenus multimédias par le système et dans un second temps au moment de l'exécution d'une requête de l'utilisateur. Outre le fait d'intégrer l'usage dans la création des métadonnées, nous nous sommes intéressés à la réduction de la consommation des ressources dans le cadre d'un système distribué de gestion des contenus multimédias. La réalisation de cette indexation a amené plusieurs propositions :

- la création d'un résumé de métadonnées ;
- la définition d'un algorithme de sélection des algorithmes d'indexation en fonction d'une requête de l'utilisateur ;
- la prise en compte des conditions environnementales pour la sélection des algorithmes d'indexation ;
- la prise en compte de l'historique des requêtes exécutées dans le système pour l'adaptation de l'indexation aux besoins des utilisateurs.

Nous avons étendu le format de métadonnées défini dans le Chapitre 4, par une structure d'usage et l'ajout du concept de température, pour prendre en compte l'usage que les utilisateurs font des documents.

Nous avons introduit deux types de structures pour les documents multimédias : la structure d'usage et la structure dynamique. La structure d'usage crée des liens entre les différentes parties du document multimédia, de la même façon que la structure hypermédia. En ce qui concerne la structure dynamique, elle peut être associée à tout type de structure.

Nous avons défini un algorithme d'augmentation de la température dans le temps en fonction du comportement des utilisateurs au moment de l'interaction avec le système et avec les documents multimédias. L'algorithme d'augmentation de la température peut être appliqué pour chaque indicateur d'intérêt de l'utilisateur envers un document, considéré comme étant important et fiable dans le domaine d'application. Pour respecter au mieux le principe physique de la température, nous avons proposé un algorithme de propagation de la température, mais aussi de refroidissement de celle-ci.

Nous avons appliqué l'algorithme d'augmentation de la température dans le cadre d'un système de gestion de contenus multimédias en prenant en compte les interactions des utilisateurs avec une liste de résultats, mais également avec les documents multimédias.

Dans la partie suivante de ce manuscrit, nous abordons l'implémentation et l'expérimentation de nos propositions dans le cadre de deux domaines différents : le projet LINDO et le domaine du commerce électronique.

Partie 3 : Les validations expérimentales

Introduction

Dans cette troisième et dernière partie du manuscrit, nous présentons l'implémentation réalisée dans le cadre de cette thèse afin de valider les propositions présentées dans le Chapitre 4 et le Chapitre 5.

Dans un premier temps, dans le Chapitre 6, nous détaillons les implémentations réalisées dans le cadre du projet LINDO. Ces implémentations nous ont permis de valider le format de métadonnées proposé et la prise en compte de l'usage dans la création des métadonnées. Ce projet nous a imposé certaines contraintes auxquelles nous avons dû nous adapter.

Parce que l'intégration de l'utilisateur dans le cadre de la gestion des métadonnées et l'analyse des interactions des utilisateurs avec les documents multimédias ne faisaient pas partie des objectifs du projet, nous avons cherché un autre cas d'utilisation pour valider l'enrichissement des métadonnées. Pour cela nous nous sommes tournés vers un autre projet qui enregistre toutes les actions que les utilisateurs réalisent sur un site de vente en ligne. Nous avons exploité ces interactions pour le calcul de la température. Ce deuxième cas d'utilisation est présenté dans le Chapitre 7.

Chapitre 6. Implémentation d'un système distribué de gestion des documents multimédias

6.1.	<i>Introduction</i>	141
6.2.	<i>L'architecture générique du système proposée dans le cadre du projet</i>	141
6.2.1.	Les composants des serveurs distants	143
6.2.2.	Les composants du serveur central	144
6.3.	<i>L'architecture et l'implémentation du démonstrateur final du projet</i>	145
6.3.1.	L'architecture logicielle du système	146
6.3.2.	Les descriptions adoptées pour les ressources	148
6.3.2.1.	La description des serveurs distants	148
6.3.2.2.	La description des algorithmes d'indexation	148
6.3.2.3.	La description des contenus multimédias	149
6.3.3.	Les algorithmes d'indexation disponibles	150
6.3.3.1.	L'algorithme d'indexation audio	150
6.3.3.2.	Les algorithmes d'indexation vidéo	151
6.3.3.3.	L'algorithme speech to text	152
6.4.	<i>L'implémentation du Metadata Engine</i>	153
6.4.1.	L'intégration des métadonnées	153
6.4.2.	Le résumé des métadonnées.....	154
6.4.3.	L'exécution d'une requête	157
6.5.	<i>Bilan</i>	158

6.1. Introduction

Dans ce chapitre, nous présentons l'implémentation réalisée dans le cadre du projet LINDO tout en insistant sur le travail que nous avons effectué, et comment il a été intégré dans l'architecture finale du projet. Notre principale contribution se situe dans la gestion de métadonnées multimédias dans des serveurs distants, au sein du module qui est dédié à cette gestion.

Nous commençons ce chapitre par la présentation générale de l'architecture distribuée définie dans le cadre du projet, tout en détaillant chaque module. Nous présentons ensuite l'architecture du démonstrateur final.

Dans la dernière partie du chapitre, nous décrivons le module dont nous étions responsables, et dans lequel nous avons développé les fonctionnalités liées à la création des métadonnées.

6.2. L'architecture générique du système proposée dans le cadre du projet

L'idée principale du projet la création d'une architecture qui permet l'utilisation de plusieurs serveurs différents en termes de fonctionnalités et de caractéristiques, et qui peut être utilisée dans plusieurs domaines d'application tels le broadcast ou la vidéosurveillance. L'architecture doit être

applicable pour des domaines du temps réel, mais aussi pour des domaines classiques où le traitement des données se fait hors connexion. Une architecture distribuée a donc été choisie, avec un serveur central qui a une vision globale sur le système. Cette solution architecturale est divisée en deux composantes principales :

- des *serveurs distants* qui acquièrent, indexent et stockent les contenus multimédias ;
- un *serveur central* qui a une vision globale de l'ensemble du système et qui orchestre les processus d'indexation et de recherche.

Dans cette architecture, les serveurs distants sont indépendants les uns des autres. Chaque serveur peut être différent en termes de contexte, de capacités, de domaine d'application, de son emplacement, mais également en termes de fonctionnalités. Par exemple, certains serveurs distants peuvent stocker et indexer les contenus multimédias en temps réel, tandis que d'autres peuvent procéder à une indexation hors connexion. Dans le cadre de cette architecture, des serveurs peuvent être enlevés ou ajoutés à tout moment, sans avoir à arrêter le système. Une autre fonctionnalité importante des serveurs distants est celle de y pouvoir y déployer et exécuter des algorithmes d'indexation alors que le système est en marche, sans arrêter le serveur.

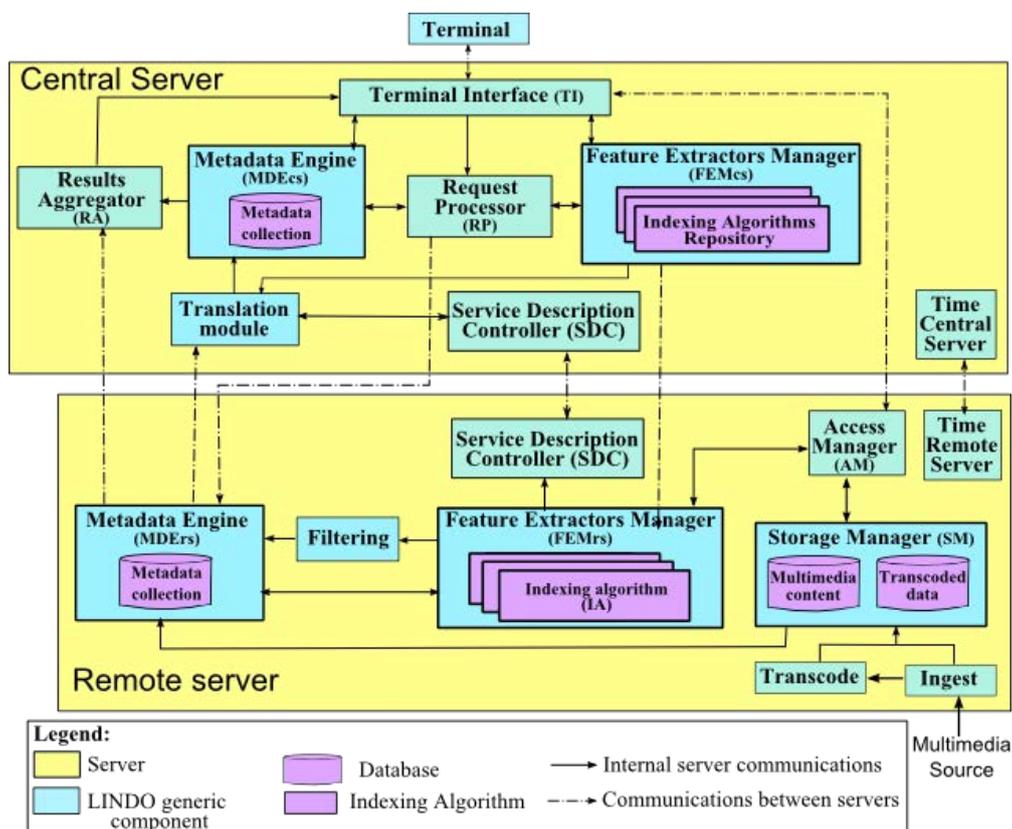


Figure 35 – L'architecture proposée par le projet LINDO (Brut, Codreanu, Dumitrescu, et al., 2011)

Pour pouvoir réaliser la diminution de la consommation des ressources, le serveur central est capable de déployer et exécuter des algorithmes d'indexation sur certains serveurs distants tout en gardant les serveurs en marche, et sans une intervention directe de l'utilisateur. Au moment de l'exécution d'une requête, le serveur central est capable de ne sélectionner que les serveurs distants qui sont susceptibles d'avoir des réponses à cette requête, et donc il n'envoie la requête qu'à ceux-ci.

Pour réaliser cette sélection, le serveur central dispose d'une vision globale du système à travers les descriptions des serveurs distants et à travers une version concise de métadonnées extraites à partir de leurs contenus.

La Figure 35 présente l'architecture proposée ainsi que les modules qui sont définis pour chaque type de serveur, qu'il soit central ou distant. La partie supérieure de la figure concerne le serveur central, tandis que la partie inférieure détaille les composants d'un serveur distant. Pour chaque module, nous avons défini une interface et des fonctionnalités bien précises. Nous présentons en quelques mots chaque module. Plus de détails sur les interfaces de chacun d'entre eux se trouvent dans les livrables réalisés dans le cadre du projet.

6.2.1. Les composants des serveurs distants

Comme nous l'avons dit, un serveur distant stocke et indexe tous les contenus multimédias acquis en temps réel ou non, et fournit des réponses à certaines requêtes reçues depuis le serveur central. Pour réaliser ces fonctionnalités, plusieurs modules ont été définis et composés :

- Le **Storage Manager (SM)** stocke les contenus multimédias acquis, en temps réel ou non. Grâce au module **Transcode**, un contenu multimédia peut être converti en plusieurs formats, avec des qualités et des codages différents, ce qui permet à un utilisateur final de télécharger différents encodages d'un contenu.
- L'**Access Manager (AM)** fournit des méthodes pour accéder aux contenus multimédias stockés dans le SM. Ce module peut également sélectionner qu'une partie d'un contenu multimédia. Par exemple, étant donné deux estampilles temporelles, le module AM peut sélectionner à partir d'une vidéo, le clip compris entre ces deux estampilles. Cette fonction est utile lorsque l'utilisateur veut voir un événement spécifique, et donc ne pas accéder à toute la vidéo et chercher par lui-même l'événement. De plus, cette fonctionnalité peut être utilisée pour sélectionner qu'une partie du document pour être indexée.
- Le **Feature Extractors Manager (FEMrs)** est en charge de la gestion et de l'exécution d'un ensemble d'algorithmes d'indexation sur le contenu multimédia acquis. A tout moment, de nouveaux algorithmes peuvent être téléchargés dans ce module, mis à jour ou supprimés, si nécessaire. Il peut soit exécuter en permanence certains algorithmes sur l'ensemble des contenus acquis soit les exécuter à la demande et uniquement sur certains contenus multimédias. Une autre fonctionnalité importante de ce module est la sélection automatique des algorithmes d'indexation qui obtiennent les meilleures performances dans le contexte d'exécution actuel. La gestion des algorithmes d'indexation est basée sur une description technique et sémantique de ces algorithmes définis dans le cadre du projet (Codreanu et al., 2011)
- Le module **Filtering** comprime les sorties d'un algorithme d'indexation qui peuvent contenir des métadonnées redondantes ou inutiles. Par exemple, si un algorithme d'indexation identifie la même personne dans un certain nombre de frames consécutifs, alors ce module concatène ces frames dans une période pour dire que la personne a été identifiée entre le frame 1 et le frame n. De cette façon la dimension de la base de données des métadonnées peut être réduite sans perte d'information.
- Le **Metadata Engine (MDErs)** recueille et agrège toutes les métadonnées extraites à partir des contenus multimédias stockés par le serveur. Les métadonnées stockées dans ce module peuvent être interrogées afin de récupérer l'information désirée. Dans le cadre du projet, le modèle de métadonnées présentées dans le Chapitre 4 a été utilisé.

- Le **Time Client** est en charge avec la synchronisation de l'heure entre les serveurs distants et le serveur central.
- Le **Service Description Controller** (*SDCrS*) stocke la description du serveur distant, comme son emplacement, ses capacités, les logiciels installés et le contexte d'acquisition des médias. En plus de cette information, ce module contient également des renseignements sur les algorithmes d'indexation qui ont été exécutés et sur quels contenus multimédias ils ont été appliqués.

Parmi ces modules, certains ne sont pas obligatoires, c'est le cas du module de Filtering, le Transcode ou le Time Client. Pour les autres modules, certaines fonctionnalités sont plus importantes que d'autres et donc elles sont obligatoires. Par exemple, le MDERS doit pouvoir exécuter des requêtes et concaténer toutes les métadonnées afférentes à un document multimédia, mais ce module a la possibilité de pas envoyer au serveur central une version concise des métadonnées qu'il gère.

6.2.2. Les composants du serveur central

Comme nous l'avons dit, le serveur central a une vision globale du système distribué. Il contrôle le processus d'indexation qui s'effectue sur les serveurs distants. Il peut soit répondre directement aux requêtes des utilisateurs, soit envoyer ces dernières à des serveurs distants qui peuvent contenir des résultats pertinents. Une différence majeure entre le serveur central et un serveur distant est que le serveur central ne stocke pas et n'indexe pas des contenus multimédias. En fait, le serveur central peut déployer, à la demande, sur les serveurs distants, des algorithmes d'indexation. Il est également en charge du filtrage du contenu multimédia qui doit être indexé, basé sur la requête de l'utilisateur. Il gère également les utilisateurs qui utilisent le système. Pour pouvoir mettre en œuvre toutes ces fonctionnalités, le serveur central est composé des éléments suivants :

- Le **Terminal Interface** (*TI*) fournit l'interface graphique du système. Dans la TI, un utilisateur peut introduire ses requêtes, en langage naturel ou en tant que mots-clés, c'est aussi dans ce module qu'est géré l'affichage des résultats de la requête. A l'intérieur de la TI, plusieurs fonctions ont été développées afin de visualiser des collections de métadonnées, d'installer, de déployer et d'exécuter à distance des algorithmes d'indexation. Un module d'identification intégré dans la TI a pour objectif d'identifier chaque utilisateur et d'afficher une interface visuelle différente en fonction du rôle de chaque utilisateur (e.g., simple utilisateur ou administrateur du système) .
- Le **Metadata Engine** (*MDEcs*) contient des informations relatives aux serveurs distants. Il contient certaines métadonnées associées aux contenus multimédias et extraites à partir des serveurs distants, des informations contextuelles sur l'ensemble du système, des informations sur les contextes d'acquisition des contenus multimédias sur les différents serveurs distants, les descriptions des serveurs distants (tels que leurs emplacements, leurs capacités), mais aussi d'autres connaissances supplémentaires qui peuvent être utilisées pour l'exécution des requêtes.
- Le **Service Description Controller** (*SDCcs*) recueille toutes les descriptions de serveurs distants, et les agrège à la collecte des métadonnées dans les MDEcs.
- Le **Feature Extractors Manager** (*FEMcs*) gère l'ensemble des algorithmes d'indexation utilisés dans le système. Il peut déployer, mettre à jour et supprimer tout algorithme d'indexation, sur un serveur distant, à tout moment. En outre, ce module peut exécuter ou arrêter à tout moment un algorithme d'indexation.

- Le **Request Processor** (*RP*) traite la requête de l'utilisateur reçue à partir du TI, afin d'extraire les caractéristiques multimédias exigées. Il exécute les requêtes sur le Metadata Engine du serveur central (MDEcs) ou il les transmet au Metadata Engine des serveurs distants (MDErs). S'appuyant sur la requête de l'utilisateur et sur les informations contenues par les MDEcs, ce module décide si une nouvelle indexation doit être exécutée sur certains serveurs distants, et, si c'est le cas, il sélectionne les meilleurs algorithmes d'indexation qui doivent être utilisés.
- Le **Results Aggregator** (*RA*) agrège les résultats reçus depuis les Metadata Engine, les MDErs et MDEcs. En fait, à partir d'une requête de l'utilisateur, il regroupe toutes les réponses disponibles à cet instant donné et les envoie au module Terminal Interface pour les afficher.
- Le **Translation** Module a pour rôle d'homogénéiser les données stockées dans le Metadata Engine du serveur central. En effet, des modèles différents peuvent être utilisés par les serveurs distants, pour stocker les métadonnées obtenues après l'indexation des contenus multimédias, ou pour décrire leurs propres caractéristiques. Par conséquent, ce module unifie toutes les descriptions afin de fournir un point de vue global du système.
- Le **Time Server** fournit un temps unique, de référence, qui est utilisé pour synchroniser tous les serveurs distants.

Parmi ces modules, il y en a certains qui ne sont pas obligatoires, c'est le cas du module de Translation ou du module Time Server.

Dans le cadre du projet nous avons défini des interfaces pour chacun des modules contenus par l'architecture, qu'ils soient au niveau du serveur central ou des serveurs distants. Ces interfaces spécifient la façon dont chaque module doit communiquer et échanger des données avec les autres modules de l'architecture. Nous nous sommes concentrés sur des interfaces génériques, en décrivant les protocoles à utiliser et les fonctionnalités attendues sans entrer dans les détails de l'implémentation effective des modules. La liste complète de ces interfaces se trouve dans (LINDO consortium, 2010).

6.3. L'architecture et l'implémentation du démonstrateur final du projet

Le projet a abouti à la réalisation d'un système fonctionnel qui intègre un serveur central et trois serveurs distants localisés dans des pays différents et surtout qui ont des domaines d'application différents. Du point de vue de l'emplacement géographique, un serveur distant était situé à Madrid, et les trois autres étaient localisés à Paris.

Le serveur de Madrid était utilisé pour stocker des contenus vidéos acquis en temps réel par une camera de vidéo surveillance située dans le couloir d'une entreprise. Un des serveurs distants de Paris gère en permanence des contenus de vidéo surveillance acquis en temps réel par une camera située dans un bureau et des bandes sonores acquises par un microphone situé dans le même endroit. Le troisième serveur stocke des vidéos d'émissions télévisées fournies par un des partenaires du projet. L'architecture de ce système proposée dans la Figure 35 présente les modules de l'architecture qui ont été utilisés pour chaque serveur.

Parmi les modules définis dans l'architecture globale, certains n'ont pas été développés car jugés inutiles dans les cas d'utilisation considérés. Sur le serveur central, le module de Translation n'a pas été mis en œuvre car le modèle de métadonnées a été utilisé sur tous les serveurs du système (central ou distants). Pour la synchronisation du temps dans tout le système, une technologie déjà existante a été utilisée et par conséquent les modules Time Server et Time Client n'ont pas été développés. En ce qui concerne l'Access Manger, certaines de ses fonctionnalités étant déjà fournies par la technolo-

gie utilisée pour le Storage Manager, ce module n'a pas été redéfini. C'est le même cas pour les modules d'Ingest et Transcode qui sont inclus dans le Storage Manager également. Sur le serveur dédié au broadcast, nous disposons d'un seul algorithme d'indexation qui ne nécessitait pas de filtrage, et par conséquent, ce module n'a pas été inclus dans ce serveur.

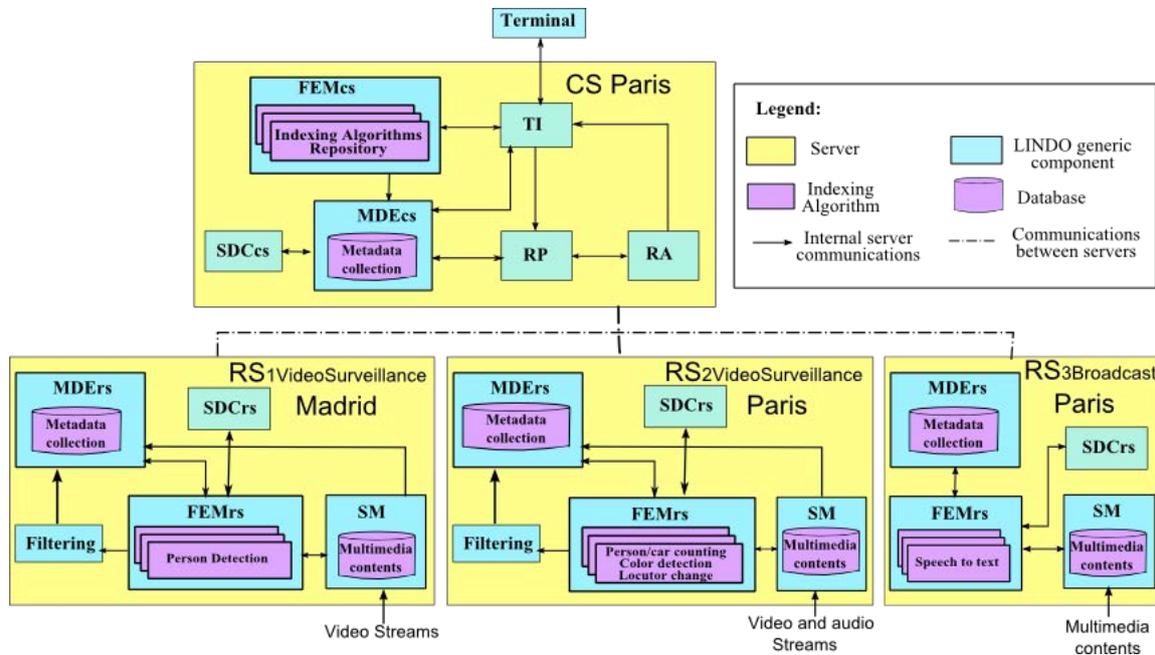


Figure 36 – L'architecture du démonstrateur final du projet LINDO (Brut, Codreanu, Dumitrescu, et al., 2011)

Par la suite nous détaillons l'architecture logicielle de ce système en présentant les technologies employées par les différents partenaires.

6.3.1. L'architecture logicielle du système

Du point de vue matériel, chaque serveur est composé par au moins un ordinateur. Sur le serveur de vidéosurveillance situé à Paris, les algorithmes d'indexation utilisés avaient des contraintes d'exécution assez fortes : un d'entre eux devait être exécuté sur le système d'exploitation Linux, et un autre sur Windows. Pour des problèmes d'accès aux ressources en temps réel, nous avons utilisé deux ordinateurs différents, un par système d'exploitation. On aurait pu employer des machines virtuelles, mais ceci a pour conséquence l'augmentation du temps d'indexation. Les modules de stockage des contenus multimédias et des métadonnées sont installés sur un autre ordinateur sous Linux (demandé par le Storage Manager). Ces choix ont été imposés par le fait que ce serveur indexe le contenu multimédia en temps réel. L'architecture matérielle de ce serveur distant est présentée dans la Figure 37.

Tous les modules du serveur de vidéosurveillance situé à Madrid sont installés sur un seul ordinateur sous Linux.

Sur l'ordinateur utilisé pour le serveur de broadcast, le système d'exploitation Windows 7 est installé. L'algorithme d'indexation et le Metadata Engine sont installés sur ce système d'exploitation. Le module Storage Manager n'est employé que pour le serveur de vidéo surveillance,

qui est exécuté sur une machine virtuelle. Cette solution était envisageable parce que l'indexation sur ce serveur est réalisée off-line.

Pour le codage des différents modules du système, des technologies variées ont été choisies par les partenaires du projet : le Storage Manager était développé en C, alors que tous les autres modules étaient codés en Java. Pour le module de gestion de métadonnées, nous avons employé du Java et pour stocker les métadonnées, nous avons utilisé Oracle Berkeley DB XML⁴⁶. Sur le serveur central, l'interface du système a été développée tant que site web.

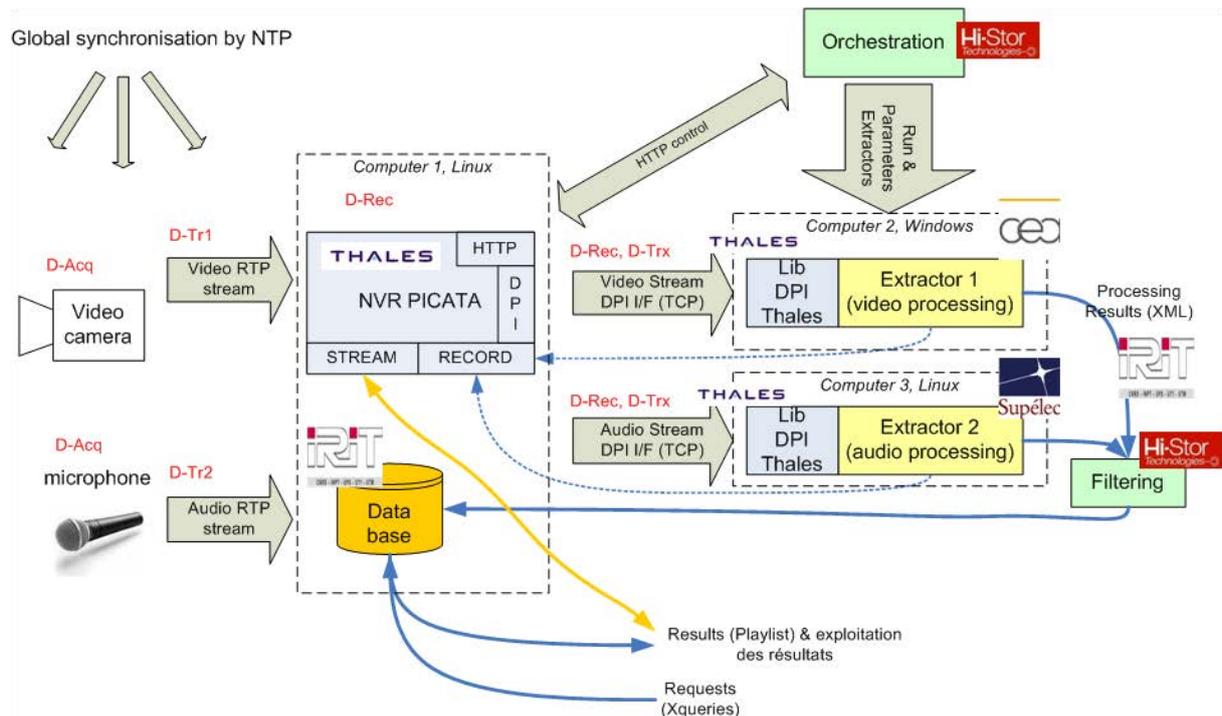


Figure 37 – L'architecture logicielle du serveur distant situé à Paris

Le module Feature Extractor Manager est basé sur la plateforme OSGi implémenté sous la licence Apache⁴⁷ pour pouvoir exécuter des algorithmes d'indexation à la demande. Pour le serveur distant situé à Paris, le FEM est réparti entre les trois serveurs.

La synchronisation des différents serveurs du système a été réalisée à travers le protocole Network Time Protocol (NTP)⁴⁸. En ce qui concerne les communications dans le cadre du système, toutes celles entre le serveur central et les serveurs distants se réalisent au travers de Services Web. L'accès au contenu multimédia stocké dans le Storage Manager s'effectue à travers les protocoles : DPI et RTP/RTCP.

La communication entre le FEM, le Filtering et le Metadata Engine est réalisée en utilisant le Java Message Service (JMS)⁴⁹. Cet API est utile surtout pour l'envoi de fichier XML d'un module à un autre. La communication entre le Storage Manager et le Metadata Engine est réalisée au travers des Services Web.

⁴⁶ <http://www.oracle.com/technology/products/berkeley-db/xml/index.html>

⁴⁷ <http://felix.apache.org/site/index.html>

⁴⁸ <http://www.ntp.org/>

⁴⁹ <http://java.sun.com/products/jms/overview.html>

6.3.2. Les descriptions adoptées pour les ressources

Un élément important pour le bon fonctionnement du système est constitué par les différentes descriptions adoptées pour les ressources.

6.3.2.1. La description des serveurs distants

Pour décrire les serveurs distants la description présentée dans la section 5.2.3 a été utilisée. Dans le cadre du projet, les algorithmes d'indexation disponibles n'avaient pas de contraintes d'exécution liées aux conditions environnementales et donc dans les descriptions des serveurs distants, il n'y a pas de paramètres de contexte.

Tableau 32 – Un extrait de la description du serveur de vidéosurveillance situé à Paris

```
<RemoteServer id="rs1" name="RS2 VideoSurveillance">
  <localisation>
    <description>bureau 322, Paris, France</description>
  </localisation>
  <description>Gère le contenu acquis par les caméras situées dans le bureau 322.</description>
  <devices>
    <camera id="c1Paris">
      <description>localisée dans le coin nord du bureau</description>
    </camera>
  </devices>
  <indexingAlgorithms>
    <indexingAlgorithm id="ia2rs1" mediaType="video" name="détection de personne"
type="implicit" status="started">
      <description>Détection des personnes à l'intérieur et leur couleur prédominante. </description>
    </indexingAlgorithm>
  </indexingAlgorithms>
</RemoteServer>
```

6.3.2.2. La description des algorithmes d'indexation

Pour une meilleure gestion des algorithmes d'indexation, il est nécessaire de connaître leurs caractéristiques et leurs contraintes d'exécution. Dans l'état de l'art, il y a des travaux qui se focalisent sur la description sémantique de ces algorithmes du point de vue de ce qu'ils prennent en entrée et ce qu'ils produisent en sortie (Roman et al., 2005), (Arndt et al., 2007). Dans un premier temps, nous avons utilisé la description offerte par l'ontologie WSMO (Roman et al., 2005) dans (Brut, Sedes, et al., 2009). Cette ontologie se focalise surtout sur la description des algorithmes sous la forme de Services Web, mais elle n'est pas facilement exploitable pour la gestion des algorithmes d'indexation qui ne sont pas définis en tant que Services Web. Pour résoudre ce problème dans le cadre du projet, (Codreanu et al., 2011) propose un modèle générique de description des algorithmes d'indexation.

Un exemple de description d'un algorithme d'indexation est fourni dans le Tableau 33. Cette description est associée à un algorithme qui détecte, dans des vidéos, des personnes et leur couleur prédominante. Cet algorithme est configurable et il prend ses paramètres depuis un fichier XML. Il produit en sortie des métadonnées. Il ne s'exécute que dans le système d'exploitation Linux. En ce qui

concerne l'exécution, une autre contrainte est qu'il n'accepte que des vidéos d'un certain format (AVI ou MPEG).

Tableau 33 – Un extrait de la description d'un algorithme d'indexation

```
<AlgorithmModel AlgoName="Détection de personne" MediaType="Vidéo" Id="Algo1">
  <InputParameters>
    <InputParamFileFormat>xml</InputParamFileFormat>
  </InputParameters>
  <OutputObject Type="Metadata">
    <MetadataObject>
      <MetadataObjectDescription> Algorithme de détection des personnes et des couleurs
    </MetadataObjectDescription>
  </MetadataObject>
</OutputObject>
<ExecutionConstraints>
  <MMConstraints>
    <DataFormat>MPEG, AVI</DataFormat>
  </MMConstraints>
  <PlatformConstraints>
    <OS>Linux</OS>
  </PlatformConstraints>
</ExecutionConstraints>
</AlgorithmModel>
```

6.3.2.3. La description des contenus multimédias

Le principal objectif de cette description est de définir un format de métadonnées qui peut être utilisé pour la description de tout type de contenu multimédia, peu importe la façon dont ce dernier a été créé. Pour réaliser cela, l'intégration des standards des métadonnées les plus utilisés dans certains domaines d'application doit être privilégiée, tout en prenant en compte le format des métadonnées qui sont extraites à partir du contenu des documents multimédias par les différents algorithmes d'indexation.

