

Adaptation d'une approche de classification et de gestion de documents multimédias hétérogènes

III.1. Introduction

La gestion des documents multimédia a été pour un bon moment sujet sur le quelle s'articule beaucoup de travaux, parmi ces travaux l'approche proposée dans [1]. Cette approche se base sur la modélisation et l'intégration des documents multimédia hétérogènes dans le but d'en faciliter leur exploitation.

Ce qui concerne la modélisation, l'originalité du modèle de représentation des documents multimédia proposé dans [1] réside dans la dichotomie entre structure logique et structure des métadonnées, la dichotomie entre structure générique et structure spécifique. Ce modèle offre une certaine souplesse lors de l'exploitation mais nous avons remarqué qu'il pose un vrai problème en matière de droits d'utilisation, par exemple l'auteur d'un document ne peut pas tolérer ou ne pas tolérer la mise à jour de la description de son document par un autre utilisateur.

En termes de gestion, la classification et l'exploitation sont assez performantes mais il est possible de leurs ajoutées d'autres fonctionnalités ou d'autres modes d'utilisation pour qu'ils donnent des meilleurs résultats.

Notre travail d'adaptation consiste a réalisé des améliorations sur son modèle de représentation des documents multimédia pour permettre la gestion des droits d'utilisation, pour la démarche de classification nous avons vue qu'elle est suffisamment performante et nous l'avons pris tels quelle est, ce qui concerne l'exploitation nous avons ajouté une démarche pour rechercher les documents dans l'entrepôt de document selon différents critères (selon les modèles de description).

Dans ce troisième chapitre, nous présentons nos travaux d'adaptation accompagnée d'exemples dans les trois phases de modélisation, classification et exploitation.


III.2. Modélisation des documents multimédia

Le modèle de représentation offre une séparation claire des informations relatives au contenu de celles relatives à la structure, aussi il regroupe les documents similaires qui sont représentés par des structures spécifiques au sein d'une même classe illustrée par une structure générique. Ce qui tolère la gestion des deux niveaux d'hétérogénéité documentaire: l'hétérogénéité intra-document, et l'hétérogénéité inter-documents.

Pour l'instanciation des modèles nous utiliserons les documents "Présentation_Martyr" et "Présentation_Trajan's_Arch" :

Martyrs' Memorial, Algeria

The **Maqam Echahid** (*Arabic*: مقام الشهيد, *Arabic pronunciation*: [mækæːm elchæːhiːd], English: **Martyrs' Memorial**) is a concrete monument commemorating the [Algerian war for independence](#). The monument was opened in 1982 on the 20th anniversary of Algeria's independence. It is fashioned in the shape of three standing palm leaves which shelter the "Eternal Flame" beneath. At the edge of each palm leaf stands a statue of a soldier, each representing a stage of Algeria's struggle.



Consisting of three stylized fins that join mid-height, the concrete monument built by the Canadian company [Lavalin](#), based on a model produced in the [Fine Art Institute of Algiers](#), under the leadership of [Bashir Yelles](#), reaches a height of 92 metres (302 ft). Above the three supporting fins, at 14 metres (47 ft) from the ground, is an Islamic style turret with a diameter of 10 metres (33 ft) and a height of 7.6 metres (25 ft), topped by a dome of 6 metres (20 ft). It rests on an esplanade that burns an "eternal flame" and includes a crypt, an amphitheater and the [National Museum of El Mujahid](#) (underground).



Figure III.1. Le document "Présentation_Martyr"

Arch of Trajan, Timgad



The **Arch of Trajan** is a [Roman triumphal arch](#) located in the city of [Timgad](#) (ancient [Thamugadi](#)) near [Batna, Algeria](#). It was built between the later 2nd century and the early 3rd century.

The three [vaulted arch](#) composed the western gate of the city, at the beginning of the [Decumanus Maximus](#) and the end of the road coming from [Lambaesis](#).

The arch together with the whole archaeological site of Timgad, has been listed as a [World Heritage Site](#) by [UNESCO](#) since 1982. The arch reaches a height of 12 metres, with a central arch of 6 metres in height which permitted the passage of vehicles that have left deep ruts in the ground under the archway. The lateral arches, each 3.75 metres high, were reserved for pedestrians.

Figure III.2. Le document "Présentation_Trajan's_Arch"

III.2.1. Gestion de l'hétérogénéité intra-document

L'hétérogénéité intra-document est due à la diversité des éléments multimédia qui peuvent coexister dans le même document (textes, sons, images, vidéos), ces éléments ont des caractéristiques spécifiques et très diverses d'un média à un autre.

C'est la séparation des informations relatives à la structure de celles relatives au contenu qui permet la gestion de l'hétérogénéité intra-document. Dans ce qui suit ces deux concepts seront utilisés : (1) le concept de structure logique pour décrire la composition globale d'un document multimédia, (2) le concept de structure des métadonnées pour pouvoir associer à la structure logique une ou plusieurs descriptions du contenu des différents éléments multimédia.

▪ Structure logique d'un document

La structure logique d'un document après son adaptation est représentée comme suit :

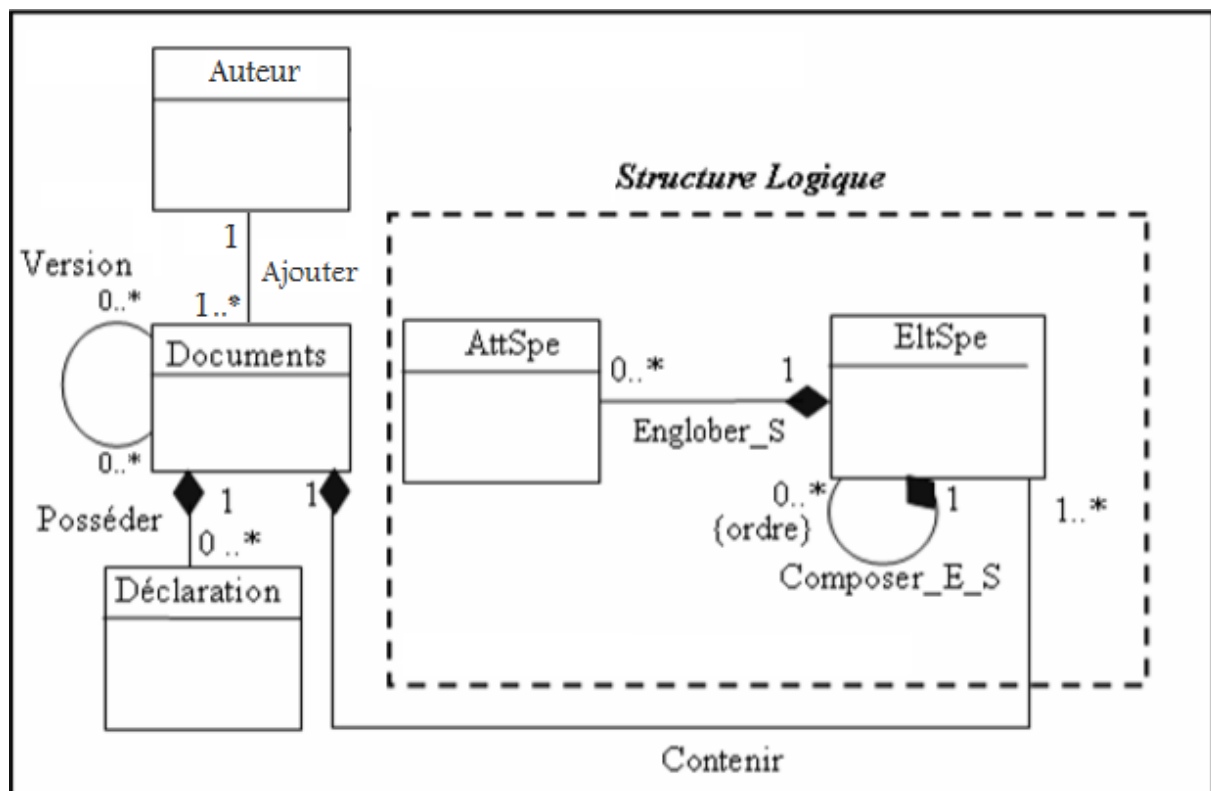


Figure III.3. Structure logique d'un document

La classe "EltSpe" représente un ensemble d'éléments logiques spécifiques, chaque élément spécifique représente un granule documentaire, qui peut être lui-même décomposé en sous granules.

La classe "AttSpe" représente les attributs spécifiques qui permettent de décrire des éléments spécifiques.

La classe "**Documents**" désigne un document particulier.

La classe "**Déclaration**" permet de garder l'information concernant d'autres caractéristiques des documents comme le "numéro de version".

La classe "**Auteur**" représente l'auteur d'un document particulier (ici l'auteur n'est pas celui qui a créé le document mais celui qui la ajouter dans la base de données).

▪ **Structure des métadonnées**

La structure des métadonnées représente le contenu des éléments, et peut servir également pour décrire le contenu d'un composant complexe. Cette structure est représentée dans la figure suivante :

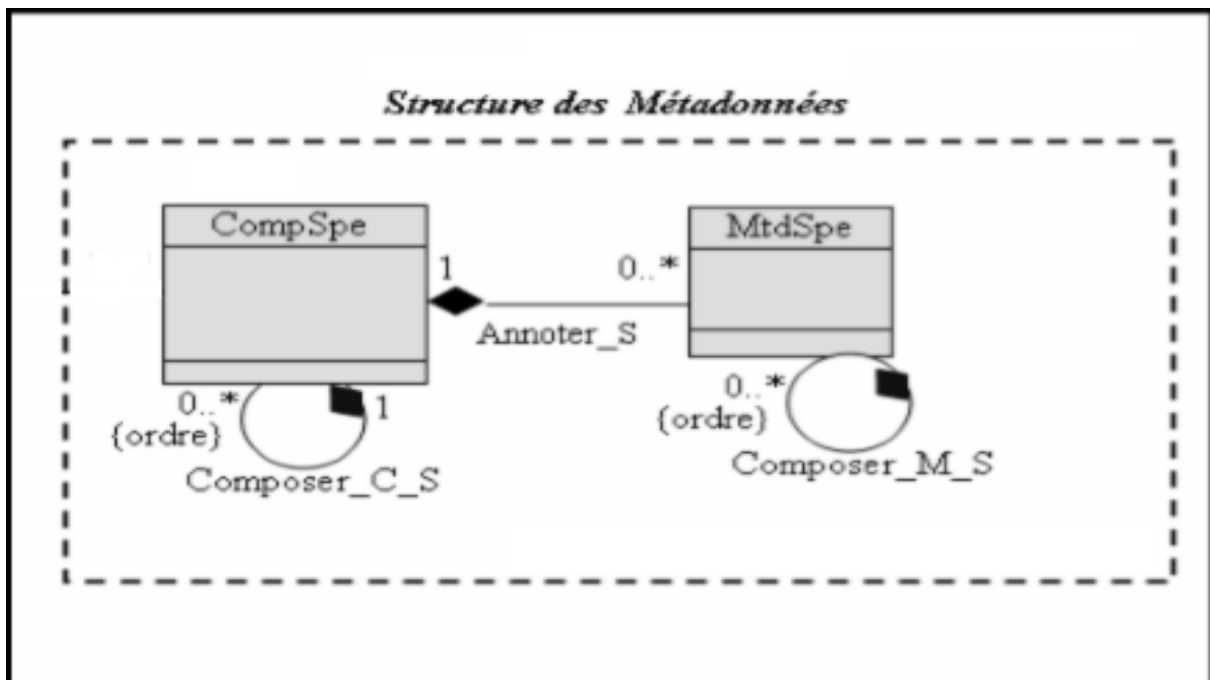


Figure III.4. Structure des métadonnées

La classe "**CompSpe**" représente des composants spécifiques qui peuvent être imbriqués, chaque composant est annoté par des métadonnées.

La classe "**MtdSpe**" représente des métadonnées spécifiques propres à chaque type de média.

