Projet ModRef : Migration de Données vers des Triplestores CIDOC-CRM

Pascaline Tchienehom

Projet ModRef : Migration de Données vers des Triplestores CIDOC-CRM

Pascaline Tchienehom [1]

Université de Paris 10 - Labex "Les passés dans le présent",
200 Avenue de la République, 92000 Nanterre, France
pkenfack@u-paris10.fr

ABSTRACT. *ModRef is a project from the laboratory Labex "Les passés dans le présent", which coordinates various projects on digital humanities. ModRef focuses more precisely on the semantic web and linked open data. The goal is to move heterogeneous data into triplestores also called data warehouses or collections of RDF files in order to improve the sharing, exchange and discovery of new knowledge. For this purpose, the CIDOC-CRM norm has been chosen since it is, at present, the reference for the semantic description of museographic or cultural heritage data. In order to realise the proof of concept of ModRef, a general architecture has been defined, a semantic modelling and data mapping of selected sub-projects of ModRef have been proposed, triplestores have also been created. A web application has been implemented and deployed. This web application describes the ModRef project, as well as it enables visualising, querying and exploring created triplestores.*

RÉSUMÉ. *ModRef est un projet du laboratoire Labex "Les passés dans le présent" qui accompagne divers projets sur des problématiques relatives aux humanités numériques. Le projet ModRef s'intéresse spécifiquement au web sémantique et aux données ouvertes et liées. Le but de ce projet est de réaliser une migration de données hétérogènes vers des triplestores encore appelés entrepôts ou collections de fichiers RDF afin d'améliorer le partage, l'échange et la découverte de nouvelles connaissances. Pour ce faire, la norme CIDOC-CRM a été choisie car elle est actuellement la norme de référence pour la description sémantique de l'information muséographique ou d'héritage culturel. Afin de réaliser la preuve conceptuelle de ModRef, une architecture générale a été définie, une modélisation sémantique et un alignement des données des trois sous projets pilotes de ModRef ont été proposés, une migration des données vers des triplestores a également été effectuée. Une application web a été développée et déployée. Cette application web décrit le projet ModRef et permet également de consulter et d'interroger les triplestores créés.*

KEYWORDS: *Digital Humanities, Semantic Web, Triplestores, CIDOC-CRM, Linked Open Data.*

MOTS-CLÉS : *Humanités Numériques, Web Sémantique, Triplestores, CIDOC-CRM, Données ouvertes et liées.*

1. Introduction

Le Labex "Les passés dans le présent" accompagne de nombreux projets en Sciences Humaines et Sociales (SHS) sur des problématiques relatives aux humanités numériques (Oldman *et al.*, 2014) : de la dématérialisation des données à la description structurée voire sémantique de ces dernières. Le projet ModRef (Modélisation, Référentiels et Culture Numérique) du Labex fédère un ensemble de sous projets pour réaliser une migration de leurs données vers des triplestores encore appelés entrepôts ou collections de fichiers RDF (Resource Description Framework) afin d'améliorer le partage, l'échange et la découverte de nouvelles connaissances. Pour ce faire, la norme CIDOC-CRM (International Committee for DOCumentation - Conceptual Reference Model) (Boeuf *et al.*, 2015) a été choisie car elle est aujourd'hui la norme de référence pour la description sémantique des informations muséographiques ou d'héritage culturel (Hooland, Verborgh, 2014). Il s'agit généralement de passer de données non structurées ou semi structurées vers des données structurées puis vers des données sémantiques. Le web sémantique propose une solution pour réaliser ces migrations.

Le web sémantique (Shadbolt *et al.*, 2006) (Berners-Lee *et al.*, 2001) n'est pas qu'un concept mais également une architecture validée et de plus en plus éprouvée sous la forme d'un ensemble de couches indépendantes mais qui s'interfacent les unes avec les autres pour réaliser différentes tâches. Cette architecture décrit les données de leur représentation à leur exploitation via des applications ou agents web sémantique. Ainsi, de nombreuses normes de représentation de données pour le web sémantique existent. Le CIDOC-CRM est un exemple de norme sémantique et est plus spécifiquement un modèle conceptuel de référence ou une ontologie. Le but de la sémantique et donc des nombreux langages de métadonnées ou normes qui la composent est de fournir un cadre homogène de description et d'interrogation de sources de données hétérogènes afin de réduire le silence informationnel et d'améliorer la découverte de connaissances. Ainsi, le projet ModRef a pour but de réaliser une migration de données vers des triplestores CIDOC-CRM en s'appuyant sur des sources de données hétérogènes tant sur le contenu que sur la structure logique initiale (tableurs, bases de données relationnelles, fichiers XML).

Dans cet article, nous présentons le projet ModRef au travers : d'une description générale de la norme CIDOC-CRM ; de l'architecture du projet ModRef ; de la modélisation sémantique CIDOC-CRM et de l'alignement des données des trois sous projets pilotes de ModRef ; de la migration des données vers des triplestores CIDOC-CRM ; de la visualisation et de l'exploitation des triplestores avec l'application web http://triplestore.modyco.fr qui a été développée et déployée ; d'une procédure d'évaluation et des résultats obtenus.

2. Présentation générale de la norme CIDOC-CRM

Il existe plusieurs modèles de représentation de données basés sur de la sémantique (Scarinci, Myers, 2014) qui utilisent des langages de métadonnées qui décrivent des concepts et/ou des liens entre concepts ou instances de concepts (Dublin Core, RDF,

RDFS, OWL, FOAF, Wordnet, CIDOC-CRM). Le *CIDOC-CRM* (Boeuf *et al.*, 2015) (cf. http://www.cidoc-crm.org/) est un modèle conceptuel de référence pour l'information muséographique ou d'héritage culturel. La version de la norme CIDOC-CRM qui a été utilisée est la version 6.2 de mai 2015. Elle comporte 94 classes et 168 propriétés. Il faut noter que les travaux sur le CIDOC-CRM ont débuté en 1996 et c'est en 2006 que le CIDOC-CRM est devenu une norme ISO 21127. Cette norme permet de décrire les caractéristiques globales des objets (identifiant, type, titre, matériau, dimension, note) mais également leur historique au travers des évènements ou activités (transfert de garde -localisations anciennes, localisation actuelle-, origine, découverte, conservation, affectation de valeur, mesure) ainsi que les relations qui existent entre objets ou parties d'objets (bibliographie, composition, similarité, autre représentation -photo, dessin, tableau-, inscription). Une implémentation OWL (Ontology Web Language) du CIDOC-CRM de l'Université d'Erlangen-Nuremberg est disponible à l'adresse suivante : http://www.erlangen-crm.org/. Notons que l'espace de nom de cette implémentation du CIDOC-CRM est généralement préfixé par "*ecrm*".