En amont du projet, une étude a été réalisée par les membres industriels du projet pour sélectionner les standards de métadonnées les plus utilisés dans leurs domaines respectifs d'application. Suite à cette étude, nous avons sélectionné les standards suivants : DC, EXIF, XMP, DICOM, MXF, les métadonnées enregistrées dans les fichiers créés par les logiciels Adobe et Microsoft. Le modèle de métadonnées que nous avons défini dans le cadre du projet a été présenté dans le Chapitre 4. Ce modèle de métadonnées est utilisé dans les modules MDers sur les serveurs distants.

Tableau 34 – Un extrait des métadonnées associées à une vidéo

```
<document src="stream1">
  <video capturedBy="c1Paris">
    <object type="Personne" id="0">
      <localisation confidence="89">
        <period start_time="2010-07-28T11:07:35" end_time="2010-07-28T11:07:55"/>
        <area>entré du bureau</area>
      </localisation>
    </object>
  </video>
</document>
```

```
<property name="color">rouge</property>
</object>
</video>
</document>
```

Dans le Tableau 34 est présentée une partie des métadonnées extraites à partir d'un flux vidéo enregistré sur le serveur distant RServer-1 (sa description est présentée dans le Tableau 32) par la camera d'identifiant c1Paris. Ces métadonnées ont été extraites par l'algorithme d'indexation Algo1 (sa description est présentée dans le Tableau 33). Cet algorithme a identifié, avec une confiance de 89%, une personne à l'entrée du bureau. Cette personne, habillée en rouge, est identifiée pendant 20 secondes.

6.3.3. Les algorithmes d'indexation disponibles

Dans le cadre du projet, certains partenaires ont fourni des algorithmes d'indexation qui traitent des contenus audio et vidéo.

6.3.3.1. L'algorithme d'indexation audio

Cet algorithme a été fourni par l'équipe de SUPELEC⁵⁰. Il prend en entrée un flux audio et détermine le changement de locuteur. L'algorithme proposé ne peut pas retrouver l'identité des personnes qui parlent dans le contenu audio, mais il peut dire si c'est une même personne qui parle en appliquant un algorithme de clustering. Cet algorithme ne fonctionne qu'en différé sous le système d'exploitation Linux.

Le Tableau 35 fournit un exemple de sortie obtenue par l'exécution de cet algorithme d'indexation sur un contenu audio. Trois personnes ont été retrouvées dans le contenu, la première fait trois interventions, tandis que les deux autres n'ont fait qu'une intervention.

Tableau 35 – Exemple de sortie fournie par l'algorithme de traitement des contenus audio

```
<document src="sons/PAROLE3.wav">
<audio>
  <object ID="ID001" type="speaker">
    <localisation confidence="100">
      <period end_time="000:05.52" start_time="000:00.00"/>
      <segment matching="1" num="1"/>
    </localisation>
    <localisation confidence="9">
      <period end_time="000:37.32" start_time="000:05.52"/>
      <segment matching="1" num="2"/>
    </localisation>
  </object>
  <object ID="ID002" type="speaker">
    <localisation confidence="40">
      <period end_time="001:01.92" start_time="000:37.32"/>
      <segment matching="3" num="3"/>
    </localisation>
  </object>
</audio>
</document>
```

⁵⁰ <http://www.supelec.fr/metz>

```

</localisation>
</object>
</audio>
</document>

```

6.3.3.2. Les algorithmes d'indexation vidéo

Les algorithmes qui traitent les vidéos en temps réel, ont été fournis par CEA LIST⁵¹ : deux algorithmes qui sont dédiés pour le traitement de vidéos prises à l'intérieur et trois autres qui extraient des informations à partir de vidéos acquises à l'extérieur. Tous ces algorithmes fournissent en sortie des métadonnées qui sont déjà représentées dans le modèle que nous avons proposé. Le Tableau 36 présente un petit aperçu de ces algorithmes d'indexation. Ces algorithmes ne fonctionnent que sous Windows XP.

Tableau 36 – Les algorithmes d'indexation de contenus vidéo

	A l'intérieur	A l'extérieur
Intrusion	- Présence de personnes (valeurs : empty, présence)	- Présence de personnes et de véhicules (valeurs : empty, présence)
Comptage	- Nombre de personnes - Couleur dominante de la partie supérieure du corps (valeurs : red / yellow / green / blue / magenta / dark gray / light gray)	- Nombre de personnes - Couleur dominante de la partie supérieure (valeurs : red / yellow / green / blue / magenta / dark gray / light gray) - Nombre de véhicules - Couleur dominante du véhicule (valeurs : red / yellow / green / blue / magenta / dark gray / light gray)
Exemple de détection		

Tableau 37 – Première sortie XML pour la détection d'une intrusion

```

<document src="dpi://192.17.12.93/CamDemoCea_VIDEO_MP4V-ES_0">
  <video capturedBy="CamDemoCea">
    <object type="Person Intrusion" id="6" value="presence">
      <localisation confidence="100">
        <period start_time="2010-07-28T11:07:42.92"/>
        <area>bureau 322</area>
      </localisation>
    </object>
  </video>
</document>

```

Tableau 38 – Seconde sortie XML pour la détection d'une intrusion

```

<document src=" dpi://192.17.12.93/ CamDemoCea_VIDEO_MP4V-ES_0">
  <video capturedBy=" CamDemoCea">
    <object type="Person Intrusion" id="6" value="presence">
      <localisation confidence="100">
        <period start_time ="2010-07-28T11:07:42.92" end_time="2010-07 28T11:07:52.40"/>
        <area>bureau 322</area>
      </localisation>
    </object>
  </video>
</document>

```

⁵¹ <http://www-list.cea.fr/>

```
</localisation>
</object>
</video>
</document>
```

Ces algorithmes produisent un fichier XML en sortie dès qu'ils perçoivent un changement dans les détections qu'ils réalisent. Par exemple, pour l'intrusion, un fichier XML est généré dès qu'il y a quelqu'un dans la zone surveillée (Tableau 37) et un autre est généré au moment où il n'y a plus personne (

Tableau 38). Dans la gestion des métadonnées, nous devons prendre en compte cette particularité des algorithmes d'indexation.

6.3.3.3. L'algorithme speech to text

Cet algorithme a été développé par la société SGT⁵². Il prend en entrée, la partie audio associée à une vidéo et il transforme le discours en texte pour pouvoir le rechercher plus facilement. Les meilleures performances sont obtenues pour les contenus enregistrés dans des studios de télévision. Cet algorithme ne fonctionne que sur le système d'exploitation Windows 7. En ce qui concerne la langue du contenu, l'algorithme fonctionne aussi bien pour l'anglais que pour le français.

Un fichier XML, dans le format que nous avons proposé, est produit pour l'analyse de chaque minute du contenu. Cette façon de générer les métadonnées doit être prise en compte au moment de leur gestion par le Metadata Engine. Un petit extrait est présenté dans le Tableau 39.

Tableau 39 – Un extrait des métadonnées obtenues par l'algorithme speech-to-text

```
<document src="test22050_audio_PCMA_0">
  <audio>
    <Episode>
      <Section startTime="0,000" endTime="54,287">
        <Turn startTime="0,000" endTime="54,287">
          <Sync time="0,000"/>À quelques kilomètres de là en reportage de Sébastien an dernier
maillon fils sont 8h00 ce matin 300 C mètre dans le rôle de l'hôpital Damien ils portent des 500 atti-
rer l'attention ils seront au deuxième étage dans le service de chirurgie orthopédique
          <Sync time="19,097"/>un détenu âgé de 36 ans hospitalisés la veille
        </Turn>
      </Section>
    </Episode>
  </audio>
</document>
```

Par la suite, nous présentons plus en détails l'implémentation de la Metadata Engine pour la gestion des métadonnées.

⁵² <http://www.sgt.eu/index.php/en/>

6.4. L'implémentation du Metadata Engine

Pour le stockage des métadonnées et des descriptions des algorithmes d'indexation, nous avons utilisé système de gestion de base de données Oracle Berkeley DB XML, parce que c'est un système natif XML qui permet de stocker les documents XML et d'exécuter des requêtes XQuery. Ce module a été développé en Java. Pour la communication avec les autres modules du système, des Services Web ont été développés et le JMS a été inclus également.

6.4.1. L'intégration des métadonnées

Comme nous avons vu dans la section précédente, les algorithmes d'indexation utilisés dans le projet ont une sortie représentée dans le format de métadonnées que nous avons proposé. Pour être plus générique, nous avons développé une série de fichier XSLT avec des règles de transformation depuis d'autres formats de métadonnées vers notre format. En plus des transformations réalisées entre le modèle proposé et les standards de métadonnées que nous avons pris en compte au moment de la réalisation du modèle, nous avons créé des transformations depuis d'autres formats de métadonnées XML qui pourraient être obtenus par des algorithmes d'indexation.

Par exemple, nous avons pris le format de sortie du logiciel SimpleLPR⁵³ qui extrait les plaques minéralogiques des voitures. Le module est configurable dans le sens où, avant l'inclusion des métadonnées dans la base de données, une transformation XSLT peut être appliquée automatiquement, si nécessaire, pour assurer la compatibilité avec le modèle proposé. Le Tableau 40 présente un exemple de sortie XML produite par ce logiciel et sa transformation dans notre format de métadonnées.

Pour pouvoir gérer le fait que plusieurs algorithmes différents peuvent extraire, dans le cadre d'un document, des informations supplémentaires sur le même objet, nous avons mis au point un fichier XSLT qui réalise la fusion de toutes ces métadonnées. Chaque fois que des métadonnées doivent être incluses dans la base de données, cette opération de fusion est accomplie, sans ralentir le système qui fonctionne en temps réel.

Tableau 40 – Exemple de sortie du logiciel SimpleLPR et sa transformation

<pre><Results country="France"> <pic path="..\voiture1.jpg" lp="1234 AB 56" /> </Results></pre>	<pre><document src="..\voiture1.jpg"> <image> <object type="license plate" value="1234 AB 56"> <property name="country">France</property> </object> </image> </document></pre>
a) XML obtenu par le logiciel SimpleLPR	b) XML obtenu après la transformation

Dans le cas de la vidéosurveillance en général, il y a des lois qui stipulent que les contenus multimédias ne peuvent pas être stockés plus d'une certaine période de temps (e.g., 2 mois). Pour garder une cohérence entre les données du Storage Manager et les métadonnées du Metadata Engine, nous avons développé une opération d'effacement des métadonnées associées à un contenu multimé-

⁵³ <http://www.warelogic.com/>

dia, qu'il s'agisse de toutes les métadonnées ou seulement d'une partie. Cette fonctionnalité est mise à disposition du Storage Manager sous la forme d'un Service Web.

6.4.2. Le résumé des métadonnées

Avant d'inclure le résumé des métadonnées dans l'architecture du projet, nous avons voulu étudier son impact sur le traitement des requêtes dans un environnement distribué.

Dans un premier temps, nous avons implémenté notre proposition en Java en utilisant JENA⁵⁴ pour la gestion des descriptions RDF, et ARQ⁵⁵ pour interroger ces descriptions à l'aide du langage SPARQL. Pour être le plus exhaustif possible, nous générons, de façon aléatoire, les descriptions RDF, en précisant leur taille ainsi que leur densité, en utilisant RDFizer⁵⁶.

Dans le cadre de cette expérimentation, pour simuler le fait que les métadonnées sont liées à un serveur distant, les URI des contenus multimédias sont liés à chaque URI des serveurs distants qui leur sont associés. Afin de préserver cette information, pour chaque description RDF engendrée de façon aléatoire, nous avons sélectionné au hasard un élément de celle-ci et lui avons attribué le type Serveur.

Par la suite, nous avons évalué l'efficacité du résumé de métadonnées en enregistrant le temps de réponse aux requêtes spécifiées par un utilisateur. Nous proposons d'exécuter :

- une requête initiale sur une collection centralisée de métadonnées qui fusionne l'ensemble des collections distantes n'ayant pas été résumé. Nous exécutons également cette même requête sur une seule collection distante.
- une requête relaxée sur la collection centralisée de métadonnées ainsi que sur un résumé de celle-ci.
- une requête relaxée qui identifie les serveurs susceptibles de contenir l'information recherchée. Cette requête est exécutée sur la collection centralisée résumée de métadonnées.

Le Tableau 41 présente une requête SPARQL initiale (Requête 1), une requête relaxée (Requête 2) qui est obtenue de la Requête 1 en supprimant des éléments, et une requête relaxée qui cherche les serveurs distants contenant l'information recherchée (Requête 3). Nous avons choisi ces trois requêtes pour être le plus générique et exhaustif possible. La méthode utilisée pour relaxer la requête est très simple, le but étant de constater son influence sur le temps de réponse. Comprenant plus de variables, les trois autres requêtes présentées (Requêtes 4, 5 et 6) sont encore plus complexes. Le Tableau 42 et le Tableau 43 contiennent le temps de réponse, en ms, obtenu lors de l'exécution des requêtes sur différentes descriptions RDF.

Dans le Tableau 42, les résultats des expérimentations montrent que la taille des descriptions RDF a une forte influence sur le temps d'exécution de la requête. Par exemple, pour la Requête 1, on peut constater que le temps de réponse à cette requête sur une seule collection de métadonnées est inférieur au temps de réponse à la même requête sur l'ensemble des descriptions centralisées. Ce résultat ne fait que confirmer les travaux décrits dans (Pérez et al., 2006) qui démontre que la complexité d'évaluation d'une requête SPARQL (pour des requêtes simples) dépend non seulement de la taille des descriptions RDF mais aussi du nombre de variables de la requête.

⁵⁴ <http://jena.sourceforge.net>

⁵⁵ <http://jena.sourceforge.net/ARQ/>

⁵⁶ http://simile.mit.edu/wiki/Random_RDFizer/

Tableau 41 – Les requêtes utilisées pour l'expérimentation

SELECT * WHERE { ?s1 ?p1 ?o1 . ?o1 ?p2 ?o2 . ?o2 ?p3 ?o3 . } (a) Requête 1 : requête initiale.	SELECT * WHERE { ?s1 ?p1 ?o1 . ?o1 ?p2 ?o2 . } (b) Requête 2 : Requête 1 relaxée.	SELECT ?s1 WHERE { ?s1 <type> Serveur . ?s1 ?p1 ?o1 . ?o1 ?p2 ?o2 . } (c) Requête 3 : Requête 1 relaxée qui cherche des serveurs.
SELECT * WHERE { ?s1 ?p1 ?o1 . ?o1 ?p2 ?o2 . ?o2 ?p3 ?o3 . ?o3 ?p4 ?o4 . } (d) Requête 4 : requête initiale.	SELECT * WHERE { ?s1 ?p1 ?o1 . ?o1 ?p2 ?o2 . ?o2 ?p3 ?o3 . } (e) Requête 5 : Requête 4 relaxée.	SELECT ?s1 WHERE { ?s1 <type> Serveur . ?s1 ?p1 ?o1 . ?o1 ?p2 ?o2 . ?o2 ?p3 ?o3 . } (f) Requête 6 : Requête 4 relaxée qui cherche des serveurs.

Tableau 42 – Résultats de l'évaluation des requêtes 1, 2 et 3 en ms.

Nombre de serveurs	Taille d'une collection distante	Requête 1		Requête 2		Requête 3
		Résumé central degré=n	Collection distante	Résumé central degré=n	Résumé central degré=3	Résumé central degré=3
2	10	65	14	6	3	2
	20	284	90	20	6	4
	30	1030	453	62	11	4
	40	2960	1253	114	23	6
	50	6213	3099	216	45	7
5	10	96	11	10	5	4
	20	523	86	44	13	5
	30	2551	520	149	29	7
	40	6380	1243	284	37	9
	50	15663	3166	560	117	17

L'évaluation de la Requête 2 montre également que répondre à une requête relaxée se fait de manière plus rapide que répondre à une requête non-relaxée. De plus, répondre à une requête relaxée sur le résumé de métadonnées proposé est d'autant plus efficace que répondre à la même requête sur l'ensemble des métadonnées centralisées.

En ce qui concerne la Requête 3, les résultats montrent que lorsqu'une variable est fixée dans une requête relaxée (par exemple, pour chercher des serveurs) son temps d'évaluation est encore plus rapide, surtout lorsque celle-ci est évaluée sur le résumé proposé.

Le Tableau 43 présente l'évaluation de requêtes SPARQL contenant plus de variables, c'est-à-dire les requêtes 4, 5 et 6. Ces tests montrent l'évolution du temps de réponse en fonction de la complexité des requêtes. Il est important de souligner que le résumé de métadonnées est également efficace pour ce type de requêtes.

Tableau 43 – Résultats de l'évaluation des requêtes 4, 5 et 6 en ms.

Nombre de serveurs	Taille d'une collection distante	Requête 4		Requête 5		Requête 6
		Résumé central degré=n	Collection distante	Résumé central degré=n	Résumé central degré=3	Résumé central degré=3
2	10	150	47	15	4	3
	20	2037	886	183	7	3
	30	15083	7596	871	115	5
	40	54177	25573	2524	217	6
	50	170010	84516	6203	733	9
5	10	266	51	37	15	2
	20	5165	1160	412	61	4
	30	34653	7153	2057	188	7
	40	141310	27366	6372	922	13
	50	409200	80402	15265	1436	17

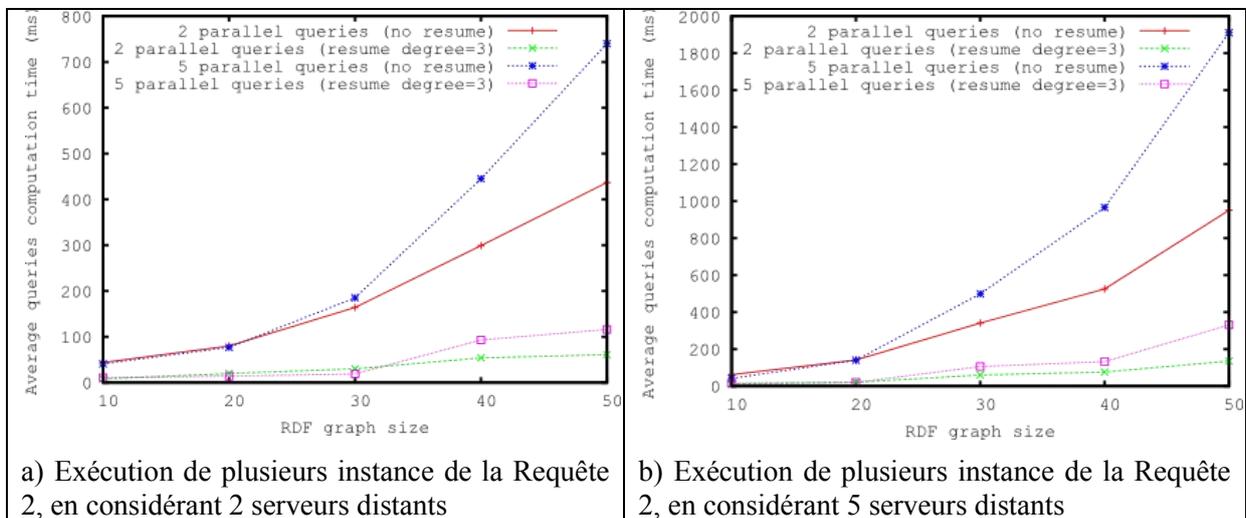


Figure 38 – Graphique représentant l'exécution en parallèle de plusieurs requetes

Les graphiques présentés dans la Figure 38 démontrent l'efficacité du résumé des métadonnées lorsque plusieurs requêtes sont exécutées simultanément sur le serveur central, c'est le cas en particulier, de la Requête 2. En outre, l'exécution de cinq requêtes en parallèle sur un résumé de métadonnées centralisé est plus efficace que l'exécution de deux requêtes en parallèle sur une collection de métadonnées centralisée qui n'a pas été résumée.

En conclusion, cette expérimentation montre que, l'utilisation d'un résumé sur le serveur central dans le cadre de métadonnées réparties sur différents serveurs distants, améliore les performances de réponse d'une requête. Par la suite, nous avons introduit le résumé de métadonnées dans l'implémentation du projet.

Pour la réalisation du résumé, nous avons implémenté des stratégies différentes pour chaque type de domaine d'application. Pour les serveurs distants de vidéo surveillance, le résumé est calculé toutes les demi-heures et est constitué par le nombre de types d'objet qui ont été détectés dans chaque contenu multimédia pendant la dernière demi-heure. Par exemple, le résumé calculé entre 10h00 et 10h30 est composé par le fait que 10 intrusions ont été détectées. Pour le serveur de broadcast, le résumé est calculé pour des intervalles égaux à 10% de chaque contenu. Il est constitué par les mots les plus utilisés dans cette période de temps.

6.4.3. L'exécution d'une requête

Comme déjà présenté dans la section 5.2.1, l'exécution de la requête commence sur le serveur central. L'interface graphique permet la spécification de la requête à travers la sélection des paramètres désirés. La Figure 39 présente l'interface développée dans le cadre du projet, où nous pouvons voir les différents éléments de la requête.

Chaque élément de la requête est traité de manière différente dans le cadre du processus d'analyse et exécution de la requête. La localisation n'est utilisée que dans l'identification des serveurs distants qui contiennent des contenus multimédias acquis dans un lieu spécifié. Les dates et les heures sont utilisées dans la sélection des contenus multimédias qui doivent être indexés pendant l'indexation explicite, mais ils sont aussi inclus dans la requête exécutée sur les métadonnées.

Pour optimiser le processus de création de la requête XQuery, nous avons choisi de traiter le texte libre de la requête sur le serveur central afin d'obtenir les mots clés, en utilisant Wordnet. Ces mots clés sont utilisés sur le serveur central pour affiner la liste des serveurs, en les exécutant sur la collection de métadonnées gérée par le MDEcs (elle contient les résumés des métadonnées des serveurs distants, les descriptions des serveurs, les descriptions des algorithmes d'indexation et d'autres connaissances supplémentaires (e.g., les horaires des bus). Une fois les serveurs sélectionnés, les mots clés sont envoyés aux serveurs à travers un Service Web. C'est le MDers qui est en charge de la création de la requête XQuery à partir de ces mots clés.

Parce que sur chaque serveur nous avons des algorithmes d'indexation différents, les métadonnées générées sont aussi différentes, même si elles respectent le modèle proposé. Pour limiter le temps d'exécution de la requête sur la collection des métadonnées, la requête ne doit rechercher l'information que dans les éléments du modèle qui ont été renseignés par les algorithmes d'indexation. Par exemple, sur le serveur de broadcast, sur lequel seulement l'algorithme speech-to-text est installé, il ne sert à rien de rechercher de l'information dans l'élément objet. Pour être le plus générique possible, nous avons développé un module de génération de requêtes XQuery qui prend en compte les sorties des algorithmes d'indexation installés sur le serveur.

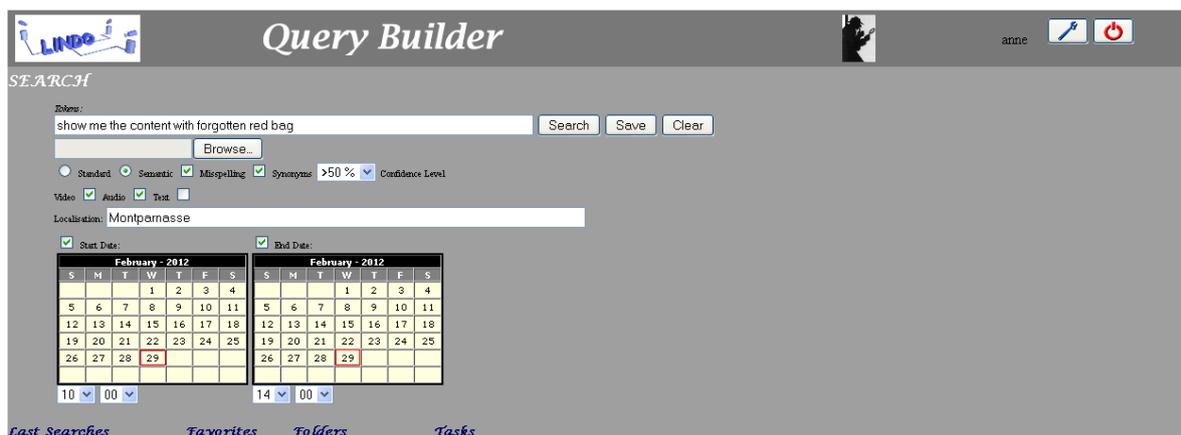


Figure 39 – La spécification d'une requête par l'utilisateur

Les éléments de métadonnées étant typés, ou pouvant n'avoir que certaines valeurs, des contraintes sont ainsi imposées pour la génération de la requête. C'est, par exemple, le cas des dates ou genre. Nous sommes allés encore plus loin dans l'analyse des valeurs que les éléments peuvent avoir, en regardant le type de mot que l'on peut retrouver dans un élément donné.

Le Tableau 44 présente un extrait de cette étude. Nous avons utilisé des règles de correspondance, à travers un fichier de configuration, pour générer la requête XQuery. De cette façon, nous utilisons le même code, en ne changeant seulement que les éléments de métadonnées extraites par les algorithmes d'indexation installés sur le serveur, qu'ils soient implicites ou explicites.

Ce module contient deux bases de données : une pour les métadonnées et une autre pour les descriptions des algorithmes d'indexation. La gestion de ces descriptions est assurée également par le module (en termes d'ajout, d'effacement, de mise-à-jour). En nous basant sur ces descriptions, nous avons développé l'algorithme de sélection des algorithmes d'indexation pour une requête.

Dans le cadre du projet, le Metadata Engine a été utilisé pour la gestion des métadonnées sur les serveurs distants mais aussi sur le serveur central. La seule différence entre les instances utilisées est le fait que sur les serveurs distants, la sélection des algorithmes d'indexation n'est pas active.

Tableau 44 – Extrait des correspondances entre les éléments du modèle et le type de mot contenu

L'expression Xpath de l'élément de métadonnées considéré	Le type de mots contenus
//general/filename	Substantif
//general/description	N'importe quel type de mot (e.g., substantif, adjectif)
//general/language	Seulement les mots qui contenus dans le fichier languages.txt
//genera/author	Substantif
//general/creationDate	Date
//object/localisation/period/@start_time	Date
//object/@type	Seulement les mots contenus dans le fichier concepts.txt. Nous considérons surtout les éléments définis par LSCOM
/audio/Episode/Section/Turn	N'importe quel type de mot (e.g., substantif, adjectif)
/audio/Episode/Section/Turn/@startTime	Nombre réel
/audio/Episode/Section/Turn/Sync/@time	Nombre réel

6.5. Bilan

Dans ce chapitre nous avons présenté les implémentations de nos contributions dans le cadre du projet LINDO : l'utilisation du format de métadonnées et la réalisation de l'indexation en deux temps (implicite et explicite).

Ce projet a abouti à la réalisation d'un système qui englobe des serveurs de domaines différents (vidéosurveillance et broadcast) et dans lequel certains serveurs indexent en temps réel.

Nous avons d'abord présenté l'architecture proposée dans le cadre du projet. Nous nous sommes intéressés ensuite à la réalisation physique et logicielle du système final. Dans cette partie du chapitre, nous avons présenté les différents algorithmes d'indexation qui ont été fournis par les partenaires du projet, en insistant sur leur format de sortie et sur leurs caractéristiques.

Une partie importante de ce chapitre a été dédiée aux implémentations que nous avons réalisées dans le cadre du projet pour intégrer notre format de métadonnées et gérer de manière optimale les métadonnées générées en temps réel.

Chapitre 7. L'enrichissement des métadonnées par l'usage

7.1.	<i>Introduction</i>	159
7.2.	<i>L'architecture du prototype proposé</i>	160
7.3.	<i>L'analyse du site web et de la base de données d'interactions</i>	161
7.3.1.	Le choix des mots clés	161
7.4.	<i>La production des métadonnées</i>	167
7.5.	<i>La production des métadonnées d'usage et leur mise à jour continue</i>	169
7.6.	<i>L'exploitation des métadonnées</i>	171
7.7.	<i>Les expérimentations</i>	172
7.8.	<i>L'évaluation des résultats</i>	172
7.9.	<i>Bilan</i>	178

7.1. Introduction

Notre travail de thèse a été réalisé principalement dans le contexte du projet industriel ITEA2 LINDO. Ce projet nous a permis de valider le format de métadonnées proposé dans nos travaux. Le projet n'ayant pas comme objectif d'étudier l'utilisateur et de prendre en compte son comportement dans la gestion de documents multimédias, nous avons dû rechercher un autre cas d'étude pour valider la prise en compte du comportement de l'utilisateur dans la gestion des documents multimédias. Nous nous sommes intéressés au domaine du commerce électronique pour pouvoir exploiter les données d'usage déjà collectées dans le cadre d'un autre projet.

Dans ce chapitre nous présentons les implémentations et expérimentations que nous avons menées pour valider l'extension du modèle de métadonnées par l'ajout du concept de température et de son évolution dans le temps.

Dans le cas d'étude choisi, nous avons étudié l'évolution des interactions de l'utilisateur que nous avons utilisée pour calculer la température de différentes pages de présentation de produits disponibles sur le site considéré. Une fois les températures calculées, nous les avons utilisées pour réaliser le reclassement des résultats obtenus après l'exécution d'une requête de l'utilisateur. Pour valider notre approche, nous avons comparé la performance du moteur de recherche avant et après l'application du reclassement.

Lors nos expérimentations, nous avons travaillé sur le site de vente en ligne de compléments alimentaires et produits naturels NaturaMundi.com⁵⁷. Ce site est développé en utilisant la plateforme Magento⁵⁸, une des plus utilisées dans le monde⁵⁹. Nous avons choisi ce site, car c'est un client de la

⁵⁷ <http://www.naturamundi.com/>

⁵⁸ <http://www.magentocommerce.com/fr/>

⁵⁹ <http://www.creation-site-lyon.com/2012/06/12/magento-vs-prestashop-vs-oscommerce/>

société Devatics⁶⁰. Les technologies développées par cette entreprise toulousaine permettent d'enregistrer toutes les interactions d'un utilisateur réel avec un site web.

Dans un premier temps, nous présentons l'architecture du prototype proposé en insistant sur les technologies que nous avons utilisées et sur l'emplacement des données et des métadonnées.

Dans un deuxième temps, nous présentons les étapes de notre expérimentation en détaillant chaque étape. Enfin, nous analysons les résultats que nous avons obtenus suite à des tests.

7.2. L'architecture du prototype proposé

L'architecture de l'application implémentée pour réaliser ces expérimentations s'appuie sur Java (Figure 40).

L'élément clés de notre expérimentation se situe au niveau des interactions des utilisateurs avec les contenus du site web. Ces interactions sont enregistrées à travers un proxy sur un serveur différent de celui où se trouve le contenu du site.

Toutes les interactions de l'utilisateur avec le site web (clics, déplacements de souris, défilements, appuis sur les touches du clavier) sont enregistrées par le biais d'un module Javascript générique, inséré sur toutes les pages du site. Lorsqu'un événement est détecté par les écouteurs qui se greffent dans l'arbre DOM de chaque page, des informations sont récoltées en utilisant les propriétés de l'arbre DOM (e.g. l'élément HTML sur lequel l'interaction a été effectuée ou l'URL de la page) et transférées vers le serveur en utilisant un appel asynchrone (Baccot et al., 2011). Ces interactions sont enregistrées pour une page du site, mais aussi pour tous les éléments de la page.