▪ **Structure sémantique**

La structure sémantique du document est formée par l'association entre la structure logique et la structure des métadonnées, il est possible de rattacher à une même structure logique plusieurs structures de métadonnées, la structure sémantique est représentée comme le suivant :

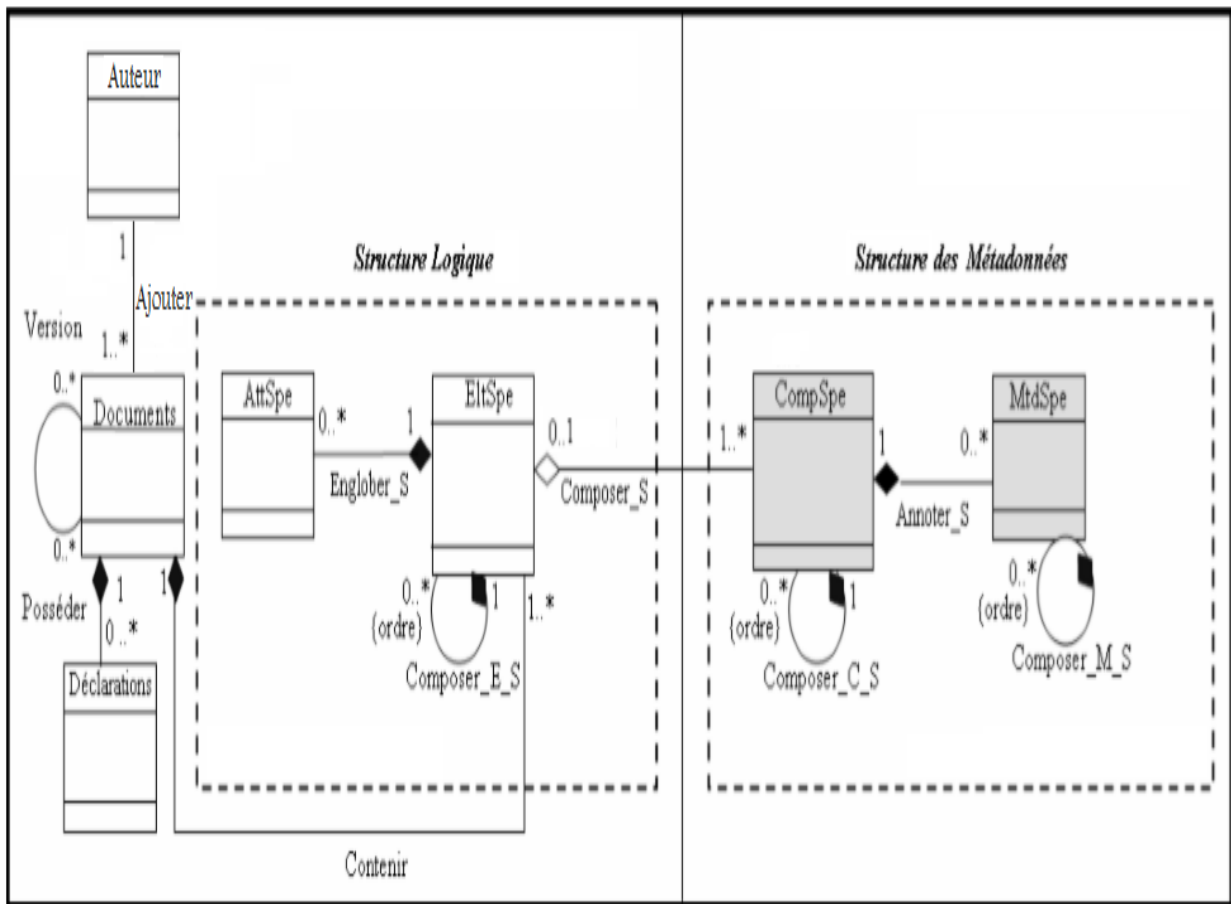


Figure III.5. Structure sémantique d'un document

La figure III.6 Montre la structure sémantique dans le cas du document "Présentation-
_Martyr".

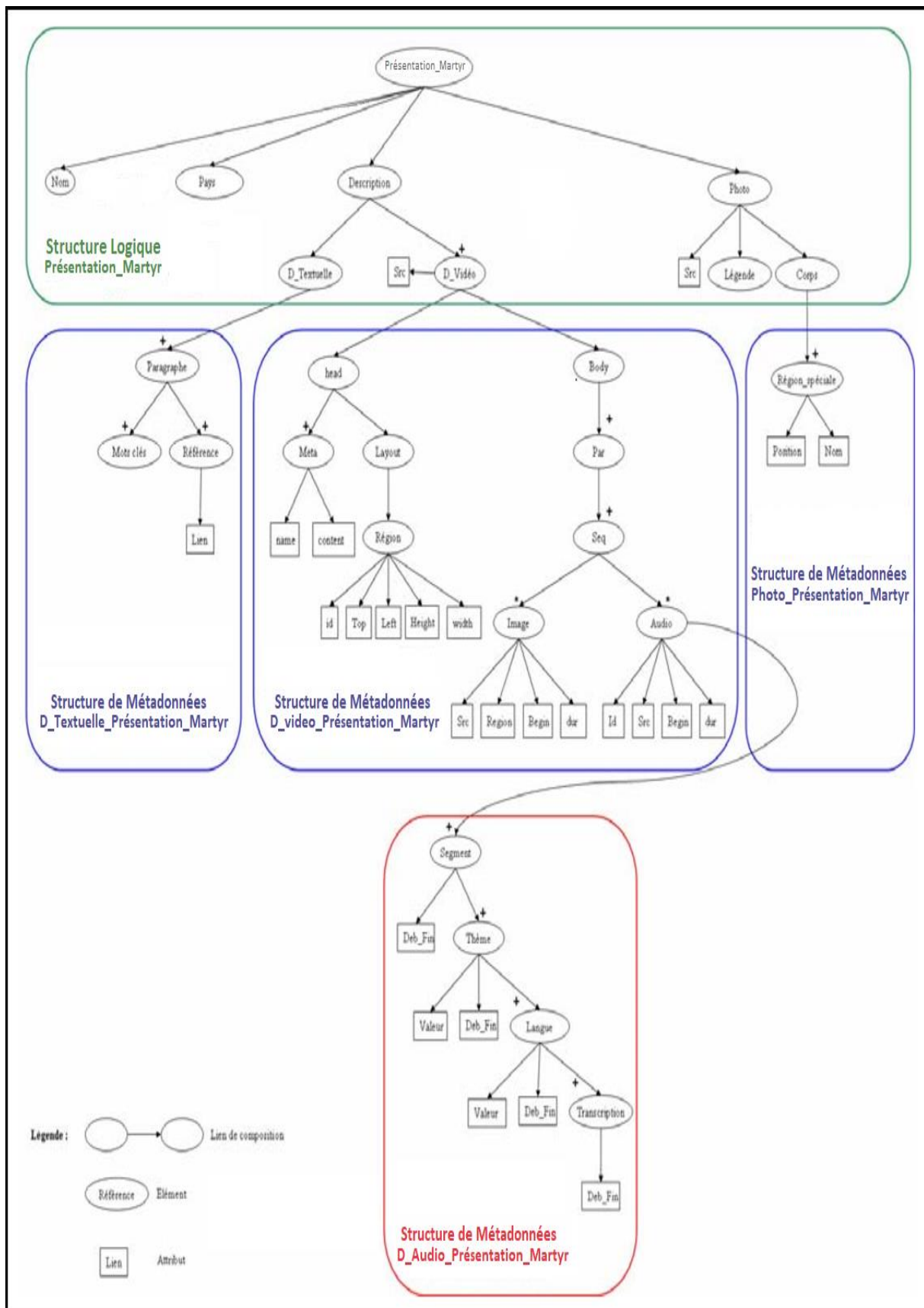


Figure III.6. Structure sémantique dans le cas du document "Présentation_Martyr"

La structure sémantique dans le cas du document "Présentation_Martyr" montre les éléments spécifiques qui appartiennent à ce document, ces éléments spécifiques représentent la classe "EltSpe". Par exemple le document contient l'élément spécifique "Description" qui est composé des deux éléments spécifiques "D_Textuelle" et "D_Vidéo", ce dernier englobe l'attribut spécifique "Src" qui représente à son tour la classe "AttSpe". L'élément "D_Vidéo" est rattaché à la structure des métadonnées "D_video_Présentation_Martyr", car il est composé des deux composants spécifiques "Head" et "Body", le composant spécifique "Body" est composé d'autres composants spécifiques, parmi eux le composant "Audio" qui est rattaché à la structure de métadonnées "D_Audio_Présentation_Martyr". "Audio" est alors un composant complexe.

La cardinalité " " désigne un et un seul, "+" : un ou plusieurs, "*" : zéro ou plusieurs, "?" : zéro ou un.

▪ Structure spécifique d'un document

Une structure spécifique caractérise l'organisation d'un document particulier. Elle se base sur la fragmentation, car la fragmentation permet d'éviter la redondance de stockage des granules.

Aussi il ne se limite pas à des structures prédéfinies, la structure logique du document joue le rôle de structure pivot qui peut prendre des formes différentes. La structure sémantique spécifique est représentée comme suit :

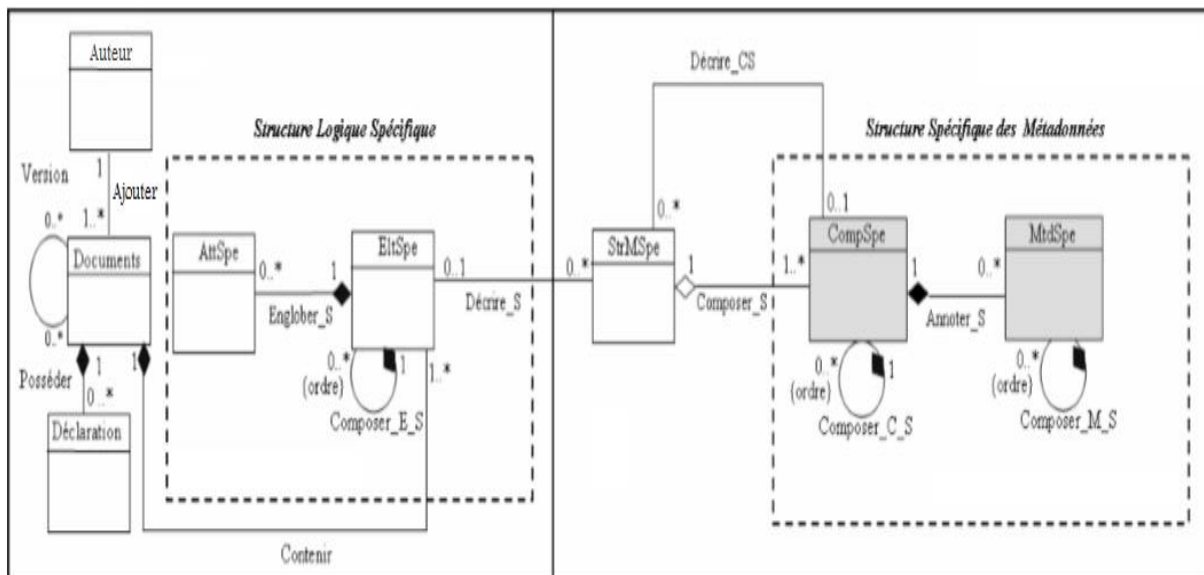


Figure III.7. Structure sémantique spécifique d'un document

La classe "StrMSpe" représente une structure des métadonnées spécifique qui permet de rattacher d'une manière transparente une ou plusieurs descriptions des métadonnées à un élément ou à un composant.

La structure sémantique spécifique dans le cas du document "Présentation_Martyr" (seule une partie est représentée) est la suivante :

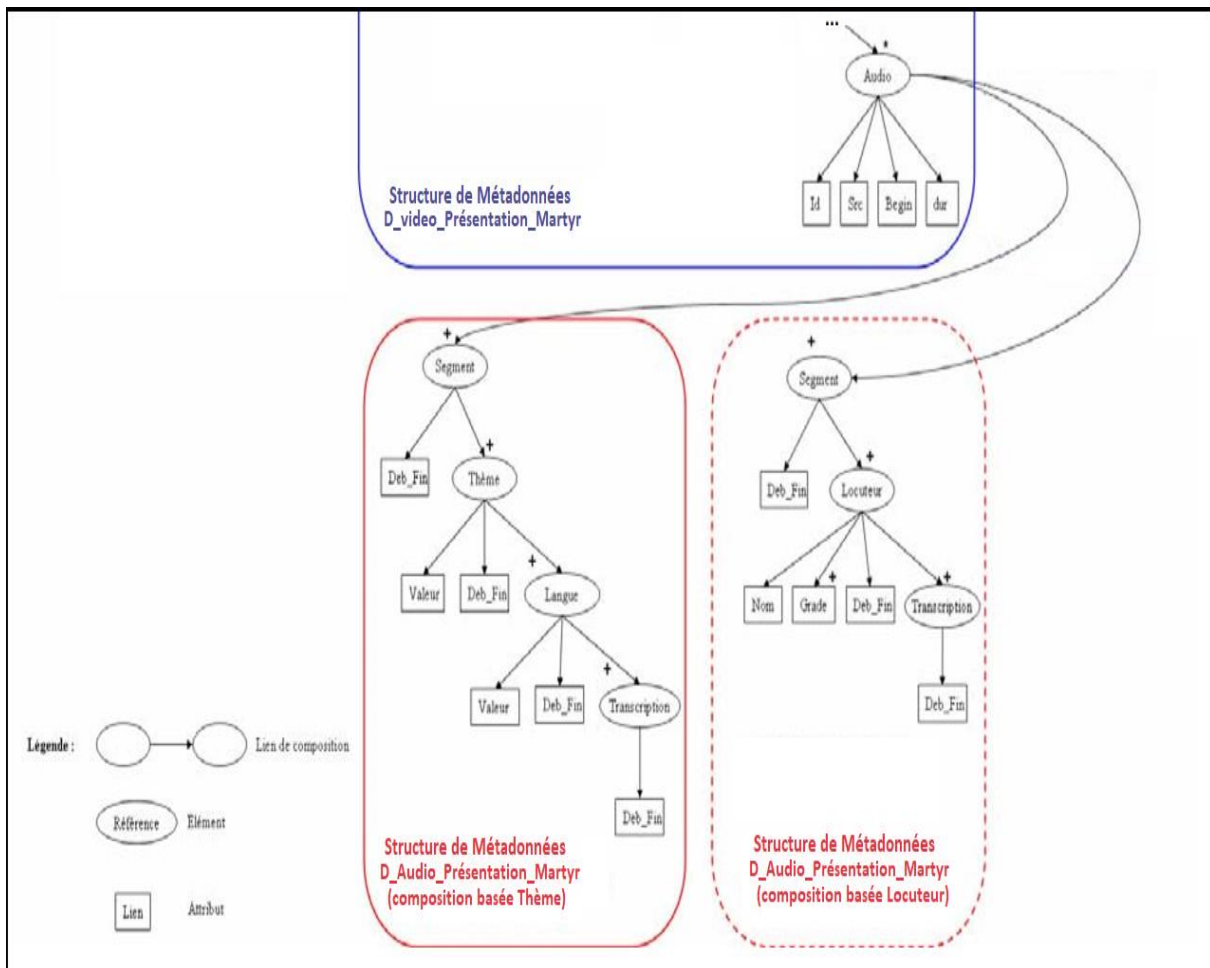


Figure III.8. Structure sémantique spécifique dans le cas du document "Présentation- Martyr"

Le composant complexe "Audio" est rattaché à **deux** structures des métadonnées grâce à la classe "StrMSpe" dans la structure sémantique spécifique : "D_Audio_Présentation_Martyr" (composition basée Thème) et "D_Audio_Présentation_Martyr" (composition basée Locuteur). Ici nous trouvons la notion de "**documents multi-structurés**" (cf. Section I.2.2).