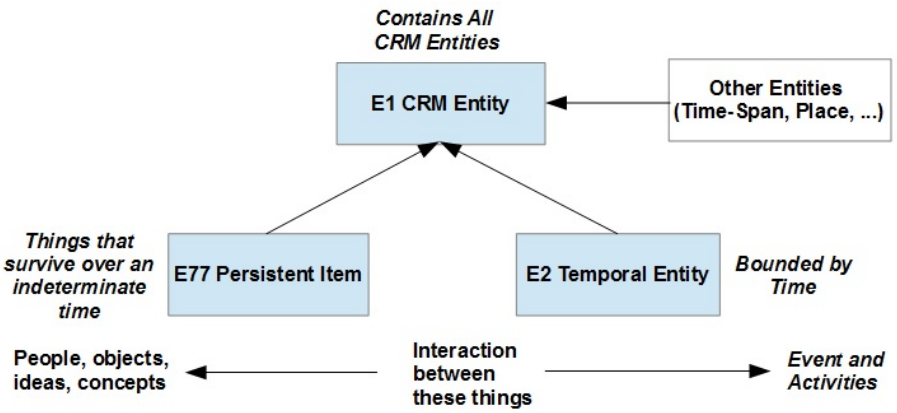


Figure 1. Structure générale des entités du CIDOC-CRM.

La structure générale du CIDOC-CRM est celle de la figure 1. La classe mère de toutes les entités du CIDOC-CRM est la classe *E1 CRM Entity* et elle se subdivise en sous-classes directes dont les deux principales sont : (1) *E77 Persistent Item* qui est la classe la plus générique des entités dites persistantes. Une entité persistante est une entité qui est capable de survivre pendant une période indéterminée, comme par exemple : les personnes, les objets, les idées, les concepts. Ce sont généralement des entités pouvant avoir un début ou une fin d'existence (destruction, par exemple) ; (2) *E2 Temporal Entity* qui est la classe la plus générique des entités dites temporelles. Une entité temporelle est une entité qui est limitée dans le temps, comme : un évènement, un début d'existence, une fin d'existence, une activité, une création, une production, une modification, un transfert de garde, une conservation, une mesure.

Les autres sous-classes directes de la classe racine *E1 CRM Entity* sont les classes *E52 Time-Span, E53 Place, E54 Dimension, E92 Spacetime Volume*. En général, le CIDOC-CRM décrit des entités mais également les interactions qui peuvent exis-

ter entre ces entités : interactions entre entités persistantes ; interactions entre entités temporelles ; interactions entre entités persistantes et entités temporelles ; interactions générales entre entités (par exemple, les interactions qui existent entre entités persistantes ou temporelles avec des entités qui décrivent des durées, des lieux ou des dimensions). Il existe aussi des interactions entre entités et valeurs primitives (chaîne de caractères, nombre, date heure).

D'autre part, plusieurs projets dans le monde s'intéressent à la migration de données vers des triplestores (CIDOC-CRM ou non) : (1) Le *British Museum* (cf. http://collection.britishmuseum.org/) qui est un musée sur l'histoire et la culture et qui utilise le CIDOC-CRM ; (2) *Arches* (cf. http://www.getty.edu/conservation/our_projects/field_projects/arches/) qui est une collaboration entre le Getty Conservation Institute (GCI) et le World Monuments Fund (WMF) sur l'héritage culturel immobilier (monuments, ponts) et qui utilise le CIDOC-CRM ; (3) *DBPedia* (cf. http://www.dbpedia.org/sparql) qui est une encyclopédie en ligne largement utilisée (Ruan *et al.*, 2016) et qui utilise des langages de métadonnées comme : *dbpedia, foaf, umbel, schema.org, dublin core, geo* ; (4) *Nakala* (cf. http://www.nakala.fr/sparql) qui est un service en ligne pour déposer, documenter et diffuser des données (muséographiques) et qui utilise des langages de métadonnées comme : *foaf, skos, dublin core, vcard*.

La spécificité de notre application web est qu'elle traite de sources de données hétérogènes tant sur le contenu que sur la structure logique initiale (bases de données, fichiers XML) de ces données. Les données migrées dans des triplestores sont totalement ouvertes via notre application web. Cette application permet de visualiser les triplestores sous trois différents formats : *rdf, triplets, résumé attribut-valeur*. L'application permet aussi d'interroger les triplestores via des *"Endpoint Sparql"* (interface de saisie et d'exécution de requêtes Sparql - Sparql étant le langage de référence actuel d'interrogation de fichiers RDF) et via des *"formulaires généraux"* qui s'avèrent être utiles si on ne connait pas le Sparql (Haase *et al.*, 2004) et le CIDOC-CRM.

3. Architecture du projet ModRef

L'architecture du projet ModRef, illustrée dans la figure 2, décrit les différents processus de numérisation des données depuis la phase de création de ces données numériques à partir des connaissances d'un expert, jusqu'à son interrogation et sa visualisation par un usager. Les données peuvent ainsi subir de nombreuses transformations avant d'être finalement exploitables via des triplestores.

Ainsi, on peut passer de données non structurées ou semi structurées (notes, rapports, livres, sites web) vers des données structurées décrites par une structure logique. Cette structure logique peut être une structure à plat sous la forme *attribut-valeur* ou *tableur*, mais elle peut aussi être plus fortement structurée sous la forme de *bases de données relationnelles* ou de *fichiers XML* qui, dans notre contexte, sont des fichiers XML-EAD (Encoded Archival Description). Ces différentes descriptions font généralement usage de thésaurus (vocabulaire contrôlé de termes descripteurs ou non). À partir de ces descriptions structurelles, on va construire une description sémantique

des données sous forme de graphe sémantique RDF en s'appuyant sur des référentiels standards ou normes. Dans notre contexte, nous avons utilisé la norme CIDOC-CRM pour générer nos triplestores à partir d'un alignement de nos données avec le graphe sémantique CIDOC-CRM. Ces triplestores vont pouvoir être exploités dans diverses applications web sémantique ou via des "Endpoint Sparql".