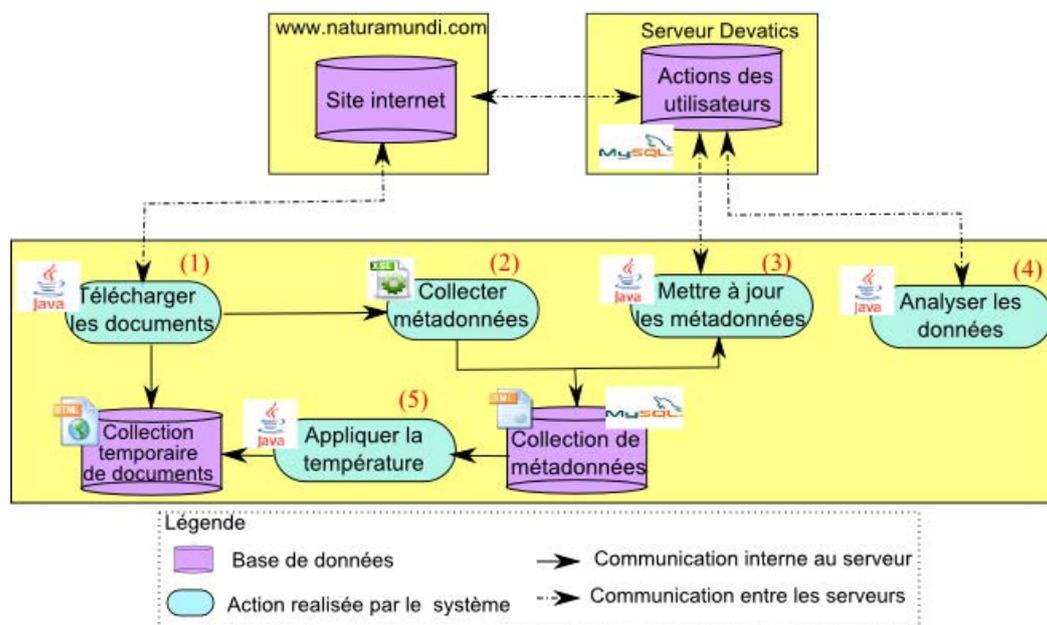


Figure 40 – Architecture de l'application

Pour pouvoir exploiter ces interactions, nous devons tout d'abord extraire les métadonnées associées à chaque page. Pour réaliser ceci, nous avons téléchargé les pages du site pour un traitement

⁶⁰ <http://www.devatics.com/fr>

plus rapide (module (1) de la Figure 40). Sur chaque page, nous avons appliqué l'extraction des métadonnées en utilisant des transformations XSLT (module (2) de la Figure 40). Les métadonnées obtenues après cette étape d'extraction sont stockées dans une base de données. Ces métadonnées ne contiennent pas d'informations liées à l'usage. La fonction de mise-à-jour de métadonnées (module (3) de la Figure 40) s'occupe de l'enrichissement des métadonnées en créant les liens indirects entre les différentes pages et en calculant la température (cf. section 5.3). Une fois la température calculée, nous avons mis en place un module (module (5) de la Figure 40) qui la prend en compte pour réaliser le reclassement des résultats d'une requête de l'utilisateur.

Dans cette architecture, nous n'avons pas la possibilité de filtrer les interactions qui sont enregistrées dans la base de données. Comme elle ne contenait pas assez d'informations sur les utilisateurs pour pouvoir établir des groupes, nous avons donc considéré que les utilisateurs forment un seul groupe. Par conséquent, une seule valeur de la température est calculée pour chaque élément de métadonnées.

Avant de pouvoir exploiter cette base d'interactions, nous devons analyser son contenu pour identifier les éléments qui nous seront utiles (module (4) de la Figure 40).

7.3. L'analyse du site web et de la base de données d'interactions

Une analyse macroscopique de la base de données des interactions nous permet de constater qu'elle contient toutes les interactions réalisées par un utilisateur avec le site et qui sont capturables par le navigateur pendant une session :

- les déplacements de souris sur chaque élément d'une page ;
- les clics sur des éléments de la page, qu'ils soient lien ou pas ;
- la séquence des pages visitées ;
- les défilements qu'il réalise pour consulter le contenu de la page qui est affiché en dessous la ligne de flottaison ;
- les requêtes, à base de mots clés, que l'utilisateur a réalisées sur le site.

La base de données contient les requêtes des utilisateurs et les clics qu'ils ont réalisés dans les listes des résultats obtenues pour ces requêtes. Nous pouvons donc utiliser ces informations comme référentiel dans nos expérimentations. Plus précisément, nous :

1. déterminons les mots les plus recherchés par les utilisateurs réels du site,
2. utilisons ces mots pour les tests utilisateur,
3. comparons les résultats des tests avec les informations obtenues directement des utilisateurs du site à travers leurs interactions.

Pour pouvoir avoir une comparaison fiable entre les résultats des utilisateurs réels du site et nos utilisateurs test, nous devons avoir le plus d'informations possibles pour les utilisateurs réels du site, c'est pour cela que nous choisissons les requêtes les plus utilisées.

7.3.1. Le choix des mots clés

Pour le choix du référentiel, nous prenons en compte une période de 3 mois, entre le 1^{er} février et le 30 avril.

Pour choisir les mots clés les plus utilisés, nous n'utilisons que les informations qui sont stockées sur le serveur Devatics. Les requêtes des utilisateurs sont incluses dans les liens vers les

pages de résultats que l'utilisateur consulte. Suite à la façon dont le site génère les pages de résultats pour une requête de l'utilisateur, dans la base de données il y a deux types de liens. Par exemple, pour la recherche du mot « rhodiola » on obtient les liens présentés dans Tableau 45.

Tableau 45 – Exemples de liens obtenus pour la recherche du mot « rhodiola »

http://www.naturamundi.com/catalogsearch/result/?q=rhodiola
http://www.naturamundi.com/catalogsearch/result/index/?q=rhodiola

Nous avons réalisé un classement des requêtes en fonction du nombre de fois qu'elles ont été exécutées pendant la période de temps considérée. Pour ce faire, nous avons éliminé : les accents, les espaces présents au début et à la fin de chaque mot, et nous avons transformé en minuscules tous les mots.

Pour la période considérée, nous avons obtenu 2332 mots, avec leurs nombre d'utilisations. Le Tableau 46 présente les 100 premiers mots les plus utilisés.

Tableau 46 – Les 100 mots les plus recherchés sur le site

Nb.	Le mot	Nb Utilisations
1	<i>Rhodiola</i>	499
2	Kudzu	249
3	Griffonia	198
4	Spiruline	156
5	tisane acidose	126
6	Sommeil	93
7	Eleutherocoque	90
8	tisane hépatique	81
9	Baccharis	78
10	Acidose	77
11	Maca	70
12	Chrysanthellum	68
13	griffe du chat	67
14	<i>Rodiola</i>	66
15	6134	61
16	Desmodium	60
17	Harpagophytum	59
18	huile visage	59
19	kudzu tabac	59
20	chrysantellum americanum	58
21	rhodiola rosea	56
22	rhodiola 120 gelules	55
23	acerola	52
24	candida	50
25	huiles essentielles	46
26	tisane	46
27	eau de rose	45
28	7128b	42
29	astragale	42
30	probiotiques	41
31	bourrache	37
32	<i>rodhiola</i>	36
33	millepertuis	33
34	olivier	33
35	baccharis 120	32
36	baccharis bio	32
37	huile essentielle lavande	32
38	lapacho	32
39	cordyceps	31
40	ginseng	31
41	pieere alun	30
42	Probiotique	30
43	rhodiola 200 gelules	30
44	6114	29
45	Magnesium	29
46	griffonia simplicifolia	28
47	Hépatique	28
48	Echinacea	27
49	<i>Rhodolia</i>	27
50	tisane digestive	27
51	Bacopa	26
52	5105b	25
53	huile essentielle	25
54	Curcumine	24
55	eau de bleuet	24
56	Guimauve	24
57	aubier de tilleul	23
58	collagene marin	23
59	tisane dépurative	23
60	Cassis	22

61	Curcuma	22
62	Foie	22
63	Huile	22
64	Noni	22
65	Bardane	21
66	tisane cellulite	21
67	algue marine	20
68	gelee royale	20
69	Cholesterol	19
70	Focus	19
71	omega 3	19
72	tisane equilibrante renale	19
73	desmodium ascendens	18
74	huile cellulite	18
75	tisane cholesterol	18
76	4186	17
77	chrysantellum	17
78	dormez	17
79	ginseng rouge	17
80	glucosamine	17

81	jeune	17
82	lithothamne	17
83	muira puama	17
84	the vert	17
85	vigne rouge	17
86	baccharis 120	16
87	chardon marie	16
88	glucosamine-chondroïtine	16
89	ortie	16
90	savon	16
91	6106b	15
92	arpagophytum	15
93	aubepine	15
94	bois bande	15
95	collagene marin enrichi	15
96	prostate	15
97	sabal	15
98	visage	15
99	6134n	14
100	arthrose	14

En analysant les résultats obtenus après cette étape, on remarque qu'il y a plusieurs variantes d'écriture pour un mot. Par exemple, le mot *rhodiola* a été recherché en plusieurs orthographes (i.e., les lignes en italique dans le Tableau 46: 1, 14, 32, 49).

En effectuant les recherches sur le site pour certains des mots les plus utilisés, on a constaté l'existence de certaines des requêtes ayant très peu de résultats. Par exemple, « *rhodiola* » ne donne que 18 résultats, tandis que « griffe de chat » donne 151 résultats. Pour pouvoir choisir les mots à utiliser dans la suite des expérimentations, nous avons créé une application qui exécute sur le site toutes les requêtes obtenues à l'étape précédente et enregistre la liste des résultats obtenus pour chaque requête dans des fichiers différents. A partir des fichiers obtenus, les différentes requêtes sont ordonnées par rapport au nombre de résultats affichés à l'utilisateur. Le Tableau 47 présente un aperçu des 100 mots qui ont le plus de résultats. La première colonne de ce tableau représente le rang que le mot de la deuxième colonne a dans le Tableau 46. La troisième colonne indique le nombre de fois que le mot a été utilisé en tant que recherche. La dernière colonne constitue le nombre de résultats retournés par le moteur de recherche du site pour un mot.

On observe que les 10 mots les plus recherchés n'ont pas un nombre élevé de résultats (e.g., 18 résultats pour *rhodiola*, 3 résultats pour *kudzu*, 2 résultats pour *griffonia*). Afin de pouvoir valider l'utilité du reclassement dans ces conditions, nous allons choisir 10 mots qui ont le plus de résultats et qui sont parmi les plus utilisés 30 mots. Les mots sélectionnés à cette étape sont mis en bleu dans le Tableau 47: *eau de rose*, *griffe de chat*, *baccharis bio*, *tisane hépatique*, *tisane acidose*, *sommeil*, *huile visage* et *rhodiola*. Nous avons privilégié les mots clés qui n'ont pas une signification trop générale, pour lesquels nous pouvons trouver facilement les différents produits correspondants. Dans le cas du mot « tisane », il peut être utilisé comme requête pour n'importe quel type de tisane, et il est impossible d'exploiter les informations de la base de données comme base de comparaison pour nos tests utilisateurs.

Tableau 47 – Le nombre de résultats par mot recherché

Rang	Mot clé	Utilisations	Résultats
57	aubier de tilleul	23	214
55	eau de bleuet	24	214
27	eau de rose	45	214
13	griffe du chat	67	177
36	baccharis bio	32	120
71	omega 3	19	110
84	the vert	17	68
66	tisane cellulite	21	42
59	tisane depurative	23	38
75	tisane cholesterol	18	37
50	tisane digestive	27	37
8	tisane hepatique	81	36
26	tisane	46	35
5	tisane acidose	126	35
72	tisane equilibrante renale	19	35
74	huile cellulite	18	33
37	huile essentielle lavande	32	28
22	rhodiola 120 gélules	55	23
43	rhodiola 200 gélules	30	23
58	collagene marin	23	21
95	collagene marin enrichi	15	21
6	sommeil	93	21
18	huile visage	59	20
63	huile	22	19
53	huile essentielle	25	19
1	rhodiola	499	18
21	rhodiola rosea	56	18
94	bois bande	15	13
87	chardon marie	16	10
35	baccharis 120	32	8
86	baccharis 120	16	8
79	ginseng rouge	17	8
67	algue marine	20	7
60	cassis	22	7
69	cholesterol	19	7
7	eleutherocoque	90	7
85	vigne rouge	17	7
29	astragale	42	5
19	kudzu tabac	59	5
12	chrysanthellum	68	4
70	fucus	19	4
25	huiles essentielles	46	4
81	jeune	17	4
89	ortie	16	4
96	prostate	15	4
93	aubepine	15	3
9	baccharis	78	3
61	curcuma	22	3
40	ginseng	31	3
47	hepatique	28	3
2	kudzu	249	3
42	probiotique	30	3
30	probiotiques	41	3
97	sabal	15	3
90	savon	16	3
4	spiruline	156	3
15	6134	61	2
28	7128b	42	2
23	acerola	52	2
10	acidose	77	2
51	bacopa	26	2
31	bourrache	37	2
62	foie	22	2
68	gelee royale	20	2
3	griffonia	198	2
46	griffonia simplicifolia	28	2
82	lithothamne	17	2
34	olivier	33	2
98	visage	15	2
76	4186	17	1
44	6114	29	1
52	5105b	25	1
91	6106b	15	1
92	arpagophytum	15	1
65	bardane	21	1
20	chrysantellum americanum	58	1
54	curcumine	24	1
16	desmodium	60	1
73	desmodium ascendens	18	1

78	dormez	17	1
48	echinacea	27	1
80	glucosamine	17	1
88	glucosamine-chondroïtine	16	1
17	harpagophytum	59	1
38	lapacho	32	1
11	maca	70	1
45	magnesium	29	1
33	millepertuis	33	1
83	muira puama	17	1

41	pieere alun	30	1
99	6134n	14	0
100	arthrose	14	0
24	candida	50	0
77	chrysantellum	17	0
39	cordyceps	31	0
56	guimauve	24	0
64	noni	22	0
49	rhodolia	27	0
32	rodhiola	36	0
14	rodiola	66	0

Ce tableau nous permet d'analyser aussi la qualité du moteur de recherche, interne à Magento, utilisé par le site. Si on ne regarde que les lignes qui contiennent le mot rhodiola, on peut faire les remarques suivantes :

- Il n'y a pas de correction orthographique la requête : pour le mot rhodiola le site affiche 18 résultats, mais pour rhodolia, rodhiola ou encore rodiola il n'y a aucun résultat.
- Le moteur de recherche utilise une recherche par mot, en utilisant l'opérateur 'ou' entre les mots, et il ignore les nombres : pour rhodiola il trouve 18 résultats, mais pour rhodiola 120 gélules et rhodiola 200 gélules, il y a 23 résultats.

Cette analyse ne fait que confirmer le fait que le moteur de recherche proposé par la plateforme Magento ne possède que les fonctionnalités basiques, et donc l'utilité d'une fonction de reclassement des résultats apparaît.

Avant de finaliser la liste des mots utilisés pour la suite des expérimentations, nous avons essayé de voir comment les requêtes sont utilisées pendant les deux mois choisis pour nos expérimentations. Le résultat de cette étude est présenté dans la Figure 41. Nous observons une utilisation constante de certaines de ces mots clés en tant que requêtes, surtout le mot « rhodiola », qui est le plus recherché pendant la période considérée. Donc l'utilisation de ces mots pour la suite semble prometteuse pour pouvoir analyser l'évolution de la température.

Un autre élément important à considérer est la distribution des accès des utilisateurs aux différentes pages du site. En analysant les actions des utilisateurs fournies par la base de données, nous obtenons la distribution présentée dans la Figure 42. Cette distribution respecte bien la forme de « longue traîne » définie par (Anderson, 2009). Cette distribution montre le fait que la somme des accès aux pages qui se trouvent vers la fin cette longue traîne est aussi important que la somme des accès aux premières pages – les plus populaires. Selon Anderson, l'exploitation des accès à ces pages est une bonne stratégie pour obtenir un meilleur chiffre d'affaire. En outre, les moteurs de recherche commencent à regarder plus en détails, ce problème de longue traîne dans la représentation des requêtes. Dans cette même optique, nous cherchons à faire remonter certaines pages de la longue traîne, en tête des résultats de recherche, si ces pages sont pertinentes. A travers notre proposition nous faisons remonter certaines des pages qui se trouvent vers la fin de cette longue traîne.

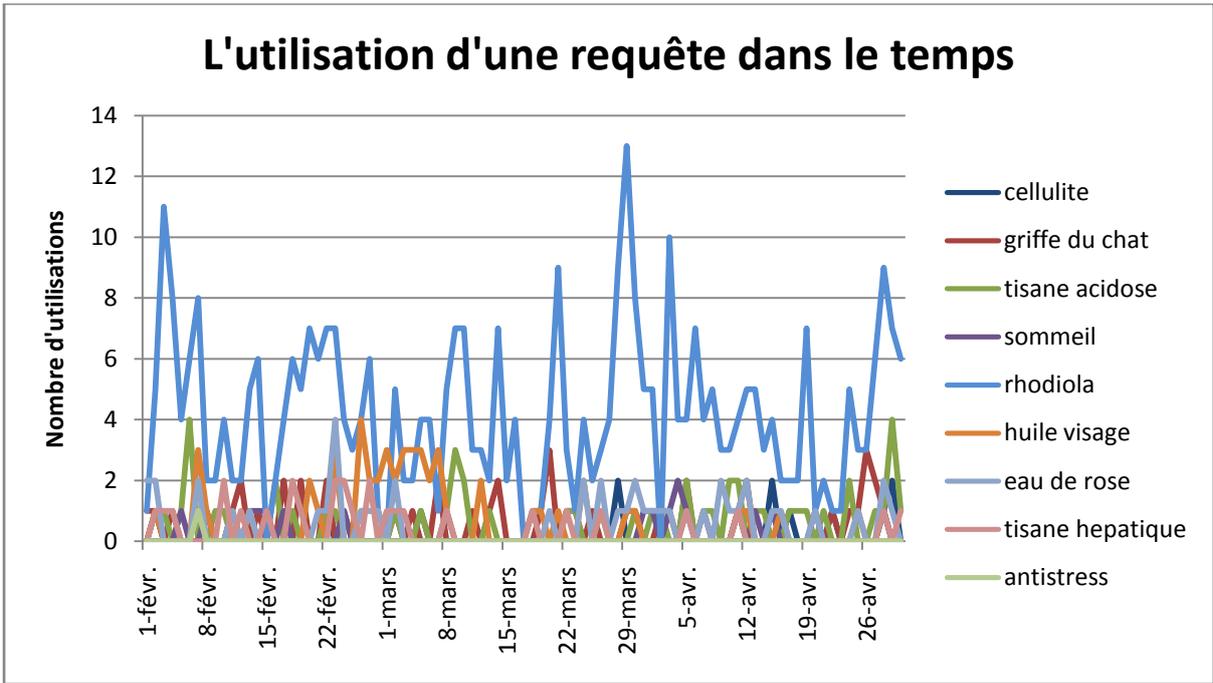


Figure 41 – Le nombre d'utilisations de chaque requête

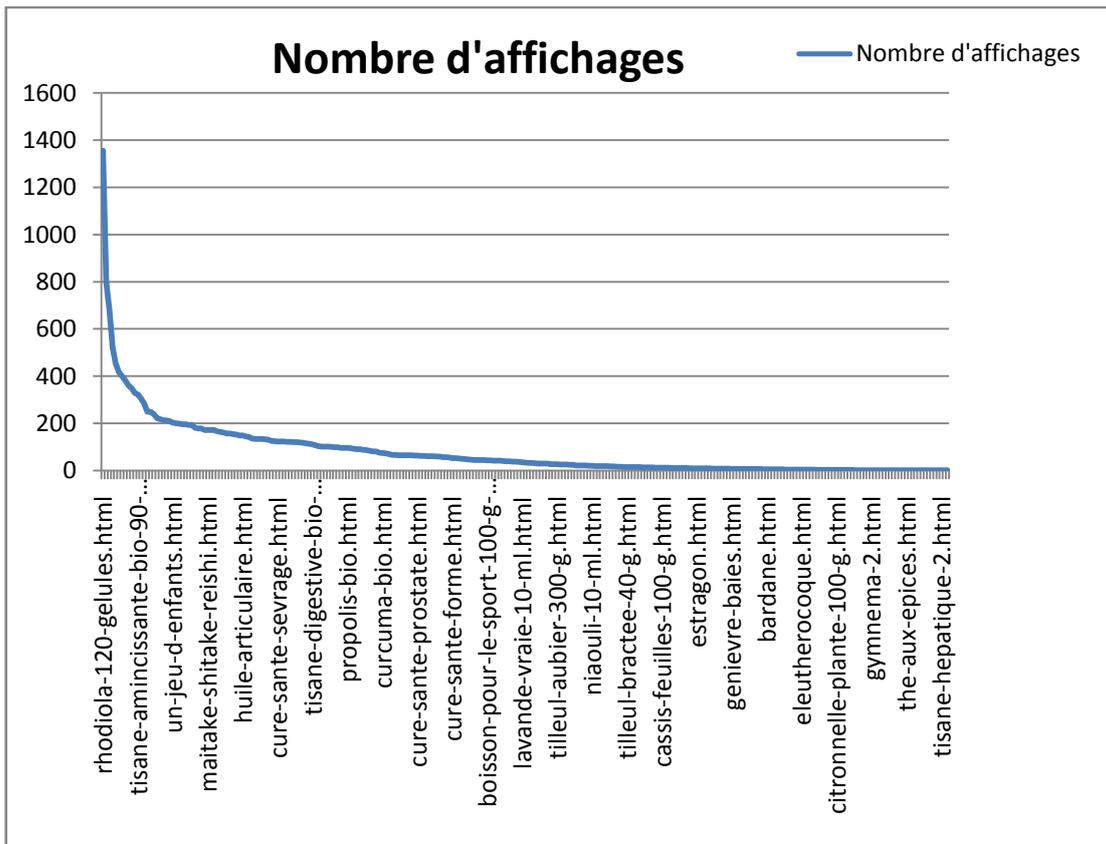


Figure 42 – La distribution du nombre d'accès aux pages du site

7.4. La production des métadonnées

Une fois les mots choisis, nous pouvons passer à l'étape suivante, la génération des métadonnées afférentes à chaque page html de présentation d'un produit disponible sur le site. Ces métadonnées sont générées en plusieurs étapes présentées dans la Figure 43.

Pour pouvoir exploiter la richesse des interactions sauvegardées dans la base de données, nous devons associer des métadonnées à chaque élément d'une page avec lequel l'utilisateur est susceptible d'interagir. C'est notamment le cas des images, des liens et des paragraphes de texte.

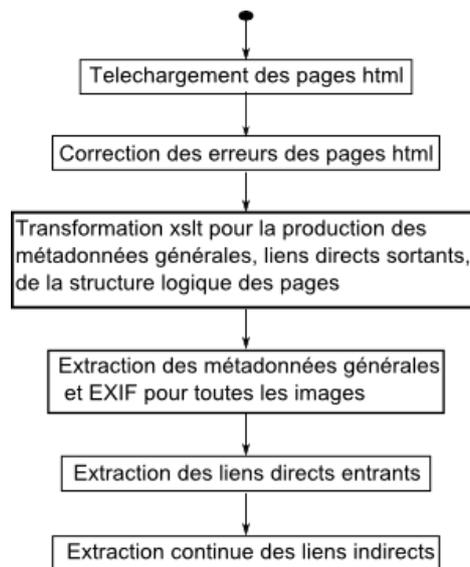


Figure 43 – Les étapes de l'extraction des métadonnées

Premièrement, chaque page html est téléchargée automatiquement. Certaines de ces pages ne sont pas bien formées du point de vue XML. L'erreur la plus rencontrée est le fait que les balises ne sont pas fermées dans le bon ordre. Par exemple, on peut avoir «
 » à la place de « </br> », ou des balises qui ne sont jamais fermées. Ces erreurs sont corrigées pour avoir une page valide. Une fois les pages corrigées, l'extraction des métadonnées peut commencer. Cette extraction est réalisée automatiquement, de la manière suivante :

- Une transformation XSLT est produite à travers une feuille de style écrite préalablement. Après cette étape, les métadonnées générales sont extraites en exploitant les balises meta présentes dans l'entête de la page. La structure logique de la page est extraite aussi, et les liens sortants de la structure hypermédia sont générés.
- Les métadonnées obtenues sont enrichies avec les métadonnées extraites pour chaque image présente dans le fichier html. Chaque image est téléchargée et ses métadonnées EXIF sont extraites en utilisant une application Java qui utilise la librairie metadata-extractor.jar. Les liens entrants pour chaque page html sont créés également.
- La structure dynamique est créée en ajoutant la température à chaque élément de métadonnées.
- La structure d'usage est, quant à elle, créée à fur et à mesure du temps et de l'utilisation de la page.

Pour illustrer les métadonnées obtenues suite à ce processus, nous prenons la page de présentation du produit Rhodiola 120 gélules (Figure 44). Par soucis de lisibilité et d'espace, nous ne présentons que des parties de ces métadonnées, un exemple plus complet est présenté dans l'Annexe Q.

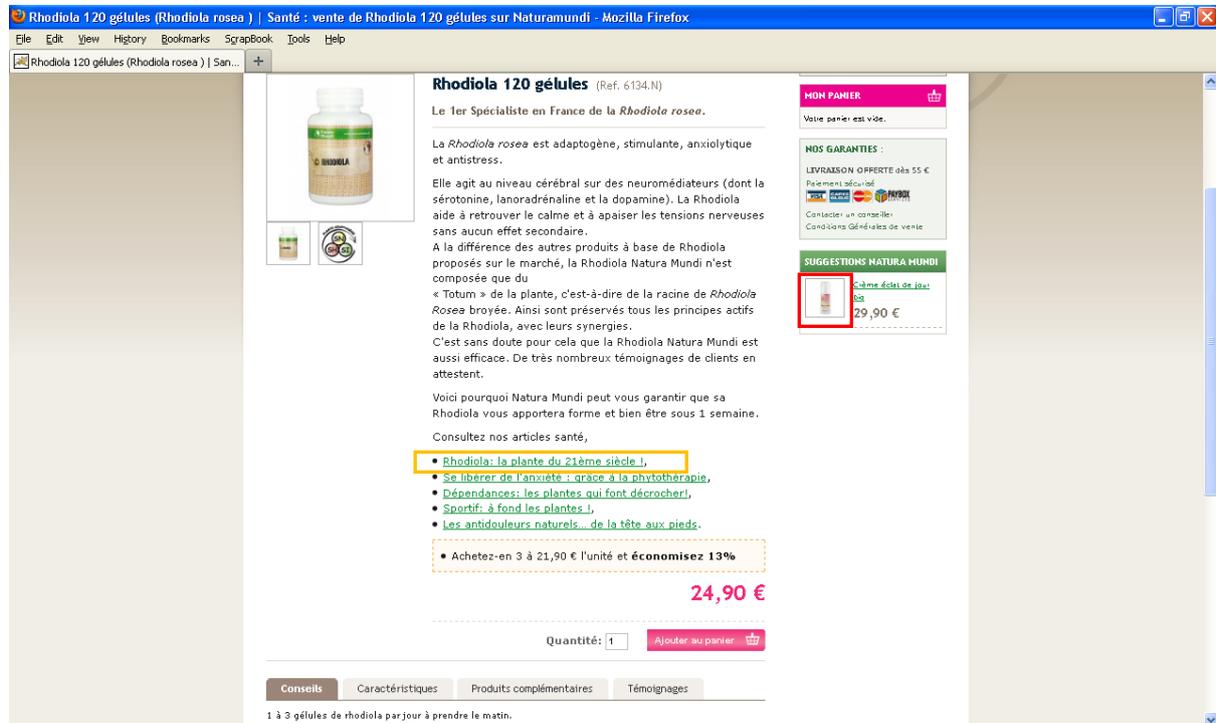


Figure 44 – La page de présentation du produit Rhodiola 120 gélules

Les métadonnées extraites pour l'image entourée en rouge dans la Figure 44 se trouvent dans le Tableau 48. Comme nous pouvons le constater les métadonnées qui sont extraites ne sont pas trop riches. Dans le cas de cette image, cela est dû au fait que c'est une vignette. En regardant les métadonnées des autres images utilisées par le site, nous constatons la même chose pour toutes, sans exception, parce qu'elles ne contiennent pas de métadonnées EXIF pour avoir une dimension plus petite pour se télécharger plus vite.

En ce qui concerne les métadonnées associées à l'hyperlien, entouré par le rectangle jaune, elles sont présentées dans le Tableau 49. Pour cet élément de la page, sont créés un premier élément TextUnit contenant une référence vers un lien direct sortant (d'identifiant U54) et un second élément TextUnit pour le texte affiché.

Tableau 48 – Métadonnées extraites pour une image

```
<image ID="U31" avgTemperature="0.0">
  <generalInformation ID="U32" avgTemperature="0.0">
    <filename avgTemperature="0.0" ID="U33">
      <Temperature userGroupID="GR1">0.0</Temperature>
      <value>http://www.naturamundi.com/media/catalog/product/cache/1/thumbnail/
50x/9df78eab33525d08d6e5fb8d27136e95/5/1/5124B_2.jpg</value>
```

```

</filename>
<description avgTemperature="0.0" ID="U34">
  <Temperature userGroupID="GR1">0.0</Temperature>
  <value>Crème de jour bio 50 ml</value>
</description>
<Temperature userGroupID="GR1">0.0</Temperature>
</generalInformation>
<EXIF ID="U530" avgTemperature="0.0">
  <TiffIfd ID="U531" avgTemperature="0.0">
    <Temperature userGroupID="GR1">0.0</Temperature>
  </TiffIfd> <Temperature userGroupID="GR1">0.0</Temperature>
</EXIF> <Temperature userGroupID="GR1">0.0</Temperature>
</image>

```

Tableau 49 – Métadonnées associées à un hyperlien

```

<TextUnit ID="U202" definitionRef="&lt;li&gt;&lt;a " avgTemperature="0.0">
  <TextUnit ID="U203" definitionRef="&lt;a" avgTemperature="0.0">
    <refs>U54</refs>
    <TextUnit ID="U204" definitionRef="Rhodiola: la plante du 21ème siècle !" avgTemperature="0.0">
      <Temperature userGroupID="GR1">0.0</Temperature>
    </TextUnit> <Temperature userGroupID="GR1">0.0</Temperature>
  </TextUnit> <Temperature userGroupID="GR1">0.0</Temperature>
</TextUnit>

```

7.5. La production des métadonnées d'usage et leur mise à jour continue

Une fois les métadonnées extraites, la température est calculée pour chaque page en utilisant l'algorithme présenté dans la Section 5.3. Les résultats sont stockés dans une base de données locale, pour pouvoir être utilisés pour le reclassement des résultats.

Pour ces expérimentations, nous avons considéré les valeurs suivantes pour les paramètres :

- $\alpha_{\text{clics}} = 0.5$
- $\alpha_{\text{mouseOver}} = 0.25$
- $\beta = 0.5$
- $\delta = 0.5$
- NBPERIODES = 2 ;
- $\gamma = 0.5$

Les températures des métadonnées sont augmentées en plusieurs étapes, jour après jour ($\Delta t=1$).

Pour illustrer l'évolution de la température, nous présentons, dans les figures suivantes, les interactions des utilisateurs qui ont induit cette évolution. Pour des soucis de lisibilité, nous ne considérons dans ces exemples que les interactions des utilisateurs pendant le premier mois de notre expérimentation (e.g., février) avec trois éléments de la page de présentation du produit Rhodiola 120 gé-lules (Figure 44) :

- « Les compléments alimentaires ne se substituent en aucun cas à des » - situé en bas de la page ;

- « Forme galénique et conditionnement » - situé dans l'onglet « caractéristiques »
- « Mon compte » - situé dans la partie supérieure de la page

La Figure 45 présente les clics et les mouse-over réalisés par les utilisateurs pendant le mois de février sur ces éléments. Ces interactions sont en forme de dents de scie, ce qui nous permet de voir aussi le refroidissement de ces éléments.

