



HAL
open science

Accessing and using complex multimedia documents in a digital library

Anh Tuan Ly

► To cite this version:

Anh Tuan Ly. Accessing and using complex multimedia documents in a digital library. Other [cs.OH].
Université Paris Sud - Paris XI, 2013. English. NNT : 2013PA112123 . tel-00871651

HAL Id: tel-00871651

<https://theses.hal.science/tel-00871651>

Submitted on 10 Oct 2013

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

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



UNIVERSITE PARIS-SUD

ÉCOLE DOCTORALE : *Informatique de Paris-Sud*
Laboratoire de *Recherche en Informatique*

DISCIPLINE Informatique

SYNTHÈSE DE LA THÈSE

soutenue le 09/07/2013

par

Anh Tuan LY

Accès et utilisation de documents multimédia
complexes dans une bibliothèque numérique

Directeur de thèse :

Nicolas SPYRATOS

Professeur, Université Paris Sud, France

Composition du jury :

Président du jury :

Rapporteurs :

Examineurs :

Philippe RIGAUX
Dominique LAURENT
Peter STANCHEV
Chantal REYNAUD
Francois GOASDOUE

Professeur, CNAM, France
Professeur, Université Cergy-Pontoise, France
Professeur, Kettering University, USA
Professeur, Université Paris Sud, France
MdC Habilité, Université Paris Sud, France

1. Introduction et état de l'art

Il existe de nombreuses définitions de la bibliothèque numérique. Chaque définition offre une perspective différente sur le concept. Aux fins de cette thèse, nous considérons d'une bibliothèque numérique comme un système d'information qui devrait soutenir le stockage des ressources numériques et à offrir des services pour accéder et manipuler ces ressources [1][2]. Ressources gamme numérique d'objets multimédias simples (textes numériques, images, vidéos et autres) à des objets composites de toutes sortes.

Les services de base attendus d'une bibliothèque numérique devraient permettre aux utilisateurs d'effectuer les tâches suivantes:

- *identifier* une ressource d'intérêt, dans le sens de l'assignation d'un identifiant;
- *accéder* à une ressource;
- *décrire* une ressource d'intérêt selon un vocabulaire;
- *créer* une ressource numérique complexe en ré-utilisant les ressources existantes;
- *découvrir* des ressources d'intérêt en fonction de leurs métadonnées.

Afin de concevoir et mettre en œuvre un système de bibliothèque numérique prenant en charge les services ci-dessus, il faut un modèle de données et un langage de requête.

Dans cette thèse, notre point de départ est le modèle et le langage de requête mis au point par notre équipe de recherche dans le cadre de trois projets européens:

- 2002-2004: SeLeNe: Self Réseaux d'apprentissage en ligne (Contrat IST-2001-39045) (<http://www.dcs.bbk.ac.uk/selene/>).
- 2004-2007: FP6 DELOS réseau d'excellence pour les bibliothèques numériques (Contrat: G038-507618) (<http://www.delos.info/partners.html>)
- 2010-2012 Projet ASSETS: Recherche avancée Services et Amélioration des solutions de technologies pour la bibliothèque numérique européenne (CIP-ICT PSP-2009-3, Accord de subvention n° 250527) (<http://www.assets4europeana.eu/>)

Les aspects théoriques du modèle et le langage de requêtes ont été présentées dans [3][4]. L'objectif de cette thèse est double:

1/ Montrer la faisabilité du modèle théorique en concevant et réalisant une forme simplifiée du modèle et le langage de requête. Tel était l'objectif principal.

2/ Contribuer au modèle théorique elle-même par l'étude d'un important service communautaire, la génération à savoir le contenu grâce à la réutilisation.

Le modèle théorique utilisé dans cette thèse est inspiré par le Web, qui constitue une solide base universellement acceptée pour les notions et les services attendus d'une bibliothèque numérique. Plus précisément, la notion de ressource a été introduite dans le contexte du Web et constitue la pierre angulaire de l'architecture du Web [5], ainsi que les notions d'identifiant et de représentation d'une ressource. En fait, aujourd'hui, le Web prend en charge l'identification des ressources (par URI), leur accès (via des serveurs Web), leur création (sous la forme de documents HTML) et leur découverte (par le biais des moteurs de recherche).