La première phase de transformation des données (données non structurées ou semi structurées vers données structurées) est réalisée au sein de chaque sous projet tandis que le projet ModRef lui intervient principalement dans la deuxième phase de transformation des données (données structurées vers données sémantiques).

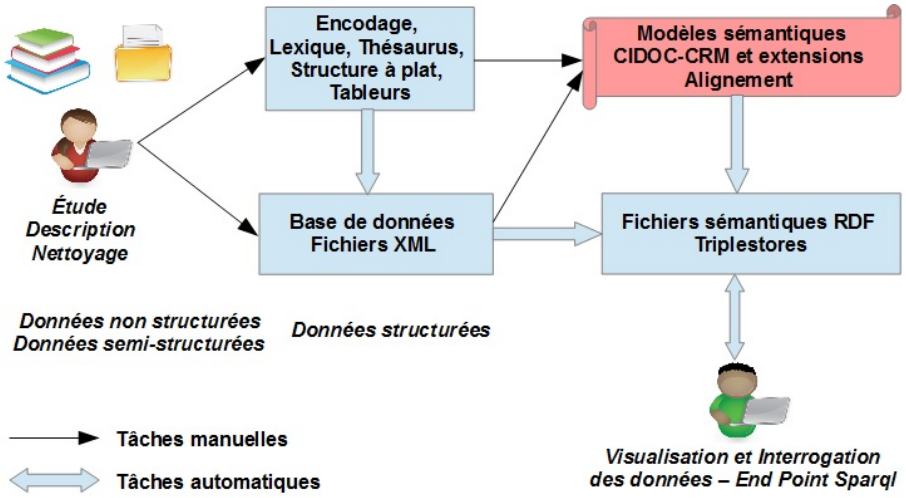


Figure 2. Architecture du projet ModRef.

Par ailleurs, pour réaliser la preuve conceptuelle de ModRef, trois projets pilotes ont été sélectionnés :

1. *CDLI* : conservatoire numérique ou musée virtuel de l'ensemble des documents rédigés en écriture cunéiforme (cf. http://www.cdli.ucla.edu) ;

2. *ObjMythArcheo* : corpus numérique d'objets archéologiques à iconographie mythologique (cf. http://www.limc-france.fr et http://medaillesetantiques.bnf.fr) ;

3. *BiblioNum* : bibliothèque numérique sur l'histoire de France du 20e siècle (cf. http://www.argonnaute-u.paris10.fr).

La table 1 compare les données des trois projets pilotes de ModRef sur 5 critères : taille des textes, nombre d'objets, type de la structure logique, nombre d'éléments de la structure logique et langue de description des données.

4. Modélisation CIDOC-CRM de ModRef et alignement des données

Nous avons identifié les classes CIDOC-CRM utiles (dont au moins un chemin conduit vers une valeur non nulle) pour modéliser les données de nos trois projets pi-

Table 1. Comparaison des données des projets pilotes de ModRef.

	CDLI	**ObjMythArcheo**	**BiblioNum**
Taille textes	300 Mo	100 Mo	100 Mo
Nombre d'objets	313 332 tablettes	17 424 objets	77 collections - 62 392 fichiers
Structure logique	Base de données de type Tableur	Base de données relationnelle	XML-EAD
Nombre d'éléments de structure	1 table avec 61 attributs	59 tables	146 éléments XML-EAD
Langue	Anglais	Français-Anglais	Français

lotes. Cela représente des extraits de graphes relatifs aux quatre thèmes suivants : (1) caractéristiques générales (identifiant, type, titre, matériau, dimension, note/description), bibliographie, composition et similarité d'objets ; (2) évènements de début d'existence (origine) et de fin d'existence ; (3) activités diverses (transfert de garde, mesure, conservation) ; (4) inscriptions et autres représentations (photo, dessin, tableau).

De façon générale, ces extraits sont stables pour tout projet car dans le CIDOC-CRM, il est possible d'identifier tous les chemins possibles pour obtenir une information donnée sur un objet. En effet, un graphe sémantique est un ensemble de nœuds et d'arcs orientés ou relations qui obéissent à un certain nombre de contraintes et règles (raccourci, héritage, inverse, symétrie, transitivité). Ce sont ces contraintes et règles qui définissent la cohérence et la validité d'un modèle.

Dans cette section, nous décrirons nos différents thèmes de modélisation de graphe CIDOC-CRM ainsi qu'un exemple d'alignement. En effet, le principe d'alignement est globalement le même pour tous les thèmes et pour tous les projets pilotes.

4.1. Modélisation des caractéristiques générales

Les caractéristiques générales d'un objet s'obtiennent le plus souvent par des interactions, avec des chemins de graphe assez courts, entre entités. Elles permettent de définir pour un objet les éléments suivants : identifiant, type (catégorisation), titre, matériau, dimension, note.

La modélisation des caractéristiques générales des objets du projet ModRef est illustrée dans la figure 3. Dans cette figure, on peut observer l'existence de deux chemins de graphes différents pour la définition des dimensions d'un objet : (1) un *chemin plus court* ou *raccourci* qui relie la classe *E70 Thing* à la classe *E54 Dimension* avec la propriété *P43 has dimension*, soit le triplet *[E70 Thing, P43 has dimension, E54 Dimension]* ; (2) un *chemin plus long* qui contient davantage de nœuds informationnels à remplir. Ce chemin est décrit par les triplets suivants : *[E1 CRM Entity, P39i was measured by, E16 Measurement], [E16 Measurement, P40 observed dimension, E54 Dimension]*. Avec ce chemin, on peut en plus remplir des informations concernant l'activité de mesure *E16 Measurement*. En effet, la classe *E16 Measurement* est

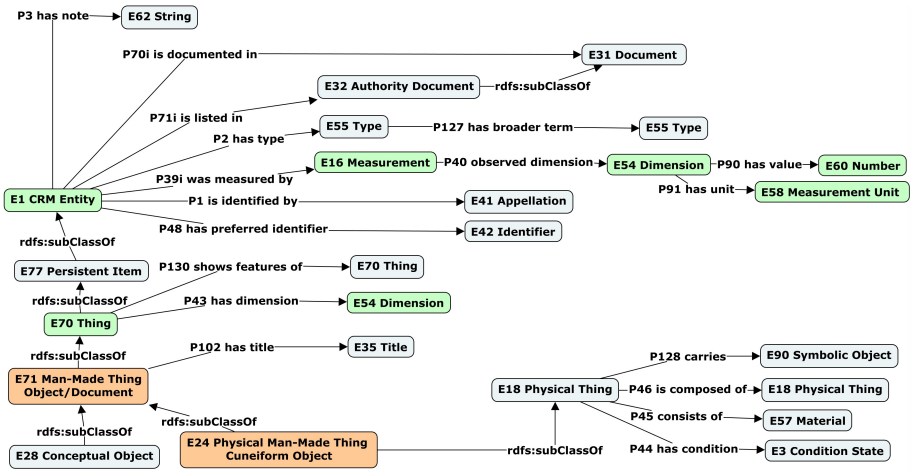


Figure 3. Modélisation des caractéristiques générales.

un type d'activité car les classes *E13 Attribute Assignment*, *E7 Activity* et *E5 Event* font partie de sa hiérarchie (cf. figure 4).