III.2.2. Gestion de l'hétérogénéité inter-documents

L'hétérogénéité inter-documents concerne les différences qui peuvent être détectées entre les contenus (domaines très divers, langues différentes) d'un ensemble de documents ou entre leurs formes de représentation (différences de structures, de formats, de standards ou normes), ces différences de contenu peuvent être dues à des variations autour d'un type de document, ou encore à des évolutions des versions d'un même document. Cela peut engendrer un grand nombre de structures documentaires qui sont relativement semblables.

Le regroupement des documents similaires permet de gérer l'hétérogénéité inter-documents, il consiste à identifier des documents ayant des structures similaires ou identiques afin de les

regrouper dans une même classe afin d'optimiser la recherche ultérieure de tel ou tel type de documents, chaque classe sera caractérisée par une "structure générique" qui représente une collection de "structures spécifiques". Une structure spécifique caractérise l'organisation d'un document particulier, la liste des structures génériques peut être utilisée ainsi comme index pour diminuer le temps d'accès et de recherche dans les structures spécifiques. Ainsi, les utilisateurs de la base peuvent avoir un accès direct à une collection particulière qui répond à leurs besoins.

▪ Structure générique

Une structure générique représente une classe ou une collection de structures spécifiques, les structures spécifiques ayant des structures similaires ou identiques seront regroupées dans une même classe.

Les structures sémantiques génériques différentes peuvent avoir la même structure logique et/ou une ou plusieurs structures de métadonnées en commun.

Afin de regrouper les structures spécifiques autour d'une structure générique, il s'avère nécessaire de rattacher les fragments spécifiques aux fragments génériques par un lien de **conformité** pour éviter la perte des noms des fragments, et éviter aussi le risque d'avoir des champs vides. La structure sémantique générique est représentée comme le suivant :

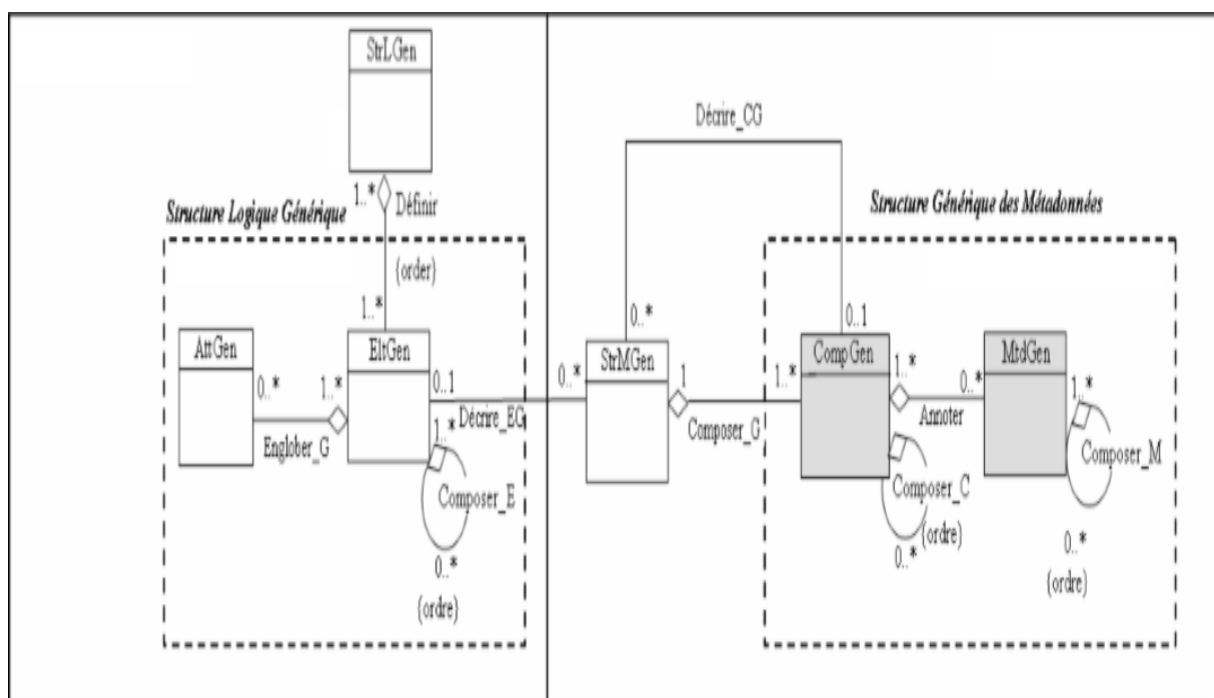


Figure III.9. Structure sémantique générique d'un document

La classe "**StrLGen**" désigne la structure logique générique commune à un ensemble de documents.

La structure sémantique générique dans le cas des deux documents de base "Présentation_Martyr" et "Présentation_Trajan's_Arch" est comme suit :

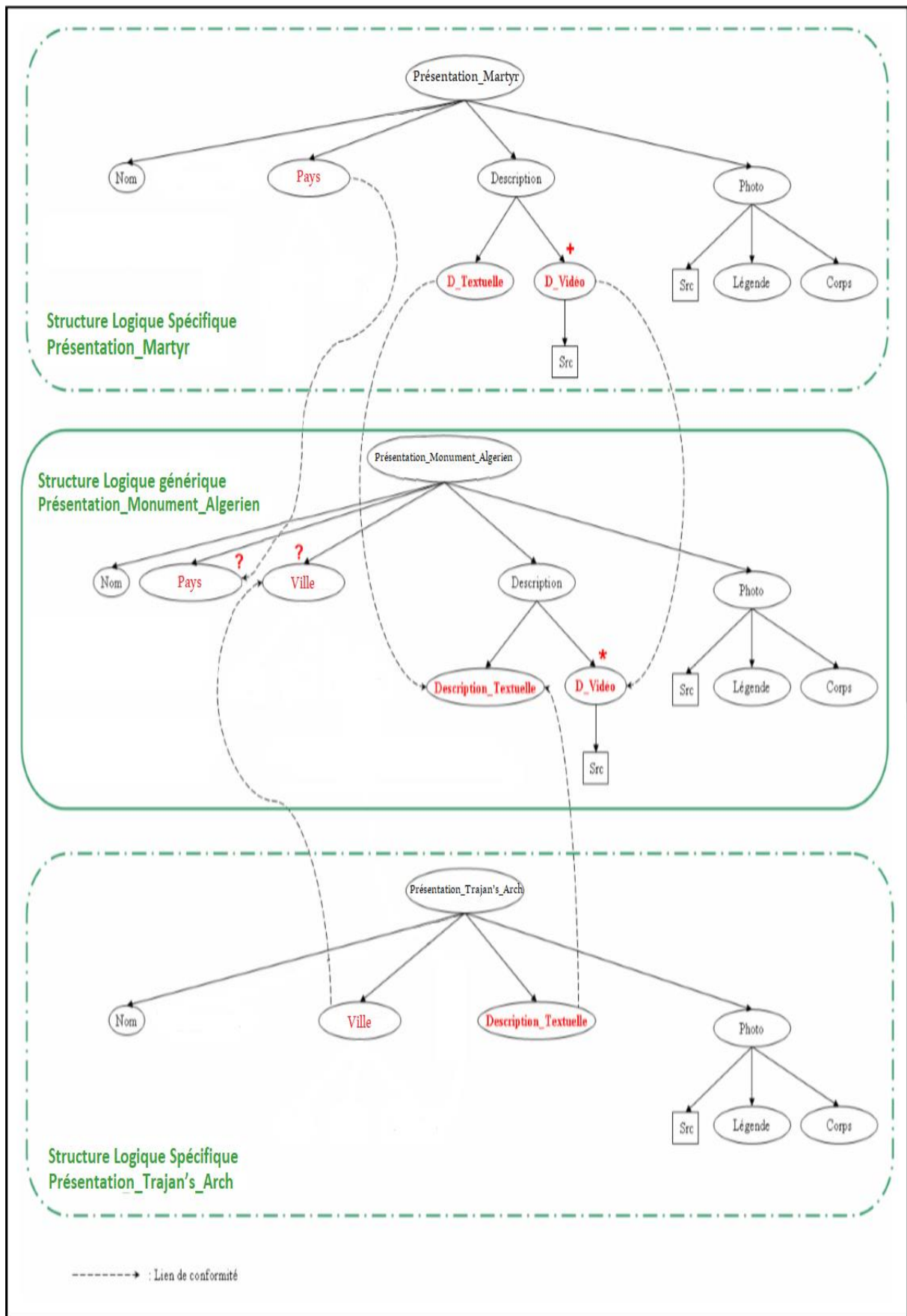


Figure III.10. Structure logique générique dans le cas des documents "Présentation_Martyr" et "Présentation_Trajan's_Arch"


Les structures logiques spécifiques des deux documents "Présentation_Martyr" et "Présentation_Trajan's_Arch" sont regroupées au sein d'une même classe appelée "Présentation_monument_algerien", cette classe est représentée par une structure logique générique.

L'élément "Pays" de la structure spécifique du document "Présentation_Martyr" est attachée à l'élément "Pays" de la structure générique par le lien de conformité, ce dernier est accompagné d'une cardinalité "?" pour dire qu'il est optionnel, ce qui permet d'éviter d'avoir un champ vide dans la structure logique spécifique du document "Présentation_Trajan's_Arch" qui ne contient pas cet élément. C'est le même cas pour les deux éléments "Ville" et "D_Vidéo", mais pour l'élément "Ville" c'est le document "Présentation_Martyr" qui ne contient pas cet élément. Ce qui concerne l'élément "D_Vidéo", il est accompagné d'une cardinalité "*" au lieu de "?" parce que l'élément dans la structure spécifique à une cardinalité "+".

Le lien de conformité entre l'élément "D_Textuelle" et "Description_Textuelle" de la structure générique permet de garder le nom du fragment "D_Textuelle" donné par le créateur du document "Présentation_Martyr". Pour le lien qui attache l'élément "Description_Textuelle" du document "Présentation_Trajan's_Arch", c'est parce qu'il a un chemin différent de celui dans la structure générique.

III.2.3. Modèle de représentation global

Le modèle de représentation réalisé englobe tous ce qu'on a vu précédemment, il est basé sur l'intégration de plusieurs niveaux d'organisation imbriqués :

- Une description structurelle qui correspond à la modélisation de structures logiques, et une description de métadonnées qui décrit le contenu des éléments logiques, Ces deux descriptions forment la structure sémantique du document.
- Une structure spécifique qui est associée à un seul document, les structures spécifiques similaires sont liées à une même structure générique (la structure générique représente une classe ou un groupe) à travers des liens de **conformité**, le lien de conformité est représenté comme le suivant : 

Le modèle de représentation de documents multimédia est le suivant :

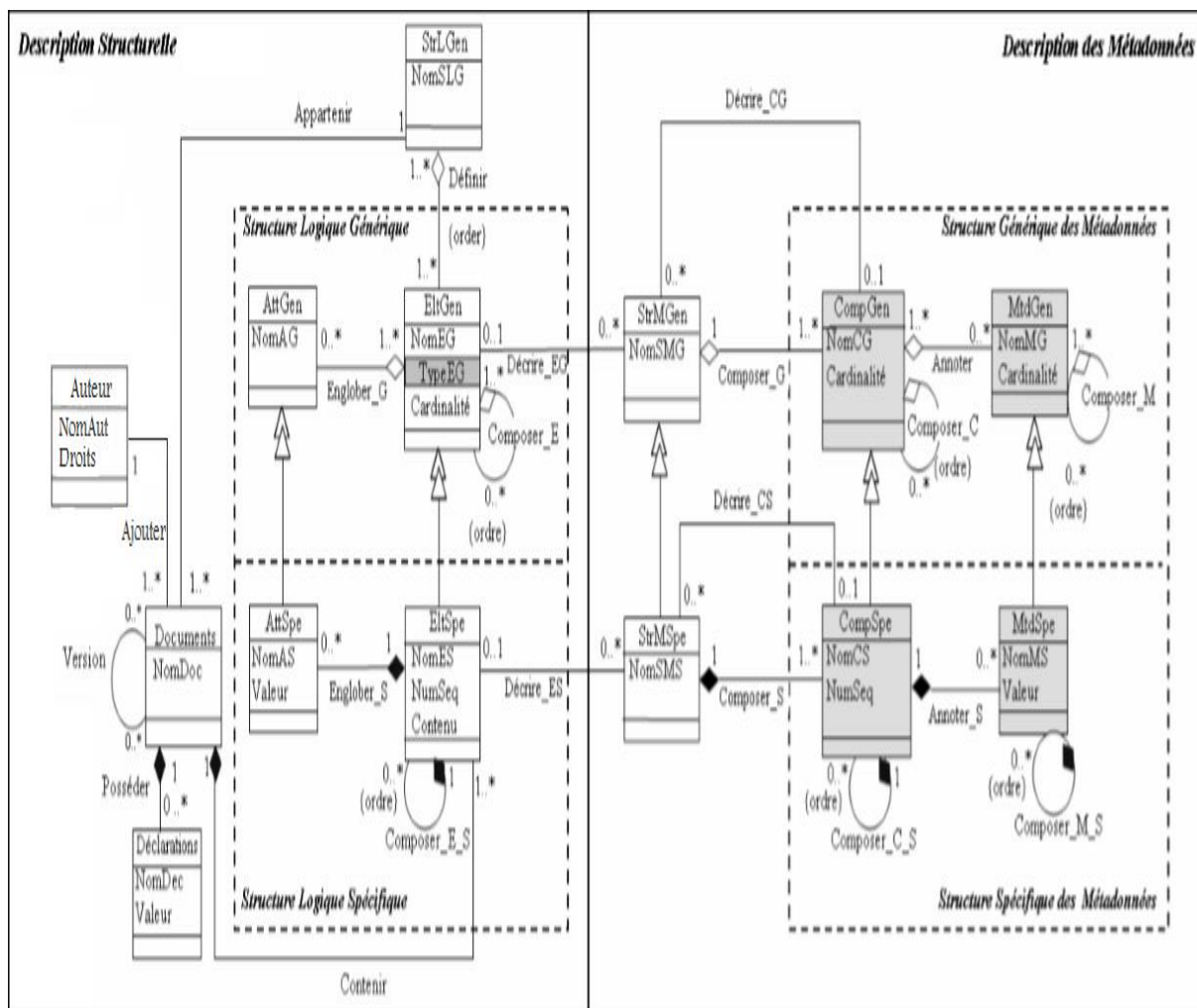


Figure III.11. Modèle de représentation de documents multimédia

■ Description structurelle

L'auteur d'un document est caractérisé par son nom "**NomAut**", et un champ "**Droits**", à travers ce champ l'auteur peut tolérer ou ne pas tolérer la mise à jour de la description de son document par un autre utilisateur. C'est en insérant la classe "**Auteur**" et en ajoutant le champ "**Droits**" que nous avons pu régler le problème des droits d'utilisation.