Toutefois, lorsque l'URL est utilisée comme un identifiant, l'accès et l'identification sont brouillées par le Web, comme l'URL fournit à la fois de la référence à la ressource et un chemin d'accès. Plus important encore, une notion de base des bibliothèques numériques est absent de l'architecture du Web actuel, à savoir la notion de description d'une ressource. Le Web sémantique¹ fournit langues pour exprimer descriptions, et des mécanismes permettant d'interroger ces descriptions, mais ces langues et de mécanismes d'interrogation ne jouent pas un rôle significatif dans le contexte du Web, à l'heure actuelle, parce que dans l'architecture du Web actuel, il n'existe aucun moyen pour relier les ressources et les descriptions.

Notre modèle tente de remédier à ces inconvénients par:

- (a) faire une distinction claire entre l'identification d'une ressource et l'accès;
- (b) fournir un moyen de descriptions de modélisation des ressources indépendantes;
- (c) concernant les descriptions des ressources décrites et
- (d) fournir un langage de requête pour la découverte de ressources en fonction de leurs descriptions.

Dans notre modèle, nous utilisons des relations d'exprimer les faits de base stockés dans une bibliothèque numérique, et de s'appuyer sur une théorie du premier ordre pour obtenir des informations implicites dans les faits donnés, sous certains axiomes. Le choix de la logique est motivée par le désir de généralité, qui comprend la liberté de toute contrainte technologique.

Pour démontrer la faisabilité du modèle développé par notre équipe de recherche et son adéquation pour des applications pratiques, nous offrons une traduction complète du modèle de RDF [6] et du langage de requêtes SPARQL à [7]. Le choix de RDF est dû au fait qu'il s'agit d'un langage de représentation généralement accepté dans le cadre des bibliothèques numériques et le Web sémantique.

Dans ce contexte, comme indiqué précédemment, l'un des principaux objectifs de cette thèse est de concevoir et mettre en œuvre une forme simplifiée du modèle mis au point par notre équipe

¹ SemanticWeb - W3C. www.w3.org/standards/semanticweb/

de recherche. Un système de bibliothèque numérique RDF et SPARQL-fondé a été développé en utilisant les technologies Java, une API Java pour RDF appelé Jena² et le cadre de Google Web Toolkit³. L'architecture du système est constitué de:

- *Une couche de stockage* qui comprend un stockage pour stocker les ressources numériques à l'intérieur de la bibliothèque numérique, une table de référence pour stocker et gérer des associations entre des identificateurs et des ressources numériques, et une base de méta-données pour stocker les métadonnées et le contenu des ressources numériques. Dans la mise en œuvre effective, les faits triplet RDF formats sont stockés dans un magasin RDF.
- *Une couche de logique métier* qui contient la logique métier de l'application. Il se compose de quatre modules de base comme suit: Module triple manipulation, le module d'évaluation requête, les moteurs de traducteur, module de gestion de persistance.
- *Une couche de service* qui commande la communication entre la logique de client, et la logique de serveur, par l'exposition d'un ensemble de services (exploitation) pour les composants côté client. Par exemple, les services de manipulation sous-jacentes telles que la création, l'édition, mise à jour et la suppression des prédicats, ainsi que des services requête qui fournissent les méthodes de base pour l'évaluation des requêtes conjonctives.
- *Une interface utilisateur* qui permet à l'utilisateur de la bibliothèque numérique: recherche et/ou parcourir les collections de ressources; voir les enregistrements de métadonnées qui décrivent une ressource; créer, lire, mettre à jour et supprimer des métadonnées des ressources, ainsi que des requêtes conjonctives express dans notre langage de requête pour découvrir des informations à partir de la bibliothèque numérique.

Pendant les travaux de thèse, le système a été construit, testé et déboguer localement, puis déployé sur Google App Engine (GAE)⁴. Dans l'avenir, il peut être étendu pour devenir un système de gestion de bibliothèque numérique à part entière.