Il est tout à fait possible de remplir différents chemins donnant une même information dans un graphe. Cependant, on peut être amené à faire un choix entre deux possibilités de chemins lorsque l'on ne dispose pas des informations nécessaires pour décrire un chemin donné. Ceci est le cas le plus souvent lorsqu'une entité temporelle fait partie du chemin. D'autre part, la figure 3 permet aussi d'illustrer d'autres interactions entre entités persistantes comme : *P70i is documented in* pour les références bibliographiques, *P46 is composed of* pour la composition d'objets, *P130 shows features of* pour la similarité entre objets, *P128 carries* pour la relation entre un objet et une entité qui se trouve sur l'objet, comme une inscription par exemple.

4.2. Modélisation des évènements de début et de fin d'existence

Une activité importante concernant les informations muséographiques consiste à décrire leur origine : à les dater, à définir leur lieu d'origine et éventuellement les participants à leur création. La modélisation des évènements de début et de fin d'existence des objets du projet ModRef est illustrée dans la figure 4. Le CIDOC-CRM permet ainsi de définir la date, le lieu et les participants de chaque évènement.

Pour le début d'existence (origine), on utilise l'évènement *E63 Beginning of Existence* et les patterns de triplets suivants : *[E77 Persistent Item, P92i was brought into existence by, E63 Beginning of Existence], [E2 Temporal Entity, P4 has time-span, E52 Time-Span], [E52 Time-Span, P78 is identified by, E49 Time Appellation], [E4 Period, P7 took place at, E53 Place], [E5 Event, P11 had participant, E39 Actor], [E63 Beginning of Existence, rdfs : subClassOf, E5 Event], [E5 Event, rdfs : subClassOf, E4 Period], [E4 Period, rdfs : subClassOf, E2 Temporal Entity]*. Par ailleurs, on

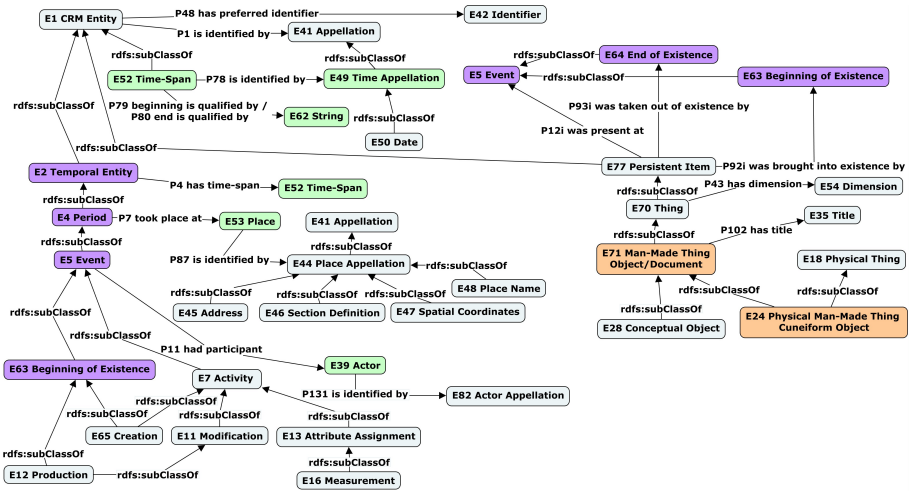


Figure 4. Modélisation des évènements de début et de fin d'existence.

peut aussi partir des activités *E65 Creation* ou *E12 Production* qui ont pour superclasses les classes *E63 Beginning of Existence* et *E7 Activity* (cf. figure 4).

Pour la fin d'existence, on utilise la classe *E64 End of Existence* ou une de ses sous-classes. Ainsi, on va pouvoir définir également la date, le lieu et les participants à la fin d'existence d'un objet.

4.3. Modélisation des activités

La figure 5 illustre un extrait de notre modèle pour la description des activités en général, et de l'activité de *transfert de garde* en particulier. Ainsi, pour rattacher un objet à une activité de transfert de garde, on va utiliser la propriété *P30 transferred custody of* (ou son inverse *P30i custody transferred through*) entre l'activité (*E10 Transfer of Custody*) et l'objet physique (*E18 Physical Thing*). De plus, pour un transfert de garde, on peut décrire les différents protagonistes du transfert (*P29 custody received by, P28 custody surrendered by*) et décrire éventuellement aussi un historique des différents transferts de garde relatifs à un objet ou à un document donné. Notons qu'il existe également un chemin de raccourci qui ne passe pas par l'activité de transfert de garde et qui permet de définir les gardiens ou propriétaires anciens ou actuels d'un objet (*P49 has former or current keeper, P50 has current keeper, P51 has former or current owner, P52 has current owner*).

De façon générale, pour un évènement ou une activité, on va décrire la date, le lieu et les participants ou acteurs de l'évènement ou de l'activité. Plus spécifiquement pour une activité (transfert de garde, assignation de valeur à un attribut, mesure, conservation), on va pouvoir en plus décrire : la procédure utilisée (*P33 used specific technique, P32 used general technique*), les objets utilisés (*P16 used specific object,*

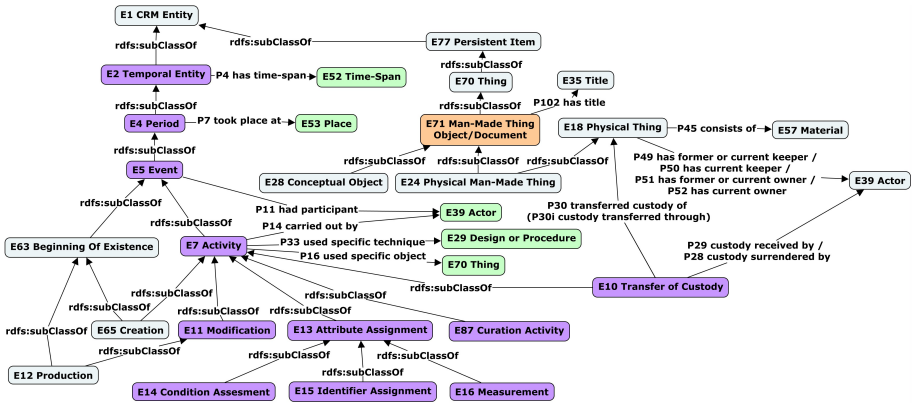


Figure 5. Modélisation des activités.