Le niveau spécifique : Un élément spécifique est caractérisé par son nom "**NomES**" qui permet de sauvegarder le nom de la balise qui existe dans le document s'il est différent de celui indiqué dans la structure logique générale, d'un numéro de séquence "**NumSeq**" dans la composition de l'élément générale.

Le niveau générale : Un élément générale est caractérisé par un nom générale "**NomEG**", un type générale "**TypeEG**", et une "**Cardinalité**" qui indiquera les différentes occurrences possibles d'un élément (" " : un et un seul, "+" : un ou plusieurs, "*" : zéro ou plusieurs, "?" : zéro ou un).

▪ Description des métadonnées

Au niveau spécifique : la structure spécifique des métadonnées présente la composition spécifique d'un document particulier et détaille la description de son contenu, elle est adéquate à une structure générique des métadonnées.

Au niveau générique : la structure générique des métadonnées permet d'extraire des composants génériques implicites, ces composants ne peuvent pas être déduits directement de la structure logique générique. Un composant générique peut être unité textuelle, une région d'une image, une séquence vidéo, un segment (parole et/ou musique). Un composant générique peut être fragmenté en d'autres composants génériques, il est possible aussi d'attribuer à chaque composant générique des métadonnées génériques.

La description structurelle dans le cas du document "Présentation_Martyr" est dans la figure III.12, la description des métadonnées du fragment "D_Textuelle" est dans la figure III.13.

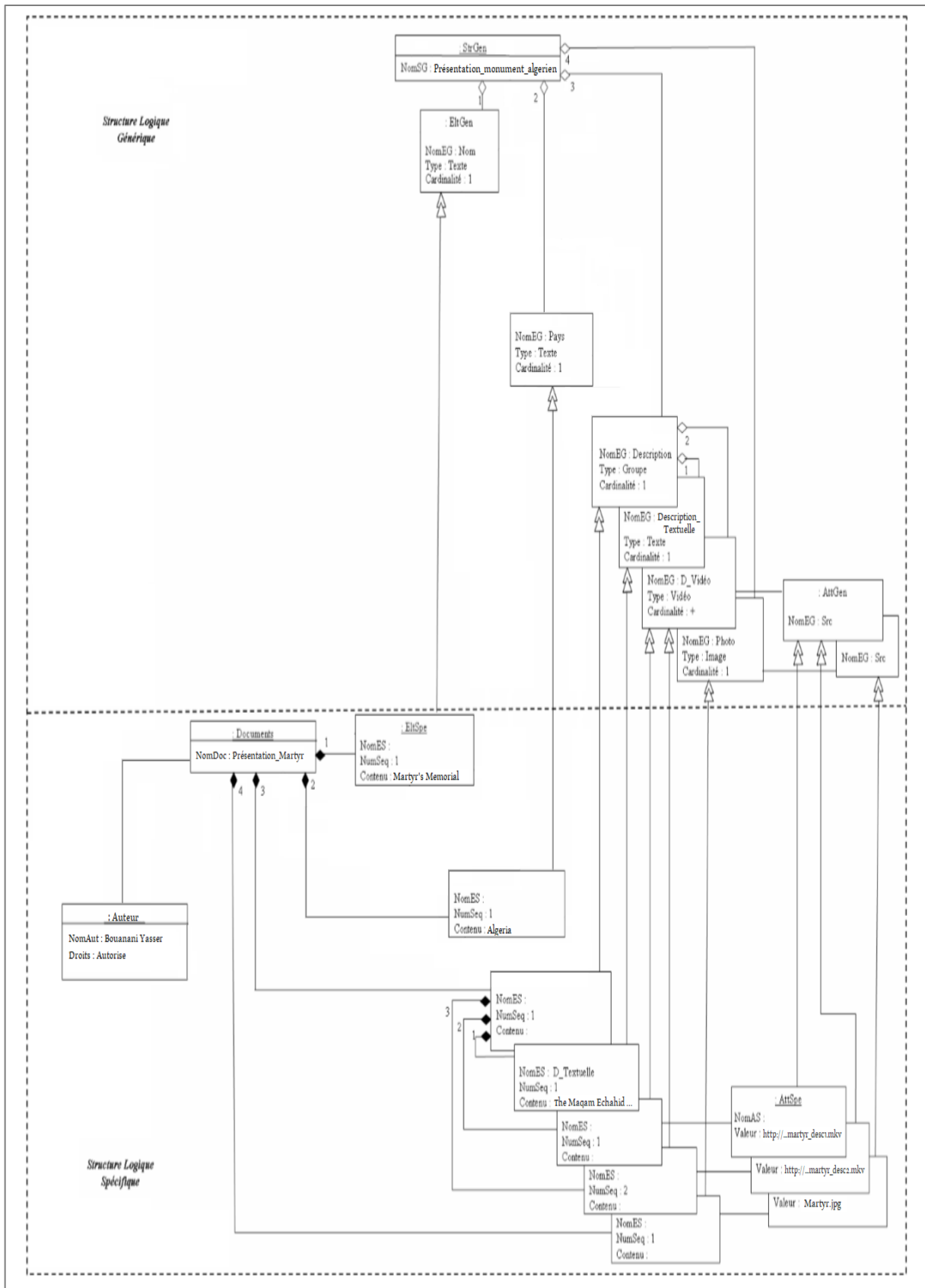


Figure III.12. Description structurelle dans le cas du document "Présentation_Martyr"

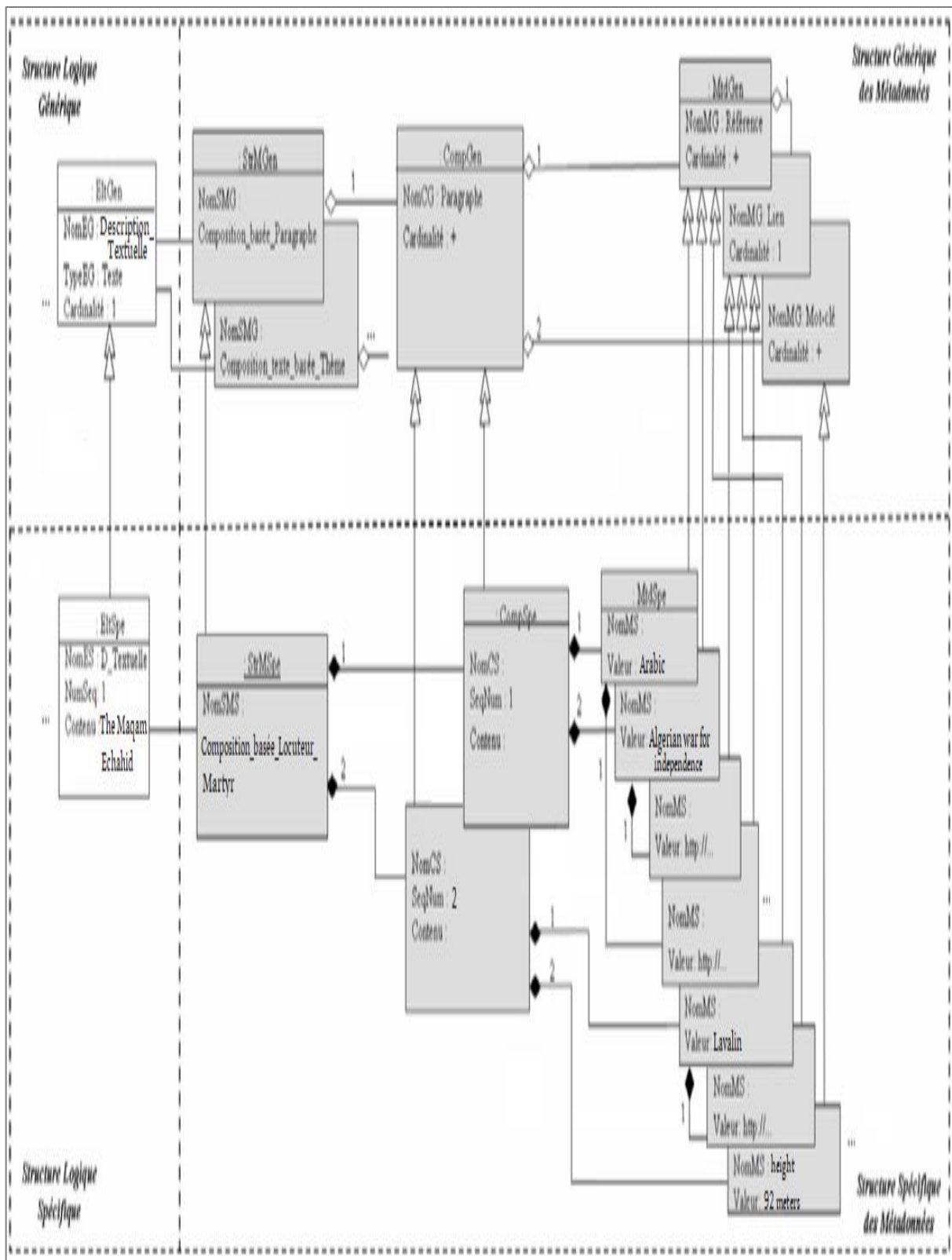


Figure III.13. Description des métadonnées du fragment "D_Textuelle"

Dans la description structurale (figure III.12) on peut voir le nom de l'auteur ("NomAut") qui a ajouté le document "Présentation_Martyr" est "Bouanani Yasser", à travers le champ

"Droits" cet auteur autorise la mise à jour de la description de son document par un autre utilisateur. Le nom de l'élément spécifique ("NomES") "D_Textuelle" est sauvegardé parce qu'il est différent de celui indiqué dans la structure logique générique ("Description_Textuelle"). La valeur de numéro de séquence "NumSeq" de l'élément attaché à l'attribut qui a la valeur "http//... martyr_desc2.mkv" est 2 car c'est le deuxième élément spécifique rattaché à l'élément générique "D_Vidéo".

Dans la description des métadonnées du fragment "D_Textuelle" (figure III.13) on peut voir la structure générique des métadonnées rattachée à l'élément générique "Description_Textuelle" de la structure générique "Présentation_monument_algerien", ces structures reflètent les différentes façons utilisées pour décrire le contenu de cet élément générique : "composition basée paragraphes", "composition basée thèmes". On peut voir aussi la structure spécifique des métadonnées qui décrit le contenu de la description textuelle du document "Présentation_Martyr", ce contenu est décrit seulement selon la "composition basée paragraphes".

III.3. Classification des documents multimédia

La démarche de classification repose sur l'extraction de la structure spécifique de document, sur la comparaison de cette structure avec l'ensemble des structures générique de la base pour pouvoir rattacher le nouveau document à une classe de documents. Ce rattachement peut être effectué directement après la phase de comparaison ou suite à une adaptation de l'une des deux structures (générique ou spécifique), le rattachement peut être assuré également suite à la création d'une nouvelle structure générique.