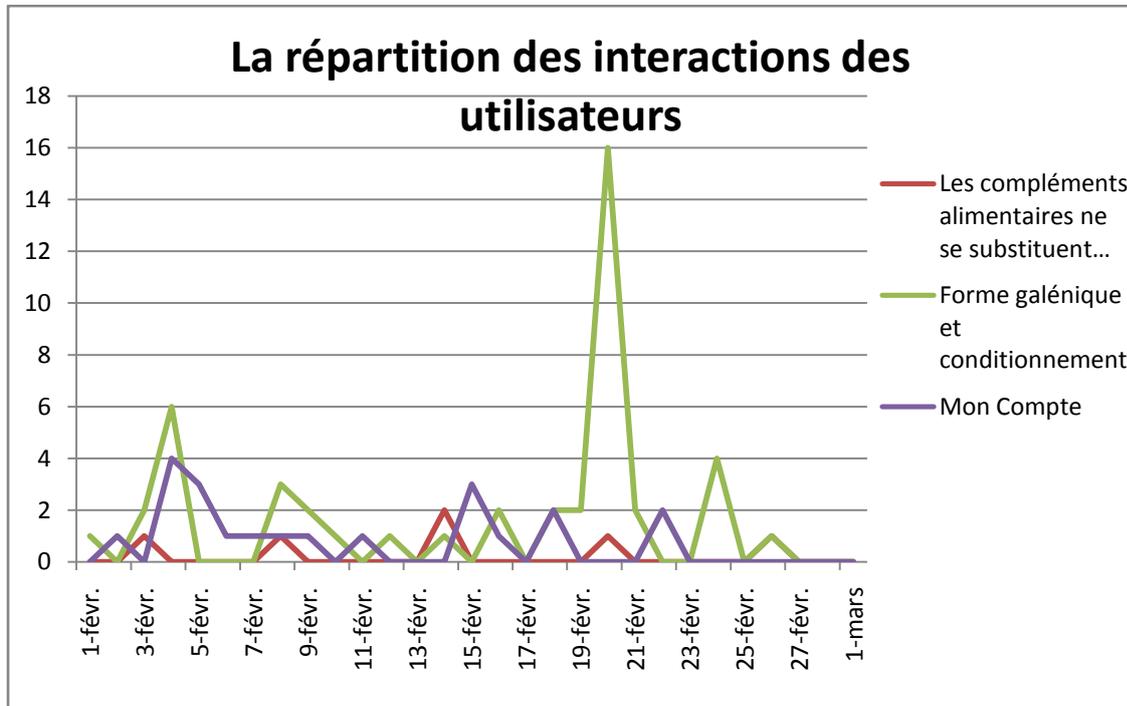


Figure 45 – La répartition des interactions (clics et mouse-over) des utilisateurs dans le temps

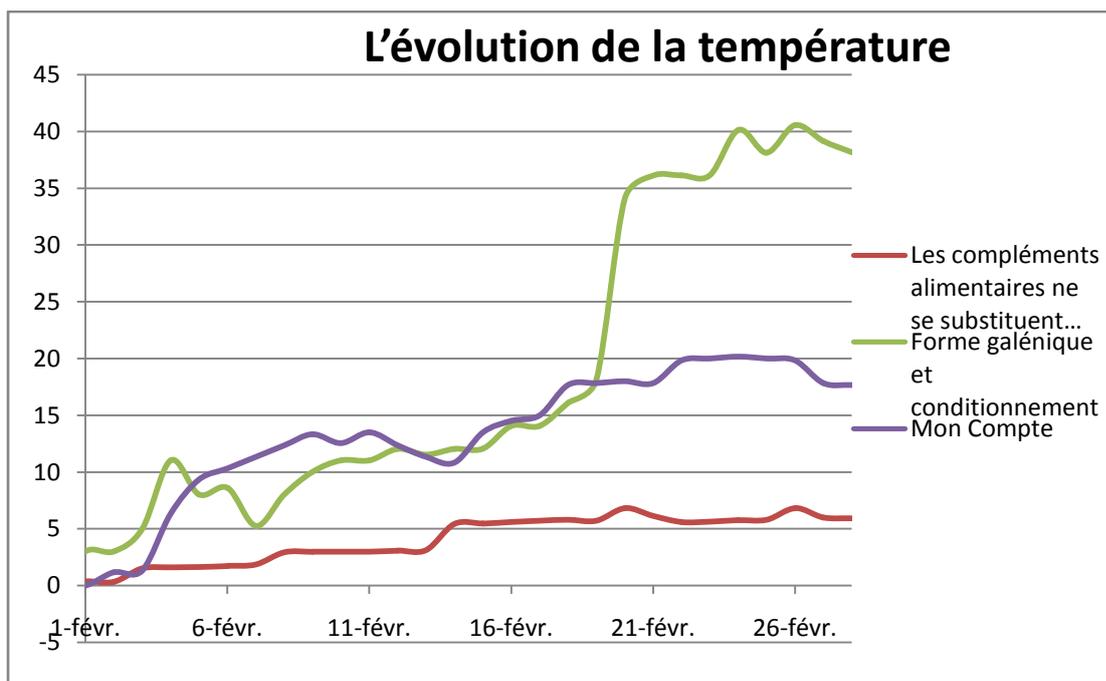


Figure 46 – L'évolution de la température

La Figure 46 présente l'évolution de la température des trois éléments de la page, après un mois d'utilisation. En analysant les courbes vertes, correspondant à l'élément « Forme galénique et conditionnement », on se rend compte qu'au moment où l'élément n'est pas utilisé, sa température diminue. Par exemple entre le 5 et le 7 février, cet élément n'est pas utilisé, donc sa température diminue progressivement. On peut observer que le 6 février, la température de cet élément augmente légèrement, même s'il n'est pas utilisé. Cela est dû au fait que l'onglet « caractéristiques », qui contient cet élément, est utilisé, et le changement de température de cet élément est propagé à tous les éléments contenus. Le pic d'utilisation de cet élément du 20 février se traduit par une augmentation significative de la température.

7.6. L'exploitation des métadonnées

Pour pouvoir évaluer l'ajout de la température, nous l'avons utilisée pour réaliser le reclassement des résultats retournés par le moteur de recherche du site. En effet, pour une requête donnée, le moteur de recherche standard ne retourne que la liste de résultats dans un ordre décroissant de leur pertinence pour la requête (sans le score).

Pour chaque requête de l'utilisateur, nous avons récupéré la liste de résultats retournés par le site et nous avons réalisé un nouveau classement des résultats, par ordre décroissant par rapport à la température de chaque document. Nous avons fusionné ces deux classements (celui du site et celui basé sur la température) en utilisant une formule classique (11).

$$\text{reclassementScore}(doc) = \frac{1}{\frac{1}{\text{rankSite}(doc)} + \frac{1}{\text{rankTemperature}(doc)}} \quad (11)$$

<p>Catégorie Santé (16) Minceur (1) Anti-âge (7) Beauté (1) Bien-être (13) Herboristerie (2) Liste complète (+)</p> <p>Indications Anti-âge et mémoire (9) Antioxydants (6) Antistress (15) Beauté de la peau (1) Cellulite (1) Circulation sanguine (2) Confort articulaire (1) Cosmétiques bio (1) Défenses immunitaires (4) Détente-sommeil (6) Forme et vitalité (10) Ménopause (1) Minceur (4) Prostate (1) Sexualité (5) Tension (1) Liste complète (+)</p> <p>Plantes & principes naturels Aunée (2) Baccharis trimera (1) Cynorrhodon (2) Eleuthérocoque (1) Framboisier (2) Ginseng (1) Griffonia (1) Kudzu (1) Margolaine (2) Menthe douce (2) Myrtille (2) Olivier (1) Probiotiques (1) Rhodiola rosea (16) Sabal serrulata (1)</p>	<p>Résultats de recherche pour 'rhodiola'</p> <p>Voir 30 par page 18 article(s)</p> <p>Trier par Pertinence ↓</p> <p>Duo minceur 74,70 € 69,90 € Associez la rhodiola au baccharis pour la minceur. En savoir plus Voir</p> <p>Duo articulations 94,70 € 89,90 € Associez la rhodiola au collagène marin enrichi pour les articulations. En savoir plus Voir</p> <p>Duo prostate 82,70 € 77,90 € Associez la rhodiola au sabal pour la prostate. En savoir plus Voir</p> <p>Duo pour entretenir sa mémoire 49,90 € 44,90 € Associez la rhodiola au bacopa pour entretenir sa mémoire. En savoir plus Voir</p> <p>Duo face aux addictions 74,70 € 69,90 € Associez la rhodiola au kudzu face aux addictions. En savoir plus Voir</p> <p>Duo vitalité 49,90 € 44,90 €</p>	<p>Résultats de recherche pour 'rhodiola'</p> <p>Voir 30 par page 18 article(s)</p> <p>Trier par Pertinence ↓</p> <p>Rhodiola 120 gélules 24,90 € Le 1er Spécialiste en France de la Rhodiola rosea. En savoir plus A partir de: 21,90 € Voir</p> <p>Duo minceur 74,70 € 69,90 € Associez la rhodiola au baccharis pour la minceur. En savoir plus Voir</p> <p>Duo sommeil serein 49,90 € 44,90 € Associez la rhodiola à la griffonia un sommeil serein. En savoir plus Voir</p> <p>Duo articulations 94,70 € 89,90 € Associez la rhodiola au collagène marin enrichi pour les articulations. En savoir plus Voir</p> <p>Duo ménopause 46,90 € 39,90 € Associez la rhodiola à la</p>
<p>a) Classement standard du site pour la requête rhodiola</p>	<p>b) Reclassement des résultats pour la requête rhodiola ; Entouré un résultat remonté depuis la deuxième page du classement standard</p>	

Figure 47 – Un exemple de liste de résultats reclassée

La Figure 47 présente un exemple de résultats fournis par le moteur de recherche pour la requête rhodiola et la liste reclassée suite à la prise en compte de la température.

7.7. Les expérimentations

Afin de ne pas perturber le site de production et de pouvoir contrôler les conditions d'expérimentation, les expérimentations ne se sont pas déroulées sur le site en production, mais sur une copie de celui-ci, qui a été réalisée en utilisant les pages utilisées pour la création des métadonnées. L'objectif de ces expérimentations est de valider la pertinence du reclassement des résultats.

Pour ces expérimentations, nous avons considéré les requêtes choisies après l'analyse du site et de la base d'interactions que nous avons présentée dans la section 7.3. Le processus du test utilisateur s'est déroulé de la manière suivante :

1. Chaque utilisateur donne son avis pour chacune des requêtes sélectionnées auparavant.
2. Pour chaque requête, nous avons :
 - a. présenté à l'utilisateur la liste des résultats reclassée.
 - b. demandé à l'utilisateur de trouver les résultats (environ 5 produits) qui lui semblent les plus pertinents pour la requête.
 - c. sauvegardé les interactions (i.e., clics) de l'utilisateur sur les liens de la liste des résultats en tant qu'indicateur de leur intérêt envers les documents.
 - d. demandé à l'utilisateur de donner une note de 0 à 10 à chaque document qu'il a sélectionné par clic, en fonction du degré de pertinence de ce document pour la requête

Nous avons comparé les résultats obtenus suite à cette expérimentation avec les résultats que nous avons collectés sur site de production. Pendant la période choisie, nous avons eu un nombre d'environ 15719 utilisateurs différents, parmi lesquels 3373 utilisateurs se sont connectés au moins 2 fois et 2969 utilisateurs ont effectué au moins une requête sur le site.

Nous avons effectué cette expérimentation sur un groupe de 14 utilisateurs, avec un âge compris entre 23 et 45 ans, avec une parité hommes – femmes. Tous les utilisateurs ont l'habitude de faire des achats sur internet.

7.8. L'évaluation des résultats

Avant d'analyser les résultats obtenus suite aux tests utilisateurs, nous avons analysé les clics des utilisateurs réels du site pendant la période considérée. Pour chaque mot-clé choisi auparavant, nous obtenons le nombre de fois que chaque lien de la liste des résultats a été cliqué.

L'objectif a été d'étudier la répartition des clics en fonction de la position des résultats dans la liste des résultats. La Figure 48 montre que les utilisateurs ne regardent que les premiers résultats. Toutefois, certains utilisateurs ont consulté la deuxième page de résultats pour trouver le produit qui les intéressait. C'est le cas surtout pour les mots : rhodiola et sommeil. Par exemple, pour le mot rhodiola, le produit présent à la 12ème position est le plus cliqué parmi tous les produits affichés. Pour cette requête, nous retrouvons aussi des clics sur les produits affichés à la 15ème et même 16ème position. Tous ces produits se trouvent sur la deuxième page des résultats.

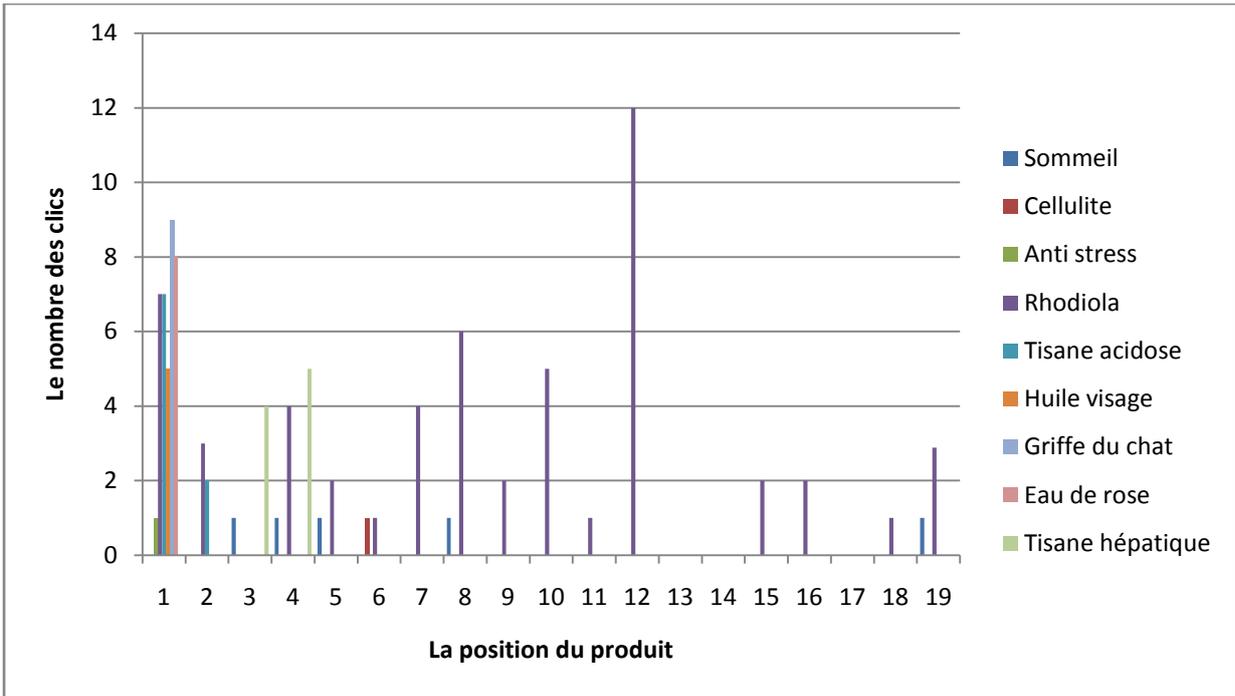


Figure 48 – Le nombre de fois que chaque résultat a été cliqué par les utilisateurs du site de production

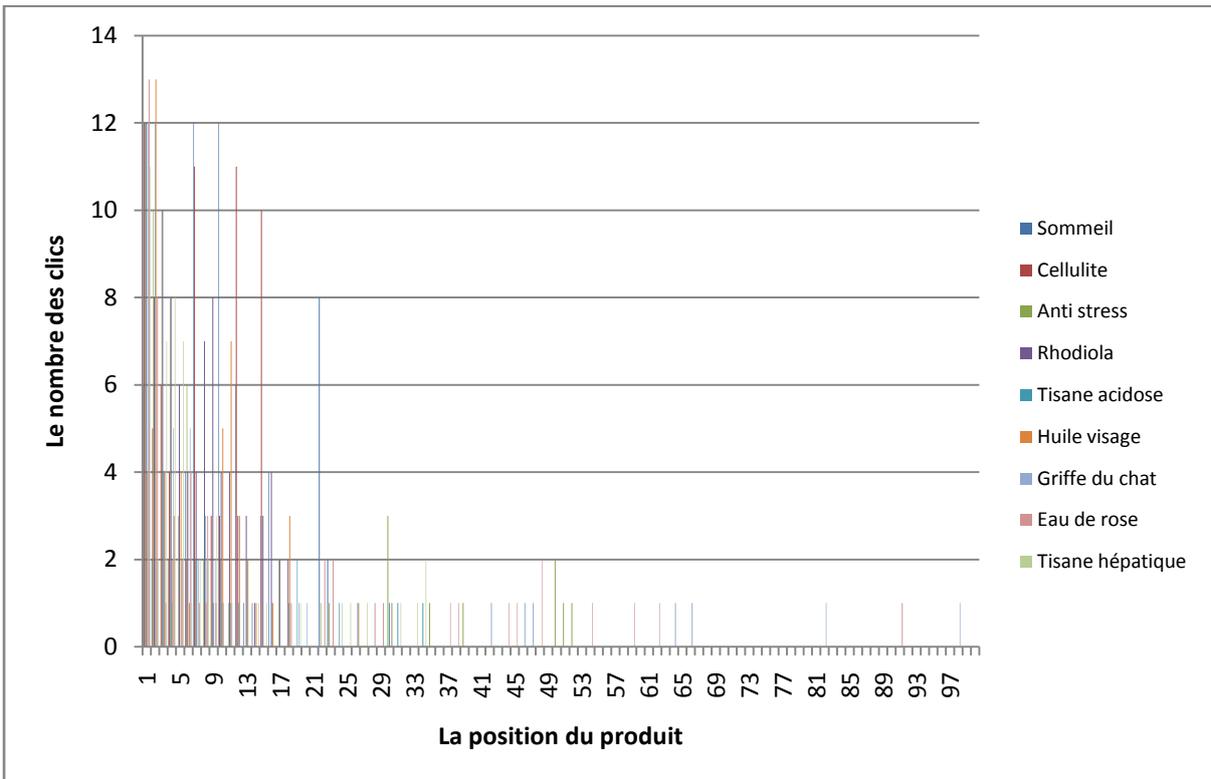


Figure 49 – La répartition des clics des utilisateurs pendant les tests, sur la copie du site

En analysant le cadre de tests, les clics des utilisateurs dans les différentes listes de résultats dans, présentés dans la Figure 49, on peut voir que seulement quelques utilisateurs ont consulté des

résultats qui ne se trouvent pas sur la première page. La distribution de leurs clics respecte bien la même forme de longue traîne (Anderson, 2009), tout comme la distribution des clics des utilisateurs réels du site (Figure 42).

Nous pouvons constater une différence importante de la distribution des clics entre la Figure 48 et la Figure 49. Elle est probablement causée par le fait que les utilisateurs réels du site sont des utilisateurs plus avisés, qui connaissent mieux les produits, alors que les utilisateurs des tests les connaissent peu (seulement 64% des utilisateurs des tests connaissent les compléments alimentaires, et seulement 29% en prennent). Par conséquent, nos utilisateurs ont cliqué plus de résultats pour pouvoir décider si le produit est pertinent pour la requête ou non. En outre, le fait d'avoir demandé de trouver plusieurs produits pertinents pour les requêtes, engendre plus de clics aussi.

Pour éviter cet inconvénient, nous n'avons considéré que les produits qui ont obtenu une note supérieure à 5, comme étant pertinents pour une requête. La Figure 50 présente la répartition des produits pertinents.

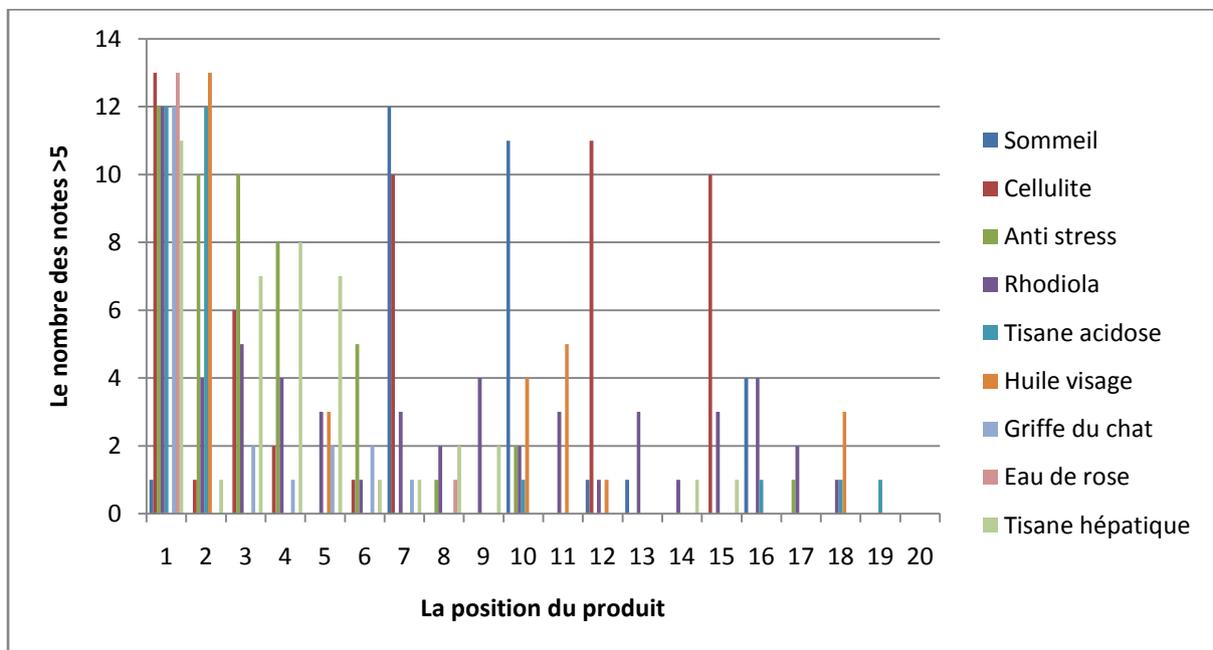


Figure 50 – La répartition des produits considérés pertinents

En analysant la Figure 49 et la Figure 50, on se rend compte que suite à notre reclassement des résultats, les clics des utilisateurs et les résultats pertinents sont situés dans la première partie de la liste des résultats.

Dans la Figure 48, on observe un pic de clics, pour rhodiola à la 12ème position, donc sur la deuxième page. Nous avons comparé les résultats de nos tests pour cette requête (Figure 51). Cette comparaison montre bien que la prise en compte de la température fait remonter les produits pertinents vers le début de la liste de résultats.

Les différents pics d'utilisation obtenus dans le cas de nos tests pour les requêtes cellulite et sommeil dans la deuxième partie de la liste des résultats sont une conséquence du fait que pendant la période que nous avons considérée, ces deux requêtes n'ont pas eu beaucoup de clics réalisés sur leurs

différents résultats. . En conséquence, les produits de la liste de résultats de ces requêtes ne sont pas suffisamment chauds pour être remontés dans la liste.

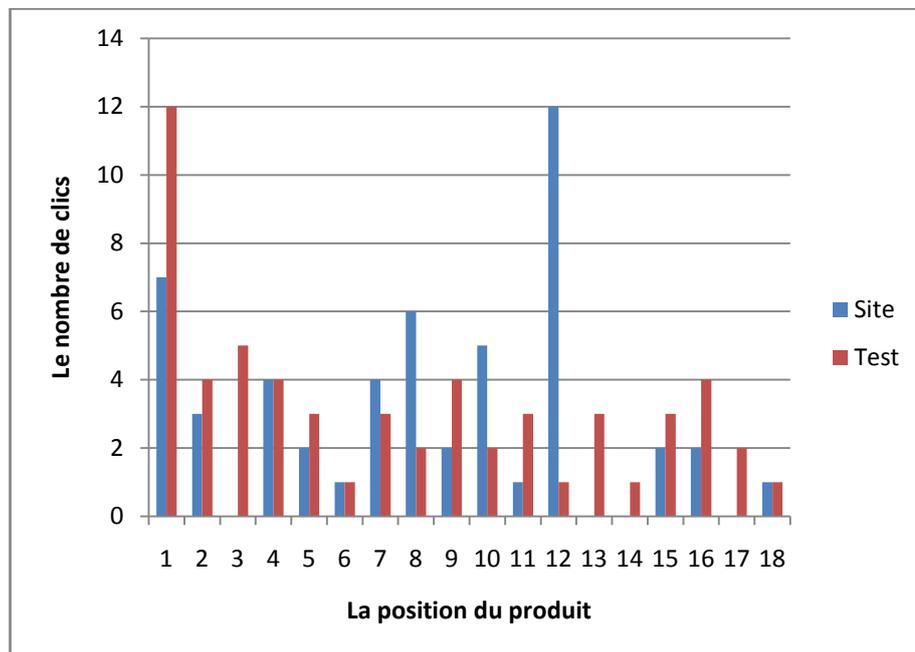


Figure 51 – La comparaison pour la requête rhodiola

Pour pouvoir évaluer l’apport de notre contribution par rapport à la performance du moteur de recherche du site, nous réalisons une comparaison de la précision des deux approches (e.g., la liste initiale des résultats du moteur de recherche et celle reclassée). Pour le calcul de la précision de chaque variante, la Précision à k résultats affichés (P@k) est utilisée. Pour le calcul de cette précision, l’équation (12) est employée.

$$P@k = \frac{1}{k} * \sum_{i=1}^k rel(d_i), \text{ où } rel(d_i) = \begin{cases} 1, & \text{si } d_i \text{ est relevant} \\ 0, & \text{si } d_i \text{ n'est pas relevant} \end{cases} \quad (12)$$

Dans notre cas, nous n’avons pas une mesure de base qui indique quels sont les résultats pertinents pour une requête parmi ceux retournés par le moteur de recherche. Pour établir quels sont les résultats pertinents pour une requête, nous utilisons les clics des utilisateurs. Pour chaque requête :

1. Nous considérons tous les clics des utilisateurs sur les résultats, indépendamment de leur ordre de réalisation.
2. Nous récupérons le maximum de clics (MAX) réalisés sur un résultat de la liste.
3. Nous considérons comme pertinents, tous les résultats qui ont été cliqués au moins MAX/2 fois. De cette façon, nous favorisons le consensus entre les utilisateurs qui ont consulté la liste des résultats.

Nous appliquons le même processus pour les clics des utilisateurs sur le site de production récupérés depuis la base de données des interactions, et pour les clics enregistrés pendant les tests

réalisés. Le Tableau 50 présente les résultats obtenus pour le calcul de la précision P@k pour le site de production et le Tableau 51 pour les tests.

Tableau 50 – Les résultats du calcul du P@k pour le site de production

	P@5	P@10	P@15	P@20
Sommeil	0.6	0.4	0.2667	0.25
Cellulite	0	0.1	0.0667	0.05
Anti stress	0.2	0.1	0.0667	0.05
Rhodiola	0.2	0.2	0.2	0.15
Tisane acidose	0.2	0.1	0.0667	0.05
Huile visage	0.2	0.1	0.0667	0.05
Griffe du chat	0.2	0.1	0.0667	0.05
Eau de rose	0.2	0.1	0.0667	0.05
Tisane hépatique	0.4	0.2	0.1333	0.1

Tableau 51 – Les résultats du calcul du P@k pour les tests

	P@5	P@10	P@15	P@20
Sommeil	0.2	0.3	0.2667	0.2
Cellulite	0.4	0.3	0.3333	0.25
Anti stress	0.8	0.5	0.3333	0.25
Rhodiola	1	0.7	0.4667	0.35
Tisane acidose	0.4	0.2	0.1333	0.1
Huile visage	0.2	0.1	0.1333	0.1
Griffe du chat	0.2	0.1	0.0667	0.05
Eau de rose	0.4	0.2	0.1333	0.1
Tisane hépatique	0.8	0.4	0.2667	0.2

En analysant ces résultats, nous observons que la P@5 est meilleure pour toutes les requêtes pour les résultats affichés à l'utilisateur après notre reclassement. Ce qui signifie que parmi les 5 premiers résultats affichés pour chaque requête, il y a plus de résultats pertinents dans la liste de résultats reclassée que dans celle qui ne l'est pas. Pour la requête « griffe du chat », nous obtenons les mêmes précisions dans les deux cas : les utilisateurs du site de production et ceux qui ont participé à notre test, n'ont trouvé que le premier résultat affiché pertinent pour la requête (i.e.griffe-du-chat.html).

En analysant plus en détails les résultats pour la requête « rhodiola » (présentée dans la Figure 52), nous observons que la P@5 est maximale (P@5=1) pour nos tests. Cela veut dire que pour les utilisateurs de test, les 5 résultats affichés sont pertinents pour la requête. Pour le site de production, la précision est constante pour P@5, P@10 et P@15, parce que parmi les 5 premiers résultats, il n'y a qu'un qui est pertinent, dans les 10 premiers, il y a 2 et dans les 15 premiers, il y en a 3.

Pour mieux comprendre ce qui se passe avec cette requête, nous présentons dans le Tableau 52, les 5 premiers résultats affichés aux utilisateurs pendant les tests pour cette requête. Nous constatons que les 3 résultats pertinents pour les utilisateurs du site de production sont remontés vers les trois premières positions de la liste reclassée présentée aux utilisateurs pendant le test. En outre de ces

trois résultats, les utilisateurs de test ont considéré que les deux autres résultats affichés parmi les 5 premiers sont pertinents, d'où la précision de 100%.

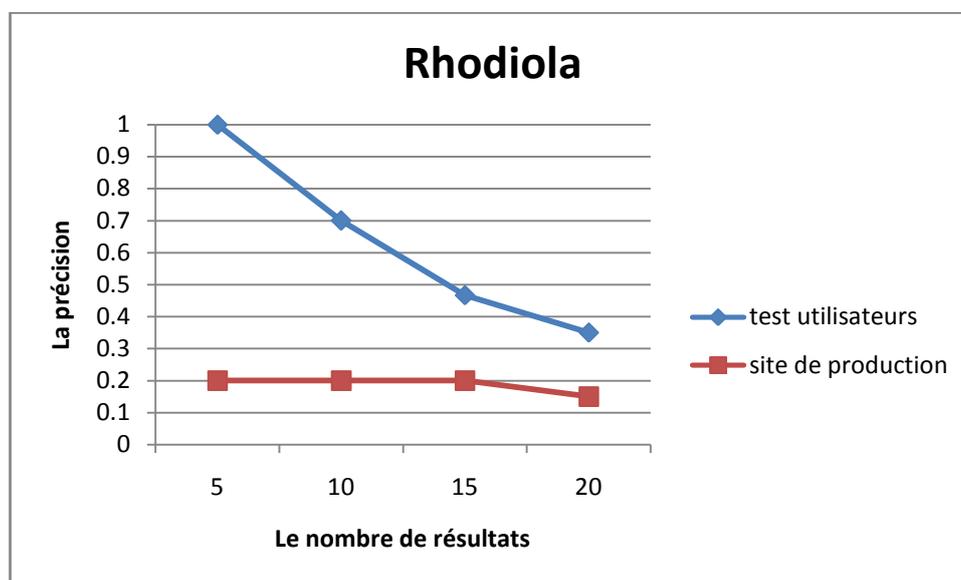


Figure 52 – Comparaison entre le P@k pour le site de production et le P@k des tests utilisateurs pour la requête « rhodiola »

Le produit	Le site de production		Test utilisateurs	
	Position	Pertinent	Position	Pertinent
rhodiola-120-gelules.html	12	Oui	1	Oui
duo-minceur.html	1	Oui	2	Oui
duo-sommeil-serein.html	8	Oui	3	Oui
duo-articulations.html	2	Non	4	Oui
duo-menopause.html	7	Non	5	Oui

Tableau 52 – Les premiers résultats pertinents pour la requête « rhodiola »

Ces résultats expérimentaux montrent que le reclassement des résultats obtenus pour une requête, en tenant compte de la température des résultats, améliore la précision de la requête. Le rappel reste le même, car nous ne recherchons pas de nouveaux résultats pertinents pour la requête.

En ce qui concerne les réponses des utilisateurs aux questions liées à leur avis sur le fait que leurs actions sont enregistrées et analysées pour aider les autres à retrouver l'information plus vite, les avis sont assez divisés. Il y en a certains qui sont d'accord, mais d'autres qui trouvent qu'ils sont « flingués » et ils sont gênés :

- « mes recherches sont ma propriété et je ne souhaite pas les céder ou qu'elles soient usurpées »
- « cela peut être intéressant si l'utilisateur a le choix et si les données sont anonymes. »
- « c'est bénéfique pour gagner le temps et l'effort fait pour les recherches. En plus, j'appartiens sans aucun doute à une communauté qui me ressemble »

- « je pense que cela permet aux autres utilisateurs d’avoir une idée sur la tendance générale, donc cela apporte à l’utilisateur plus d’information sur la vente d’un produit ou les produits les plus visités. »
- « Good idea ! Human is a social being, so help one another ! »

7.9. Bilan

Dans le cadre de ce chapitre nous avons présenté l’implémentation que nous avons réalisée afin de valider la chaîne de production des métadonnées d’usage ainsi que leur utilité dans l’amélioration de la pertinence des résultats de recherche.