P125 used object of type), l'objectif de l'activité (*P20 had specific purpose, P21 had general purpose*).

4.4. Modélisation des inscriptions et autres représentations d'un objet

La figure 6 illustre un extrait de notre modèle pour la description des inscriptions sur des objets ou la description d'autres représentations (photos, dessins, tableaux) de ces objets.

Ainsi, pour rattacher un objet à son inscription, on va utiliser la propriété *P128 carries* entre un objet physique (*E18 Physical Thing*) et un objet symbolique (*E90 Symbolic Object*) qui se trouve sur l'objet et qui est ici notre inscription. Cela permet donc de retrouver, par exemple, les objets qui portent une certaine inscription (sceau, signature).

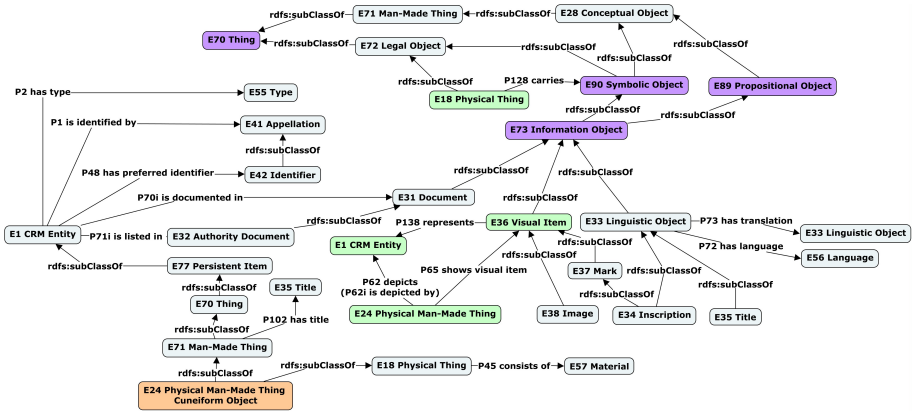


Figure 6. Modélisation des inscriptions et autres représentations d'un objet.

Par ailleurs, pour relier une photo (ou dessin ou tableau) à un objet (physique ou conceptuel), on peut utiliser un ensemble de propriétés : (1) *P62 depicts* pour décrire le lien entre la photo ou le dessin ou le tableau (ici, *E24 Physical Man-Made Thing*) et l'entité *E1 CRM Entity* (objet physique ou conceptuel) *représentée* par la photo ou le dessin ou le tableau. Cette propriété ne se rapporte pas aux inscriptions ou autres informations encodées sur un objet ; (2) *P65 shows visual item* permet de rattacher la photo ou le dessin ou le tableau à une représentation visuelle (*E36 Visual item*) de l'objet *représenté* par la photo ou le dessin ou le tableau ; (3) *P138 represents* permet de relier une représentation visuelle (*E36 Visual item*) d'un objet à l'objet en question (*E1 CRM Entity*). Notons cependant que la propriété *P62 depicts* est un raccourci des propriétés *P65 shows visual item et P138 represents*. La photo ou dessin ou tableau étant généralement décrite avec la classe *E24 Physical Man-Made Thing*.

4.5. Alignement des données

La migration de données vers des triplestores nécessite une phase d'alignement des données avec les extraits de graphe sémantique CIDOC-CRM proposés. Cet alignement est indispensable du fait de l'hétérogénéité initiale de la description des données, conséquence aussi de la diversité des projets pilotes de ModRef. Cet alignement n'est pas une tâche programmatique mais fait appel à des détails de structure logique propre au modèle de description de données choisi par chaque projet. C'est une tâche à mi-chemin entre la modélisation et l'implémentation qu'elle permet d'entrevoir un peu plus clairement. Notons que cette tâche ne doit pas être confondue avec l'alignement entre ontologies (fichiers owl/rdf) (Faria *et al.*, 2013) étant donné que notre alignement est plutôt entre l'ontologie du CIDOC-CRM et des données brutes provenant de bases de données ou de fichiers XML (dans notre contexte, des fichiers XML-EAD).

L'alignement va consister principalement à remplir les nœuds du graphe sémantique. Les nœuds terminaux vont être remplis par des valeurs extraites des structures logiques des données des projets correspondants et les nœuds non-terminaux ou intermédiaires seront remplis avec des URIs qui définissent ainsi des chemins vers les nœuds terminaux. Notons qu'une rigueur particulière doit être apportée à la construction des URIs, à la fois pour leur lisibilité mais également pour la cohérence des chemins dans le graphe, afin d'éviter des conflits de chemins et garantir ainsi l'unicité d'un chemin donné par rapport à un autre. La figure 7 illustre un extrait d'alignement de données, initialement au format XML-EAD (cf. https://www.loc.gov/ead/tglib/element_index.html) et correspondant au premier thème de notre modélisation sémantique (cf. figure 3). En XML-EAD, pour obtenir par exemple les dimensions d'un objet on utilise les chemins d'accès xpath "*/ead/archdesc/did/physdesc/dimensions*" ou *"/ead/archdesc/dsc/c/did/physdesc/dimensions*", selon que l'on est au niveau de la collection ou du document. Ainsi, pour décrire les dimensions d'un objet, on utilise une succession de triplets de la forme :

[http ://www.modref.org/biblionum/document_id/e70_thing, rdf : type, ecrm : E70_Thing],