La figure suivante résume cette démarche de classification :

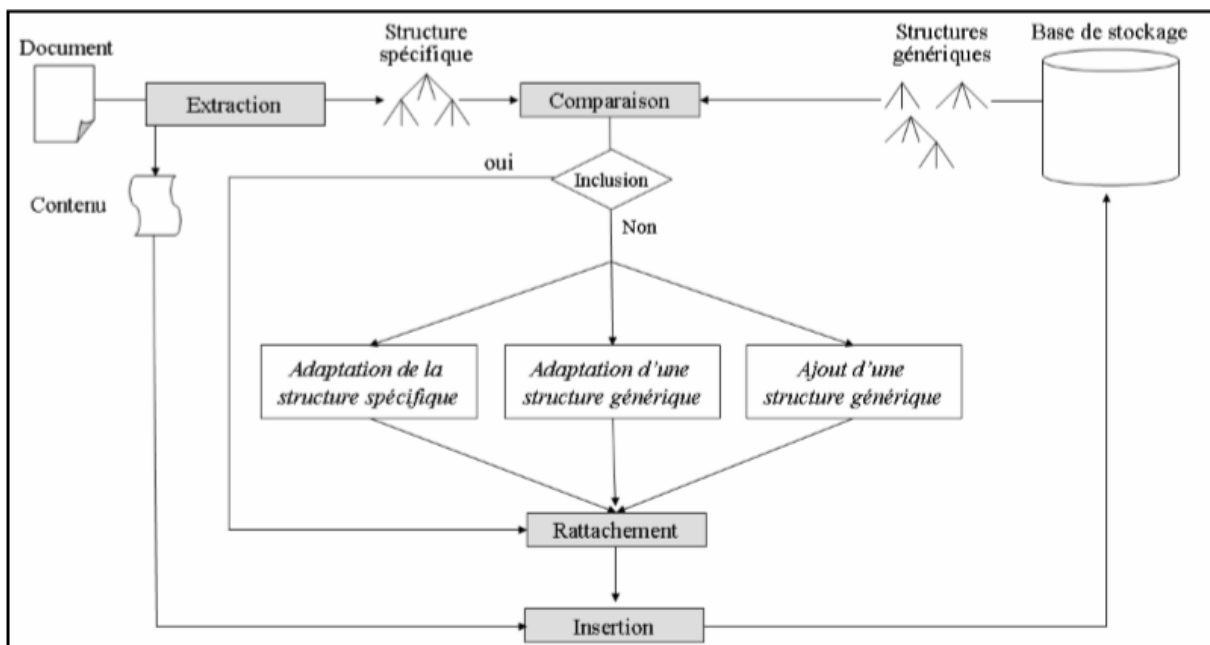


Figure III.14. Démarche de classification de structures

III.3.1. Extraction de structures spécifiques

L'extraction de la structure spécifique du document consiste à séparer sa structure et son contenu, la structure spécifique est extraite à partir de ces balises d'annotation (granules documentaires) en ayant recours à un moteur de réécriture de chaînes.

Le résultat de cette phase est une structure arborescente ordonnée et étiquetée où chaque nœud de cette arborescence peut être mono-valué, optionnel ou multi-valué, les étiquettes d'une arborescence correspondent aux différents noms des balises extraites du document (éléments ou composants ou attributs ou métadonnées).

Le processus d'extraction des structures spécifiques se déroule en neuf étapes :

- **Etape 1** : extraction des balises du document (sans contenu).
- **Etape 2** : extraction des attributs des différentes balises, les noms des attributs sont préfixés par "A_".
- **Etape 3** : chaque balise ouvrante immédiatement suivie de sa balise fermante sera remplacée par un élément spécifique de niveau "1" et de cardinalité "1", le nom de l'élément spécifique est préfixé par "Ex_" (x constitue le numéro de son niveau) et suivi de sa cardinalité.
- **Etape 4** : si des éléments spécifiques de même nom se succèdent, ils sont alors remplacés par un seul élément spécifique dont la cardinalité devient "+" au lieu de "1", la liste des attributs de cet élément est formée par l'union de toutes les listes d'attributs des éléments que vont être remplacés, l'attribut qui ne figure pas dans toutes les listes sera enregistré avec la cardinalité "?".
- **Etape 5** : si entre une balise ouvrante et sa balise fermante se trouve que des éléments spécifiques au travers des étapes précédentes, ces deux balises (ouvrante et fermante) seront remplacées par un nouvel élément spécifique, son niveau prend la valeur 1, alors que le niveau de ses sous-éléments doit être incrémenté de 1.
- **Etape 6** : si des éléments spécifiques de même nom et de même niveau ne sont séparés que par d'autres éléments de niveau supérieur (que par des fils), ils seront alors remplacés par un seul élément spécifique dont la cardinalité devient "+" et qui garde le même ordre de niveau, les listes des attributs et des éléments qui seront rattachés à cet élément sont formées par l'union des listes des attributs des éléments qui sont rattachés aux éléments que vont être remplacés, les éléments et les attributs qui n'existent pas dans toutes les listes seront enregistrés dans la liste finale avec la cardinalité "?" (ou "*" s'ils possèdent déjà la cardinalité "+").
- **Etape 7** : répéter les étapes 1 à 6 jusqu'à obtenir un fichier qui ne contienne que des éléments spécifiques.
- **Etape 8** : remplacer les lignes qui ont la forme Ex_"nom"_"cardinalité" par <!Element "nom" ("fils1" "cardinalité_fils1", ..., "filsn" "cardinalité_filsn") >, la liste des fils est formée par le nom des éléments de niveau (x+1) rencontrés avant de trouver un autre élément de niveau (x), x étant le niveau de l'élément en cours de traitement. "Cardinalité_fils*" est la cardinalité de fils en cours d'ajout, la liste éventuelle des attributs présentée entre accolades suite de l'étape précédente sera effacée et remplacée par une liste de lignes

(une ligne par attribut), chaque ligne sera éditée sous la forme suivante : < !Attribut "nom_attribut">.

- **Etape 9** : s'il existe dans la liste des fils d'un élément père des éléments qui se répètent sans se succéder, alors la sous-séquence qui comporte des doublons sera remplacée par une autre sous séquence formée par des éléments uniques et qui sont séparés par l'opérateur "OU" : "|", la cardinalité de cette sous séquence sera "+".

Comme exemple d'exécution de ce processus nous procédant à l'extraction de la structure des métadonnées du composant complexe "audio" nommée "D_Audio_Présentation_Martyr" (composition basée Locuteur) déjà vue dans la figure III.8 :

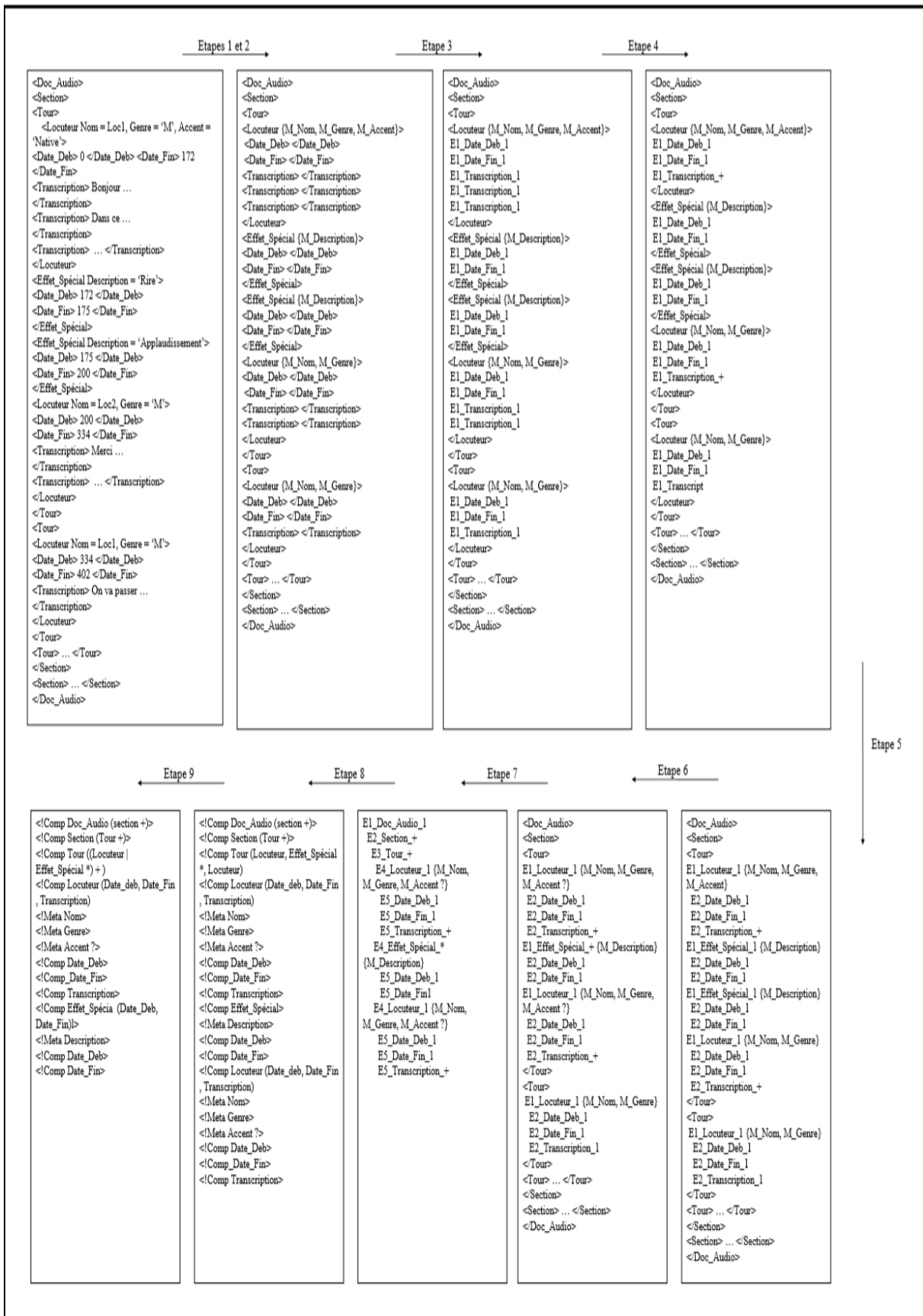


Figure III.15. Exécution du processus d'extraction dans le cas composant complexe "audio" (composition basée Locuteur)

III.3.2. Comparaison de structures

La comparaison de deux structures exige une comparaison séparée de leur structures logiques et les structures de métadonnées, elle consiste à comparer leurs arborescences respectives, et tenir en compte de l'ordre des éléments et des composants en largeur et en profondeur, par contre le non-respect n'est pas pénalisant au niveau des attributs et des métadonnées (car il ne modifie pas la sémantique du contenu des balises).

La figure suivante résume le processus de comparaison :

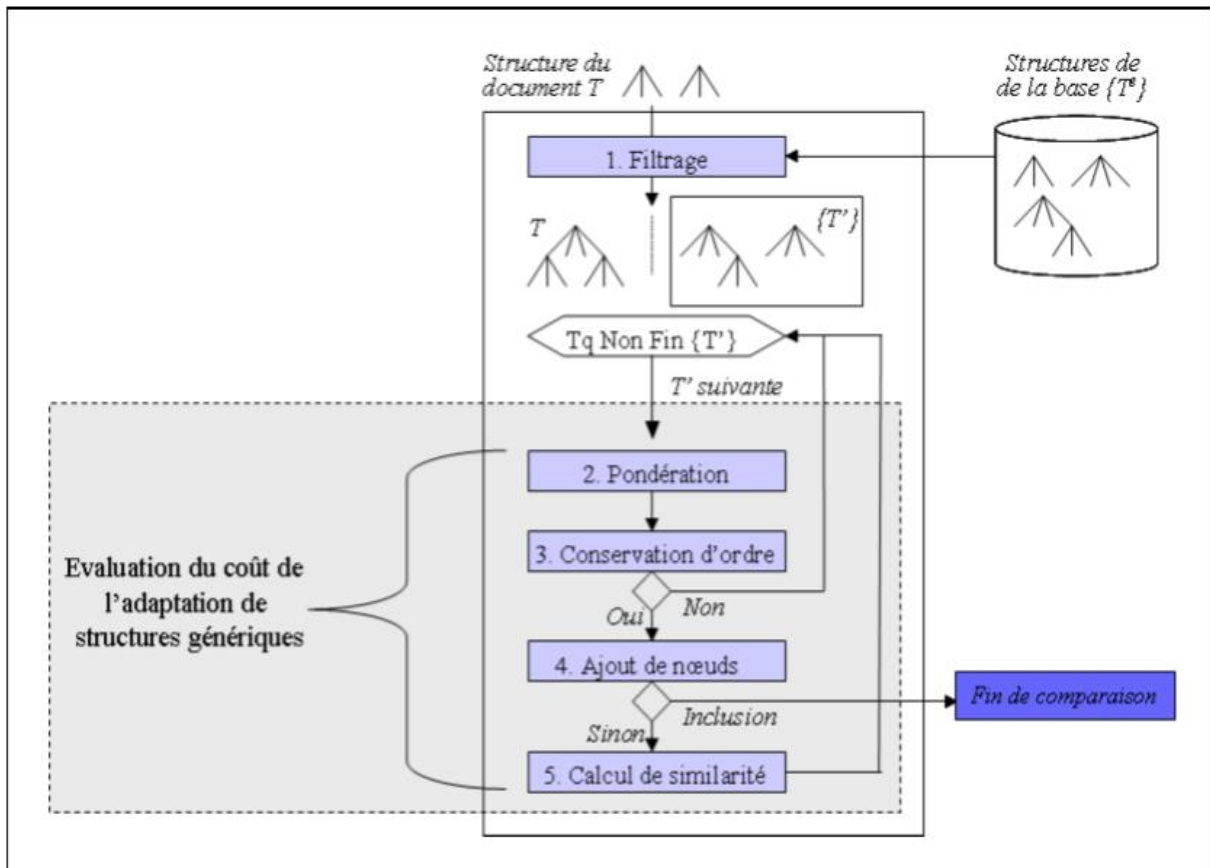


Figure III.16. Algorithme de comparaison

Les concepts de base utilisés dans le processus de comparaison sont les suivants :

$T = (V, E)$ une arborescence ordonnée et étiquetée.

$V = \{v_1, \dots, v_n\}$ l'ensemble des nœuds de $T / \forall i \in [1, 2, \dots, n], v_i = (Etiqu_i, ordre_i)$.

$E = \{e_1, \dots, e_m\}$ l'ensemble des arcs de $T / \forall i \in [1, 2, \dots, m], \forall j, k \in [1, 2, \dots, n], j \neq k, e_i = (v_j, v_k)$.

$v_{EC} = \{v_1, \dots, v_k\}$ les éléments ou les composants de T .

$v_{AM} = \{v_1, \dots, v_n\}$ les attributs ou les métadonnées de T .

v_0 est un nœud unique n'appartenant à aucune arborescence appelé nœud vide.