2. Un modèle de données pour les bibliothèques numériques

Officieusement, nous considérons d'une bibliothèque numérique comme un système d'information qui se compose de deux éléments: (a) un ensemble de ressources numériques que les magasins de bibliothèques numériques (objets multimédias généralement complexes), et (b) les connaissances sur les ressources qui sont stockées dans le numérique bibliothèque, et

² Apache Jena. jena.apache.org/

³ Google Web Toolkit (GWT). developers.google.com/web-toolkit/

⁴ Google App Engine. developers.google.com/appengine/

éventuellement sur les ressources qui se trouvent en dehors de la bibliothèque numérique, mais sont pertinents aux fins de la bibliothèque numérique. Chacun de ces deux éléments peuvent être absents de la bibliothèque numérique. Par exemple, on peut concevoir une bibliothèque numérique stocker seulement un ensemble de ressources numériques, sans enregistrer aucune connaissance sur les ressources. Un album de photos personnelles téléchargées à partir d'un appareil photo numérique sans aucune annotation, est un exemple d'une telle bibliothèque - même si la seule chose que l'utilisateur peut faire avec elle, consiste à parcourir (c'est à dire l'accès) les images.

De même, on peut concevoir une bibliothèque numérique qui contient seulement des connaissances sur les ressources de stockage sans aucune ressource. Un exemple notable d'une telle bibliothèque numérique Europeana⁵ est, la bibliothèque numérique européenne, qui (au moins dans sa forme actuelle) enregistre uniquement des métadonnées.

Dans cette thèse, nous supposons que d'une bibliothèque numérique stocke à la fois, des ressources numériques et des connaissances sur les ressources stockées (ainsi que les connaissances sur les ressources non stockés).

Afin d'exprimer les connaissances sur les ressources , nous avons besoin de moyens pour y faire référence. Ces moyens sont les identifiants, nous appelons les identificateurs des ressources numériques utilisés comme des références à d'autres ressources dans une bibliothèque numérique. "Un identifiant incarne l'information nécessaire pour distinguer ce qui est identifié à partir de toutes les autres choses dans son champ d'identification" [8].

Pour atteindre son but, un identifiant ne peut pas se référer à deux ressources différentes en même temps. Ceci met en place une fonction à partir d'identifiants aux ressources visées. Nous appellerons cette fonction, la fonction de référence de la bibliothèque numérique. Cependant, toutes les associations de référence peuvent être stockés dans la bibliothèque numérique. Nous nous référons à l'ensemble des associations de référence stockées que le tableau de référence de la bibliothèque numérique.

La table de référence est l'une des deux principales composantes d'une bibliothèque numérique. L'autre composante est la base de métadonnées. La base de métadonnées d'une bibliothèque numérique contient deux types de connaissances que l'utilisateur peut exprimer d'une ressource: les métadonnées et le contenu de la ressource.

- Les métadonnées d'une ressource consiste en une ou plusieurs descriptions de la ressource.
- Le contenu d'une ressource consiste en une ou plusieurs autres ressources qui composent la ressource.

⁵ www.europeana.eu

Afin de s'acquitter de ses tâches, la bibliothèque numérique doit constituer, stocker et traiter un certain nombre d'informations sur les ressources de son champ d'application. Nous ne sommes pas intéressés à la provenance de cette information, comment il est capturé, ou comment il arrive à la bibliothèque numérique. Nous ne nous intéressons à la modélisation de l'information elle-même sous la forme de ce que nous appelons une description. Par ailleurs, nous découplons descriptions des ressources dont ils pourraient être associés à, en fournissant un mécanisme distinct pour associer descriptions des identifiants de ressources.

Dans notre modèle, nous descriptions de modèles à l'aide de deux relations DescCl(d, s, c) qui exprime le fait que la classe c sur schéma s appartient à la description d, et DescPr(d, s, p, i) qui exprime le fait que la paire propriété-valeur (p, i), où p est plus schéma s, appartient à d. Alors, l'association entre les ressources et leur description est modélisé par la relation DescOf(d, i) qui exprime le fait que d identifie une description de la ressource identifiée par i. Nous saisissons également la connaissance dans un schéma en utilisant d'autres relations.

En informatique, le terme "contenu" est utilisé avec beaucoup de significations différentes. Dans ce travail, par le contenu d'une ressource r, nous entendons l'ensemble des autres ressources qui composent r d'un point de vue applicatif, chaque ressource est appelée une partie de r. Par exemple, chaque chapitre d'un livre (vu comme une ressource à part entière) peut être considéré comme une partie de ce livre. De même, chaque tableau dans une exposition de peintures peut être considéré comme une partie de l'exposition.