Nous avons réalisé la validation de ce travail dans le domaine du commerce électronique. Nous avons bénéficié d’une base de données, des interactions des utilisateurs d’un site de vente en ligne, fournis par une société toulousaine. Cette base nous a permis de calculer la température des pages web du site en fonction des interactions des utilisateurs avec le site.

Une fois la température des pages web calculée, nous l’avons utilisée pour réaliser le reclassement des résultats retournés par le moteur de recherche du site pour des requêtes.

Nous avons réalisé une première validation de l’intérêt de l’exploitation de l’usage dans le reclassement des résultats d’une requête afin d’obtenir une précision plus grande. Cette validation, relativement simple à mettre en œuvre, souffre toutefois de nombreuses limitations. Nous pouvons citer :

- La façon dont les résultats sont considérés comme pertinents pour une requête. Nous avons pris en compte seulement les clics des utilisateurs sur le résultat affiché dans la liste. Si cette façon est considérée insuffisante, alors nous pouvons utiliser aussi :
 - Le nombre d’interactions après le clic sur le produit ;
 - Le produit est mis au panier ou non ;
 - Le produit est acheté ou non.
- Le nombre d’utilisateurs de tests est trop faible ;
- La réalisation des tests sur une copie du site ;
- La période considérée pour le test (e.g., seulement 3 mois). Une période plus longue peut être envisagée, pour avoir encore plus d’interactions et plus d’utilisateurs, mais avec le risque de changements (i.e., modification de l’ergonomie du site, du contenu, des produits).

Conclusions générales et perspectives

Synthèse des propositions

Dans le cadre de cette thèse, nous nous sommes intéressés au processus de gestion des documents multimédias et plus précisément à la prise en compte de l'utilisateur dans ce processus. En effet, l'interaction de l'utilisateur avec ces documents est constante et à plusieurs niveaux : via le système, avec les documents, mais aussi avec les métadonnées qui décrivent ces documents.

Avant de nous intéresser à l'intégration de l'utilisateur, nous avons étudié le processus de gestion des documents multimédias en nous focalisant dans un premier temps sur la création des métadonnées. La multitude de standards de métadonnées existants et n'étant pas interopérables, ils ne sont pas facilement intégrables dans le même système. Pour faire face à ce problème d'interopérabilité, nous avons proposé un format de métadonnées qui intègre les standards de métadonnées les plus utilisés, qui soit extensible en structure et en vocabulaire. Ce format est adaptable à plusieurs domaines d'application et permet de décrire les documents multimédias avec une granularité plus ou moins fine, en fonction du besoin.

La création des métadonnées est très importante dans la gestion des documents et très consommatrice de ressources du système. Nous avons proposé d'intégrer l'utilisateur dans le processus de création des métadonnées à travers l'exécution de l'indexation en deux étapes : au moment de l'acquisition du contenu (i.e., indexation implicite) et au moment de l'exécution de la requête de l'utilisateur, si besoin (i.e., indexation explicite). Pour l'indexation implicite, nous avons eu recours à un ensemble fixe d'algorithmes d'indexation qui extraient des métadonnées générales. Pour l'indexation explicite, nous avons employé des algorithmes plus spécifiques, qui sont choisis en fonction de la requête de l'utilisateur. Pour améliorer la sélection des algorithmes, nous avons intégré les contextes d'acquisition des contenus multimédias, les contraintes d'exécution des algorithmes d'indexation et les conditions dans lesquelles les algorithmes ont des résultats optimaux. Nous avons proposé également de prendre en compte l'historique des requêtes des utilisateurs pour adapter la liste des algorithmes implicites aux besoins récurrents des utilisateurs.

Les modes d'interaction de l'utilisateur avec le système de gestion des contenus multimédias ne se résument pas qu'aux requêtes des utilisateurs. Il existe différentes façons d'interagir avec un système, à travers : les clics, les déplacements de la souris, le défilement, le regard, etc. En outre, l'usage collectif d'un système tisse des liens entre les documents auxquels on n'aurait pas pensé au moment de la création des documents.

La dernière contribution de cette thèse se situe dans l'enrichissement des métadonnées à travers l'usage. Nous avons étendu le format de métadonnées proposé en ajoutant le concept de température et des liens d'usage. La température est utilisée pour représenter la popularité des métadonnées : plus un document/une métadonnée est utilisée, plus elle devient « chaude ». La température reflète bien l'évolution de la popularité. Nous avons proposé un algorithme de calcul de la température dans le temps, en fonction des interactions de l'utilisateur avec le document et les métadonnées.

Nous avons introduit deux types de structures pour les documents multimédias : la structure d'usage et la structure dynamique. La structure d'usage crée des liens entre les différents blocs du

document multimédia, de la même façon que la structure hypermédia. En ce qui concerne la structure dynamique, elle peut être associée à tout type de structure.

Cette extension du format de métadonnées proposée peut être utilisée de plusieurs manières. Les liens indirects peuvent être créés à tout moment pendant l'interaction de l'utilisateur avec le document, le type du lien étant établi en fonction du besoin identifié dans le domaine d'application. Dans l'extension proposée, nous avons attribué une température par groupe d'utilisateurs, mais il n'est pas obligatoire d'avoir plusieurs groupes d'utilisateurs. Au moment de l'utilisation du format dans un certain domaine, il est possible de n'avoir qu'une température pour tous les utilisateurs, en ignorant leur appartenance à un ou plusieurs groupes.

Nous avons validé nos propositions dans le cadre de deux domaines différents. Dans un premier temps, nous avons implémenté le format de métadonnées, l'indexation implicite/explicite et le choix des algorithmes d'indexation en fonction de la requête de l'utilisateur dans le cadre du projet LINDO. Le projet a abouti à la réalisation d'un système de gestion de vidéosurveillance distribué, dans lequel ce ne sont pas les contenus multimédias qui sont transférés vers des serveurs d'indexation, mais ce sont les bons algorithmes d'indexation qui sont déployés sur les serveurs où le contenu est stocké, en optimisant ainsi la consommation des ressources.

Dans un deuxième temps, nous avons validé l'enrichissement des métadonnées par l'usage dans le domaine du commerce électronique. Nous avons exploité les interactions des utilisateurs avec un site de vente de produits naturels. Nous avons utilisé ces interactions pour calculer la température des différentes pages du site sur une période de deux mois. Nous avons utilisé cette température pour faire le reclassement des résultats obtenus suite à l'exécution des requêtes des utilisateurs. Afin de valider l'usage de la température, nous avons effectué des tests utilisateurs. Ces tests ont démontré que l'utilisation de la température augmente la performance du système en remontant dans la liste, les résultats qui ont été les plus consultés pendant la période de temps considérée, et qui sont donc les plus populaires. Nos résultats sont fortement dépendants du comportement des utilisateurs : plus il y a d'utilisateurs qui consultent des résultats situés sur plusieurs pages de résultats d'une requête, plus performant est le reclassement des résultats en utilisant la température.

Pour conclure, nous pouvons dire que dans cette thèse, nous avons proposé une première prise en compte des interactions de l'utilisateur dans le cadre de l'extraction et de l'enrichissement des métadonnées associées aux documents multimédias. Cette prise en compte permet d'un côté, de mieux utiliser les ressources du système et de l'autre, d'augmenter sa performance.

Perspectives de recherche

Nos propositions ouvrent plusieurs perspectives scientifiques à court et à plus long terme. Nous soulignons ici certaines de ces perspectives.

A court terme, nous voulons réaliser une validation à grande échelle de la prise en compte de l'utilisateur dans le cadre de la création des métadonnées. Pour réaliser cette validation, nous avons besoin de plusieurs algorithmes d'indexation qui extraient différentes caractéristiques des contenus multimédias et qui ont des performances différentes en fonction des caractéristiques physiques et techniques des contenus à indexer. Pour chacun de ces algorithmes, un travail de mappage de leur sortie vers le format proposé doit être réalisé.

Nous avons appliqué cette approche dans le cadre d'un cas de vidéo surveillance sans prendre en compte les problèmes de confidentialité et d'accès aux données que l'on retrouve dans ce domaine. Une étude est en cours sur l'intégration de ces contraintes dans notre approche afin de restreindre l'accès des utilisateurs, aux données et aux ressources du système en fonction de leur contexte (Al Kukhun et al., 2012). pour spécifier quels droits possède chaque utilisateur dans un contexte donné. Par exemple, un agent de sécurité ne peut écouter les enregistrements audio que s'il utilise le système depuis le centre de commande. Mais dans le domaine de la vidéo surveillance, il y a des situations d'urgence où, offrir des réponses alternatives à l'utilisateur, s'il n'a pas le droit de consulter certaines ressources, est très important. Par exemple, dans le cas d'un vol à l'arraché, un agent de sécurité sur le terrain peut avoir besoin d'écouter certaines conversations enregistrées par le système, mais il n'a pas accès à ce type de contenu en dehors du centre de commande. Dans la situation courante de l'agent, avoir des réponses alternatives aux contenus audio peut être important pour l'aider à poursuivre son investigation, donc dans ce cas, nous pouvons, par exemple, lui fournir une partie des transcriptions des conversations qui l'intéresse. Afin de réaliser cette prise en compte de la situation et du contexte de l'utilisateur, nous intégrons le système PSQRS (Al Kukhun and Sèdes, 2008) dans l'architecture LINDO. Ce système permet effectivement de chercher des solutions alternatives dans le cas d'un refus d'accès à une ressource, en fonction du contexte et la situation de l'utilisateur.

Dans le cadre d'un système distribué, tel que LINDO, où plusieurs moteurs de recherche peuvent être utilisés sur des serveurs distants afin de réaliser l'indexation et la recherche des contenus multimédias stockés sur le serveur, la fusion des résultats d'une requête exécutée sur plusieurs serveurs doit être réalisée. Nous allons étudier ce problème pour voir comment on peut combiner, d'un côté, la sémantique de la requête et des métadonnées associées aux résultats (Stegmaier et al., 2010) et de l'autre côté, la température des documents mais aussi des métadonnées.

En ce qui concerne l'enrichissement des métadonnées, nous devons réaliser une étude qualitative et quantitative plus poussée pour valider notre proposition. Dans les expérimentations que nous avons menées, nous avons considéré les clics des utilisateurs comme étant un indicateur de l'intérêt des utilisateurs envers les documents, mais pour valider la proposition jusqu'au bout, et sans ambiguïté, il faudra surtout considérer le retour en termes de ventes via le site.

Dans les expérimentations menées, nous nous sommes surtout intéressés à la validation du concept de température sans nous attarder sur la formule de calcul pour le reclassement des résultats d'une requête. Dans la formule utilisée, nous n'avons considéré que la température globale des documents, et pas celle des liens indirects. Nous devons analyser le graphe complet qui est créé par les liens indirects pour identifier un classement des documents en fonction de la température de chaque lien du graphe.

Dans le cadre de l'algorithme d'augmentation de la température, nous n'avons considéré que les clics des utilisateurs comme indicateurs de leur intérêt pour un document/partie de document. Nous pouvons envisager deux perspectives. Une à court terme porte sur la prise en compte d'autres indicateurs, tels que le temps que l'utilisateur passe sur un document, le suivi de son regard ; une à plus long terme pour intégrer l'ordre de réalisation des clics dans le calcul de la température. Comme (Joachims et al., 2007) l'ont démontré, les clics des utilisateurs sur une liste de résultats sont biaisés, et ils ne sont pas de bons indicateurs de la préférence absolue des utilisateurs envers un document, ils indiquent plutôt des préférences relatives d'un document par rapport à un autre.

En analysant le calcul et la propagation de la température, nous pouvons observer qu'ils ne respectent pas fidèlement l'évolution de la température d'un objet réel. Dans un premier temps, nous avons considéré que la température ne se propage que du « père au fils » et vice-versa, de l'élément contenant vers l'élément contenu et vice-versa. En réalité, il y a aussi une propagation de la tempéra-

ture entre les éléments qui se trouvent à proximité. Nous envisageons donc de réaliser une propagation de la température aux éléments qui se trouvent sur le même niveau. Dans un deuxième temps, après la propagation de la température du père au fils, le cumul de la température propagée à tous les éléments contenus par le « père » est supérieur à la température d'augmentation du « père », ce qui ne respecte pas la réalité. Nous allons donc modifier les formules utilisées pour le calcul de la température pour prendre en compte cette propriété de la température.

Dans nos expérimentations, nous avons considéré que les utilisateurs appartiennent à un seul groupe. A plus long terme, nous voulons prendre en compte le profil des utilisateurs (Tchuente et al., 2011) pour pouvoir déterminer le groupe auquel ils appartiennent. Nous devons aussi prendre en compte dans l'application de la température, le fait qu'un utilisateur peut appartenir à plusieurs groupes.

L'algorithme de calcul de la température contient plusieurs paramètres. Dans le cadre des expérimentations, nous avons pris pour ces paramètres, des valeurs déterminées après plusieurs tests. Pour choisir ces paramètres d'une manière plus pertinente, nous devons appliquer des algorithmes d'apprentissage (Kotsiantis, 2007) pour chaque domaine d'application. Il est possible d'obtenir des valeurs différentes pour chaque cas d'utilisation.

A plus long terme, nous voulons mettre en commun nos deux propositions de prise en compte de l'utilisateur (dans le choix des algorithmes d'indexation et dans l'enrichissement des métadonnées). Cela peut être réalisé, d'un côté par une augmentation de la température au moment de l'exécution de la requête de l'utilisateur (la température des métadonnées utilisées pour répondre à la requête est augmentée) et d'un autre côté, par une prise en compte de la température des métadonnées au moment de la sélection des algorithmes.

Nous envisageons aussi d'appliquer le concept de température à d'autres domaines, par exemple les sites communautaires, les réseaux sociaux, les systèmes de recommandation, pour pouvoir avoir une validation plus complète de notre approche.

Références

- Abiteboul, S., 1997. Querying Semi-Structured Data, in: ICDT. pp. 1–18.
- Abreu, B., Botelho, L., Cavallaro, A., Douxchamps, D., Ebrahimi, T., Figueiredo, P., Macq, B., Mory, B., Nunes, L., Orri, J., Trigueiros, M.J., Violante, A., 2000. Video-based multi-agent traffic surveillance system, in: Intelligent Vehicles Symposium, 2000. IV 2000. Proceedings of the IEEE. pp. 457–462.
- Adobe Developers Association, 1992. TIFF Revision 6.0 (Standard).
- Adobe Systems Incorporated, 2010a. XMP Specification – Adding Intelligence to Media, XMP specification part 1 data model, serialization, and core properties (Standard).
- Adobe Systems Incorporated, 2010b. XMP Specification – Adding Intelligence to Media, XMP specification part 2, additional properties (Standard).
- Adobe Systems Incorporated, 2010c. XMP Specification – Adding Intelligence to Media, XMP specification part 3, storage in files (Standard).
- Agichtein, E., Brill, E., Dumais, S., 2006. Improving web search ranking by incorporating user behavior information, in: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '06. ACM, New York, NY, USA, pp. 19–26.
- Agosti, M., Buccio, E.D., Nunzio, G.M.D., Ferro, N., Melucci, M., Miotto, R., Orio, N., 2007. Distributed Information Retrieval and Automatic Identification of Music Works in SAPIR, in: Ceci, M., Malerba, D., Tanca, L. (Eds.), Proceedings of the 15th Italian Symposium on Advanced Database Systems (SEBD). pp. 479–482.
- Al Kukhun, D., Codreanu, D., Manzat, A.-M., Sèdes, F., 2012. Applying Pervasive and Flexible Access Control to Distributed Multimedia Retrieval, in: International Workshop on Information Management for Mobile Applications in conjunction with VLDB 2012. CEUR Workshop Proceedings, pp. 41–48.
- Al Kukhun, D., Sèdes, F., 2008. Adaptive Solutions for Access Control within Pervasive Healthcare Systems, in: Helal, S., Mitra, S., Wong, J., Chang, C.K. (Eds.), International Conference On Smart homes and health Telematics (ICOST), LNCS. Springer, pp. 42–53.
- ALA/ALCTS/CCS Committee, 2004. Cataloging: Description and Access - Task Force on Metadata: Summary Report.
- Alboody, A., 2011. Réception des données spatiales et leurs traitements: analyse d'images satellites pour la mise à jour des SIG par enrichissement du système de raisonnement spatial RCC8. (Thèse de l'Université Paul Sabatier).
- Allen, J.F., 1983. Maintaining Knowledge About Temporal Intervals. Communications of the ACM 26, 832–843.
- Allen, J.F., 1991. Time and time again: The many ways to represent time. International Journal of Intelligent Systems 6, 341–355.
- Amir, S., 2011. Un système d'intégration des métadonnées dédiées au multimédia (Thèse de l'Université Lille 1).
- Amous, I., Jedidi, A., Sèdes, F., 2004. Modeling and querying multimedia document, in: International Conference on Intelligent Access of Multimedia Documents on Internet (MediaNet'2004). HERMES.

- Amous, I., Jedidi, A., Sèdes, F., 2005. A Contribution to Multimedia Document Modeling and Querying. *Multimedia Tools and Applications* 25, 391–404.
- Anderson, C., 2009. *La Longue Traîne*, 2e édition.
- ANSI/NISO, 2007. The Dublin Core Metadata Element Set (Standard No. ANSI/NISO Z39.85). ANSI/NISO.
- Apers, P.M.G., Blanken, H.M., Houtsma, M.A.W., 1997. *Multimedia Databases in Perspective*. Springer-Verlag Berlin and Heidelberg GmbH & Co. K.
- Arndt, R., Troncy, R., Staab, S., Hardman, L., Vacura, M., 2007. COMM: Designing a Well-Founded Multimedia Ontology for the Web, in: 6th International Semantic Web Conference, LNCS. Springer, pp. 30–43.
- Auffret, G., Carrive, J., Chevet, O., Dechilly, T., Ronfard, R., Bachimont, B., 1999. Audiovisual-based hypermedia authoring: using structured representations for efficient access to AV documents, in: *Proceedings of the Tenth ACM Conference on Hypertext and Hypermedia*: Returning to Our Diverse Roots: Returning to Our Diverse Roots, HYPERTEXT '99. ACM, New York, NY, USA, pp. 169–178.
- Baccot, B., Grigoras, R., Charvillat, V., 2011. Reinforcement Learning for Online Optimization of Banner Format and Delivery, in: *Online Multimedia Advertising*. IGI Global, pp. 13–31.
- Baeza-Yates, R., Navarro, G., 2011. Queries: Languages and Properties, in: *Modern Information Retrieval: The Concepts and Technology Behind Search*. Addison-Wesley, pp. 255–280.
- Baeza-Yates, R., Ribeiro-Neto, B., 2011. Relevance Feedback and Query Expansion, in: *Modern Information Retrieval: The Concepts and Technology Behind Search*. Addison Wesley, pp. 177–202.
- Baker, T., Blanchi, C., Brickley, D., Duval, E., Heery, R., Johnston, P., Kalinichenko, L., Neuroth, H., Sugimoto, S., 2002. Principles of metadata registries (White paper). DELOS Network of Excellence on Digital Libraries.
- Barras, C., Geoffrois, E., Wu, Z., Liberman, M., 2001. Transcriber: Development and use of a tool for assisting speech corpora production. *Speech Commun.* 33, 5–22.
- Batko, M., Falchi, F., Lucchese, C., Novak, D., Perego, R., Rabitti, F., Sedmidubsky, J., Zezula, P., 2010. Building a web-scale image similarity search system. *Multimedia Tools and Applications* 47, 599–629.
- Bender, M., Michel, S., Triantafillou, P., Weikum, G., Zimmer, C., 2005. MINERVA: Collaborative P2P Search. Presented at the 31st International Conference on Very Large Data Bases (VLDB), pp. 1263–1266.
- Bhokal, J., Macfarlane, A., Smith, P., 2007. A review of ontology based query expansion. *Inf. Process. Manage.* 43, 866–886.
- Bloehdorn, S., Petridis, K., Saathoff, C., Simou, N., Tzouvaras, V., Avrithis, Y., Handschuh, S., Kompatsiaris, Y., Staab, S., Strintzis, M.G., 2005. Semantic Annotation of Images and Videos for Multimedia Analysis, in: Gómez-Pérez, A., Euzenat, J. (Eds.), *Proceedings of the Second European Semantic Web Conference (ESWC)*. Springer, pp. 592–607.
- Böhm, K., Rakow, T.C., 1994. Metadata for Multimedia Documents. *SIGMOD RECORD* 23, 21–26.
- Boisson, F., Crucianu, M., Vodislav, D., 2008. Publication Framework for Content-Based Search in Heterogeneous Distributed Multimedia Databases (Scientific Rapport No. CEDRIC-08-1585). CEDRIC laboratory, CNAM-Paris, France.
- Bouquillard, D., 2001. Le multimédia, une approche audiovisuelle. Impacts: Regards croisés sur le Multimédia 35, 39–69.
- Broder, A., 2002. A taxonomy of web search. *SIGIR Forum* 36, 3–10.

- Brunie, L., Pierson, J.-M., Coquil, D., 2002. Semantic collaborative Web caching, in: Proceedings of the Third International Conference on Web Information Systems Engineering, 2002. WISE 2002. pp. 30 – 39.
- Brusilovsky, P., 2001. Adaptive Hypermedia. *User Modeling and User-Adapted Interaction* 11, 87–110.
- Brut, M., Codreanu, D., Dumitrescu, S., Manzat, A.-M., Sedes, F., 2011. A distributed architecture for flexible multimedia management and retrieval, in: Proceedings of the 22nd International Conference on Database and Expert Systems Applications - Volume Part II, DEXA'11. Springer-Verlag, pp. 249–263.
- Brut, M., Codreanu, D., Manzat, A.-M., Sedes, F., 2011a. Adapting Indexation to the Content, Context and Queries Characteristics in Distributed Multimedia Systems, in: International IEEE Conference on Signal-Image Technologies and Internet-Based System. Presented at the SITIS, IEEE Computer Society, pp. 118–125.
- Brut, M., Codreanu, D., Manzat, A.-M., Sedes, F., 2011b. Distributed multimedia indexing and optimal resources utilization: an implementation based on metadata, context and usage. *The Journal of Multimedia Processing and Technologies, Intelligent Multimedia Processing* 2, 197–225.
- Brut, M., Laborie, S., Manzat, A.-M., Sedes, F., 2009a. A generic metadata framework for the indexation and the management of distributed multimedia contents, in: Proceedings of the 3rd International Conference on New Technologies, Mobility and Security, NTMS'09. IEEE Press, pp. 383–387.
- Brut, M., Laborie, S., Manzat, A.-M., Sedes, F., 2009b. A Framework for Automatizing and Optimizing the Selection of Indexing Algorithms, in: Sartori, F., Sicilia, M.Á., Manouselis, N. (Eds.), *Metadata and Semantic Research*. Presented at the MTSR, Springer Berlin Heidelberg, pp. 48–59.
- Brut, M., Sedes, F., 2010. An ontology-based approach of multimedia information personalized search, in: Proceedings of the 8th International Conference on Adaptive Multimedia Retrieval: Context, Exploration, and Fusion, AMR'10. Springer-Verlag, pp. 31–45.
- Brut, M., Sedes, F., Manzat, A.M., 2009. A Web Services Orchestration Solution for Semantic Multimedia Indexing and Retrieval, in: 2010 International Conference on Complex, Intelligent and Software Intensive Systems. IEEE Computer Society, pp. 1187–1192.
- Cabanac, G., 2008. *Fédération et amélioration des activités documentaires par la pratique d'annotation collective* (Thèse de l'Université Paul Sabatier).
- Cabanac, G., Chevalier, M., Chrisment, C., Julien, C., 2008. Exploiting the Annotation Practice for Personal and Collective Information Management, in: International Workshop on Model Driven Information Systems Engineering: Enterprise, User and System Models. CEUR-WS, pp. 67–78.
- Callan, J.P., Lu, Z., Croft, W.B., 1995. Searching distributed collections with inference networks, in: Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '95. ACM, pp. 21–28.
- Camera & Image Products Association, Japan Electronics and Information Technology Industries Association, 2010. Exchangeable image file format for digital still cameras: Exif Version 2.3 (Standard).
- Candillier, L., Jack, K., Fessant, F., Meyer, F., 2009. State-of-the-Art Recommender Systems, in: Chevalier, M., Julien, C., Soule-Dupuy, C. (Eds.), *Collaborative and Social Information Retrieval and Access*. IGI Global, pp. 1–22.
- Chan, L.M., Zeng, M.L., 2006. Metadata Interoperability and Standardization - A Study of Methodology Part I. *D-Lib Magazine* 12.

- Charhad, M., Quenot, G., 2004. Semantic video content indexing and retrieval using conceptual graphs, in: 2004 International Conference on Information and Communication Technologies: From Theory to Applications, 2004. Proceedings. pp. 399 – 400.
- Chen, S.-C., Shyu, M.-L., Zhao, N., 2004. SMARXO: towards secured multimedia applications by adopting RBAC, XML and object-relational database, in: Proceedings of the 12th Annual ACM International Conference on Multimedia, MULTIMEDIA '04. ACM, New York, NY, USA, pp. 432–435.
- Chrisment, C., Sèdes, F., 2001. Document multimédia: indexation et recherche, in: CIFED/CIDE'2000, Colloque International Francophone Sur l'Écrit Et Le Document , Lyon,. Eds. Lyon, pp. 10–20.
- Ciaccia, P., Patella, M., Zezula, P., 1997. M-tree: An Efficient Access Method for Similarity Search in Metric Spaces, in: Proceedings of the 23rd International Conference on Very Large Data Bases, VLDB '97. Morgan Kaufmann Publishers Inc., pp. 426–435.
- Claypool, M., Le, P., Wased, M., Brown, D., 2001. Implicit interest indicators, in: Proceedings of the 6th International Conference on Intelligent User Interfaces, IUI '01. ACM, pp. 33–40.
- Clementini, E., Sharma, J., Egenhofer, M.J., 1994. Modelling topological spatial relations: Strategies for query processing. *Computers & graphics* 18, 815–822.
- Codreanu, D., Laborie, S., Sèdes, F., 2011. Modelling Multimedia Indexing Algorithms Description for Implicit and Explicit Indexation, in: 2011 5th FTRA International Conference on Multimedia and Ubiquitous Engineering (MUE). pp. 232 –238.
- Conan, V., Ferrané, I., Joly, P., Vasserot, C., 2003. KLIMT \square : Intermediations Technologies and Multimedia Indexing, in: Proceedings of the Third International Workshop on Content-Based Multimedia Indexing (CBMI). INRIA, pp. 11–18.
- Conklin, J., 1987. Hypertext: An Introduction and Survey. *Computer* 20, 17–41.
- Cuenca-Acuna, F.M., Peery, C., Martin, R.P., Nguyen, T.D., 2003. PlanetP: using gossiping to build content addressable peer-to-peer information sharing communities, in: 12th IEEE International Symposium on High Performance Distributed Computing, 2003. Proceedings. pp. 236 – 246.
- Day, H., 1966. Looking time as a function of stimulus variables and individual differences. *Perceptual and Motor Skills* 22, 423–428.
- De Sutter, R., Notebaert, S., Van de Walle, R., 2006. Evaluation of Metadata Standards in the Context of Digital Audio-Visual Libraries, in: Gonzalo, J., Thanos, C., Verdejo, M., Carrasco, R. (Eds.), *Research and Advanced Technology for Digital Libraries, Lecture Notes in Computer Science*. Presented at the 10th European Conference on Digital Libraries, Springer Berlin / Heidelberg, pp. 220–231.
- Digital Imaging Group, 2000. DIG35 Specification — Metadata for Digital Images (Version 1.0) (Standard).
- Digital Imaging Group, 2001. DIG35 Specification — Metadata for Digital Images (Version 1.1) (Standard).
- Digital Library Federation, 2010. <METS>Metadata Encoding and Transmission Standard: primer and reference manual, Version 1.6 (Standard).
- Djemal, K., 2010. De la modélisation à l'exploitation des documents à structures multiples.
- Döller, M., Stegmaier, F., Kosch, H., Tous, R., Delgado, J., 2010. Standardized interoperable image retrieval, in: Proceedings of the 2010 ACM Symposium on Applied Computing, SAC '10. ACM, pp. 880–886.

- Dou, Z., Song, R., Wen, J.-R., 2007. A large-scale evaluation and analysis of personalized search strategies, in: *Proceedings of the 16th International Conference on World Wide Web, WWW '07*. ACM, pp. 581–590.
- Egenhofer, M.J., 1994. Spatial SQL: A Query and Presentation Language. *IEEE Trans. on Knowl. and Data Eng.* 6, 86–95.
- Faw, T., Nunnally, J., 1967. The effects on eye movements of complexity, novelty, and affective tone. *Attention, Perception, & Psychophysics* 2, 263–267.
- Flickner, M., Sawhney, H., Niblack, W., Ashley, J., Huang, Q., Dom, B., Gorkani, M., Hafner, J., Lee, D., Petkovic, D., Steele, D., Yanker, P., 1995. Query by Image and Video Content: The QBIC System. *Computer* 28, 23–32.
- Ford, R., Robson, C., Temple, D., Gerlach, M., 2000. Metrics for Shot Boundary Detection in Digital Video Sequences. *Multimedia Systems* 8, 37–46.
- Forum Multimédia 92, 1992. Le livre blanc du multimédia : l'intégration numérique du son, des textes, des images et des données : 2930 septembre - 1er octobre 1992, CNIT Paris-La Défense. CAPRIC.
- Fourrel, F., 1998. Modélisation, indexation et recherche de documents structurés (Thèse de l'Université Joseph-Fourier).
- Fox, S., Karnawat, K., Mydland, M., Dumais, S., White, T., 2005. Evaluating implicit measures to improve web search. *ACM Trans. Inf. Syst.* 23, 147–168.
- García, R., Tsinaraki, C., Celma, Ó., Christodoulakis, S., 2008. Multimedia Content Description Using Semantic Web Languages, in: Kompatsiaris, Y., Hobson, P. (Eds.), *Semantic Multimedia and Ontologies*. Springer London, London, pp. 17–54.
- Gauvain, J.L., Lamel, L., Adda, G., 2001. Audio Partitioning and Transcription for Broadcast Data Indexation. *Multimedia Tools and Applications* 14, 187–200.
- Ghazal, M., Sedes, F., 2010. Système de recherche d'informations spatiales basé sur le croquis à main levée, in: *CORIA*. pp. 337–348.
- Giroux, P., Brunessaux, S., Brunessaux, S., Doucy, J., Dupont, G., Grilhères, B., Mombrun, Y., Saval, A., 2008. Weblab: An integration infrastructure to ease the development of multimedia processing applications, in: *International Conference on Software and System Engineering and Their Applications (ICSSEA)*.
- Gravano, L., García-Molina, H., Tomasic, A., 1999. GLOSS: text-source discovery over the Internet. *ACM Trans. Database Syst.* 24, 229–264.
- Greenberg, J., 2002. Metadata and the World Wide Web. *Encyclopedia of Library and Information Science* 42, 244–261.
- Gyllstrom, K., 2009. Enriching personal information management with document interaction histories (Thèse de University of North Carolina at Chapel Hill).
- Halvey, M.J., Keane, M.T., 2007. Analysis of online video search and sharing, in: *Proceedings of the Eighteenth Conference on Hypertext and Hypermedia, HT '07*. ACM, pp. 217–226.
- Hardman, L., Obrenović, Ž., Nack, F., Kerhervé, B., Piersol, K., 2008. Canonical processes of semantically annotated media production. *Multimedia Systems* 14, 327–340.
- Harvey, N.J.A., Jones, M.B., Saroiu, S., Theimer, M., Wolman, A., 2003. SkipNet: a scalable overlay network with practical locality properties, in: *Proceedings of the 4th Conference on USENIX Symposium on Internet Technologies and Systems*. USENIX Association, pp. 9–9.
- Haslhofer, B., Klas, W., 2010. A survey of techniques for achieving metadata interoperability. *ACM Comput. Surv.* 42, 7:1–7:37.