T[v] est la sous-arborescence de T enracinée au nœud v de T, c'est l'arborescence de racine v que nous obtenons en effaçant l'arc qui pointe sur le nœud v dans T et en conservant l'arbre connexe issu de v.

fil[s] est la liste des fils du nœud v, le nombre de fils du nœud v est noté n_v .

anc[v] est l'ensemble des ancêtres du nœud v dans T, v non inclus.

chm[v] est la liste des nœuds qui composent le chemin allant de la racine r vers v de T, v inclus ($chm[v] = \{v\} + anc[v]$).

des[v] est l'ensemble des descendants du nœud v dans T.

pere[v] est un nœud représentant le père du nœud v.

eti[s] est l'étiquette associée au nœud v, les étiquettes sont les noms des balises.

card(X) est le nombre de nœuds appartenant à X, X est une arborescence.

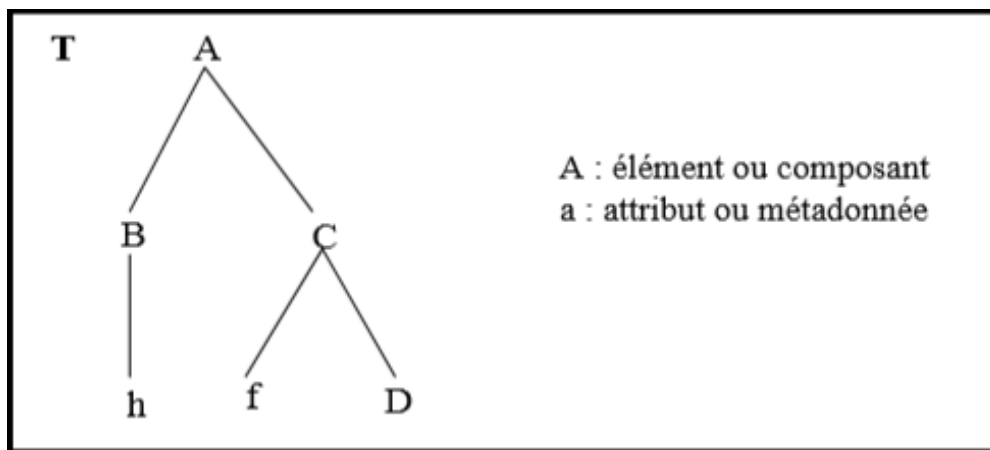


Figure III.17. Exemple de représentation de la structure T

▪ Filtrage

Un filtrage des structures génériques consiste à sélectionner un ensemble de structures génériques candidates à la comparaison, ces structures ont un certain degré de ressemblance avec la structure spécifique, ce degré est évalué sur la base d'un pourcentage de nœuds communs entre ces structures.

Une fonction d'alignement **m** permet de déterminer la sous-arborescence commune à la structure spécifique et à la structure générique, cette fonction consiste à associer à chacun des nœuds d'une arborescence V un nœud de l'arborescence V' possédant la même étiquette ou une étiquette synonyme ou le nœud vide (v_0) dans le cas contraire.

Fonction d'alignement **m**

$$\forall v \in V \text{ et } \forall v' \in V' \cup \{v_0\} \Rightarrow \begin{cases} m(v) = v' \text{ Si } \text{Etiqu}(v) = \text{Etiqu}(v') \text{ Ou } \text{Etiqu}(v) = \text{synonyme} \text{Etiqu}(v') \\ m(v) = v_0 \text{ Sinon} \end{cases}$$

$$\forall v' \in V' \text{ et } \forall v \in V \cup \{v_0\} \Rightarrow \begin{cases} m(v') = v \text{ Si } \text{Etiqu}(v') = \text{Etiqu}(v) \text{ Ou } \text{Etiqu}(v') = \text{synonyme} \text{Etiqu}(v) \\ m(v') = v_0 \text{ Sinon} \end{cases}$$

Le degré de ressemblance C_f doit être supérieur au seuil fixe par expérimentation S_f pour que la structure générique soit candidate, l'équation pour calculer C_f est la suivante :

$$C_f = \left(\frac{\text{card}(v_{EC}) * \text{card}(v''_{EC}) + \text{card}(v_{AM}) * \text{card}(v''_{AM})}{\text{card}(v_{EC})^2 + \text{card}(v_{AM})^2} + \frac{\text{card}(v'_{EC}) * \text{card}(v''_{EC}) + \text{card}(v'_{AM}) * \text{card}(v''_{AM})}{\text{card}(v'_{EC})^2 + \text{card}(v'_{AM})^2} \right) / 2$$

La figure qui suit montre un exemple d'exécution de la fonction d'alignement :

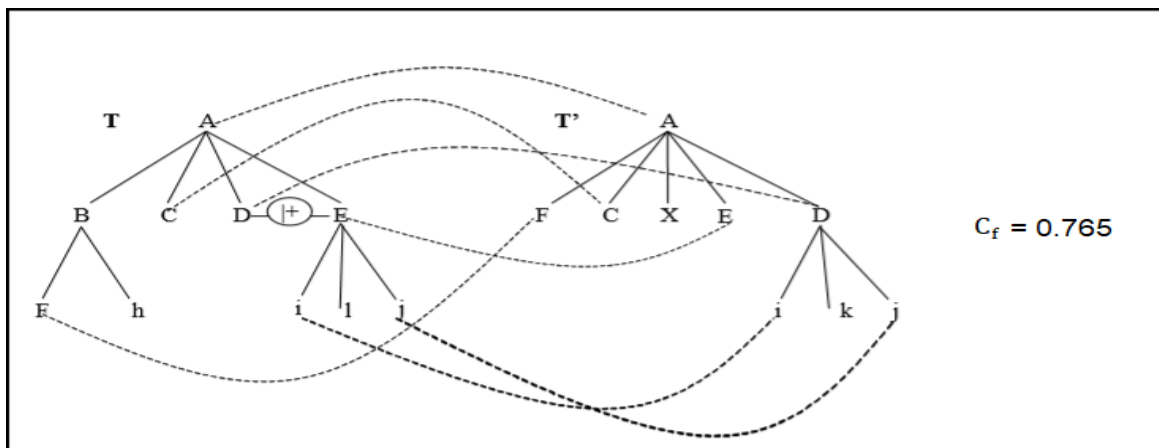


Figure III.18. Exemple d'alignement de deux structures

▪ Pondération

L'étape de pondération consiste à attribuer des poids aux nœuds de la structure de manière à tenir compte d'une part de la profondeur du nœud et d'autre part de son ordre parmi ses frères.

La fonction $P_n : V \rightarrow [1, 2[$ associe à chaque nœud v un poids V :

$$P_n(v) = \begin{cases} 1, \text{ si } v = r / r : \text{ racine de l'arborescence } V (\text{anc}[r] = \emptyset) \\ P_n(\text{pere}(v)) + \frac{\beta}{N^{\alpha-1}}, \text{ sinon } (\text{anc}[v] \neq \emptyset) \end{cases}$$

Telque:

α : est la profondeur du nœud u dans l'arborescence, la profondeur de la racine est égale à 1, cette valeur sera incrémentée de un pour chaque niveau en se rapprochant des feuilles.

β : un ordre qui peut avoir comme valeur

- Numéro d'ordre du nœud v dans l'ensemble des fils de père (v) (sans compter les attributs et les métadonnées).
- Zéro si v est un attribut ou une métadonnée.
- Le même ordre attribué au nœud v_1 si le nœud v participe à une combinaison d'alternance avec v_1 ($v | v_1$).

P_n : permet de gérer toute structure de document avec la contrainte que $\text{card}(\text{fils}(u)) < N$ pour tout nœud u non feuille.

n : nombre maximum de fils pour un même père défini par $10^x, \forall x \in [1, \dots, n]$.

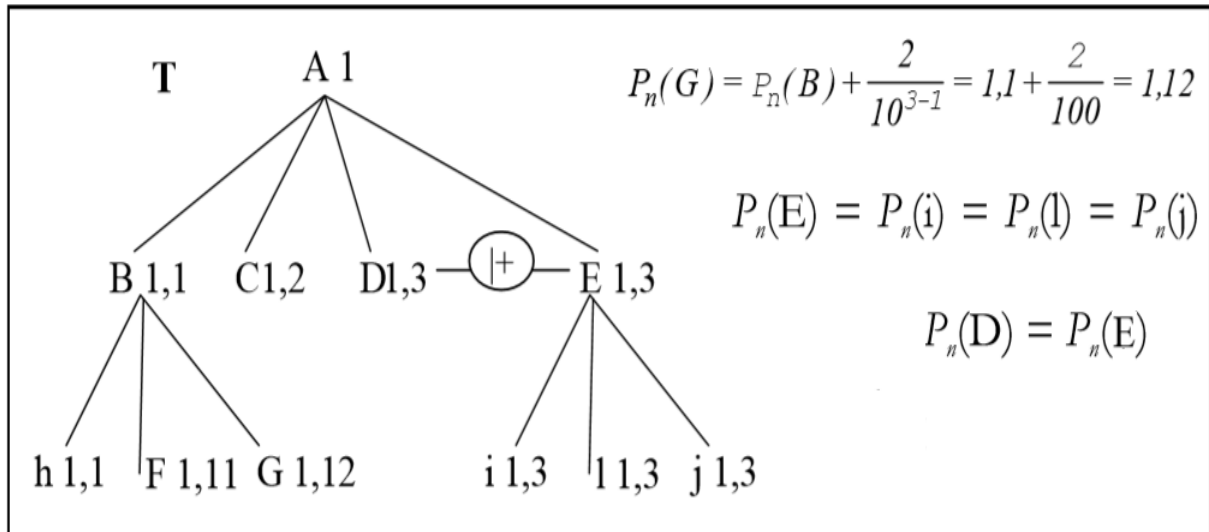


Figure III.19. Exemple de pondération des nœuds d'une arborescence

▪ Conservation d'ordre

La conservation d'ordre permet d'étudier le placement des nœuds alignés, dans les structures à comparer. Deux règles doivent être vérifiées : la conservation d'ancestralité et la conservation d'ordre des fils d'un nœud père. Si l'une des deux règles n'est pas vérifiée, les deux structures arborescentes sont considérées comme suffisamment différentes, cette structure ne sera pas gardée dans la liste des structures candidates, cette conservation concernera uniquement les éléments et les composants, mais pas les attributs et les métadonnées.

• Conservation d'ancestralité :

La règle de conservation d'ancestralité permet de s'assurer que deux nœuds alignés ont également leurs ancêtres alignés.

$$\left. \begin{array}{l} \forall \{v_1, v_2\} \in V \text{ et } \forall \{v'_1, v'_2\} \in V', v_1 \in \text{anc}[v_2] \\ m(v_1) = v'_1, v'_1 \neq v_0 \text{ et } m(v_2) = v'_2, v'_2 \neq v_0 \end{array} \right\} \Rightarrow v'_1 \in \text{anc}[v'_2]$$

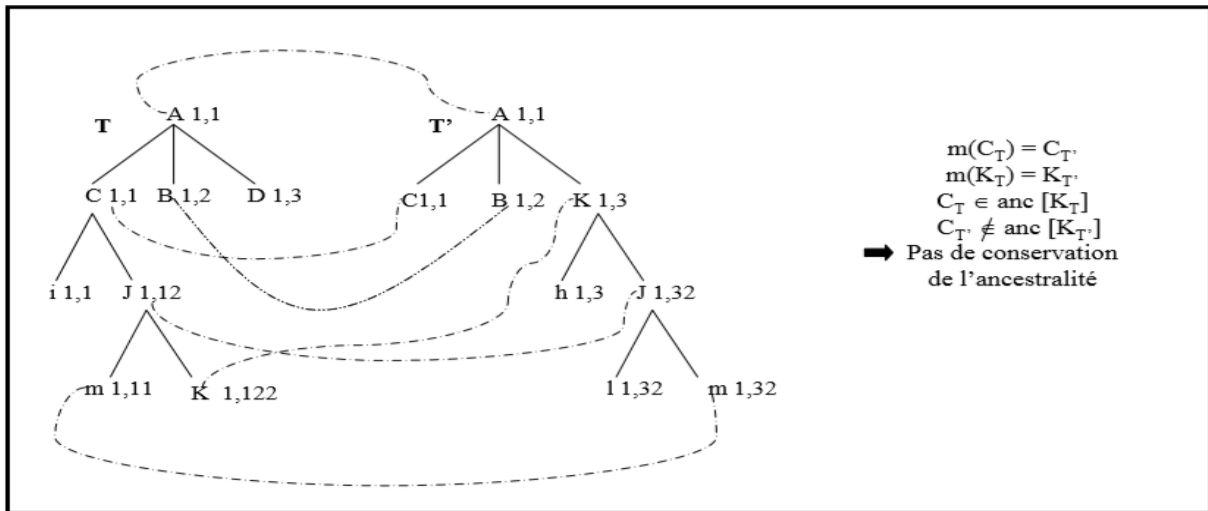


Figure III.20. Exemple de non-conservation d'ancestralité

• Conservation d'ordre des fils :

La règle de conservation d'ordre des fils permet de s'assurer du respect de l'ordre des nœuds fils alignés de deux pères alignés.

$$\left. \begin{array}{l} \forall \{v_1, v_2\} \in V \text{ et } \forall \{v'_1, v'_2\} \in V' \\ m(v_1) = v'_1, v'_1 \neq v_0 \text{ et } m(v_2) = v'_2, v'_2 \neq v_0 \\ \text{pere}(v_1) = \text{pere}(v_2) \text{ et } \text{pere}(v'_1) = \text{pere}(v'_2) \\ P_n(v_1) \leq P_n(v_2) \end{array} \right\} \Rightarrow P_n(v'_1) \leq P_n(v'_2)$$

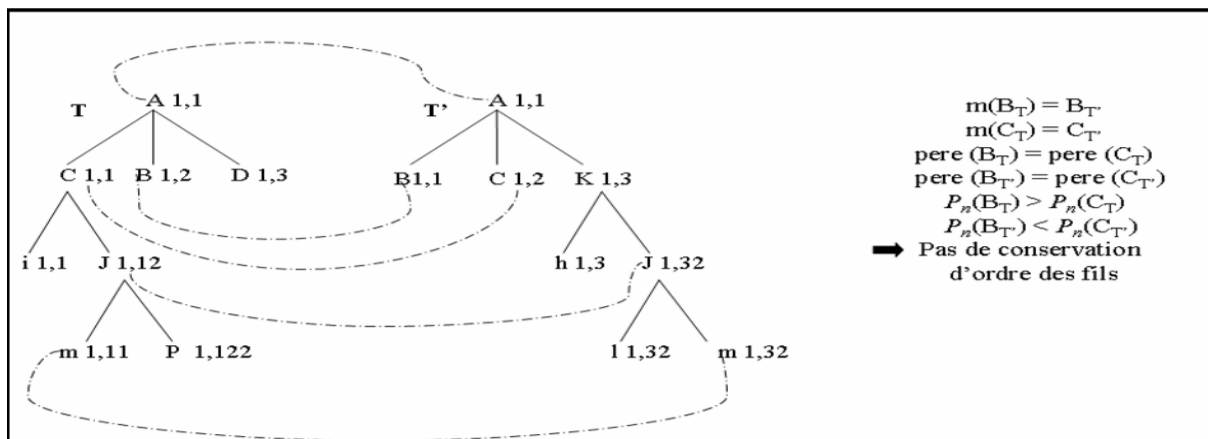


Figure III.21. Exemple de non-conservation d'ordre des fils

▪ **Ajout de nœuds**

Consiste à vérifier, en insérant fictivement des nœuds en largeur ou en profondeur à la structure générique, si on se rapproche ou non de la structure spécifique à intégrer. Il existe deux types d'insertion : (1) ajout d'ancêtres, (2) ajout de fils.