Pas toutes les ressources peuvent avoir des parties. Nous supposons que les classes, les propriétés, les schémas et les descriptions ne peuvent pas avoir des parties, ni être parties d'autres ressources. Nous appellerons tous les autres types de ressources composable. Notez que aussi les ressources physiques sont composables (à travers leurs identifiants).

Dans notre modèle, le contenu est représenté par la PartOf de relation (i, j) qui exprime le fait que i identifie une ressource composable qui fait partie de la ressource composable identifié par j.

3. Technologies de fond

Dans notre travail, nous avons utilisé les normes et les technologies suivantes pour la mise en œuvre de l'application:

- AJAX, la technologie de pointe dans le Web 2.0.
- La technologie Java Servlet, la solution Java pour fournir des services basés sur le Web, qui est utilisé pour le développement de notre application.

- RDF, SPARQL et mise à jour SPARQL qui définissent le modèle de données, un langage de requête et de son extension pour le Web sémantique, respectivement.
- Triplestore, une base de données spécialement conçu pour le stockage et la récupération de Resource Description Framework métadonnées (RDF).
- Jena, un framework Java pour créer des applications Web sémantique.
- Google Web Toolkit, qui a été utilisé pour implémenter le côté client du système.

4. Mise en œuvre du modèle basé sur RDF et SPARQL

Jusqu'à présent, nous avons adopté une approche logique pour la modélisation d'une bibliothèque numérique. Ce choix était délibéré afin d'être en mesure d'exprimer les concepts de base des bibliothèques numériques sans être contraints par des considérations techniques. Cependant, l'objectif de notre travail est de contribuer à la technologie des bibliothèques numériques. Par conséquent, nous considérons maintenant comment le modèle peut être mis en œuvre. Nous rappelons que d'une bibliothèque numérique se compose de deux parties, la table de référence et la base de métadonnées.

En ce qui concerne la mise en place de la table de référence, on remarque que son entretien est généralement assurée par le système d'exploitation, et en particulier par le système de fichiers. Par conséquent, nous ne considérons pas le maintien de la table de référence plus loin. Au contraire, nous nous concentrons notre attention sur la mise en œuvre de la base de métadonnées, et nous considérons deux scénarios différents.

- Le premier scénario consiste dans la mise en oeuvre de la base de méta-données en tant que base de données relationnelles et le calcul de l'achèvement d'une bibliothèque numérique par l'intermédiaire d'un système d'enregistrement de données. Interrogation peut être implémentée en associant notre langage de requête SQL pour, et d'exécuter les requêtes résultant sur la base de données relationnelle. Cette mise en œuvre est conceptuellement simple (comme la base de métadonnées est en fait une base de données relationnelle).
- Le deuxième scénario comprend la mise en oeuvre de la base de métadonnées comme un graphe RDF, et en utilisant un moteur d'inférence RDF pour calculer l'achèvement de la bibliothèque numérique. Interrogation peut être implémentée en associant notre langage d'interrogation SPARQL pour. Cette mise en œuvre est conceptuellement beaucoup plus exigeante, car elle nécessite de traduire les relations et les axiomes de notre modèle de RDF.

Le premier scénario exploite la technologie relationnelle bien établie, y compris le traitement de la requête optimisée de SQL. Cela garantit l'évolutivité et la robustesse de la mise en œuvre. Le

deuxième scénario bénéficie du fait que RDF est un langage de représentation généralement accepté dans le cadre des bibliothèques numériques et le Web sémantique. Bien que RDF n'a pas encore atteint la maturité de la technologie relationnelle, outils de gestion des graphes RDF ont été intensément étudié et développé dans la dernière décennie, et atteignent maintenant un niveau élevé de maturité technologique⁶. Ces outils comprennent des systèmes de la persistance de grands graphes RDF dans le stockage secondaire (dite magasins triples), moteurs d'inférence RDF et les moteurs de traitement des requêtes optimisées pour SPARQL⁷.

Le choix d'une de ces deux scénarios dépend de facteurs contextuels. Comme nous l'avons déjà observé, la traduction du modèle d' relationnel est simple, tout en traduisant à partir du modèle de RDF est plus exigeante. Cependant, dans cette thèse, nous choisissons le second scénario en raison de la capacité d' assurer l'interopérabilité entre les applications de RDF et la capacité de permettre le traitement automatisé des ressources Web de celui-ci.