- Hijikata, Y., 2004. Implicit user profiling for on demand relevance feedback, in: Proceedings of the 9th International Conference on Intelligent User Interfaces, IUI '04. ACM, pp. 198–205.
- Hopfgartner, F., 2010. Personalised video retrieval: application of implicit feedback and semantic user profiles (Thèse de University of Glasgow).
- Hunter, J., 2001. Adding Multimedia to the Semantic Web: Building an MPEG-7 ontology, in: International Semantic Web Working Symposium (SWWS 2001). pp. 261–283.
- International Federation of Library Associations and Institutions, 1997. Functional Requirements for Bibliographic Records (Standard). IFLA.
- International Press Telecommunications Council, 1999. IPTC-NAA Information interchange Model Version No. 4.1 (Standard).
- International Press Telecommunications Council, 2010. IPTC Standard Photo Metadata, IPTC Core, IPTC Extension (Standard).
- Jameson, A., 2008. Adaptive Interfaces and Agents, in: Sears, A., Jacko, J.A. (Eds.), The Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications. CRC Press, Boca Raton, FL, pp. 433–458.
- Jansen, B.J., Spink, A., Koshman, S., 2007. Web searcher interaction with the Dogpile.com meta-search engine. *J. Am. Soc. Inf. Sci. Technol.* 58, 744–755.
- Japan Electronic Industry Development Association, 1995. Digital Still Camera Image File Format Standard (Exif) Version 1.0 (Standard).
- Järvelin, K., Kekäläinen, J., Niemi, T., 2001. ExpansionTool: Concept-Based Query Expansion and Construction. *Information Retrieval* 4, 231–255.
- Jaspers, E.G.T., Wijnhoven, R.G.J., Albers, A.H.R., Desurmont, X., Barais, M., Hamaide, J., Lienard, B., 2005. Candela-Storage, Analysis and Retrieval of Video Content in Distributed Systems: Real-Time Video Surveillance and Retrieval, in: Multimedia and Expo, 2005. ICME 2005. IEEE International Conference On. pp. 1553–1556.
- Jedidi, A., 2005. Modélisation générique de documents multimédia par des métadonnées - Mécanismes d'annotation et d'interrogation (Thèse de l'Université Paul Sabatier).
- Jiang, J.J., Conrath, D.W., 1997. Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy. *the International Conference on Research in Computational Linguistics* 19–33.
- Jiang, Y.-G., Yang, J., Ngo, C.-W., Hauptmann, A.G., 2010. Representations of Keypoint-Based Semantic Concept Detection: A Comprehensive Study. *IEEE Transactions on Multimedia* 12, 42–53.
- Joachims, T., Granka, L., Pan, B., Hembrooke, H., Radlinski, F., Gay, G., 2007. Evaluating the accuracy of implicit feedback from clicks and query reformulations in Web search. *ACM Transactions on Information Systems (TOIS)* 25, 7–es.
- Joachims, T., Radlinski, F., 2007. Search Engines that Learn from Implicit Feedback. *Computer* 40, 34–40.
- Kahan, J., Koivunen, M.-R., Prud'Hommeaux, E., Swick, R.R., 2002. Annotea: an open RDF infrastructure for shared Web annotations. *Computer Networks* 39, 589–608.
- Kankanhalli, M.S., Chua, T.-S., 2000. Video modeling using strata-based annotation. *IEEE Multimedia* 7, 68–74.
- Kaplan, A., Falchi, F., Allasia, W., Gallo, F., Mamou, J., Mass, Y., Miotto, R., Orio, N., Hagège, C., 2007. Deliverable D3. 1 Common Schema for Feature Extraction (Deliverable). SAPIR Project.
- Kelly, D., Teevan, J., 2003. Implicit feedback for inferring user preference: a bibliography. *SIGIR Forum* 37, 18–28.

- Kemp, C., Ramamohanarao, K., 2002. Long-Term Learning for Web Search Engines, in: Proceedings of the 6th European Conference on Principles of Data Mining and Knowledge Discovery, PKDD '02. Springer-Verlag, pp. 263–274.
- Kofler, C., Lux, M., 2009. Dynamic presentation adaptation based on user intent classification, in: Proceedings of the 17th ACM International Conference on Multimedia, MM '09. ACM, pp. 1117–1118.
- Kosch, H., 2003. Distributed Multimedia Database Technologies Supported by MPEG-7 and MPEG-21. CRC Press.
- Kosch, H., Boszormenyi, L., Doller, M., Libsie, M., Schojer, P., Kofler, A., 2005. The Life Cycle of Multimedia Metadata. *IEEE MultiMedia* 12, 80–86.
- Kotsiantis, S.B., 2007. Supervised Machine Learning: A Review of Classification Techniques, in: Proceedings of the 2007 Conference on Emerging Artificial Intelligence Applications in Computer Engineering: Real World AI Systems with Applications in eHealth, HCI, Information Retrieval and Pervasive Technologies. IOS Press, pp. 3–24.
- Laborie, S., Euzenat, J., 2008. An Incremental Framework for Adapting the Hypermedia Structure of Multimedia Documents, in: Wallace, M., Angelides, M., Mylonas, P. (Eds.), *Advances in Semantic Media Adaptation and Personalization, Studies in Computational Intelligence*. Springer Berlin / Heidelberg, pp. 157–176.
- Laborie, S., Manzat, A.-M., Sedes, F., 2009a. Managing and querying efficiently distributed semantic multimedia metadata collections. *IEEE MultiMedia Special Issue on Multimedia-Metadata and Semantic Management* 16, 12–21.
- Laborie, S., Manzat, A.-M., Sedes, F., 2009b. Création et utilisation d'un résumé de métadonnées pour interroger efficacement des collections multimédias distribuées, in: *INFORSID*. IRIT Press, pp. 227–242.
- Laborie, S., Manzat, A.-M., Sedes, F., 2010. A generic framework for the integration of heterogeneous metadata standards into a multimedia information retrieval system, in: 9th International Conference on Adaptivity, Personalization and Fusion of Heterogeneous Information, RIAO'10. Paris, France, France, pp. 80–83.
- Lagger, C., Lux, M., Marques, O., 2012. What Makes People Watch Online Videos: An Exploratory Study. *ACM Transactions on Computers in Entertainment*.
- Lagoze, C., Hunter, J., 2001. The ABC ontology and model. *Journal of Digital Information. Special Issue on Metadata* 2, 1–18.
- Lamb, J., 2001. Sharing Best Methods and Know-How for Improving Generation and Use of Metadata, in: *New Techniques and Technologies for Statistics and Exchange of Technology and Know-how*. pp. 175–194.
- Lambolez, P.Y., Queille, J.P., Chrisment, C., 1995. EXREP: A Generic Rewriting Tool for Textual Information Extraction. *Ingénierie des Systèmes d'Information* 3, 471–487.
- Lassila, O., 1998. Web metadata: a matter of semantics. *IEEE Internet Computing* 2, 30–37.
- Learning Technology Standards Committee, 2002. *IEEE Standard for Learning Object Metadata (Standard No. 1484.12.1-2002)*. IEEE.
- Lee, W., Bailer, W., Burger, T., Champin, P.-A., Evain, J.-P., Malaise, V., Michel, T., Sasaki, F., Söderberg, J., Stegmaier, F., Strassner, J., 2012. *Ontology for Media Resources 1.0 (Recommendation)*. W3C.
- Limam, L., Coquil, D., Kosch, H., Brunie, L., 2010. Extracting User Interests from Search Query Logs: A Clustering Approach, in: Proceedings of the 2010 Workshops on Database and Expert Systems Applications, DEXA '10. Presented at the 7th International Workshop on Text-based Information Retrieval (TIR '10) in conjunction with the 21st International Conference

- on Database and Expert Systems Applications (DEXA '10), IEEE Computer Society, Washington, DC, USA, pp. 5–9.
- LINDO consortium, 2010. LINDO Interface Design Documents.
- Liu, J., Dolan, P., Pedersen, E.R., 2010. Personalized news recommendation based on click behavior, in: Proceedings of the 15th International Conference on Intelligent User Interfaces, IUI '10. ACM, New York, NY, USA, pp. 31–40.
- Loboda, T.D., Brusilovsky, P., Brunstein, J., 2011. Inferring word relevance from eye-movements of readers, in: Proceedings of the 16th International Conference on Intelligent User Interfaces, IUI '11. ACM, New York, NY, USA, pp. 175–184.
- Lu, J., Callan, J., 2003. Content-based retrieval in hybrid peer-to-peer networks, in: Proceedings of the Twelfth International Conference on Information and Knowledge Management, CIKM '03. ACM, New York, NY, USA, pp. 199–206.
- Lux, M., Kofler, C., Marques, O., 2010. A classification scheme for user intentions in image search, in: CHI '10 Extended Abstracts on Human Factors in Computing Systems, CHI EA '10. ACM, New York, NY, USA, pp. 3913–3918.
- Ma, W., Manjunath, B.S., 1999. NeTra: a toolbox for navigating large image databases. *Multimedia Systems* 7, 184–198.
- Manning, C.D., Raghavan, P., Schütze, H., 2008. *Introduction to Information Retrieval*. Cambridge University Press.
- Manola, F., Miller, E., 2004. *RDF Primer (Recommendation)*. W3C.
- Manzat, A.-M., Grigoras, R., Sèdes, F., 2010. Towards a User-aware Enrichment of Multimedia Metadata (regular paper), in: Workshop on Semantic Multimedia Database Technologies (SMDT). CEUR Workshop Proceedings, Saarbrücken, Germany, pp. 30–41.
- Marcoux, Y., 1994. Les formats normalisés de documents électroniques. *Intelligence artificielle et sciences cognitives au Québec* 6, 56–65.
- Martinez, J., 1996. The design of an extensible multimedia library for an OODBMS, in: 7th International Workshop on Database and Expert Systems Applications. IEEE Computer Society, pp. 208–213.
- Martinez, J.M., 2004. MPEG-7 Overview (version 10) (Standard No. ISO/IEC JTC1/SC29/WG11N6828). ISO/IEC.
- Mauclair, J., Pinquier, J., 2004. Fusion of Descriptors for Speech/Music Classification, in: Hlawatsch, F., Matz, G., Rupp, M., Wistawel, B. (Eds.), XII European Signal Processing Conference. CRYPTAS, Vienna, Autriche, pp. 1285–1288.
- Mbarki, M., 2008. *Gestion de l'hétérogénéité documentaire: le cas d' un Entrepôt de documents multimédia* (Thèse de l'Université Paul Sabatier).
- McDonald, D.W., Javanmardi, S., Zachry, M., 2011. Finding patterns in behavioral observations by automatically labeling forms of wikiwork in Barnstars, in: Proceedings of the 7th International Symposium on Wikis and Open Collaboration, WikiSym '11. ACM, pp. 15–24.
- Memmel, M., Dengel, A., 2007. Sharing Contextualized Attention Metadata to Support Personalized Information Retrieval, in: Duval, E., Naijar, J., Wolpers, M. (Eds.), Int'l ACM/IEEE Workshop on Contextualized Attention Metadata (CEUR'07): Personalized Access to Digital Resources. pp. 19–26.
- Metadata Working Group, 2009. *Guidelines for handling image metadata*.
- Miezianko, R., Pokrajac, D., 2008. Detecting and recognizing abandoned objects in crowded environments, in: Proceedings of the 6th International Conference on Computer Vision Systems, ICVS'08. Springer-Verlag, pp. 241–250.

- MIX, 2006. Data Dictionary - Technical Metadata for Digital Still Images (Standard No. Z39.87-2006). ANSI/NISO.
- MIX, 2008. NISO Metadata for Images in XML (Standard). NISO.
- Morita, M., Shinoda, Y., 1994. Information filtering based on user behavior analysis and best match text retrieval, in: Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '94. Springer-Verlag New York, Inc., New York, NY, USA, pp. 272–281.
- Naphade, M., Smith, J.R., Tesic, J., Shih-Fu Chang, Hsu, W., Kennedy, L., Hauptmann, A., Curtis, J., 2006. Large-scale concept ontology for multimedia. *Multimedia*, IEEE DOI - 10.1109/MMUL.2006.63 13, 86–91.
- National Information Standards Organization, 2004. Understanding metadata.
- Nichols, D., 1998. Implicit Rating and Filtering, in: Proceedings of the Fifth DELOS Workshop on Filtering and Collaborative Filtering. Budapest, Hongrie, pp. 31–36.
- Notaise, J., Barda, J., Dusanter, O., 1995. Dictionnaire du multimédia, audiovisuel informatique télécommunications. AFNOR.
- Novak, D., Zezula, P., 2006. M-Chord: a scalable distributed similarity search structure, in: Proceedings of the 1st International Conference on Scalable Information Systems, InfoScale '06. ACM, p. 19.
- Oard, D.W., Kim, J., 1998. Implicit feedback for Recommender Systems, in: Proceedings of the AAAI Workshop on Recommender Systems. pp. 81–83.
- OASIS, 2003. A brief Introduction to XACML [WWW Document]. URL www.oasis-open.org/committees/download.php/2713/Brief_Introduction_to_XACML.html
- Oostendorp, A., Berlyne, D.E., 1978. Dimensions in the perception of architecture:II. Measures of exploratory behavior. *Scandinavian Journal of Psychology* 19, 83–89.
- Papadias, D., Theodoridis, Y., 1997. Spatial relations, minimum bounding rectangles, and spatial data structures. *International Journal of Geographical Information Science* 11, 111–138.
- Papapetrou, O., Michel, S., Bender, M., Weikum, G., 2005. On the Usage of Global Document Occurrences in Peer-to-Peer Information Systems, in: Meersman, R., Tari, Z. (Eds.), *On the Move to Meaningful Internet Systems 2005: CoopIS, DOA, and ODBASE*, Lecture Notes in Computer Science. Springer Berlin / Heidelberg, pp. 310–328.
- Pedersen, E.R., McDonald, D.W., 2008. Relating documents via user activity: the missing link, in: Proceedings of the 13th International Conference on Intelligent User Interfaces, IUI '08. ACM, pp. 389–392.
- Pereira, F., Vetro, A., Sikora, T., 2008. Multimedia Retrieval and Delivery: Essential Metadata Challenges and Standards. *Proceedings of the IEEE* 96, 721 –744.
- Pérez, J., Arenas, M., Gutierrez, C., 2006. Semantics and Complexity of SPARQL, in: Proceedings of the 5th International Semantic Web Conference. pp. 30–43.
- Petkovic, M., Jonker, W., 2003. *Content-Based Video Retrieval: A Database Perspective*, Multimedia Systems and Applications. Springer.
- Pietarila, P., Westermann, U., Järvinen, S., Korva, J., Lahti, J., Löthman, H., 2005. CANDELA – Storage, Analysis, and Retrieval of Video Content in Distributed Systems, in: Proceedings of the IEEE Int. Conf. on Multimedia and Expo (ICME). IEEE, pp. 1557–1560.
- Pinquier, J., André-Obrecht, R., 2006. Audio Indexing: Primary Components Retrieval - Robust Classification in Audio Documents. *Multimedia Tools and Applications* 30, 313–330.

- Pleşca, C., Charvillat, V., Grigoras, R., 2008. User-aware adaptation by subjective metadata and inferred implicit descriptors, in: Granitzer, M., Lux, M., Spaniol, M. (Eds.), *Multimedia Semantics — The Role of Metadata*. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 127–147.
- Pouillet, L., Pinon, J.-M., Calabretto, S., 1997. Semantic structuring of documents, in: *Proceedings of the Third Basque International Workshop on Information Technology, 1997. BIWIT '97*. IEEE, pp. 118–124.
- Ratnasamy, S., Francis, P., Handley, M., Karp, R., Shenker, S., 2001. A scalable content-addressable network, in: *ACM SIGCOMM '01 Proceedings of the 2001 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*. pp. 161–172.
- Rayner, K., 1998. Eye movements in reading and information processing: 20 years of research. *Psychological Bulletin* 124, 372–422.
- Rocchio, J., 1971. Relevance Feedback in Information Retrieval, in: Salton, G. (Ed.), *The SMART Retrieval System*. Prentice Hall, pp. 313–323.
- Roman, D., Keller, U., Lausen, H., de Bruijn, J., Lara, R., Stollberg, M., Polleres, A., Feier, C., Busler, C., Fensel, D., 2005. Web Service Modeling Ontology. *Applied Ontology* 1, 77–106.
- Rose, D.E., Levinson, D., 2004. Understanding user goals in web search, in: *Proceedings of the 13th International Conference on World Wide Web, WWW '04*. ACM, pp. 13–19.
- Roth, V., Peters, J., Pinsdorf, U., 2006. A distributed content-based search engine based on mobile code and web service technology. *Scalable Computing: Practice and Experience* 7, 101–117.
- Rowstron, A., Druschel, P., 2001. Pastry: Scalable, Decentralized Object Location, and Routing for Large-Scale Peer-to-Peer Systems, in: Guerraoui, R. (Ed.), *Middleware 2001*. Springer Berlin Heidelberg, pp. 329–350.
- Ruthven, I., 2003. Re-examining the potential effectiveness of interactive query expansion, in: *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval, SIGIR '03*. ACM, pp. 213–220.
- Saathoff, C., Scherp, A., 2010. Unlocking the semantics of multimedia presentations in the web with the multimedia metadata ontology, in: *Proceedings of the 19th International Conference on World Wide Web, WWW '10*. ACM, pp. 831–840.
- Salazar, F., Valero, F., 1995. Analyse Automatique de documents vidéo (Rapport de recherche No. 95-28-R). IRIT, Université Paul Sabatier, Toulouse.
- Salton, G., McGill, M., 1986. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., New York, NY, USA.
- Savage, T.M., Vogel, K.E., 2008. *An Introduction to Digital Multimedia*. Jones and Bartlett Publishers.
- Schafer, J.B., Frankowski, D., Herlocker, J., Sen, S., 2007. Collaborative filtering recommender systems, in: Brusilovsky, P., Kobsa, A., Nejdl, W. (Eds.), *The Adaptive Web*. Springer-Verlag, Berlin, Heidelberg, pp. 291–324.
- Schoeffmann, K., Hopfgartner, F., Marques, O., Boeszoermenyi, L., Jose, J.M., 2010. Video browsing interfaces and applications: a review. *SPIE Reviews* 1, 1–35.
- Sears, A., Jacko, J.A., 2007. *The Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies, and Emerging Applications (Human Factors and Ergonomics Series)*. L. Erlbaum Associates Inc., Hillsdale, NJ, USA.
- Sedes, F., 1998. Bases documentaires-Hyperbases □: Proposition d'un modèle générique et contribution à la spécification d'un langage pour l'intégration et la manipulation d'informations semi-structurées. (HDR de l'Université Paul Sabatier).

- Sellen, A.J., Harper, R.H., 2003. *The Myth of the Paperless Office*. MIT Press, Cambridge, MA, USA.
- Seo, Y.-W., Zhang, B.-T., 2000. Learning user's preferences by analyzing Web-browsing behaviors, in: *Proceedings of the Fourth International Conference on Autonomous Agents, AGENTS '00*. ACM, New York, NY, USA, pp. 381–387.
- Skobeltsyn, G., Junqueira, F., Plachouras, V., Baeza-Yates, R., 2008. ResIn: a combination of results caching and index pruning for high-performance web search engines, in: *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '08*. ACM, pp. 131–138.
- Smeulders, A.W.M., Worring, M., Santini, S., Gupta, A., Jain, R., 2000. Content-Based Image Retrieval at the End of the Early Years. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22, 1349–1380.
- Smith, J.R., Schirling, P., 2006. Metadata standards roundup. *IEEE MultiMedia* 13, 84 – 88.
- Snoek, C.G.M., Worring, M., 2005. Multimodal Video Indexing: A Review of the State-of-the-art. *Multimedia Tools and Applications* 25, 5–35.
- Snoek, C.G.M., Worring, M., 2009. Concept-Based Video Retrieval. *Foundations and Trends® in Information Retrieval* 2, 215–322.
- Soukkarieh, B., 2010. *Vers un système d'information Web fournissant des services Web sensibles au contexte* (Thèse de l'Université Paul Sabatier).
- Soukkarieh, B., Sèdes, F., 2006. Interrogation graphique pour l'aide à la formulation de requête XQuery, in: *Journées de travail interdisciplinaire autour des documents multi structurés (JournéesDMS)*. INRIA, <http://www.inria.fr>, pp. 53–57.
- Stegmaier, F., Burger, T., Doller, M., Kosch, H., 2010. Knowledge Based Multimodal Result Fusion for Distributed and Heterogeneous Multimedia Environments: Concept and Ideas, in: *LNCS. Presented at the 8th International Workshop Adaptive Multimedia Retrieval: Context, Exploration, and Fusion*, Springer Berlin Heidelberg, pp. 61–73.
- Stoica, I., Morris, R., Liben-Nowell, D., Karger, D.R., Kaashoek, M.F., Dabek, F., Balakrishnan, H., 2003. Chord: a scalable peer-to-peer lookup protocol for Internet applications. *IEEE/ACM Transactions on Networking* 11, 17 – 32.
- Suel, T., Mathur, C., Wu, J., Zhang, J., Delis, A., Kharrazi, M., Long, X., Shanmugasundaram, K., 2003. ODISSEA: A Peer-to-Peer Architecture for Scalable Web Search and Information Retrieval, in: *WebDB'03*. pp. 67–72.
- Sun, J.-T., Zeng, H.-J., Liu, H., Lu, Y., Chen, Z., 2005. CubeSVD: a novel approach to personalized Web search, in: *Proceedings of the 14th International Conference on World Wide Web, WWW '05*. ACM, pp. 382–390.
- Tanenbaum, A.S., Steen, M.V., 2001. *Distributed Systems: Principles and Paradigms*. Prentice Hall PTR, Upper Saddle River, NJ, USA.
- Tannier, X., 2006. *Extraction et recherche d'information en langage naturel dans les documents semi-structurés* (Thèse de l'École Nationale Supérieure des Mines de Saint-Etienne).
- Taylor, A.G., 2004. *The organization of information*. Libraries Unlimited, Westport, Conn.
- Tchuente, D., Jessel, N., Canut, M.-F., 2011. Accès à l'information dans les réseaux socionumériques. *Hermès, Ces réseaux numériques dits sociaux* 59, 59–64.
- Tchuente, D., Jessel, N., Péninou, A., Canut, M.-F., Sèdes, F., 2012. Visualizing the relevance of social ties in user profile modeling. *Web Intelligence and Agent Systems* 10, 261–274.
- Teevan, J., Dumais, S.T., Horvitz, E., 2010. Potential for personalization. *ACM Trans. Comput.-Hum. Interact.* 17, 4:1–4:31.

- Text Encoding Initiative, 1990. TEI P1: Guidelines for the Encoding and Interchange of Machine Readable Texts. (Standard). ACH-ALLC-ACL Text Encoding Initiative.
- Text Encoding Initiative, 2006. TEI Lite: Encoding for Interchange: an introduction to the TEI Revised for TEI P5 release (Standard).
- Text Encoding Initiative, 2011. TEI P5: Guidelines for the Encoding and Interchange of Machine Readable Texts, version 2.0 (Standard).
- Trippe, B., 2001. XMLegant answers: do XML editors matter? *Transform Magazine* 10, 27–27.
- Troncy, R., Celma, Ò., Little, S., García, R., Tsinaraki, C., 2007. MPEG-7 based Multimedia Ontologies: Interoperability Support or Interoperability Issue?, in: 1st Workshop on Multimedia Annotation and Retrieval Enabled by Shared Ontologies. Genova, Italy.
- Tzouvaras, V., Troncy, R., Pan, J.Z., 2007. Multimedia Annotation Interoperability Framework (No. W3CTR-XGR-interoperability-20070814/). W3C.
- van Zwol, R., Garcia, L., Ramirez, G., Sigurbjornsson, B., Labad, M., 2008. Video Tag Game, in: 17th International World Wide Web Conference (WWW Developer Track). ACM.
- Vazirgiannis, M., Theodoridis, Y., Sellis, T., 1998. Spatio-temporal composition and indexing for large multimedia applications. *Multimedia Systems* 6, 284–298.
- Viaud, M.-L., Thièvre, J., Goëau, H., Saulnier, A., Buisson, O., 2008. Interactive components for visual exploration of multimedia archives, in: Proceedings of the 2008 International Conference on Content-based Image and Video Retrieval (CIVR). ACM Press, pp. 609–616.
- Wang, Y., Galanis, L., de Witt, D.J., 2003. Galanx: An efficient peer-to-peer search engine system (Technical report). University of Wisconsin, Madison, USA.
- Wolpers, M., Najjar, J., Verbert, K., Duval, E., 2007. Tracking Actual Usage: the Attention Metadata Approach. *Educational Technology & Society* 10, 106–121.
- Yang, B., Mei, T., Hua, X.-S., Yang, L., Yang, S.-Q., Li, M., 2007. Online video recommendation based on multimodal fusion and relevance feedback, in: Proceedings of the 6th ACM International Conference on Image and Video Retrieval, CIVR '07. ACM, pp. 73–80.
- Zayani, C., 2008. Contribution à la définition et à la mise en oeuvre de mécanismes d'adaptation de documents semi-structurés (Thèse de l'Université Paul Sabatier).
- Zayani, C., Péninou, A., Canut, M.-F., Sèdes, F., 2006. An adaptation approach: query enrichment by user profile, in: Signal-Image Technology & Internet-Based Systems (SITIS). IEEE, pp. 24–35.
- Zeng, M.L., Chan, L.M., 2006. Metadata Interoperability and Standardization - A Study of Methodology Part II. *D-Lib Magazine* 12.
- Zhao, B.Y., Huang, L., Stribling, J., Rhea, S.C., Joseph, A.D., Kubiatoiwicz, J.D., 2004. Tapestry: A resilient global-scale overlay for service deployment. *IEEE Journal on Selected Areas in Communications* 22, 41–53.

Annexes

ANNEXE A. : TAGGED IMAGE FILE FORMAT	196
ANNEXE B. : EXCHANGEABLE IMAGE FILE FORMAT	197
ANNEXE C. : EXTENSIBLE METADATA PLATFORM (XMP).....	198
ANNEXE D. : IPTC – INFORMATION INTERCHANGE MODEL.....	199
ANNEXE E. : DIG35	200
ANNEXE F. : MIX.....	201
ANNEXE G. : METS	202
ANNEXE H. : TEI.....	203
ANNEXE I. : FRBR.....	204
ANNEXE J. : ID3.....	205
ANNEXE K. : MÉTADONNÉES QUICKTIME	206
ANNEXE L. : LES STANDARDS SMPTE.....	207
ANNEXE M. : MPEG-7.....	209
ANNEXE N. : DUBLIN CORE	212
ANNEXE O. : DICOM	213
ANNEXE P. : LSCOM.....	214
ANNEXE Q. : LOM.....	215
ANNEXE R. : NEWSML, EVENTSML, SPORTSML	216

Annexe A. : Tagged Image File Format

Le Tagged Image File Format (TIFF) est un format de fichier, proposé par Adobe Systems, et utilisé pour le stockage des images. Le format TIFF est largement utilisé par des applications qui manipulent des images, telles que les scanners et les télécopieurs. TIFF n'a pas eu de mise à jour majeure depuis 1992 (plusieurs notes techniques ont été publiées avec des extensions mineures de ce format. La dernière version du TIFF est la version 6.0 (Adobe Developers Association, 1992).

Au début, le format TIFF n'était qu'un format d'image binaire. Les avancées technologiques et la nécessité de la couleur ont amené des extensions de TIFF pour stocker des images en niveaux de gris, puis des images en couleur. En plus de l'image elle-même, un fichier TIFF contient également des métadonnées techniques sur ce contenu, sous forme d'une paire clé-valeur.

Chaque fichier TIFF commence par un indicateur sur 2 octets pour l'ordre des octets (i.e., little endian ou big endian). Les 2 octets suivants représentent toujours le nombre 42, le nombre choisi pour représenter le format TIFF. Les 4 octets suivants contiennent l'adresse du premier IFD (Image File Directory) où se trouvent les métadonnées. L'IFD contient plusieurs entrées représentées sur 12 octets, chacune correspondant à un élément de métadonnées. L'IFD commence par le nombre d'entrées et finit avec l'adresse de l'IFD suivant.

La structure d'une entrée de répertoire est fixe, elle contient le tag, le type de l'élément, sa dimension et la valeur de l'élément si elle peut être représentée sur 4 octets, sinon on retrouve dans le répertoire, l'adresse où se trouve la valeur. Le tag est représenté par un nombre hexadécimal qui correspond à une métadonnée (e.g., 256 correspond à « ImageWidth », 315 correspond à « Artist »)⁶¹. Les informations qui peuvent être stockées dans le fichier TIFF sont assez limitées. Il y a environ 30 balises qui sont définies dans la norme de base, et encore 60 dans l'extension. Mais toutes ces métadonnées sont de bas niveau. Ces balises peuvent être réparties en 4 catégories :

- tags liés à *la structure de l'image* : la largeur, la hauteur, l'orientation ;
- tags liés à *l'emplacement des données* : la localisation, le nombre de lignes par bande, le nombre d'octets par bande comprimée ;
- tags liés aux *caractéristiques de l'image* : la fonction de transfert, la palette de couleurs ;
- *D'autres balises* : le titre, la date et l'heure de la création du fichier, le modèle d'appareil photo, le logiciel utilisé, la personne qui a créé l'image.

Des extensions à la norme ont permis d'intégrer, autant que possible, les nouvelles technologies dans le domaine du traitement, du stockage et de la création d'image. Parmi ces extensions nous pouvons citer : l'archivage des documents (numéro de page, nom de la page, positions X et Y), la compression LZW, la compression JPEG.

Au début, cette norme a été élaborée en tant que fichier utilisé par les scanners. Mais maintenant, ce dernier n'est plus employé tout seul. En revanche, les balises qu'il définit, sont utilisées dans d'autres normes.

TIFF est l'un des premiers formats de fichier qui essaie de rassembler dans le même fichier les données et les métadonnées correspondantes.

⁶¹ Une liste complète des tags peut être trouvée sur le site <http://www.awaresystems.be/imaging/tiff/tifftags.html>

Annexe B. : Exchangeable image file format

Exif (Exchangeable image file format) est un standard qui spécifie le format des images, du son et des métadonnées associées à ces images et qui sont créées par des appareils photos numériques. Ce standard a été créé par Japan Electronic Industries Development Association (JEIDA), et sa dernière version, 2.3, a été mise au point en 2010 (Camera & Image Products Association, Japan Electronics and Information Technology Industries Association, 2010). De nos jours, tout appareil photo numérique peut stocker dans le même fichier, le contenu de l'image et les métadonnées Exif associées à ce contenu, enregistrées en temps réel au moment où l'image est prise.

La spécification des métadonnées Exif réutilise des éléments de métadonnées déjà définis dans d'autres formats et standards différents : JPEG – pour des images compressées –, TIFF – pour les images non compressées –, WAV et RIFF – pour les fichiers audio. En plus de ces métadonnées, le standard contient des éléments spécifiques.

Comme pour le format TIFF, les métadonnées Exif sont sauvegardées dans le fichier de l'image sous la forme d'une association clé-valeur. Les éléments de ce format représentent des métadonnées de bas niveau, en contenant surtout des informations sur les paramètres de l'appareil photo au moment de la prise de vue (e.g., exposure time, F number, Aperture, Flash). La liste complète de tous les tags utilisés par la norme, leur nom, leur type, leur définition mais aussi les valeurs que chacun peut prendre peuvent être retrouvés dans (Camera & Image Products Association, Japan Electronics and Information Technology Industries Association, 2010). En ce qui concerne la structure du fichier, Exif combine les structures des normes JPEG et TIFF.

Les métadonnées définies dans la norme peuvent être classées en 5 catégories :

- *Date et heure* : les appareils photo enregistrent automatiquement la date courante, en tant que métadonnées ;
- *Paramètres de l'appareil* : incluent des informations statiques – qui ne varient pas d'une image du même appareil à l'autre –, telles que le modèle et la version de l'appareil, et des informations qui varient d'une image à l'autre, comme orientation, aperture, shutter speed, focal length, metering mode ;
- *Vignette et métadonnées associées à cette vignette* ;
- *Descriptions et informations administratives* ;
- *Localisation GPS* ;
- *Éléments réservés à chaque fabricant d'appareil photo.*

Parmi les éléments définis dans la norme Exif certains sont obligatoires, d'autres ne le sont pas. Par exemple, les coordonnées GPS ne sont pas obligatoires parce qu'il n'y a qu'un nombre limité d'appareils photo équipés d'un capteur GPS.