- Ajout d'ancêtres :

Pour chaque élément feuille v d'une arborescence dont il existe un élément v' correspondant dans l'autre arborescence, récupérer son chemin jusqu'à la racine ($chm[v]$) ainsi que celui de son homologue ($chm[v']$), pour les comparer et éventuellement insérer des nœuds ancêtres.

Soit $v \in V, v' \in V', m(v) = v', fils[v] = \emptyset$ $\forall v', v'' \in chm[v']$ et $\forall vk \in chm[v]$ $pos(v'i) = pos(v'j) + 1, Pn(vk) \geq Pn(m(v'i))$ et $Pn(vk) \leq Pn(m(v'j))$	$\left. \vphantom{\begin{matrix} \text{Soit } v \in V, v' \in V', m(v) = v', fils[v] = \emptyset \\ \forall v', v'' \in chm[v'] \text{ et } \forall vk \in chm[v] \\ pos(v'i) = pos(v'j) + 1, Pn(vk) \geq Pn(m(v'i)) \\ \text{et } Pn(vk) \leq Pn(m(v'j)) \end{matrix}} \right\} \Rightarrow \text{Ajout de } vk \text{ entre } v'i \text{ et } v'j \text{ dans } V'$
Telque : pos est une fonction permettant de récupérer la position d'un nœud dans un chemin.	

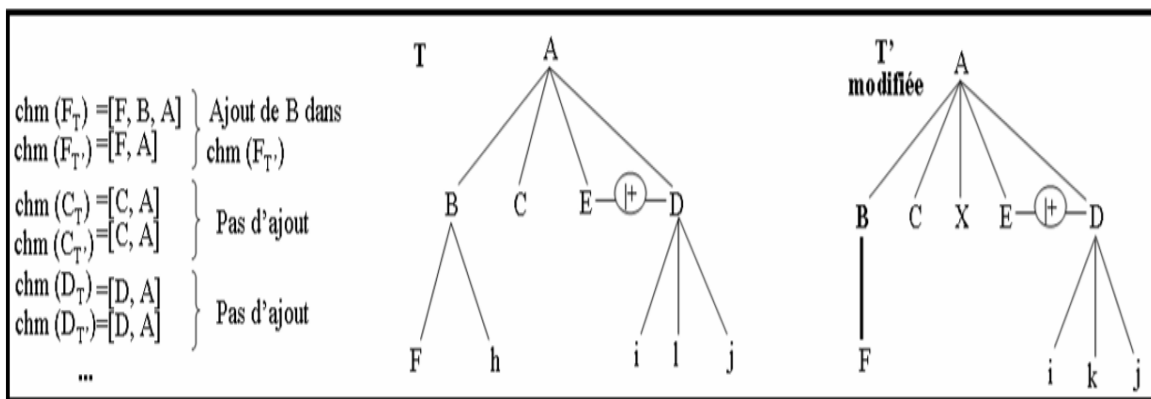


Figure III.22. Exemple d'ajout d'ancêtres

- Ajout de fils :

Pour chaque nœud v' dont l'image v est non vide, à récupérer tous les fils de son père (fils [pere (v')]) et ceux du père de son image (fils [pere (v)]), afin de les comparer et d'ajouter d'éventuels fils selon le même principe que précédemment.

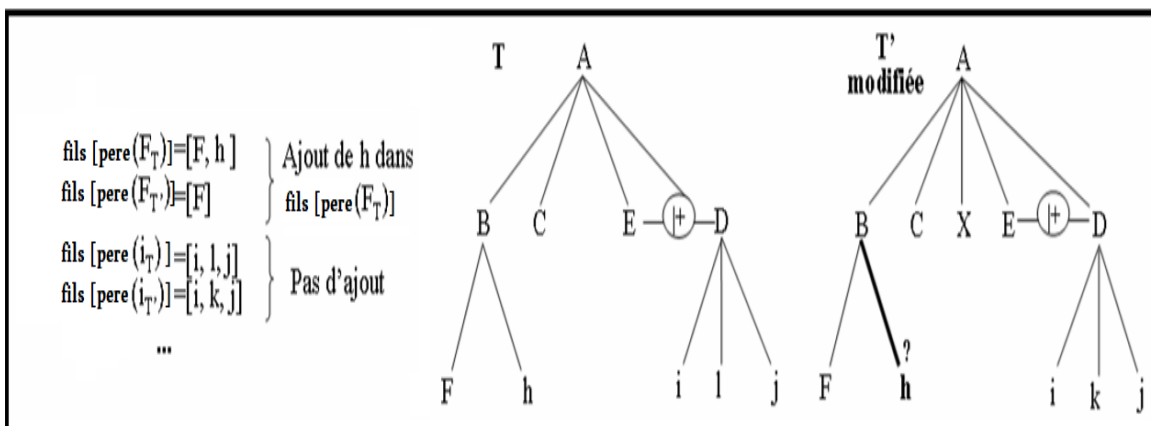


Figure III.23. Exemple d'ajout de fils

❖ Coefficient d'inclusion

Le coefficient d'inclusion C_i permet de vérifier si la structure spécifique du document à insérer est incluse dans la structure générique que l'on tente d'adapter.

Si la valeur C_i égale à 1, la structure est incluse dans la structure générique, sinon ($0 \leq C_i < 1$) les deux structures présentent des différences et un degré de similarité doit être calculé. La formule pour calculer C_i est la suivante :

$$C_i = \frac{\text{card}(v_{EC}) * \text{card}(v''_{EC}) + \text{card}(v_{AM}) * \text{card}(v''_{AM})}{\text{card}(v_{EC})^2 + \text{card}(v_{AM})^2}$$

▪ Calcul de similarité

Il s'agit de déterminer parmi les structures génériques candidates restantes celle qui est la plus proche de la structure spécifique à intégrer, et ça on calculant un degré de similarité **Sim**, ce degré est basé sur le calcul de la distance d'alignement des différents nœuds des deux structures arborescentes. Le calcul de similarité permet également de tester si nous pouvons ou pas ajouter des nœuds complexes dans la structure générique.

*Fonction de similarité **Sim***

$$Sim(T, T') = 1 - \frac{\sum_{v_j} D_{anc}(v_j)}{\sum_{v_j} P_{anc}(v_j)}$$

Où :

$\forall v_j \in V / \text{fils}[v_j] = \emptyset$ (v_j est feuille) et $\forall v_i \in \text{chm}[v_j]$ et

$$D_{anc}(v_j) = \sum_{v_i} D_n(v'_i, v_i)$$

$$P_{anc}(v_j) = \sum_{v_i} P_n(v_i)$$

D_n calcule la distance d'alignement entre deux nœuds :

Soient $v \in V \cup \{v_0\}$, $v' \in V' \cup \{v_0\}$ et $m(v) = v'$, $D_n(v, v') = |P_n(v) - P_n(v')|$ avec $P_n(v_0) = 0$

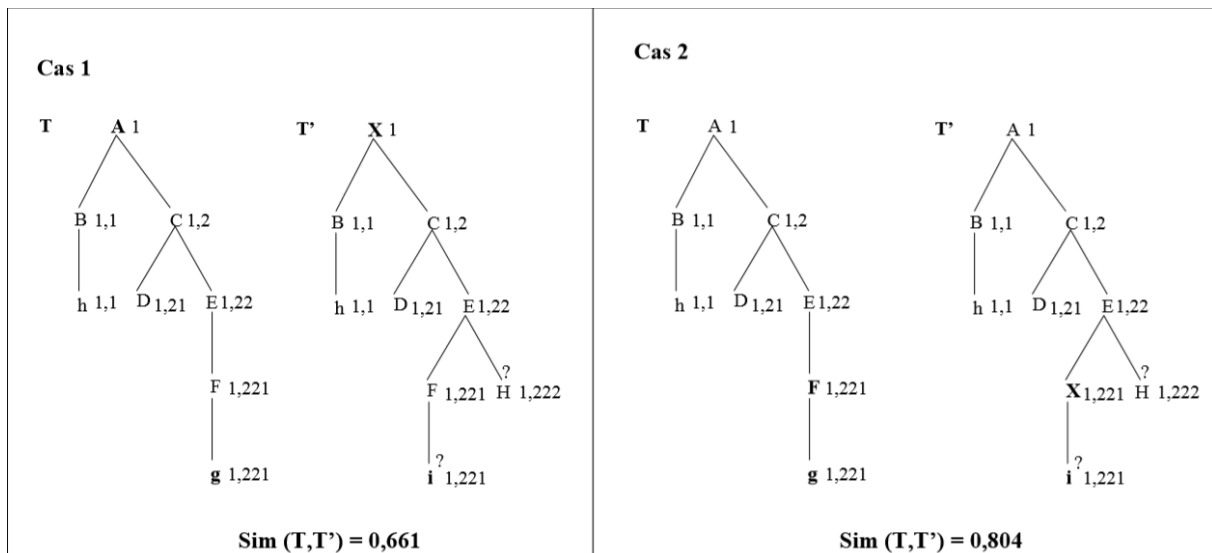


Figure III.24. Exemples de calculs de similarités

III.3.3. Adaptation des structures

L'adaptation des structures vient après avoir trouvé que la structure générique n'est pas identique à la structure spécifique (si les deux structures s'ont identiques la structure spécifique sera intégrée directement sans adaptation), cette adaptation concerne soit la structure générique de la base soit la structure spécifique du document.

- **Adaptation de structure générique**

L'adaptation d'une structure générique se fait de manière automatique, elle consiste à :

- Si une seule structure générique vérifie la contrainte d'inclusion (le coefficient d'inclusion et le calcul de similarité traduisent une proximité suffisante au sens des seuils fixés par expérimentations) les ajouts fictifs de nœuds envisagés dans la phase de comparaison seront concrétisés.
- Si plusieurs structures génériques vérifient la contrainte d'inclusion il faut prendre celle nécessitant un minimum d'ajout de nœuds.
- Si la structure spécifique n'est pas entièrement incluse dans la structure générique, il faut extraire à partir de l'ensemble des structures génériques sélectionnées celle dont le degré de similarité avec la structure du document à intégrer est le plus élevé, puis à comparer ce degré avec un seuil de similarité S_s fixé par expérimentations :
 - Si le degré de similarité est strictement inférieur à S_s , la structure générique en cours ne peut pas être adaptée.
 - Sinon, l'adaptation de la structure générique est possible, il faut effectuer des comparaisons de chemins en commençant de la racine jusqu'aux feuilles, les nœuds occupant la même position dans les deux arborescences seront interconnectés fictivement par l'opérateur "|" dans la structure générique tout en leur rattachant leurs éventuels fils.