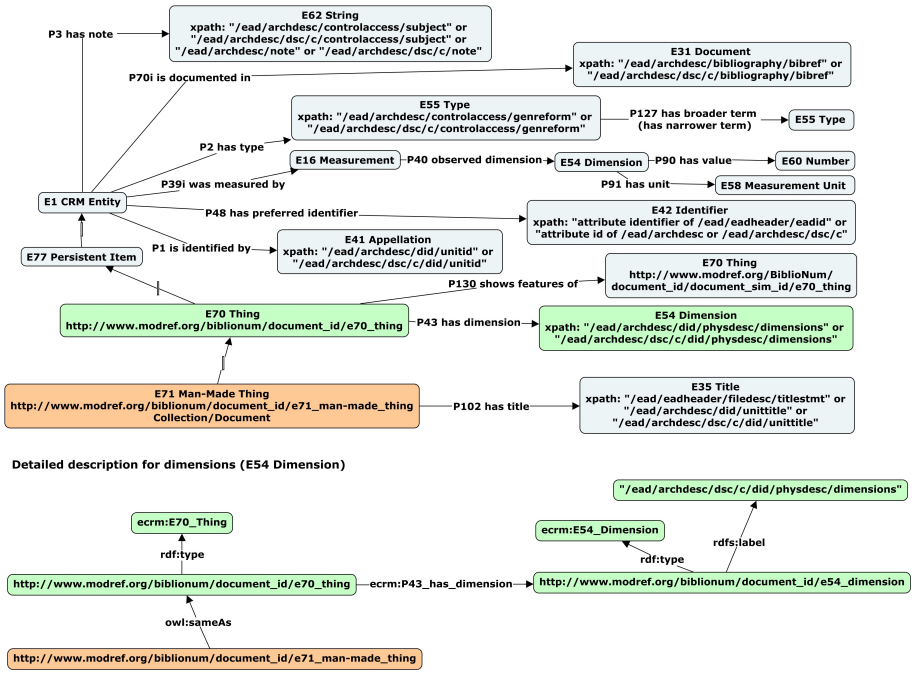


Figure 7. Exemple d'alignement de données au format XML-EAD.

[http ://www.modref.org/biblionum/document_id/e70_thing, ecrm : P43_has_dimension, http ://www.modref.org/biblionum/document_id/e54_dimension],

[http ://www.modref.org/biblionum/document_id/e54_dimension, rdf : type, ecrm : E54_Dimension],

[http ://www.modref.org/biblionum/document_id/e54_dimension, rdfs : label, "/ead/archdesc/dsc/c/did/physdesc/dimensions"],

[http ://www.modref.org/biblionum/document_id/e71_man-made_thing, owl : sameAs, http ://www.modref.org/biblionum/document_id/e70_thing].

De façon générale, l'alignement réalisé va être décrit dans une structure de données programmatique qui sera utilisée pour générer automatiquement des fichiers qui respectent la syntaxe RDF et CIDOC-CRM : c'est la migration des données ou création de nos triplestores.

5. Migration de données vers des triplestores

Une migration efficace et cohérente de données fait appel à différentes compétences. Pour assurer la pérennisation de cette procédure, une architecture générale et rigoureuse du workflow des différents types de données à manipuler doit être définie.

Cette architecture explicite la démarche globale de tout projet qui souhaite faire migrer ses données vers des triplestores. Pour ModRef, cette démarche se subdivise en différentes étapes bien identifiées : préparation des données (étude et description structurelle), modélisation sémantique et alignement des données structurées avec le graphe sémantique CIDOC-CRM, création et exposition de triplestores qui vont alors pouvoir être interrogés par des usagers ou des applications web sémantique. Initialement, les données sont souvent non-structurées ou semi-structurées (notes, rapports, livres, html) et ont besoin d'être d'abord décrites par une représentation structurée (tableurs, bases de données, fichiers XML-EAD) afin de pouvoir construire plus facilement leur représentation sémantique, par la suite. Ce continuum d'étapes fait intervenir plusieurs sous-procédures pour assurer le passage d'un format de représentation de données à un autre. Ainsi, on distingue deux phases principales permettant le passage : (1) des données non structurées ou semi structurées vers des données structurées ; (2) des données structurées vers des données sémantiques.

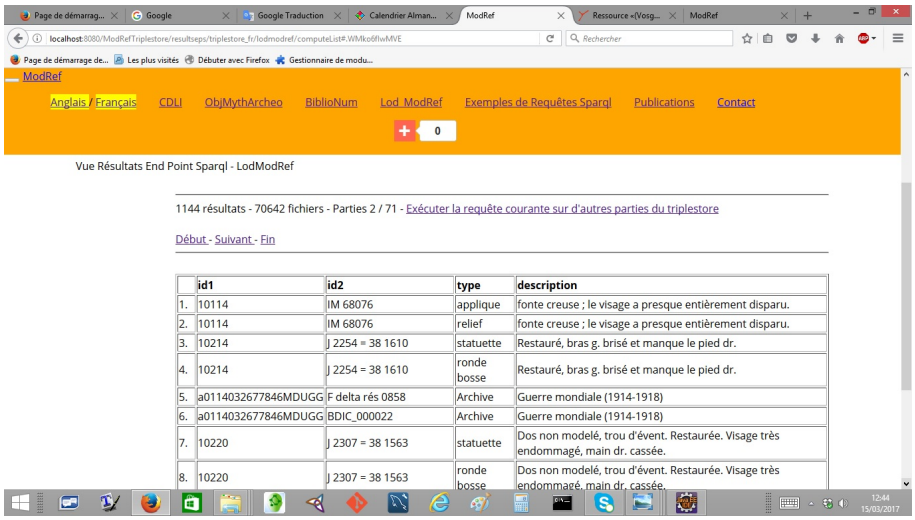
En effet, l'élément clé du processus de migration de données vers des triplestores est la modélisation et l'alignement des données avec le modèle de graphe sémantique choisi. Pour réaliser un alignement des données avec notre graphe CIDOC-CRM, nous avons effectué une mise en correspondance de certains nœuds du graphe sémantique proposé avec des informations extraites à la fois de bases de données mais aussi de collections de fichiers XML-EAD. Cette migration implique donc à la fois de la lecture de bases de données et du parsing de fichiers XML-EAD (voir la table 1).

La preuve conceptuelle du projet ModRef ou la validation de la migration de données vers des triplestores concernent donc un ensemble de tâches en amont (préparation et structuration des données, modélisation sémantique, alignement des données) et en aval (exposition, visualisation, interrogation et exploration des données) du processus de migration. Ainsi, l'exploitation des triplestores créés et les bénéfices que l'on peut en tirer est l'autre aspect majeur autour de la question de ces nouveaux entrepôts de documents RDF que sont les triplestores.