Annexe C. : Extensible Metadata Platform (XMP)

Extensible Metadata Platform (XMP) est une spécification proposée par Adobe Systems Incorporated, qui se focalise sur la création, le traitement et l'échange des métadonnées associées à différentes ressources. La spécification de ce standard a été commencée dans les années 2000, la dernière version datant de Juillet 2010 (Adobe Systems Incorporated, 2010a, 2010b, 2010c).

La norme est divisée en trois parties, qui présentent :

- *Un modèle de données*, qui décrit de manière simple et flexible les métadonnées associées aux documents (Adobe Systems Incorporated, 2010a) ;
- *Un modèle de stockage*, qui est une implémentation du modèle des données (Adobe Systems Incorporated, 2010c). La norme propose l'utilisation de structures spécifiques – nommées Paquets XMP – pour l'intégration des métadonnées dans le même fichier que le contenu du document. Pour ce stockage, la norme privilégie l'utilisation des flux RDF et XML.
- *Des schémas de métadonnées*, qui contiennent un ensemble de définitions de propriétés pertinentes pour une large variété d'applications fournies par Adobe mais également par d'autres industriels (Adobe Systems Incorporated, 2010a, 2010b).

La norme réutilise certaines métadonnées définies dans des standards tels que Dublin Core, Exif et TIFF. En plus de ces éléments, XMP définit ses propres éléments, groupés en fonction de leur rôle, sous différents namespaces : (1) *XMP* : éléments de description basique, (2) *XMP Rights Management* : éléments spécifiant les droits d'utilisation, (3) *XMP Media Management* : information liée à l'identification, la composition et l'historique de la ressource, (4) *XMP Basic Job Ticket* : la description du workflow du document, (5) *XMP Paged-text* : information sur le texte, (6) *XMP Dynamic Media* : informations sur les contenus audio et vidéos comprises dans le fichier, (7) *Adobe PDF*, (8) *Photoshop*, (9) *Camera Raw* : information sur les images brutes, prises par les appareils photo numériques.

La norme est conçue pour être extensible, permettant aux fournisseurs d'ajouter d'autres éléments de métadonnées personnalisés.

Le XMP peuvent être intégré dans un nombre important de formats de fichier, tels que JPEG, GIF, PNG, TIFF, PDF, PSD, sans interférer avec la structure du fichier. Pour les fichiers qui ne sont pas du tout compatibles avec le format XMP, les métadonnées sont stockées dans un fichier séparé avec l'extension XMP.

Annexe D. : IPTC – Information Interchange Model

L'**Information Interchange Model (IIM)** est un standard proposé par l'International Press Telecommunications Council (IPTC) et le Newspaper Association of America (NAA) dans le début des années 1990 pour améliorer l'échange de l'information (International Press Telecommunications Council, 1999). Ce standard inclut *la définition d'une structure pour l'intégration des métadonnées* dans le même fichier que le contenu auquel elles sont associées, et *un ensemble d'éléments de métadonnées*. Les métadonnées définies par cette norme peuvent être appliqués à des fichiers texte, des images et d'autres types de média. Les métadonnées IPTC sont utilisées davantage pour décrire les images. Leur popularité a grandi au moment où Adobe a choisi d'intégrer une partie des éléments définis par ce standard dans les entêtes des fichiers obtenus par son logiciel Photoshop, mais également dans la spécification du XMP. Suite à cette utilisation sur grande échelle, le standard IPTC IIM a connu une amélioration en 2010, quand « IPTC Core specifications » et « IPTC Extension Schema » ont été créées (International Press Telecommunications Council, 2010).

L'élément principal de ce standard est le modèle d'intégration des métadonnées. Ce modèle est constitué de 9 sous-ensembles qui regroupent les informations relatives à un certain aspect de l'image :

- 1 – Object Envelope Record : informations relatives à l'expédition de l'objet (e.g., File Format, Envelope Number, Date Sent)
- 2 – Application Record : description documentaire sur l'image (e.g., Editor, Keywords, Abstract, Category, By-line, City),
- 3 – Digital Newsphoto Parameter Record : caractéristiques techniques de l'image (e.g., Pixels per line, Number of lines, Color representation)
- 4, 5 – Libres pour une évolution éventuelle de la norme ;
- 6 – Méthode de description utilisée pour le traitement du document ;
- 7 – Pre-ObjectData Descriptor Record ;
- 9 – ObjectData Record ;
- 10 –Post-ObjectData Descriptor Record.

Parmi toutes les métadonnées du standard seulement 33 sont vraiment utilisées

Numéro	Nom du champ	Description
5	Object Name	Nom de l'objet
7	Edit Status	Statut éditorial
10	Urgency	Priorité
15	Category	Catégorie
20	Supplemental Category	Catégorie supplémentaire
22	Fixture Identifier	Identificateur
25	Keywords	Mots-clés
30	Release Date	Date de disponibilité
35	Release Time	Heure de disponibilité
40	Special Instructions	Instructions spéciales
45	Reference service	Service de référence (suivi les champs 47, 50)
47	Reference Date	Date de référence
50	Reference Number	Numéro de référence
55	Date Created	Date de création
60	Time Created	Heure de création
65	Originating Program	Programme ayant créé l'objet
70	Program version	Version du programme ayant créé l'objet

75	Object cycle	Cycle de l'objet
80	By-line	Créateur de l'objet
85	By-line Title	Titre du créateur
90	90 City	Ville
95	Province/State	Province/État
100	Country/Primary Location Code	Code du pays
101	Country/Primary Location Name	Libellé du pays
103	Original Transmission Reference	Référence de la transmission (code)
105	Headline	Titre
110	Credit	Le fournisseur de l'objet
115	Source	Propriétaire intellectuel
116	Copyright Notice	Copyright
118	Contact	Contact
120	Caption/Abstract	Description, Résumé
122	Writer/Editor	Auteur de la Description
130	Image Type	Type de l'image

Annexe E. : DIG35

DIG35 est un groupe d'initiative créé en 1999 par le Digital Imaging Group et certains industriels du domaine, parmi lesquels nous citons Adobe Systems Inc., Canon Inc., Eastman Kodak Company, FujiPhoto Film Corporation, Ltd., Hewlett Packard Company. L'objectif principal de cette initiative est de définir un ensemble de métadonnées standard pour les images numériques, qui permettent d'améliorer l'interopérabilité entre les appareils, les services et les logiciels qui les manipulent. Une première version du standard a été proposée en 2000 (Digital Imaging Group, 2000) et la version 1.1 a été finalisée en juin 2001 (Digital Imaging Group, 2001).

Le groupe d'initiative a réalisé une implémentation XML du standard et a défini également un modèle d'encapsulation des métadonnées dans les fichiers JPEG et TIFF, illustrée dans la Figure 53.

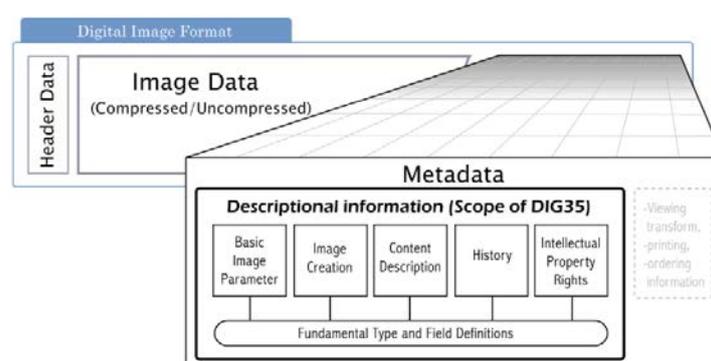


Figure 53 – La structure de fichier proposée par DIG35

Les métadonnées définies par DIG35 sont regroupées en 5 blocs en fonction de leur caractéristique principale. En plus de ces éléments, le standard contient un bloc supplémentaire, définissant les types de données utilisés. Les métadonnées sont regroupées dans les blocs suivants :

- **Basic Image Parameter** : contient des informations génériques sur l'image, telles qu'*Image size, Format, Compression*. Cette information doit être utilisée pour le décryptage de l'image seulement si le fichier ne contient pas un entête spécifique.
- **Image Creation** : définit les métadonnées liées aux outils qui ont capturé l'image, tels que les appareils photos ou les scanners. Ce sont des métadonnées techniques, qui ne sont pas modifiables. Dans ce bloc, on retrouve des métadonnées telles que *Creation_time, Image_Creator, Manufacturer, Lens information, Exposure_time, etc.*
- **Content Description** : contient de l'information descriptive sur le contenu de l'image en répondant aux questions : qui ?, quoi ?, quand ? et où ?. Ce bloc contient des métadonnées sous forme de phrases qui décrivent l'événement ou l'endroit photographié. Parmi les éléments de ce bloc on retrouve : *Caption, Location, Person description, Thing description, Participant, Role, Property, etc.*
- **History** : fournit de l'information sur les différentes modifications de l'image pour arriver dans l'état actuel. Par exemple, l'historique de l'image peut contenir les étapes suivantes : la création, la rotation de 90°, et le changement de taille.
- **Intellectual Property Rights** : décrit les différents droits de l'auteur ou fournit de l'information sur les conditions d'utilisation de l'image.

Annexe F. : MIX

Le MIX – connu sous le nom de "*NISO Metadata for Images in XML (NISO MIX)*" – représente un schéma de métadonnées qui contient les informations techniques nécessaires pour une gestion efficace des collections d'images. Ce schéma se base sur le standard proposé par ANSI/NISO en décembre 2006, *Data Dictionary - Technical Metadata for Digital Still Images (ANSI/NISO Z39.87-2006)* (MIX, 2006). La dernière version de ce schéma est 2.0, et a été publiée en Mai 2008 (MIX, 2008). Il est développé et maintenu par la librairie du congrès américain en collaboration avec le comité NISO.

Les métadonnées MIX ne s'appliquent que pour les images matricielles qui ne sont pas animées ou en mouvement. Les éléments de ce standard sont proches de celles de l'EXIF, en présentant également une description fine des caractéristiques techniques de l'image. Une différence majeure entre ces deux standards est que les métadonnées MIX sont stockées dans un fichier externe.

Les métadonnées définies par MIX sont regroupées en 5 blocs. En plus de ces éléments présents dans le standard NISO Z39.87-2006, la deuxième version du schéma MIX a été enrichie avec un bloc supplémentaire qui n'est pas présent dans le standard pour inclure de possibles extensions. Les métadonnées sont regroupées dans les blocs suivants :

- **Basic Digital Object Information** : contient des informations génériques qui peuvent être aussi appliquées pour d'autres media que les images. Ce bloc contient des informations sur *la dimension* de l'image, *le format*, la méthode de *compression*.
- **Basic Image Information** : contient des métadonnées qui peuvent être utilisées pour l'affichage de l'image. Dans ce bloc, on retrouve des métadonnées telles que *la largeur et la hauteur* de l'image (en pixels), *le modèle de couleur* utilisé pour l'image (e.g., RGB, YCbCr), *des informations supplémentaire pour certains formats* (i.e., JPEG 2000⁶², MrSID⁶³, DjVu⁶⁴).
- **Image Capture Metadata** : définit les métadonnées liées aux outils qui ont capturé l'image, tels que les appareils photos ou les scanners. Dans ce bloc on retrouve des métadonnées telles que *la date de création*, *l'auteur*, *le producteur* et *le modèle de l'appareil photo*, *le temps de pose*, *la distance focale*, etc.
- **Image Assessment Metadata** : regroupe les métadonnées représentatives pour la qualité de l'image. Parmi ces métadonnées on retrouve : *l'encodage de la couleur*, *le WhitePoint*, *PrimaryChromaticities*, etc.
- **Change History** : fournit de l'information sur les différentes modifications souffertes par l'image pour arriver dans l'état actuel. Par exemple, l'historique de l'image peut contenir les étapes suivantes : la création, la rotation de 90°, et le changement de taille.

⁶² <http://www.jpeg.org/jpeg2000/>

⁶³ http://www.lizardtech.com/files/geo/datasheets/MG4_White_Paper.pdf

⁶⁴ <http://djvu.org/>

Annexe G. : METS

Le standard METS – Metadata Encoding and Transmission Standard – représente une spécification pour l’encodage et la transmission des données, en exprimant les métadonnées nécessaires pour la gestion des objets numériques et leur partage entre les différents systèmes et institutions. Ce standard a été créé en 2001 à l’initiative du DLF (Digital Library Federation) et de la Librairie du Congrès Américain. Depuis 2004, le METS est devenu une norme NISO. La version actuelle du standard est la 1.6 (Digital Library Federation, 2010). Le standard est représenté sous forme de fichier XML. La dernière version du schéma XSD du standard est la 1.9.⁶⁵

METS se décompose en 7 sections, chacune de ces sections comprend différents types de métadonnées :

- **L’entête METS** donne des informations sur le fichier METS lui-même : e.g., la date de création du fichier, le créateur du fichier ;
- **Les métadonnées descriptives** comprennent l’information bibliographique qui décrit le document entier ou des sous parties d’un document. Ces métadonnées peuvent être incluses dans le fichier METS ou elles peuvent se trouver dans un fichier externe. Le standard ne définit pas des métadonnées descriptives, mais il encourage l’utilisation d’autres standards, tels que MARC⁶⁶, MODS⁶⁷, DC ;
- **Les Métadonnées administratives**, enferment de l’information sur la source dont le document numérique est issu, de l’information technique sur le document ou les différentes parties du document, des informations sur l’historique du document numérique, des informations sur les droits d’auteur. Ces métadonnées peuvent être incluses dans le fichier METS ou elles peuvent se trouver dans un fichier externe ;
- **Section des fichiers** fait l’inventaire de l’ensemble des fichiers qui composent le document. Il exprime les caractéristiques techniques de base de chacun de ces fichiers : taille, empreinte, emplacement du fichier ;
- **La carte de structure** est le cœur du document METS, elle dessine une structure hiérarchique pour l’objet de la bibliothèque numérique, et relie chaque élément de cette structure aux fichiers de contenu et aux métadonnées qui s’y rapportent. Plusieurs types de carte peuvent être utilisés à des niveaux de granularité différents. Il y a deux types de cartes de structure : physique, et logique. Une carte de structure physique exprime une navigation linéaire dans le document – e.g., navigation page à page pour un imprimé. Une carte de structure logique permet une navigation non linéaire (e.g., table des matières permettant de naviguer dans un ouvrage numérisé) ;
- **Lien structurels** permet d’exprimer des liens existant entre différents éléments du document, e.g., des hyperliens entre les pages pour les sites Web ;
- **Comportement** peut être utilisé pour associer des comportements exécutables au contenu d’un objet METS.

⁶⁵ <http://www.loc.gov/standards/mets/mets-schemadocs.html>

⁶⁶ <http://www.loc.gov/marc/marcdocz.html>

⁶⁷ <http://www.loc.gov/standards/mods/>

Annexe H. : TEI

Le TEI (*Text Encoding Initiative*) présente une modalité d'encodage pour les documents textuels. Il a été mis au point suite à une collaboration dans le sein du Consortium TEI⁶⁸, composé d'académiques des sciences humaines et sociales, et des linguistes. Ce consortium a préféré présenter son travail sous forme de Guidelines, règles et recommandations, parce que les éléments importants dans un document textuel peuvent être différents d'une personne à l'autre et de cette façon tout le monde est libre d'encoder les informations qu'il juge pertinentes. Le TEI a choisi une représentation XML pour ses recommandations. La première version du standard, TEI P1, est apparue en 1990 (Text Encoding Initiative, 1990). Au fil du temps le TEI a été maintenu, et il est arrivé à la version 2.0 du P5 (Text Encoding Initiative, 2011).

Le TEI définit 503 éléments et 210 attributs qui représentent différents concepts lexicaux et prosodiques sur le texte, mais aussi des métadonnées descriptives et structurelles. Ces éléments sont regroupés dans 21 modules : (1) infrastructure, (2) entête, (3) éléments pertinents pour tous les documents TEI, (4) structure par défaut du texte, (5) représentation des caractères non-standards, (6) poésie, (7) performance des textes, (8) transcriptions de la parole, (9) dictionnaires, (10) description du manuscrit, (11) représentation des sources primaires, (12) équipement critique, (13) noms, dates, personnes, et endroits, (14) tableaux, formules, et graphiques, (15) langue du corpus, (16) liens, segmentation, et alignement, (17) mécanismes analytiques, (18) propriétés des structures, (19) graphes, réseaux, et arbres, (20) certitude et responsabilité, (21) éléments de documentation. Pour simplifier l'utilisation du TEI, le consortium a proposé en 2006 une version réduite du standard, TEI LITE (Text Encoding Initiative, 2006).

Un document TEI est constitué par une balise `<teiHeader>`, qui contient toutes les métadonnées décrivant le document et une balise `<text>`, qui contient le document lui-même. L'entête contient des métadonnées bibliographiques, qui sont regroupées en 4 éléments :

- la *description du fichier* : on retrouve des éléments tels que le titre, l'auteur, les informations sur les éditeurs, la publication et la distribution du document, la dimension du fichier ;
- la *description de l'encodage*, qui offre de l'information sur la relation entre le document électronique et sa source textuelle, e.g., les corrections réalisées, la normalisation, l'interprétation, ou encore de l'information sur la classification du document, en fonction d'une certaine taxonomie.
- la *description du profil* : contient des informations non-bibliographiques sur le texte électronique, e.g., la date et le lieu de création, les mots-clés.
- la *description des révisions* : donne l'historique des modifications du document électronique, avec la date de leur réalisation et le nom de la personne qui a effectué le changement.

⁶⁸ <http://www.tei-c.org>

Annexe I : FRBR

Le FRBR (*Functional Requirements for Bibliographic Records*) contient des recommandations pour la modélisation des informations contenues dans les notices bibliographiques. Elles ont été développées par un groupe d'experts de l'International Federation of Library Associations and Institutions (IFLA) dans les années 90. La dernière version a été publiée en 1997 (International Federation of Library Associations and Institutions, 1997).

Le FRBR est conçu comme un modèle entité-relation. Une vue d'ensemble du modèle proposé par le FRBR est présentée dans la Figure 54. Dans ce modèle, on retrouve trois groupes d'entités et une série de liens entre ces entités, qui dans ce modèle portent le nom de relations :

1. Des entités associées aux produits qui sont nommés ou décrits dans les notices bibliographiques : *work*, *expression*, *manifestation*, et *item*.
2. Des entités responsables de la production et de la diffusion du contenu décrit par les notices bibliographiques : *person* et *corporate body*.
3. Des entités supplémentaires qui sont décrites dans le contenu : *concept*, *object*, *event*, et *place*.
4. Des relations entre les entités qui facilitent la navigation et la compréhension du modèle : *is realised through*, *is embodied in*, *is exemplified in*, *is created by*, *is realised by*, *is produced by*, *is owned by*, *has as subject*.
5. Une série de relations qui font le lien entre des entités du même type, par exemple entre deux « manifestations », il peut y avoir une relation « is a reproduction of », ou entre deux *works* « is a successor of ».

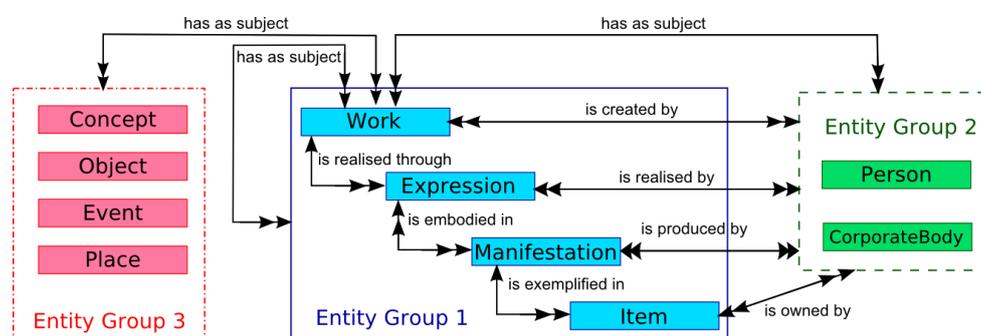


Figure 54 – Le modèle entité-relation du FRBR

La réalisation du FRBR s'est appuyée sur une étude des différents besoins qu'un utilisateur peut avoir au moment où il consulte le catalogue d'une bibliothèque. Une série d'attributs a été associée à chaque entité : (1) des attributs propres à l'entité qui peuvent être déterminés en analysant son contenu (e.g., titre, auteur, date de publication, la dimension, le résumé) et (2) des attributs externes qui font référence à des sources (e.g., le contexte, le public ciblé, l'origine de l'item, l'historique).

Les métadonnées FRBR sont les plus utilisées dans le domaine des bibliothèques pour la gestion informatique des œuvres littéraires et des différentes ressources qui peuvent leur être attachées. Le standard a été adopté par la Bibliothèque Nationale de France, qui l'a traduit en français et qui l'utilise dans la gestion de son catalogue bibliographique.

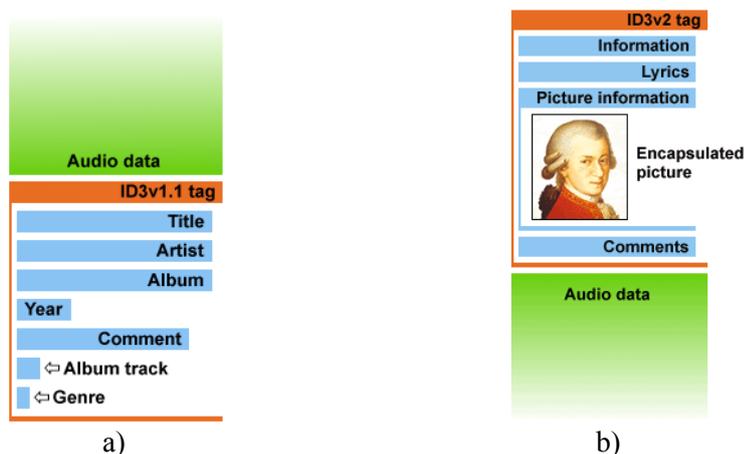
Annexe J. : ID3

L'ID3 est un acronyme utilisé pour représenter les métadonnées qui sont encapsulées dans les fichiers audio MP3. L'ID3 ne contient pas que les métadonnées mais il spécifie aussi l'endroit et la manière de stockage de ces métadonnées dans le fichier audio, appelé l'ID3 TAG.

Une première version de cette spécification a été proposée en 1996 par Eric Kemp, qui l'a appelée ID3v1. Au fil des années, cette version a été améliorée, en passant à la version 2 (ID3v2). La dernière version de ce standard est ID3v2.4, qui a apparue en 2000.

La première version de l'ID3 est assez simple. Elle a une taille fixe, de 128 octets, située à la fin du fichier audio. Elle est aussi très rigide avec les informations qui peuvent être sauvegardées dans les champs de métadonnées définies. Elle ne contient que 6 éléments (le titre de la chanson, l'interprète, le nom de l'album, l'année d'apparition, un commentaire et le genre) qui ont une taille fixe (e.g., le nom de l'interprète est sur 30 caractères). Dans la version 1.1, le numéro de piste a été introduit. Une autre limitation de cette version est le fait que les chaînes de caractères sont codées en ISO 8859-1, et donc seuls les caractères de l'alphabet latin peuvent être utilisés.

Pour surpasser ces limites et favoriser le streaming des fichiers audio, la version 2 est complètement différente de la version 1. Une comparaison des deux versions est présentée dans la Figure 55. L'ID3v2 se situe au début du fichier et il a une dimension variable qui peut aller jusqu'à 256MB. En ce qui concerne le codage des caractères, celui-ci est passé à l'Unicode. L'ID3v2 est structuré sous forme de frames qui peuvent avoir une dimension allant jusqu'à 16MB. Le standard contient une série de frames prédéfinies, telles que celle pour les paroles de la chanson, l'auteur, le compositeur, une image (e.g., la pochette de l'album).



a) b)
Figure 55 – La structure de : a) ID3v1 et b) ID3v2

Annexe K : Métadonnées QuickTime

Le QuickTime est un framework multimédia défini en 1989 par Apple pour le stockage des contenus audio-visuels. Depuis 1992, QuickTime est compatible Apple et Windows. Ce framework propose un format de fichier propre (.mov), un logiciel de visualisation et une suite de développements.

Un fichier QuickTime est composé de deux parties : une description du contenu audio-visuel et le contenu lui-même. Cette description peut être sauvegardée dans le même fichier que celui du contenu ou dans un fichier différent, ou même dans plusieurs fichiers. En interne, un fichier QuickTime est stocké sous forme d'un arbre d'atomes. Chaque atome utilise un identifiant de quatre octets pour déterminer sa structure. Un atome peut être le parent d'autres atomes ou peut contenir des données (mais il ne peut être les deux à la fois).

La structure du fichier QuickTime est présentée dans la Figure 56. La structure proposée par QuickTime a été reprise par la norme MPEG-4.

Dans cette structure hiérarchique d'atomes il y a certaines règles : toutes les métadonnées se trouvent dans l'atome identifié par l'identifiant MOOV, cet atome ne contient pas de contenu multimédia, il ne fait que référence à l'endroit où celui-là se trouve. Les valeurs des métadonnées sont enregistrées en utilisant une paire clé-valeur. Pour cela un ensemble extensible de clés a été proposé par Apple, parmi lesquels nous citons :

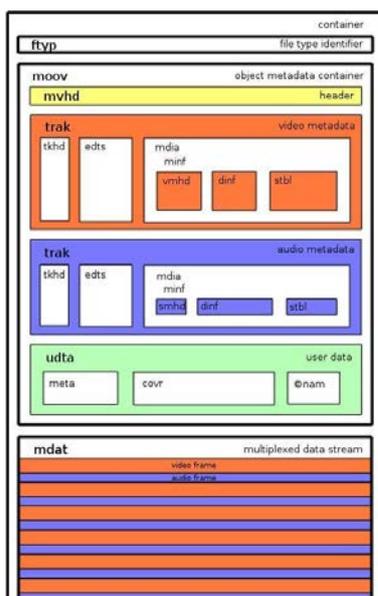


Figure 56 – La structure du fichier QuickTime

- Album : la collection dont la vidéo fait partie ;
- Artist : le nom du créateur du contenu ;
- Artwork : une image représentative du contenu ;
- Author : le nom de l'auteur du contenu ;
- Comment : un commentaire sur le contenu ;
- Copyright : les droits d'auteur sur le contenu ;
- CreationDate : la date de création du contenu ;
- Description : la description du contenu
- Title : le nom du film
- Genre : un texte libre décrivant le genre du film.
- Information : de l'information supplémentaire sur la vidéo
- Keywords : une liste de mots-clés ;
- Localisation : les coordonnées GPS de l'endroit où la vidéo a été créée. Pour affiner la localisation du contenu, l'utilisation d'éléments supplémentaires est conseillée tels que : le nom de la localisation, une description, la date ;
- Year : l'année de création du contenu original ;
- Collection.user : le nom de la collection où l'utilisateur a classé le contenu ;
- Rating.user : un numéro entre 0.0 et 5.0, en représentant la note qu'un utilisateur a donné au film.

Annexe L : Les Standards SMPTE

Tous les standards dans le domaine de l'industrie du film sont créés par le SMPTE, the Society of Motion Picture and Television Engineers. Cette société a été créée en 1916 avec l'objectif de développer des recommandations dans le domaine émergent du film et de la télévision. De nos jours le SMPTE est reconnu comme le leader dans le développement de standards pour le film, la télévision, le vidéo et le multimédia en général. Ils ont défini environ 400 standards, qui adressent des sujets variés, tels que la transmission du signal audio digital à travers les câbles coaxiaux, la synchronisation du son et de la vidéo, les paramètres de la camera ou encore un format d'échange de fichier, et des métadonnées.

Le MXF (Material eXchange Format) est un format de fichier libre, qui a pour but de faciliter l'échange de données audio-visuelles et de leurs métadonnées associées. Il a été conçu pour améliorer l'interopérabilité entre les différents acteurs qui interviennent dans le processus de création des contenus audio-visuels. Il offre une modalité de transport, dans le même fichier, des données et des métadonnées tout en étant indépendant de la méthode de compression utilisée pour sauvegarder les données audio-visuelles, et en permettant l'interopérabilité entre les équipements de différents fournisseurs et entre les applications. Le transport des métadonnées dans le même fichier que les données permet une meilleure gestion du contenu et une meilleure traçabilité des modifications effectuées sur le contenu tout au long de son cycle de vie. C'est un format qui est indépendant du système d'exploitation et qui peut être utilisé pour un transfert normal du contenu mais également pour le streaming.

Le format MXF se focalise surtout sur le transport et l'encapsulation des informations et pas sur la modélisation des métadonnées. Les métadonnées sont intégrées dans l'entête du fichier. Elles sont sauvegardées sur la forme d'un triplet KLV (key-length-value). Elles peuvent être attachées à tout le document, à un flux (e.g., le flux vidéo), ou à un segment du flux. Les fichiers MXF peuvent inclure des métadonnées statiques (qui sont figées dans le temps), ou des métadonnées dynamiques, qui changent d'un contexte à l'autre (e.g., les données GPS, les paramètres de la caméra, le mouvement d'un objet dans la vidéo).

Le SMPTE a défini un dictionnaire de métadonnées propres (SMPTE 335M-2001) qui contient des éléments décrivant le contenu audio-visuel et qui sont regroupés dans les catégories suivantes : (1) Identification : l'identification et la localisation du contenu ; (2) Administration ; (3) Interprétation : la description du contenu ; (4) Paramétrage : la configuration des équipements utilisés pour l'enregistrement du contenu ; (5) Processus : informations liées au processus de création et de traitement du contenu ; (6) Relations : entre les données ; (7) Métadonnées spatio-temporelles ; (8) Métadonnées réservées aux organisations : à usage public ou privé ; (9) Métadonnées libres : qui peuvent être créées par tout autre utilisateur, en respectant la structure du standard. Le nombre de groupes de métadonnées n'est pas figé. Le standard peut encore être enrichi, mais il y a une limite de 127 groupes qui ne peut pas être dépassée. A leur tour, les différents groupes sont encore divisés en sous-groupes, comme illustré dans le **Erreur ! Référence non valide pour un signet.**

Le dictionnaire des métadonnées a été publié sous la forme d'un tableur⁶⁹. Pour pouvoir être utilisé plus facilement le SMPTE a travaillé sur la représentation de ce dictionnaire en XML et aussi à l'intégration du XML dans le format MXF. Des travaux ont aussi pour objectif de rapprocher les métadonnées SMPTE de celles des autres schémas de description.

⁶⁹ <http://www.smp-te-ra.org/mdd/index.html>

Tableau 53 – Exemple de sous-groupes de métadonnées dans le standard SMPTE 335M-2001

Groupes de SMPTE RP210	Sous-groupes
Groupe 1: Identification et location	Globally unique identifiers
	ISO identifiers
	Device identifiers
	Local locators
Groupe 2: Administration	Supplier
	Rights
	Security
	Broadcast and repeat statistics
Groupe 3: Interprétation	Fundamental
	Descriptive (human-assigned)
	Descriptors (machine-assigned or computed)
Groupe 4: Paramétrage	Video / Audio / Data essence encoding characteristics
	Metadata encoding characteristics
	MPEG coding characteristics
	Time code characteristics
Groupe 6: Relations	Generic relationships
	Essence-to-essence / Metadata-to-essence / Metadata-to-metadata / Object-to-object / Metadata-to-object relationship
	Relationship structures
Groupe 7: Spatio-temporels	Position and space vectors
	Rate and direction of positional change
	Operational date and time information
	Absolute / Rights date and time
	Delay

Annexe M. : MPEG-7

Le **MPEG-7** est un standard ISO/IEC développé par le groupe MPEG (Moving Picture Experts Group). Aussi appelé *Multimédia Content Description Interface*, le MPEG-7 fournit une représentation standard des contenus multimédia. Il combine des métadonnées sur le contenu et des métadonnées techniques sur le fichier ainsi que des métadonnées liées à l'utilisateur. La principale ambition de MPEG -7 est « de rendre les informations multimédias aussi faciles à trouver sur le Web que le texte l'est aujourd'hui » (Martinez, 2004). Son principal objectif est de fournir l'interopérabilité entre les systèmes et applications utilisés dans la production, la gestion, la distribution et la consommation du contenu audio-visuel.