5. La mise en œuvre effective du modèle

Pour démontrer la faisabilité du modèle théorique, nous avons conçu et mis en œuvre un prototype qui est une forme simplifiée du modèle et le langage de requête.

Notre prototype illustre le modèle théorique basé sur les aspects suivants:

- Il prend en charge différents types d'identificateurs du modèle
- Elle soutient pleinement l'ensemble des relations dans le modèle qui servent à la description des modèles de ressources
- Il prend en charge différents types de variables du langage de requête
- Il prend en charge les requêtes conjonctives simples du langage de requête

Dans l'aspect pratique, notre prototype soutient la mise en œuvre du modèle basé sur RDF et SPARQL comme suit:

- Il représente les identificateurs du modèle en deux formes: la forme originale et le formulaire d'URI basée sur chacun des schémas spécifiques
- Il supporte faits traduisant représentés par les relations dans le modèle en triplets RDF pour les stocker dans le magasin RDF, ainsi que, récupérer ces données à partir du magasin RDF pour les afficher sur la lecture/l'écriture des interfaces

⁶ See, for instance: www.w3.org/2001/sw/wiki/Tools

⁷ e.g. esw.w3.org/SparqlImplementations

- Il prend en charge la traduction des requêtes conjonctives simples du langage de requête en SPARQLs et les exécute sur le store RDF. Dans lequel, les résultats retournés sont un ensemble d'identificateurs pris en charge par notre prototype.

Cependant, notre prototype prend juste requêtes conjonctives, tandis que son interface ne permet aux utilisateurs d'effectuer des requêtes atomiques. Le tableau de référence de notre prototype qui stocke les associations entre les identifiants et les ressources du système est statique. En outre, bien que l'utilisation des URI comme identifiants du modèle rend la bibliothèque numérique "cohérente", notre prototype n'a pas mis en place une base de règles d'axiomes pour assurer l'intégralité de la bibliothèque numérique encore.

Notre système peut être utilisé par les fournisseurs de contenus pour gérer la connaissance des ressources qui sont soit stockées dans la bibliothèque numérique, ou résident à l'extérieur de la bibliothèque numérique, il peut également être utilisé par l'utilisateur final pour l'interrogation de la connaissance de la bibliothèque numérique. Construit comme une application web, le système adopte le cadre de Google Web Toolkit (GWT) qui facilite le développement d'applications Web comme des applications de bureau, d'effectuer des opérations de logique métier côté serveur, ainsi que sur le côté client. La logique côté client fonctionne dans le navigateur Web s'exécutant sur l'ordinateur local de l'utilisateur, alors que la logique côté serveur fonctionne sur le serveur Web qui héberge l'application.

Le système a été conçu, testé et débogué localement, puis déployé sur Google App Engine (GAE). Le système offre toutes les fonctionnalités qui ont été demandées à l'heure de départ de ce travail et sont décrits en détail dans cette thèse. À l'avenir, le système peut être étendu pour devenir un système de gestion de bibliothèque numérique à part entière.

6. Contribution à la génération de contenu par la réutilisation

Génération de contenu par la réutilisation est le processus de création, le stockage et la réutilisation des documents composites virtuels. Ces documents virtuels peuvent ensuite être concrétisés afin qu'ils puissent être imprimés dans le format du papier, si nécessaire. Tout ce travail peut être compris comme l'élaboration d'un système de gestion de documents pour notre bibliothèque numérique. Par conséquent, notre équipe de recherche a mis au point un modèle de gestion des métadonnées qui sera utilisé dans le cadre de la gestion des métadonnées de documents virtuels [9][10]. En outre, des algorithmes et des outils logiciels aussi besoin d'être étudiés et développés pour soutenir la matérialisation d'entre eux.

Cette section présente notre contribution à la génération de contenu par les réutiliser. C'est surtout le travail théorique effectué pour compléter le modèle et le langage de requête mis au point par notre équipe de recherche. Son incorporation dans le système mis en place est laissée aux travaux futurs.