6. Visualisation et Exploitation de triplestores

Les triplestores créés sont exposés pour consultation (sous trois formes : *rdf, triplets, résumé attribut-valeur*) mais aussi pour interrogation via notre application web. L'intérêt du triplestore est qu'on a un modèle connu public et publié de représentation de l'information, ce qui permet d'interroger les triplestores indifféremment avec des procédures identiques. Nous avons défini deux procédures d'exploitation de nos triplestores : des interfaces sous forme de *"formulaires généraux"* et des *"Endpoint Sparql"* (cf. figure 8).

Les formulaires sont un moyen simple et assez intuitif, car très proche du langage naturel, pour formuler des requêtes vers nos triplestores. Nul besoin donc de compétences particulières, il suffit de remplir les rubriques du formulaire qui nous intéressent et de lancer la recherche. Une requête Sparql est automatiquement construite à partir des valeurs des champs renseignés du formulaire et c'est cette requête qui est utilisée



	id1	id2	type	description
1.	10114	IM 68076	applique	fonte creuse ; le visage a presque entièrement disparu.
2.	10114	IM 68076	relief	fonte creuse ; le visage a presque entièrement disparu.
3.	10214	J 2254 = 38 1610	statuette	Restauré, bras g. brisé et manque le pied dr.
4.	10214	J 2254 = 38 1610	ronde bosse	Restauré, bras g. brisé et manque le pied dr.
5.	a0114032677846MDUGG	F delta rés 0858	Archive	Guerre mondiale (1914-1918)
6.	a0114032677846MDUGG	BDIC_000022	Archive	Guerre mondiale (1914-1918)
7.	10220	J 2307 = 38 1563	statuette	Dos non modelé, trou d'évent. Restaurée. Visage très endommagé, main dr. cassée.
8.	10220	J 2307 = 38 1563	ronde bosse	Dos non modelé, trou d'évent. Restaurée. Visage très endommagé, main dr. cassée.

Figure 8. Application Web du projet ModRef : Endpoint Sparql.

pour interroger le triplestore. Au terme de l'exécution de la requête, une liste d'objets sélectionnés est renvoyée en résultat à l'usager qui peut les consulter également sous trois formes : *rdf, triplets, résumé attribut-valeur*. Par ailleurs, on peut aussi interroger nos triplestores via des "Endpoint Sparql". Ce deuxième mode d'interrogation nécessite la connaissance du langage Sparql qui est aujourd'hui le langage de référence pour l'interrogation de documents RDF. Sparql est un langage assez simple mais pas toujours à la portée de tous. Ainsi, les formulaires généraux peuvent être vus comme un premier point d'entrée pour l'interrogation des triplestores tandis que les "Endpoint Sparql" assurent une exploitation (interrogation et exploration) plus large de ces derniers via une formulation libre de requêtes Sparql de type "Select".

Notre application web permet de consulter, d'interroger et d'explorer nos triplestores séparément pour chaque projet pilote de ModRef mais aussi en regroupant les triplestores via le LOD (Linked Open Data) de ModRef. L'application web offre, pour chaque projet et pour le LOD de ModRef, la possibilité de consulter les données sous trois formes mais aussi celle de les interroger via des "formulaires généraux" mais aussi via des "Endpoint Sparql". Ainsi, en résultat d'une requête, le LOD permet de retrouver des informations diverses (statue/statuette, archive) provenant de différents triplestores (cf. figure 8). Plusieurs requêtes Sparql ont été exécutées pour valider la migration de données et une liste de requêtes exemples est fournie dans notre application web. Nous avons développé notre propre "Endpoint Sparql" et nous offrons également la possibilité d'interroger nos données via un "Endpoint Sparql" Virtuoso (logiciel permettant de créer un lien internet vers une instance de "Endpoint Sparql") disponible via le lien suivant : http://3s-passespresent.huma-num.fr/sparql.

Notons que la notion d'exploitation de triplestores fait appel aux notions d'interrogation et d'exploration de graphe. Ainsi, l'interrogation de triplestores consiste à

formuler une requête Sparql pré-formatée (formulaires généraux) ou libre (Endpoint Sparql) tandis que l'exploration de triplestores est une forme d'interrogation uniquement possible via des "Endpoint Sparql" qui permet de découvrir différents chemins dans un graphe sémantique vers des données précises. En effet, plusieurs chemins peuvent permettre d'obtenir une même information dans un graphe (usage de diverses notions : raccourci, raffinement, héritage, inverse) sachant que ces chemins ne sont pas toujours tous renseignés. On peut donc écrire des requêtes Sparql pour découvrir si différents chemins vers une donnée précise existent ou pour connaitre les chemins menant vers des nœuds terminaux. L'exploration est donc importante pour s'approprier un triplestore CIDOC-CRM spécifique.

7. Procédure d'évaluation et résultats

Table 2. Requêtes Sparl.

(a) Liste des triplets terminaux

```
SELECT Distinct ?subject ?predicate ?object
WHERE {
?subject ?predicate ?object .
Filter ( isLiteral( ?object) && ?object != "" )
}
```

(b) Liste des types d'objets

```
PREFIX ecrm : <...>
PREFIX rdf : <...>
PREFIX rdfs : <...>
SELECT Distinct ?type
WHERE {
 ?type_uri rdf : type ecrm : E55_Type .
 ?type_uri rdfs : label ?type .
Filter ( ?type != "" ) }
```

(c) Liste de caractéristiques provenant de l'entité "E1 CRM Entity".

```
PREFIX ecrm : <http ://erlangen-crm.org/150929/>
PREFIX rdf : <http ://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs : <http ://www.w3.org/2000/01/rdf-schema#>
SELECT Distinct ?id1 ?id2 ?type ?description
WHERE {
 ?e1_obj ecrm : P48_has_preferred_identifier ?id1_uri .
 ?id1_uri rdfs : label ?id1 .
 ?e1_obj ecrm : P1_is_identified_by ?id2_uri .
 ?id2_uri rdfs : label ?id2 .
 ?e1_obj ecrm : P2_has_type ?type_uri .
 ?type_uri rdfs : label ?type .
 ?e1_obj ecrm : P3_has_note ?description . }
```

Nous avons conçu et exécuté plusieurs requêtes Sparql pour valider les différents datasets de nos triplestores. Les requêtes sont divisées en deux groupes, un premier groupe relatif au schéma de la syntaxe RDF (*liste des concepts ou des prédicats utilisés, liste des triplets terminaux (cf. Table 2a), liste des triplets d'une ressource donnée, extraits de chemins menant vers des nœuds terminaux non vides*) et un autre groupe relatif au schéma de la norme CIDOC-CRM (*vérification de l'instanciation d'une classe spécifique, vérification des labels d'une entité ou ressource donnée (cf. Table 2b), ca-*

ractéristiques générales d'un objet (cf. Table 2c), information sur l'origine ou la garde d'un objet).