Cette norme est destinée à spécifier une description standard de divers types d'informations multimédia, telles que les images, les graphiques, les modèles 3D, les contenus audio, les discours, les vidéos, ou les présentations multimédias (e.g., une combinaison de tous ces éléments). Les descriptions MPEG-7 peuvent être associées aux contenus numériques, mais également aux contenus analogiques ou imprimés. Le standard est décomposé en 10 parties :

1. **Le système MPEG-7** : contient la description d'outils pour le transport et le stockage efficace des descripteurs.
2. **Le Langage de Définition des Descripteurs**.
3. **Le MPEG-7 Visuel** : présente les descripteurs qui ne décrivent que des caractéristiques visuelles du contenu.
4. **Le MPEG-7 Audio** : présente les descripteurs qui ne décrivent que des caractéristiques audio du contenu.
5. **Les schémas de description multimédia** : contient des descriptions génériques qui sont associées aux contenus multimédia.
6. **Le logiciel de référence MPEG-7** : contient une implémentation des parties importantes du standard, qui ont un caractère normatif.
7. **Les tests de conformité** : présentent les recommandations et les procédures pour tester la conformité des implémentations MPEG-7.
8. **L'extraction et l'utilisation des descripteurs MPEG-7** : ne contient que des recommandations sous la forme d'un Rapport Technique.
9. **Les profils et les niveaux MPEG-7**.
10. **La définition du schéma MPEG-7**.

Les principaux éléments de la norme sont :

- **Les Descripteurs (D)**, qui définissent la syntaxe et la sémantique des caractéristiques du contenu multimédia ;
- **Les Schémas de Description (SD)**, qui spécifient la structure et la sémantique des relations entre les composants du contenu, que ce soit des Ds ou des SDs ;
- **Le langage de Définition des Descripteurs (LDD)** permet la création de nouveaux Ds et SDs. Il permet également l'extension et la modification des SDs existants ;
- **Les outils système**, qui permettent la gestion et la transmission efficaces des descripteurs et des contenus multimédia, la synchronisation des descripteurs et des contenus, l'accès au contenu tout en respectant les droits intellectuels.

Les descripteurs MPEG-7 peuvent être utilisés pour décrire les contenus audio-visuels à plusieurs niveaux de granularité, selon le besoin et le domaine d'application. Un même contenu peut être décrit différemment en fonction de son contexte et de son domaine d'utilisation. Des descripteurs peuvent se trouver à côté du contenu qu'ils décrivent, ou dans le même flux que ce contenu.

Les Ds ont été conçus pour décrire des caractéristiques audio-visuelles de bas-niveau telles que la couleur, la texture, la forme, le mouvement, la fréquence, les paramètres spectraux, mais aussi

des caractéristiques générales telles que la localisation, le temps, la qualité. Les SDs proposent des structures plus complexes qui décrivent des caractéristiques de haut-niveau, telles que les régions, les segments ou encore les objets présents dans le contenu. Les SDs sont utilisés aussi pour modéliser des métadonnées liées à la création, la production et l'utilisation du contenu. Les SDs sont divisées en 5 classes. Leur organisation est présentée dans la Figure 57.

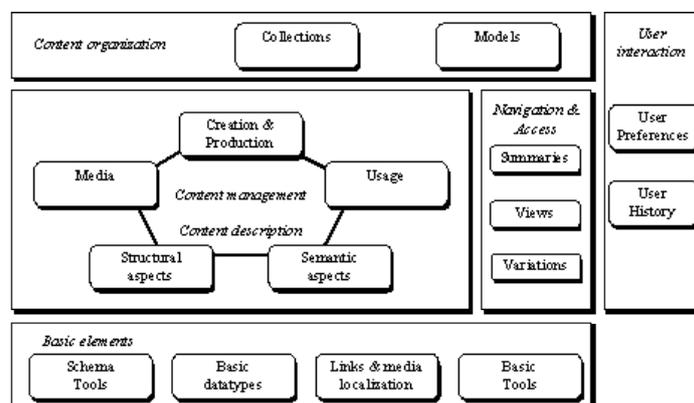


Figure 57 – L'organisation des SDs (Martinez, 2004)

Les **éléments de base** sont composés : de schémas, qui aident à la création et l'assemblage des descriptions MPEG-7 ; de type de données de base, qui étendent les types déjà existants et qui fournissent des structures mathématiques supplémentaires pour la représentation des données (e.g., les vecteurs, les matrices) ; des éléments qui réalisent les liens entre les contenus, et qui décrivent le temps, les endroits, les personnes, les groupes.

La **gestion du contenu** est réalisée à travers : (1) des métadonnées sur la création et la production du contenu, qui fournissent des informations sur la création du document (e.g., le titre, les créateurs, la date et le lieu de création), sur la classification du document (e.g., le sujet, le genre, la langue). (2) des informations sur le codage, le stockage et le format de fichier utilisé : une différence est faite entre le media original et les différentes copies qui peuvent exister (e.g., ayant des formats d'encodage différents, des qualités différentes). (3) les informations sur l'usage du document, notamment les droits d'auteur, des informations financières.

La **description du contenu** est composée des SDs de structure et des SDs sémantiques. Les descriptions de structure décrivent le contenu en termes de segments audio ou vidéo, de régions fixes ou mobiles. Cette structure est créée autour de la notion de segment, qu'il soit spatial, temporel ou spatio-temporel. Chaque segment peut être décrit en utilisant les éléments de base ou d'autres SDs. Les descripteurs sémantiques apportent des détails sur les objets, les événements et d'autres concepts de la vie réelle, qui sont surprises dans le contenu audio-visuel.

L'**organisation du contenu** fournit des descripteurs pour la modélisation et la classification des collections de contenus et de leurs descriptions afférentes.

La **navigation et l'accès** sont réalisés à travers des descripteurs qui fournissent des résumés du contenu, sous forme visuelle ou textuelle ; des morceaux et des décompositions ; et des variations du contenu, en décrivant des versions du contenu différentes en résolution, méthode de compression, langue, mais aussi type de contenu.

Les **interactions de l'utilisateur** sont représentées par deux schémas : User Preference SD et User Interaction SD. En termes de préférences le MPEG-7 ne prend en compte que celles liées au filtrage du contenu, à la recherche et à la navigation. Le User Interaction SD décrit l'historique des actions (e.g., enregistrement, pause, play) que l'utilisateur a réalisées en consommant le contenu, et le moment où l'action a été réalisée.

Le MPEG-7 fournit un langage XML standardisé pour la description des métadonnées multimédia. Pour la spécification des schémas, le langage XSD est employé. L'utilisation du XML offre la possibilité d'avoir un modèle riche et structuré, tout en profitant de la flexibilité et de l'extensibilité du langage. Pour faciliter la transmission des descripteurs dans les flux de streaming, le MPEG-7 fournit un format Binaire (BiM) qui permet la compression et le streaming de tout document XML.

La norme MPEG-7 ne spécifie pas la façon d'extraire les descriptions du contenu multimédia, pour pouvoir laisser le maximum de souplesse aux différentes applications. La manière dans laquelle les descriptions MPEG-7 devraient être utilisées pour la recherche et le filtrage du contenu n'est pas non plus normalisée, afin de laisser un maximum de flexibilité aux applications. Toutefois, comme exigence normative, la représentation de ces descriptions doit être conforme à la norme MPEG-7.

Le standard offre un très grand nombre d'éléments et de types. En total le MPEG-7 fournit : 1182 éléments, 417 attributs et 377 types complexes. Cette grande variété d'éléments rend la compréhension et l'utilisation du standard très difficiles. En outre, la norme offre plusieurs possibilités pour décrire la même chose. Cela rend le choix du bon descripteur à utiliser difficile. Cette diversité a une grande influence sur l'interopérabilité des systèmes qui utilisent le MPEG-7 comme modèle de représentation des métadonnées multimédia.

Les domaines d'application des descripteurs MPEG-7 sont très variés, tels que le broadcast, les services culturels, les bibliothèques digitales, le commerce électronique et même l'éducation, la surveillance et le domaine médical.

Annexe N. : Dublin Core

Dublin Core (DC) est un standard de métadonnées défini par ISO en 2003, sous le numéro ISO Standard 15836, et un standard ANSI/NISO, numéro Z39.85-2007 (ANSI/NISO, 2007). C'est une norme de métadonnées simple, qui a un nombre limité d'éléments, mais qui peut décrire n'importe quel type de ressource informatique ou matérielle.

Deux niveaux du standard ont été définis : le *DC Simple*, qui est composé de quinze éléments – qui sont détaillés dans le Tableau 54 –, et le *DC quantifié*, qui inclut trois éléments de plus (Audience, Provenance et RightsHolder) et des éléments de raffinement qui ajoutent de la sémantique aux éléments de base. Plus de détails sur l'utilisation de chacun de ces éléments se trouvent dans (ANSI/NISO, 2007). Aucun élément de standard n'est pas obligatoire. En outre, chaque élément peut être employé plusieurs fois dans l'instance des métadonnées associées à une ressource.

Les principaux objectifs de la norme sont :

- Une simplicité de la création et la maintenance des métadonnées, en définissant un nombre réduit d'éléments simples ;
- Une sémantique couramment comprise, en utilisant des termes qui peuvent être compris et utilisés par tout le monde ;
- Une portée internationale, à travers des versions en plusieurs langues, telles que l'anglais, le français, l'allemand, l'espagnol, le japonais, le grec.

Tableau 54 - Description des éléments du standard Dublin Core

Nom	Description
<i>Contributor</i>	L'entité (e.g., la personne, l'organisation, le service) qui a contribué au contenu de la ressource.
<i>Coverage</i>	L'étendue ou la portée spatiale, temporelle ou administrative de la ressource.
<i>Creator</i>	L'entité qui est essentiellement responsable de la création de la ressource.
<i>Date</i>	Une date associée à un certain événement dans le cycle de vie de la ressource (e.g., la création ou le disponibilité).
<i>Description</i>	Une description sommaire du contenu de la ressource, sous forme d'un texte libre, ou une table de matières.
<i>Format</i>	Une représentation physique ou digitale de la ressource, qui peut inclure le type du media ou la dimension celle-ci.
<i>Identifier</i>	Une référence claire et unique de la ressource (e.g., l'URI, l'URL, le DOI, et l'ISBN).
<i>Language</i>	Le langage de base de la ressource.
<i>Publisher</i>	L'entité responsable de la publication de la ressource.
<i>Relation</i>	Une référence à une autre ressource liée à celle-ci.
<i>Rights</i>	Informations sur les différents droits détenus sur la ressource (copyright, property rights).
<i>Source</i>	Une référence à une autre ressource qui a servi comme base pour la ressource entière ou pour seulement une partie de la ressource.
<i>Subject</i>	Le thème de la ressource, sous forme de mots-clés, ou de phrases clé.
<i>Title</i>	Le nom de la ressource.
<i>Type</i>	Le type, la nature du contenu.

Annexe O. : DICOM

Le DICOM (Digital Imaging and COmmunications in Medicine), créé en 1985 par l'ACR (American College of Radiology) et la NEMA (National Electric Manufacturers Association), est un standard pour la gestion des données issues de l'imagerie médicale. Il a pour but de standardiser les données transmises entre les différents appareils de radiologie. Ce standard définit un format de fichier mais aussi un protocole de transmission des données.

Le standard est découpé en 18 parties :

1. Introduction and Overview
2. Conformance
3. Information Object Definitions
4. Service Class Specifications
5. Data Structures and Encoding
6. Data Dictionary
7. Message Exchange
8. Network Communication Support for Message Exchange
9. Media Storage and File Format for Data Interchange
10. Media Storage Application Profiles
11. Media Formats and Physical Media for Data Interchange
12. Grayscale Standard Display Function
13. Security Profiles
14. Content Mapping Resource
15. Explanatory Information
16. Web Access to DICOM Persistent Objects (WADO)
17. Application Hosting
18. Transformation of DICOM to and from HL7 Standard

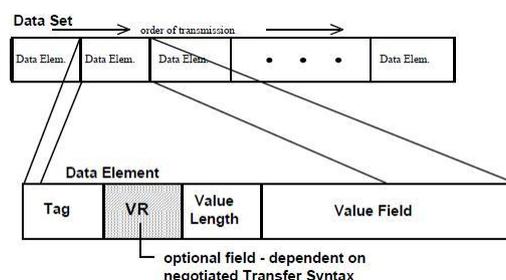


Figure 58 – Le modèle de stockage utilisé par DICOM

Comme format de stockage des données (présenté dans la Figure 58), le DICOM a choisi l'utilisation d'une suite d'éléments formés par une étiquette (tag), le type de la valeur (VR), la taille de la valeur, et la valeur elle-même. Dans ce modèle, les pixels de l'image sont une simple valeur d'un tel élément. Comme type de valeurs, le DICOM supporte une trentaine de types, parmi lesquels on cite : String, Date, Person Name, Integer, String. Les étiquettes sont composées de deux parties, chacune encodée sur 2 octets, représentée en hexa : numéro de groupe, et numéro de l'élément.

Les différents groupes d'éléments définis par DICOM dans la partie 6 sont regroupés en fonction du type d'information qu'ils offrent. Par exemple, le groupe 10 offre des informations sur le patient, son nom (tag (0010,0010)), sa date de naissance, son âge, son adresse, les allergies, son historique médical. D'autres groupes d'éléments sont dédiés pour les caractéristiques de l'appareil de radiologie et aussi pour des informations techniques sur l'image. Tous les éléments définis par le standard se trouvent dans sa Partie 6.

Annexe P. : LSCOM

Le LSCOM (Large-Scale Concept Ontology for Multimedia) est une ontologie créée entre 2004 et 2006 par un groupe de recherche composé entre autres par : Columbia University, IBM, Carnegie Mellon University mais aussi d'autres groupes de recherche publics et industriels. Le principal but de LSCOM était de créer une taxonomie qui ne contienne que des concepts qui respectent les critères d'utilité, de couverture, de faisabilité et d'observabilité.

Cette taxonomie a été créée en se basant sur le domaine du broadcast en étudiant tous les concepts (environ 10.000) qui sont décrits dans d'autres modèles de métadonnées et qui respectent les quatre critères mentionnés. Une principale condition de sélection pour chaque concept a été sa possible détection automatique. A la fin de ce travail colossal, le LSCOM est resté à 834 concepts. Ces concepts sont groupés en six catégories : objets, activités/événements, scènes/endroits, personnes, graphiques et types de programme.

Parmi ses concepts, les experts en ont choisi 39 pour représenter la version Lite de la taxonomie (Figure 59). Pour ces concepts le LSCOM fournit aussi des outils d'extraction.

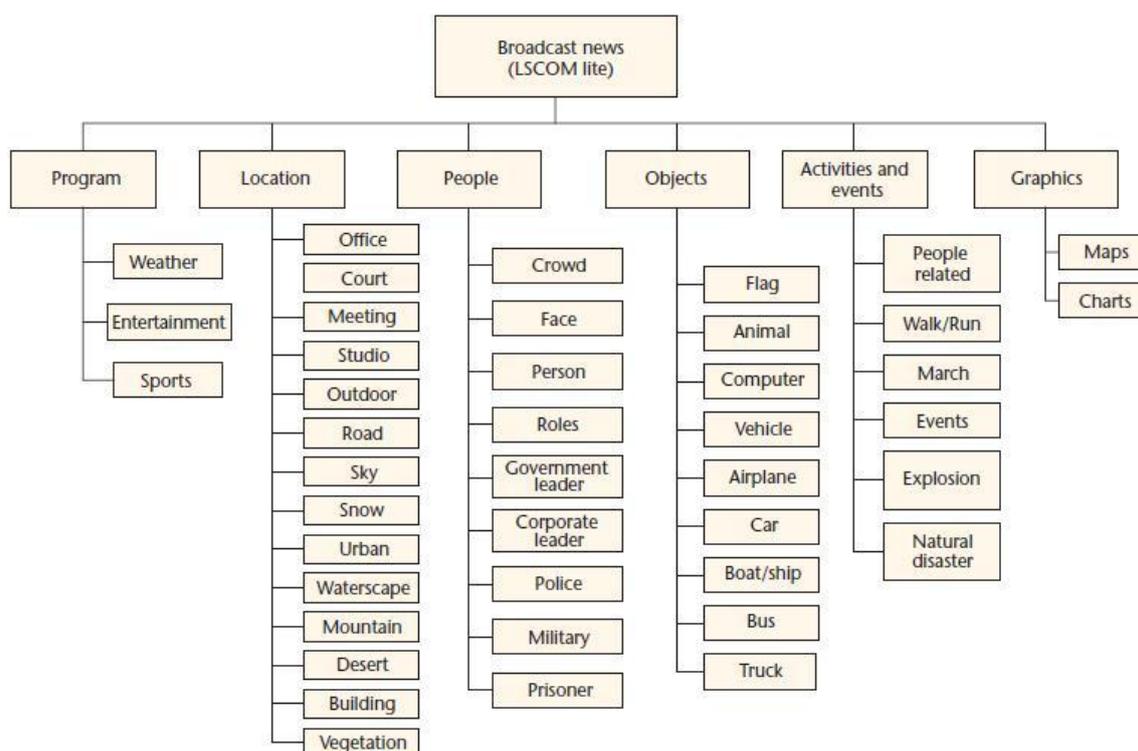


Figure 59 - Les concepts du LSCOM Lite (Naphade et al., 2006)

Une partie des concepts du LSCOM – 449 concepts – a été utilisé pour l'annotation manuelle d'environ 80 heures de video dans le cadre du TRECVID 2006.

Annexe Q. : LOM

Le LOM (Learning Object Metadata) est un standard publié en 2002 par le Learning Technology Standards Committee (LTSC) de l'IEEE. Son objectif est de proposer un modèle pour la description de n'importe quelle ressource pédagogique (i.e., toute entité – numérique ou pas – qui peut être utilisée pour l'apprentissage, l'éducation) (Learning Technology Standards Committee, 2002). Le standard est composé de quatre parties :

- IEEE 1484.12.1 – Modèle conceptuel de métadonnées ;
- IEEE 1484.12.2 – Implémentation de la norme ISO/IEC 11404 ;
- IEEE 1484.12.3 – Définition et application du schéma XML pour le LOM ;
- IEEE 1484.12.4 – Définition du cadre d'application RDF pour le LOM.

Le modèle défini par ce standard est composé de 68 éléments de métadonnées, mis dans une forme hiérarchique, qui sont regroupés dans 9 groupes (présentés dans la Figure 60) :

- **Général** – regroupe les éléments d'identification et de description d'une ressource, indépendants du contexte d'utilisation (e.g., titre, mots-clés, description).
- **Cycle de vie** – décrit l'histoire de la ressource et les contributeurs.
- **Meta-métadonnées** – donnent des informations sur les métadonnées.
- **Technique** – présente les caractéristiques techniques de la ressource (e.g., taille, format,)
- **Pédagogique** – offre une description pédagogique de la ressource (e.g., le contexte d'utilisation (scolaire, secondaire, supérieur), le type de public cible (enseignant, auteur, apprenant, gestionnaire), la difficulté, la langue des apprenants)
 - **Droits** – présentent les conditions légales d'utilisation de la ressource.
 - **Relation** – offre des informations qui lient deux ressources entre elles.
 - **Annotation** – offre la possibilité d'ajouter des informations complémentaires sur l'usage de la ressource pédagogique.
 - **Classification** – permet d'assigner des classifications à une ressource.

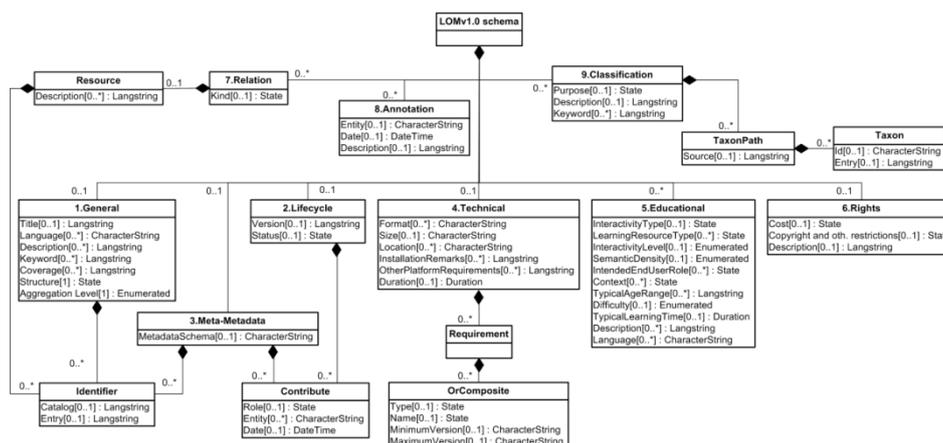


Figure 60 – Le schema UML des metadonnees LOM⁷⁰

⁷⁰ http://fr.wikipedia.org/wiki/Fichier:LOM_base_schema.png

Annexe R. : NewsML, EventsML, SportsML

Dans le début des années 2000, l'utilisation incontestable du web pour l'échange des nouvelles a déterminé l'IPTC à développer la News Architecture for G2-Standards (NAR). Cette architecture a pour but la description de tout contenu multimédia qui est utilisé dans le domaine des news, indépendamment de leur type. Le NAR est la base de tous les nouveaux standards développés par l'IPTC, tel que NewsML-G2 – pour les news –, EventsML-G2 – pour les événements –, SportsML-G2 – pour des résultats sportifs –, ProgramGuideML-G2 et WeatherML. Toutes ces spécifications se focalisent sur la description des contenus, en définissant des métadonnées spécifiques au domaine, et pas sur leur affichage. En plus de la description du contenu, des standards offrent la possibilité de décrire aussi leur gestion, leur assemblage, leur échange et les droits d'utilisation. Ces standards sont basés sur XML.

Dans cette architecture, chaque item est composé de plusieurs éléments : métadonnées générales (itemMeta) – utilisées pour la gestion –, métadonnées qui décrivent le contenu (contentMeta), le contenu ou une référence vers une source externe, une description technique du contenu. Comme éléments de métadonnées nous retrouvons : le producteur, la date de création, la date de la dernière modification, les mots-clés, le sujet, la description, la localisation.

Dans cette suite de standards, une grande importance est accordée aux droits d'utilisation de chaque partie du contenu. Chaque newsItem est accompagné par des informations sur le propriétaire du droit d'auteur, la période où le document peut être utilisé.

Les valeurs qui peuvent être renseignées dans les différents éléments de métadonnées (définis dans le NewsML par exemple) sont limitées à un certain vocabulaire spécifié par l'IPTC dans IPTC NewsCodes⁷¹. Comme genre de news nous avons, parmi d'autres : actualité, anniversaire, matériel d'archive, exclusivité.

Dans la majorité des cas, les news sont composés de texte. Mais de plus en plus on retrouve des contenus audio-visuels dans les news. Pour une meilleure gestion de ces contenus, le NewsML propose la segmentation de ceux-là en scènes. En ce qui concerne les métadonnées techniques associées à chaque contenu multimédia retrouvé dans les news, le NewsML ne s'intéresse qu'aux informations comme la taille, le code à utiliser pour le décodage du contenu, le frame rate.

Les autres standards suivent la même structure que celle du NewsML et ils utilisent les mêmes éléments de métadonnées. La différence majeure entre tous ces standards est le vocabulaire utilisé pour le remplissage de ces éléments.

⁷¹ <http://www.iptc.org/site/NewsCodes/>

Annexe P. : Exemple de métadonnées extraites pendant les expérimentations

```
<document ID="U0" src="naturaMundi/htmlFiles/rhodiola-120-gelules.html" avgTemperature="0.0">
  <generalInformation ID="U1" avgTemperature="0.0">
    <description avgTemperature="0.0" ID="U4">
      <Temperature userGroupID="GR1">0.0</Temperature>
      <value>Rhodiola 120 gélules : Le 1er Spécialiste en France de la Rhodiola rosea.</value>
    </description>
    <language avgTemperature="0.0" ID="U6">
      <Temperature userGroupID="GR1">0.0</Temperature>
      <value>French</value>
    </language>
    <Temperature userGroupID="GR1">0.0</Temperature>
  </generalInformation>
  <image ID="U31" avgTemperature="0.0">
    <generalInformation ID="U32" avgTemperature="0.0">
      <filename avgTemperature="0.0" ID="U33">
        <Temperature userGroupID="GR1">0.0</Temperature>
        <value>http://www.naturamundi.com/media/catalog/product/cache/1/thumbnail/50x/9df78eab33525d08d6e5fb8d27136e95/5/1/5124B_2.jpg</value>
      </filename>
      <description avgTemperature="0.0" ID="U34">
        <Temperature userGroupID="GR1">0.0</Temperature>
        <value>Crème de jour bio 50 ml</value>
      </description>
      <Temperature userGroupID="GR1">0.0</Temperature>
    </generalInformation>
    <EXIF ID="U530" avgTemperature="0.0">
      <TiffIfd ID="U531" avgTemperature="0.0">
        <Temperature userGroupID="GR1">0.0</Temperature>
      </TiffIfd> <Temperature userGroupID="GR1">0.0</Temperature>
    </EXIF> <Temperature userGroupID="GR1">0.0</Temperature>
  </image>
  <text ID="U70" avgTemperature="0.0">
    <TextUnit ID="U71" definitionRef="&lt;html lang=&quot;fr " avgTemperature="0.0">
      <TextUnit ID="U73" definitionRef="&lt;body" avgTemperature="0.0">
        <TextUnit ID="U140" definitionRef="&lt;div class=&quot;wrapper " avgTemperature="0.0">
          <TextUnit ID="U141" definitionRef="&lt;div class=&quot;middle-container " avgTemperature="0.0">
            <TextUnit ID="U142" definitionRef="&lt;div class=&quot;middle col " avgTemperature="0.0">
              <TextUnit ID="U143" definitionRef="&lt;div id=&quot;main" avgTemperature="0.0">
                <TextUnit ID="U161" definitionRef="&lt;div class=&quot;product-info " avgTemperature="0.0">
                  <TextUnit ID="U162" definitionRef="&lt;div class=&quot;product-essential" avgTempera-
                    ture="0.0">
                    <TextUnit ID="U173" definitionRef="&lt;div class=&quot;product-shop" avgTempera-
                    ture="0.0">
                    <TextUnit ID="U183" definitionRef="&lt;div class=&quot;product-description" avgTempera-
                    ture="0.0">
                    <TextUnit ID="U201" definitionRef="&lt;ul>&lt;li>&lt;a " avgTemperature="0.0">
                      <TextUnit ID="U202" definitionRef="&lt;li>&lt;a " avgTemperature="0.0">
                        <TextUnit ID="U203" definitionRef="&lt;a " avgTemperature="0.0">
                          <refs>U54</refs>
                          <TextUnit ID="U204" definitionRef="Rhodiola: la plante du 21ème siècle !" avgTem-
                            perature="0.0">
                            <Temperature userGroupID="GR1">0.0</Temperature>
                          </TextUnit>
                        </TextUnit>
                      </TextUnit>
                    </TextUnit>
                  </TextUnit>
                </TextUnit>
              </TextUnit>
            </TextUnit>
          </TextUnit>
        </TextUnit>
      </TextUnit>
    </TextUnit>
  </text>
</document>
```


Annexe S. : La liste des publications

Articles de revues internationales :

- Mihaela Brut, Dana Codreanu, Ana-Maria Manzat, Florence Sèdes. *Distributed multimedia indexing and optimal resources utilization: an implementation based on metadata, context and usage*. Dans : *The Journal of Multimedia Processing and Technologies*, Digital Information Research Foundation (DIRF), Numéro spécial *Intelligent Multimedia Processing*, 2(4), pp.197-225, 2011
- Sébastien Laborie, Ana-Maria Manzat, Florence Sèdes. *Managing and querying efficiently distributed semantic multimedia metadata collections*. Dans : *IEEE MultiMedia*, IEEE Computer Society, Los Alamitos - USA, Numéro spécial *Special issue on Multimedia-Metadata and Semantic Management*, Vol. 16 N. 4, p. 12-20, décembre 2009.

Conférences internationales :

- Dana Codreanu, Ana-Maria Manzat, Florence Sèdes. *Optimal resources utilization for indexing within a distributed multimedia retrieval system: an implementation for a video surveillance use case (regular paper)*. Dans : *International Conference on Distributed Multimedia Systems (DMS)*, Eden Roc Renaissance Miami Beach, USA, Knowledge Systems Institute, p. 107-112, août 2012
- Mihaela Brut, Dana Codreanu, Ana-Maria Manzat, Florence Sèdes. *Adapting Indexation to the Content, Context and Queries Characteristics in Distributed Multimedia Systems (regular paper)*. Dans : *International Conference on SIGNAL-IMAGE TECHNOLOGY & INTERNET-BASED SYSTEMS (SITIS 2011)*, Dijon, France, IEEE Computer Society, p. 118-125, novembre 2011.
- Mihaela Brut, Sébastien Laborie, Ana-Maria Manzat, Florence Sèdes. *Generic Information System Architecture for Distributed Multimedia Indexation and Management (regular paper)*. Dans : *East-European Conference on Advances in Databases and Information Systems (ADBIS 2011)*, Vienna, Austria, Springer-Verlag, p. 347-360, septembre 2011.
- Mihaela Brut, Dana Codreanu, Stefan Daniel Dumitrescu, Ana-Maria Manzat, Florence Sèdes. *A Distributed Architecture for Flexible Multimedia Management and Retrieval (regular paper)*. Dans : *International Conference on Database and Expert Systems Applications (DEXA 2011)*, Toulouse, Springer-Verlag, p. 249-263, août 2011.
- Sébastien Laborie, Ana-Maria Manzat, Florence Sèdes. *A Generic Framework for the Integration of Heterogeneous Metadata Standards into a Multimedia Information Retrieval System (short paper)*. Dans : *International Conference on Adaptivity, Personalization and Fusion of Heterogeneous Information (RIAO 2010)*, Université Paris VI, Paris, France, Centre de hautes études internationales d'Informatique Documentaire (C.I.D.), (support électronique), avril 2010.
- Mihaela Brut, Sébastien Laborie, Ana-Maria Manzat, Florence Sèdes. *Integrating Heterogeneous Metadata into a Distributed Multimedia Information System (regular paper)*. Dans : *Cognitive systems with Interactive Sensors (COGIS 2009)*, Paris (France), Société de l'Électricité, de l'Électronique et des Technologies de l'Information et de la Communication (SEE), (support électronique), novembre 2009.
- Mihaela Brut, Sébastien Laborie, Ana-Maria Manzat, Florence Sèdes. *A Generic Metadata Framework for the Indexation and the Management of Distributed Multimedia Contents (regular paper)*. Dans : *International Conference on New Technologies, Mobility and Security (NTMS 2009)*, Cairo (Egypt), IEEE Computer Society, p. 383-387, décembre 2009.

- Mihaela Brut, Sébastien Laborie, Ana-Maria Manzat, Florence Sèdes. *A Framework for Automating and Optimizing the Selection of Indexing Algorithms*. Dans : *International Conference on Metadata and Semantics Research (MTSR 2009)*, Milan, Springer-Verlag, CCIS 46, p. 48-59, septembre 2009.
- Mihaela Brut, Florence Sèdes, Ana-Maria Manzat. *A Web Services Orchestration Solution for Semantic Multimedia Indexing and Retrieval*. Dans : *International Conference on Complex, Intelligent and Software Intensive Systems, Fukuoka, Japon*, IEEE Computer Society, p. 1187-1192, mars 2009.

Workshops internationaux :

- Dana Al Kukhun, Dana Codreanu, Ana-Maria Manzat, Florence Sèdes. *Applying Pervasive and Flexible Access Control to Distributed Multimedia Retrieval (regular paper)*. Dans : *International Workshop on Information Management for Mobile Applications in conjunction with VLDB 2012*, Istanbul, Turquie, CEUR Workshop Proceedings, p. 41-48, août 2012
- Ana-Maria Manzat, Romulus Grigoras, Florence Sèdes. *Towards a User-aware Enrichment of Multimedia Metadata (regular paper)*. Dans : *Workshop on Semantic Multimedia Database Technologies (SMDT 2010)*, Saarbrücken, Germany, Vol. 680, CEUR Workshop Proceedings, p. 30-41, décembre 2010.

Conférences nationales :

- Sébastien Laborie, Ana-Maria Manzat, Florence Sèdes. *Création et utilisation d'un résumé de métadonnées pour interroger efficacement des collections multimédias distribuées*. Dans : *Congrès Informatique des Organisations et Systèmes d'Information et de Décision (INFORSID 2009)*, Toulouse, IRIT Press, p. 227-242, mai 2009.