Un utilisateur crée un nouveau document à partir de zéro ou en modifiant et en réutilisant les documents existants; dans le premier cas, le document est appelé atomique alors que dans le second composite. Dans notre approche, nous supposons qu'un document composite a une structure arborescente, dans lequel chaque noeud intérieur est un document composite et chaque noeud feuille est un document atomique, d'ailleurs, chaque noeud (que ce soit intérieur ou feuilles) dispose d'une description du contenu, il représente. Un document composite est un document virtuel en ce qu'il décrit simplement le contenu des documents du composant et la façon dont ils sont structurés de manière à compenser le document composite.

Dans ce contexte, nous discutons de la façon de concevoir et d'implémenter une interface, avec des algorithmes et d'outils logiciels permettant à l'utilisateur d'une bibliothèque numérique pour:

1/ Choisir un ensemble de n documents numériques à partir de la bibliothèque et/ou à son bureau, par exemple

$$\langle dd_1, descr_1 \rangle, \langle dd_2, descr_2 \rangle, \dots, \langle dd_n, descr_n \rangle$$

où dd_i est l'identifiant du document, et $descr_i$ est la description du document (c'est à dire un ensemble de termes d'une taxonomie).

2/ Structure des documents dans une hiérarchie avec le document $\langle dd, descr \rangle$ composite en tant que root

3/ En déduire la description documents composite $descr$ de l'expressions $descr_1, descr_2, \dots, descr_n$ des documents composant

4/ Générer la table des matières et l'index du document composite

5/ Stocker le document $\langle dd, descr \rangle$ composite dans la bibliothèque numérique

6/ Matérialiser le document composite à volonté (par exemple produire une "version papier" de celui-ci)

Notez que, deux matérialisations en deux points différents dans le temps peuvent produire différentes versions papier, reflétant les changements possibles dans les documents de composants pendant le temps écoulé.

Nous avons conçu des algorithmes qui permettent de générer la table des matières et l'index d'un document composite virtuel. Nos algorithmes décrivent comment on peut dériver une "version

papier" d'un document composite virtuel, un processus que nous appelons "matérialisation" du document composite virtuel.

7. Conclusions et Travaux futurs

Inspiré par l'architecture du Web, notre équipe de recherche a mis au point un modèle de données pour les bibliothèques numériques basées sur trois concepts fondamentaux: identifiant, ressources et description. Identificateurs désignent uniquement les ressources et les descriptions décrivent les propriétés des ressources. Les trois concepts de base de notre modèle parallèle les concepts fondamentaux de l'architecture du Web, qui sont: URI, ressources et représentation.

Le modèle généralise l'architecture du Web dans trois directions différentes:

- (a) l'ensemble des identificateurs n'est pas fixée à priori,
- (b) l'ensemble des identificateurs change dynamiquement au cours du temps, et
- (c) une description peut inclure non seulement les représentations d'une ressource, mais également d'autres propriétés.

En outre, dans le modèle, les descriptions sont définies indépendamment des ressources dont ils pourraient être associés à, alors que dans le Web il ya une dépendance stricte entre une ressource et ses représentations. Bien que rien dans l'architecture Web interdit d'utiliser la représentation d'une ressource aussi une représentation à une autre ressource, cette fonctionnalité n'est pas explicite dans le Web.

Dans le modèle, les relations sont utilisées pour exprimer les faits de base stockés dans une bibliothèque numérique, et le modèle est formalisée comme une théorie du premier ordre. Les axiomes de la théorie donnent la sémantique formelle des notions de modèle, et en même temps, donnent une définition de la connaissance qui est implicite dans une bibliothèque numérique. La théorie est ensuite traduite en un programme d'enregistrement de données qui, compte tenu d'une bibliothèque numérique, permet de compléter la bibliothèque numérique de la connaissance implicite.

Pour démontrer la pertinence du modèle pour des applications pratiques, nous offrons une traduction complète du modèle de RDF et le langage de requêtes SPARQL pour. En outre, nous introduisons le vocabulaire RDFDL, qui s'étend le vocabulaire de RDFS avec quatre nouveaux symboles qui sont nécessaires pour traduire une bibliothèque numérique en RDF.

Pour démontrer la faisabilité du modèle théorique, nous avons conçu et mis en œuvre un prototype qui est une forme simplifiée du modèle et le langage de requête. Construit comme une application web, le système adopte GWT, le cadre de Google Web Toolkit qui facilite le

développement d' applications Web comme des applications de bureau, d'effectuer des opérations de logique métier côté serveur, ainsi que sur le côté client. La logique côté client fonctionne dans le navigateur Web s'exécutant sur l'ordinateur local de l'utilisateur, alors que la logique côté serveur fonctionne sur le serveur Web qui héberge l'application.