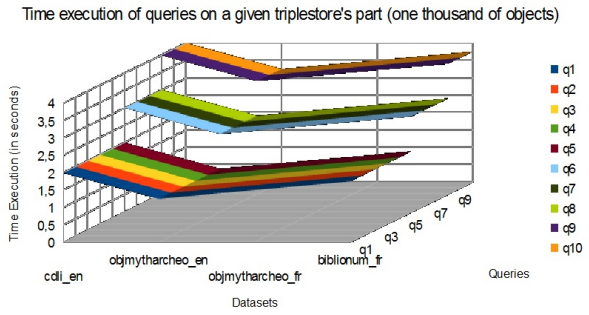


Figure 9. Exécution de requêtes Sparql pour le projet ModRef.

De plus, les triplestores sont subdivisés en parties constantes (nombre d'objets ou nombre de triplets) et les requêtes sont exécutées chaque fois sur une seule partie et puis progressivement sur les autres parties si l'utilisateur le demande. Les résultats sont donc fusionnés au fur et à mesure. L'utilisateur peut arrêter l'exécution sans avoir à exploiter tout le triplestore. Le numéro de la partie courante (sur laquelle vient de s'exécuter la requête courante) et le nombre total de parties du triplestore sont toujours affichés. La figure 9 montre que le temps moyen d'exécution (en secondes) des requêtes sur une partie de triplestore (*soit 1000 objets, approximativement 100 000 triplets*) est plutôt constant et la rapidité d'exécution de ces requêtes est tout à fait acceptable pour les usagers. Par contre, le temps d'exécution cumulatif augmente si l'on couvre davantage de parties du triplestore.

8. Conclusion

Le projet ModRef permet de réaliser une preuve conceptuelle de la migration de données vers des triplestores CIDOC-CRM à travers : une architecture générale qui identifie les différentes étapes à suivre ; la modélisation et l'alignement des données avec le graphe sémantique ; la migration des données vers les triplestores ; l'exposition des triplestores via l'application web bilingue "anglais-français" http://triplestore .modyco.fr qui permet de consulter, d'interroger et d'explorer ces triplestores.

Les perspectives qui découlent de nos travaux concernent : (1) *le partage, l'échange et la découverte de connaissances à plus grande échelle* en intégrant d'autres LOD (Linked Open data) sur internet (Beek, Rietveld *et al.*, 2016) (Daga *et al.*, 2016). Le LOD doit améliorer la découverte de nouvelles connaissances, du fait de la quantité et de la diversité des données liées mais surtout du fait de l'usage de formalismes, de langages de métadonnées, de thésaurus publiés, standardisés voire normalisés ; (2) *la comparaison de graphes sémantiques* qui décrivent des données similaires (Beek, Schlobach, Harmelen, 2016) (objets ressemblants, objets d'une même période histo-

rique, objets de même type, objets identiques) dans un contexte de LOD. Il en résultera un enrichissement mutuel des différents acteurs ou usagers des LOD.

Acknowledgements

L'auteur remercie le laboratoire Labex "Les passés dans le présent" de l'Université de Paris 10 et le projet ANR ModRef de référence ANR-11-LABX-0026-01.

Références

Beek W., Rietveld L., Schlobach S., Harmelen F. van. (2016). Lod laundromat : Why the semantic web needs centralization (even if we don't like it). *IEEE Internet Computing*, Vol. 20, N° 2, pp. 78-81.

Beek W., Schlobach S., Harmelen F. van. (2016). A contextualised semantics of owl :sameas. *In proceedings of the 13th Extended Semantic Web Conference ESWC'16*, pp. 405-419.

Berners-Lee T., Hendler J., Lassila O. (2001). The semantic web. *Scientific American*, pp. 34-43.

Boeuf P. L., Doerr M., Ore C. E., Stead S. (2015, May). Definition of the cidoc conceptual reference model, version 6.2. *Produced by the ICOM/CIDOC Documentation Standards Group, Continued by the CIDOC CRM Special Interest Group. http ://www.cidoc-crm.org/ [retrieved : March, 2017].*

Daga E., d'Aquin M., Adamou A., Brown S. (2016). The open university linked data - data.open.ac.uk. semantic web. *Semantic Web*, Vol. 7, N° 2, pp. 183-191.

Faria D., Pesquita C., Santos E., Palmonari M., Cruz I., Couto F. (2013). The agreementmakerlight ontology matching system. *The 12th International Conference on Ontologies, DataBases, and Applications of Semantics (ODBASE)*, pp. 527-541.

Haase P., Broekstra J., Eberhart A., Volz R. (2004). Comparison of rdf query languages. *In proceedings of the third International Semantic Web Conference ISWC'04*, pp. 502-517.

Hooland S. V., Verborgh R. (2014). *Linked data for libraries, archives and museums. how to clean, link and publish your matadata* (A. L. Association, Ed.).

Oldman D., Doerr M., Jong G. de, Norton B., Wikman T. (2014). Realizing lessons of the last 20 years : A manifesto for data provisioning and aggregation services for the digital humanities. http ://www.dlib.org/dlib/july14/oldman/07oldman.html [retrieved : March, 2017]. *D-Lib Magazine*, Vol. 20, N° 7/8.

Ruan T., Li Y., Wang H., Zhao L. (2016). From queriability to informativity, assessing "quality in use" of dbpedia and yago. *In proceedings of the 13th Extended Semantic Web Conference ESWC'16*, pp. 52-68.

Scarinci J., Myers T. (2014). A semantic web framework to enable sustainable lodging best management practices in the usa. *Information Technology and Tourism*, Vol. 14, N° 4, pp. 291-315.

Shadbolt N., Berners-Lee T., Hall W. (2006). The semantic web revisited. *IEEE Intelligent Systems*, Vol. 21, N° 3, pp. 96-101.