La figure suivante montre un exemple d'adaptation d'une structure générique :

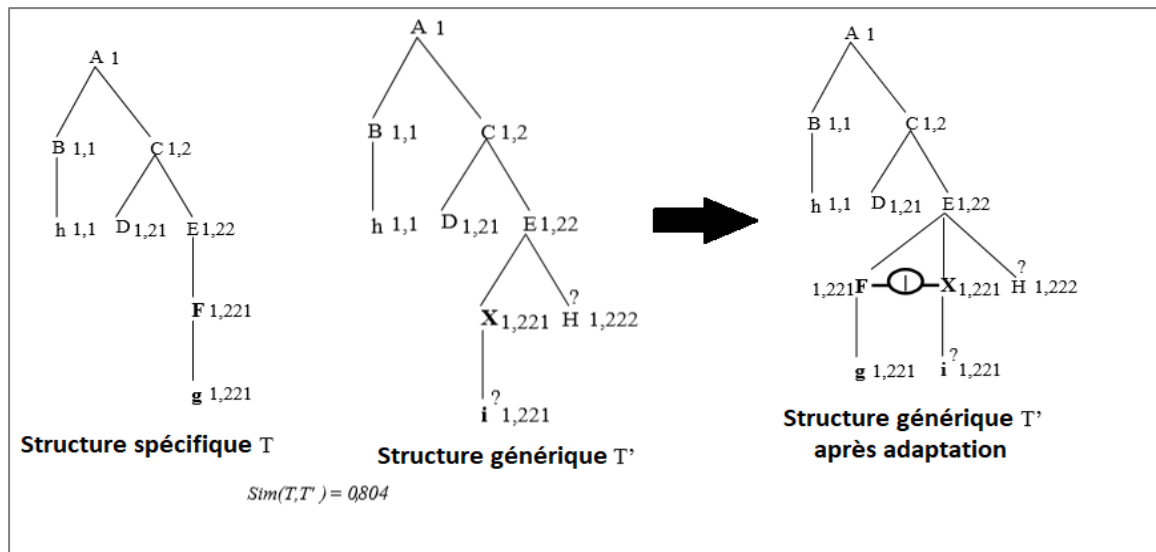


Figure III.25. Exemple d'adaptation de structure générique

▪ Adaptation de structure spécifique

Contrairement à l'adaptation des structures génériques qui consiste uniquement l'ajout des fils ou des pères, l'adaptation des structures spécifiques permet de déplacer, modifier ou supprimer des nœuds. Il est possible de faire une adaptation de structure spécifique même si le coefficient d'inclusion et le calcul de similarité ne traduisent pas une proximité suffisante (par rapport au seuil fixé) si cela est jugé utile par l'utilisateur.

L'adaptation de structure spécifique est manuelle, ce qui permet aux utilisateurs de formuler des règles personnalisées pour assurer la transformation de la structure du document vers une structure générique.

Les règles d'adaptation des structures spécifiques sont les suivantes :

❖ Règles de gestion de noms de balises :

- Règles d'ajout de balises : pour ajouter un nœud, nous devons donner son nom et son emplacement (père et fils, ou père et ordre) comme suit : **Ad** (Structure, Nœud, Père, Fils) ou **Add_leaf** (Structure, Nœud, Père, Ordre)
- Règles de déplacement : pour déplacer un nœud ou une sous-arborescence : **Move** (Structure, Nœud, Père, Ordre)
- Règles de modification : pour changer les noms des nœuds spécifiques : **Rename** (Structure, Ancien_Nom, Nouveau_Nom)
- Règles de suppression : pour éliminer des nœuds : **Remove** (Structure, Nœud)

Un exemple de déploiement de ces règles et dans la figure suivante :

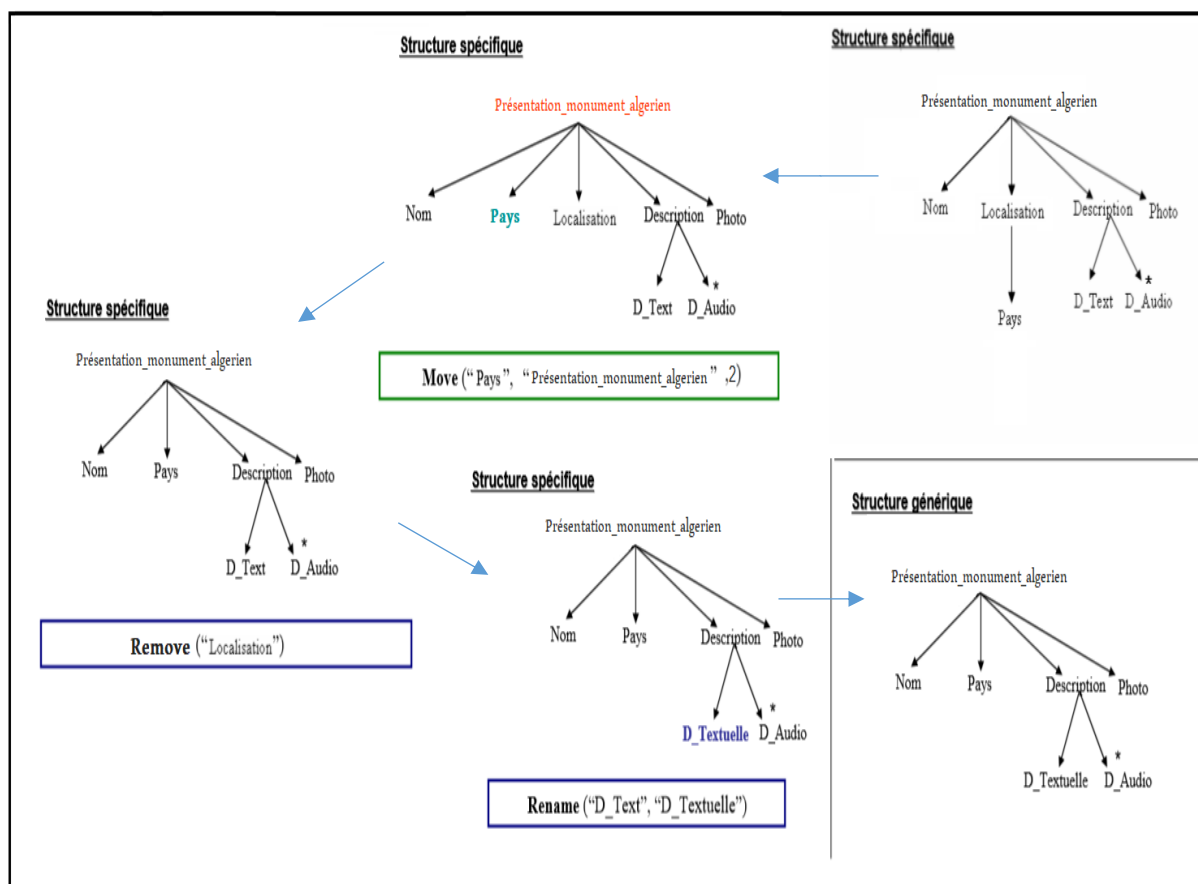


Figure III.26. Déploiement des règles de gestion de noms de balises

❖ Règles basées sur l'analyse de contenu de balises :

- la première règle permet d'ajouter les balises "<Deb-fin...>" (dates de début et de fin) et "<Trans...>" (transcription) au tour des premières transcriptions, elle consiste à : (1) détecter l'occurrence de deux balises qui commencent par <Sync time> et qui sont séparées par une seule ligne, (2) extraire deux paramètres P1 et P2 qui correspondent aux valeurs des attributs "time" de ces deux balises, (3) remplacer la première balise d'entrée par la séquence <Deb-Fin-P>P1_P2</Deb-FinP><Trans> (avec P1 = valeur du paramètre P1 et P2 = valeur du paramètre P2), (4) remplacer la deuxième balise d'entrée par </Trans>
- la deuxième règle s'intéresse au cas particulier de la dernière transcription (qui se retrouve juste avant la balise "</Turn...>"). Cette règle permet de récupérer la date fin de passage (Turn) qui correspond aussi à la fin de cette dernière transcription.

Un exemple d'application de ces règles et dans la figure suivante :

```

...
<Section type="report" topic="to1" startTime="110.268" endTime="129.105">
  <Turn speaker="spk2" startTime="110.268" endTime="129.105">
    <Sync time="110.268"/>
    de San Luiz de Malagnan
    <Sync time="112.017"/>
    avec heu un banjo algérien
    <Sync time="115.193"/>
    moi ma démarche de percussionniste heu quand j'étais au Brésil
    <Sync time="119.559"/>
  ...
  <Sync time="124.149"/>
  J'ai travaillé à fond dans la musique brésilienne je jouais déjà les tablas
</Turn>
</Section>
...

```



```

...
<Section type="report" topic="to1" startTime="110.268" endTime="129.105">
  <Turn speaker="spk2" startTime="110.268" endTime="129.105">
    <Deb-Fin -P> 110.268_112.017 </Deb-Fin-P>
    <Trans>
    de San Luiz de Malagnan
    </Trans>
    <Deb-Fin -P> 112.017_115.193 </Deb-Fin-P>
    <Trans>
    avec heu un banjo algérien
    </Trans>
    <Deb-Fin -P> 115.193_119.559 </Deb-Fin-P>
    <Trans>
    moi ma démarche de percussionniste heu quand j'étais au Brésil
    </Trans>
  ...
  <Deb-Fin -P> 124.149_129.105 </Deb-Fin-P>
  <Trans>
  J'ai travaillé à fond dans la musique brésilienne je jouais déjà les tablas
  </Trans>
</Turn>
</Section>
...

```

Figure III.27. Application sur les règles basées sur l'analyse de contenu de balises

III.3.4. Ajout d'une nouvelle structure générique

Dans le cas où aucune possibilité d'adaptation n'est envisageable, la structure spécifique du nouveau document sera intégrée et stockée dans la base comme une nouvelle structure générique (qui représente une nouvelle classe).

III.4. Exploitation des documents multimédia

Une exploitation efficace d'un entrepôt de documents multimédia se repose sur une recherche simple, flexible et surtout précise. En étudiant le processus de recherche dans [1] et son expérimentation nous avons trouvé qu'il offre des résultats assez précis et que son auteur a bien profité de la fragmentation, car le processus permet de restituer des granules documentaires (éléments, attributs, composants, métadonnées) en réponse à une requête formulée selon un ensemble des mots clés. Il permet de manipuler aussi bien la structure que le contenu (accès sémantique) des documents de l'entrepôt, mais nous avons remarqué qu'il n'exploite la dichotomie entre description structurelle et description des métadonnées qu'après la recherche et pas lors de la recherche elle-même, c'est ce qui est considéré comme un manque de flexibilité.

Nous cherchons à bénéficier de la dichotomie entre description structurelle et description des métadonnées lors de la recherche en proposant une démarche pour rechercher les documents dans la base de documents multimédia selon différents modèles de description. Nous proposons aux utilisateurs au début de la phase de recherche un mode d'utilisation "Recherche précise", ce mode offre la possibilité d'effectuer deux types de recherche : (1) une recherche dans les descriptions structurelles, (2) une recherche dans les descriptions des métadonnées.

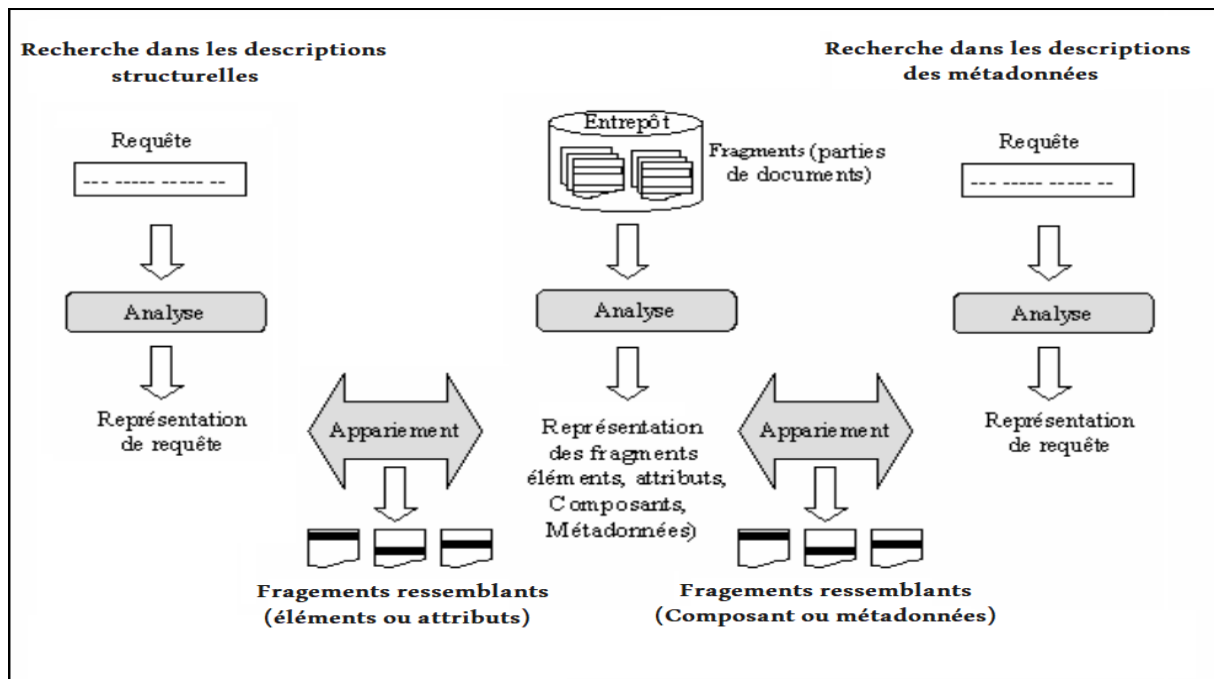


Figure III.28. Mode de recherche précise

Dans le chapitre VI nous allons voir le déploiement de ces propositions, aussi nous discuterons les résultats obtenus.

III.5. Conclusion

L'ensemble des propositions décrites dans ce chapitre vise une classification et une gestion optimale des documents multimédias hétérogènes.

En premier lieu, nous avons porté notre attention sur la modélisation des documents multimédia. Le modèle de représentation des documents adapté comporte la gestion des droits d'utilisation définis par les auteurs des documents, cette fonctionnalité dans ce modèle permet de régler un grand problème posé depuis longtemps dans les modèles de représentation, ce problème est resté inaperçu par ce qu'il n'apparaît qu'en phase d'exploitation.

En deuxième lieu, nous avons mis en place une démarche de recherche qui offre la possibilité de rechercher les documents dans l'entrepôt de document multimédia selon différents critères (selon les modèles de description), cela nous permettrait d'effectuer des recherches flexibles, fines et plus précises sur les structures et les contenus documentaires.

Nous ne prétendons pas à travers ce travail d'adaptation présenter un modèle de représentation parfait, et une démarche de recherche idéale, mais ça ne nous empêche pas de dire que ce travail porte des améliorations éminentes dans l'approche présentée par l'auteur dans [1].