Notre application est utilisée essentiellement pour faciliter la création, la récupération, mise à jour et de suppression des prédicats par les fournisseurs de contenu et de fournir aux utilisateurs finaux des services de base pour l'évaluation des requêtes conjonctives. Sa fonctionnalité principale comprend les opérations suivantes:

- Créer/supprimer/modifier/visualiser des prédicats
- Métadonnées de requête dans la base de métadonnées
- Créer/supprimer/modifier/gérer les utilisateurs

Pendant les travaux de thèse, le système a été construit, testé et déboguer localement, puis déployé sur Google App Engine (GAE). Dans l'avenir, il peut être étendu pour devenir un système de gestion de bibliothèque numérique à part entière.

Dans notre implémentation, nous avons utilisé les URI comme identifiants et donc nous n'avons pas de problème de cohérence. Toutefois, si nous voulons utiliser des objets numériques en général comme des identifiants, nous devons concevoir des algorithmes de vérification de la cohérence de la bibliothèque numérique lors des mises à jour.

Un autre aspect de notre application qui a besoin d'autres travaux sont exhaustivité. Une bibliothèque numérique est complète si sa base de métadonnées répond à trois exigences de base, comme suit: il comprend toutes les connaissances explicitement donné par les utilisateurs, il répond à tous les axiomes du modèle, et il ne comprend pas d'autres connaissances en dehors de la connaissance satisfaire les deux précédents conditions. Pour assurer l'intégralité d'une bibliothèque numérique, nous devons mettre en place un moteur d'inférence qui prend en charge tous les axiomes du modèle et permet au système de générer de nouvelles connaissances fondées sur les axiomes et les connaissances existantes dans la bibliothèque numérique.

Références

- [1] L. Candela, D. Castelli, N. Ferro, Y. Ioannidis, G. Koutrika, C. Meghini, P. Pagano, S. Ross, D. Soergel, M. Agosti, M. Dobрева, V. Katifori, H. Schuldt. The DELOS Digital Library Reference Model - Foundations for Digital Libraries. DELOS Network of Excellence on Digital Libraries, 2007.
- [2] L. Candela, D. Castelli, Y. Ioannidis, G. Koutrika, P. Pagano, S. Ross, H.-J. Schek, H. Schuldt, and C. Thanos. Setting the Foundations of Digital Libraries: The DELOS Manifesto. D-Lib Magazine, Vol. 13 No. 3/4, March/April 2007.
- [3] Carlo Meghini, Nicolas Spyrtos, Tsuyoshi Sugibuchi, and Jitao Yang. A Model for Digital Libraries and its Translation to RDF. *Submitted* for publication.
- [4] Carlo Meghini, Nicolas Spyrtos, and Jitao Yang. A data model for digital libraries. International Journal on Digital Libraries, Volume 11 Number 1, pp.41-56, March 2010.
- [5] Ian Jacobs and Norman Walsh. Architecture of the World Wide Web, Volume One. W3C Recommendation, WWW Consortium, December 2004.
<http://www.w3.org/TR/webarch/>.
- [6] Graham Klyne and Jeremy J. Carroll. Resource description framework (RDF): Concepts and abstract syntax. W3C Recommendation, WWW Consortium, February 2004.
<http://www.w3.org/TR/rdf-concepts/>.
- [7] Eric Prud'hommeaux and Andy Seaborne. SPARQL Query Language for RDF. W3C Recommendation, January 2008. <http://www.w3.org/TR/rdf-sparql-query/>.
- [8] RFC 3986 - Uniform Resource Identifier (URI): Generic Syntax.
<http://tools.ietf.org/html/rfc3986/>.
- [9] Rigaux, P., Spyrtos, N.: Metadata inference for document retrieval in a distributed repository. In: Maher, M.J. (ed.) ASIAN 2004. LNCS, vol. 3321, pp. 418–436.
- [10] Tsuyoshi Sugibuchi, Ly Anh Tuan, Nicolas Spyrtos: “Metadata Inference for Description Authoring in a Document Composition Environment” IRCDL 2012 - 8th Italian Research Conference on Digital Libraries, Bari, Italy, February 9-10, 2012.