Linked Open Data pour les Entrepôts de Données: Opportunité et Défis

Nabila Berkani*, Selma Khouri*, Ladjel Bellatreche**

*Laboratoire LCSI, Ecole Nationale Supérieure d'Informatique, Alger, Algérie
(n_berkani, s_khouri)@esi.dz
**LIAS/ISAE-ENSMA, 86960 Futuroscope, France
bellatreche@ensma.fr

Résumé. De nos jours, nous vivons dans un monde ouvert et connecté, où les petites, moyennes et grandes entreprises cherchent à s'approprier des systèmes de stockage pour une meilleure analyse des données. Les entrepôts de données ($\mathcal{ED}$) sont un exemple de ces systèmes, qui ont connu dernièrement une baisse de régime avec l'apparition des données massives et leurs différents V (Volume, Variété, Vélocité, Valeur, etc). L'arrivée de l'ère Linked Open Data ($\mathcal{LOD}$) représente une excellente opportunité pour les $\mathcal{ED}$, car elles sont porteuses de Valeur ajoutée. La prise en compte de cette Valeur dans la conception des $\mathcal{ED}$ doit faire face au problème de la Variété des sources de données et la Variété des formats de stockage cible de l'$\mathcal{ED}$. Dans cet article, nous présentons une nouvelle approche de conception d'$\mathcal{ED}$ guidée par la valeur et la variété, où différents scénarii d'intégration des $\mathcal{LOD}$ impactant les processus ETL sont étudiés. Finalement, nous exposons des expérimentations validant nos propositions.

1 Introduction

La compétitivité dans le développement des pays, des régions, des universités et des entreprises est devenue un défi majeur à l'heure de l'intégration et de la mondialisation des économies en croissance. Le besoin à l'heure actuelle pour chaque entreprise est lié à la productivité de la chaîne de valeur ajoutée. Une des dimensions porteuses de valeur dans les entreprises et concentrant d'importants investissements est liée aux technologies de traitement des données. Les entrepôts de données ($\mathcal{ED}$) et leur environnement BI ont été créés pour répondre à un besoin de création de valeur ajoutée pour l'entreprise, à travers diverses possibilités analytiques. Depuis l'apparition des données massives (Big Data) avec ses principaux Vs (Volume, Variété, Vélocité, Visibilité, Véracité et Valeur), une baisse d'intérêt pour les $\mathcal{ED}$ a été ressentie, de la même manière que les SGBD relationnels après l'apparition de NoSQL. Le cours des évènements nous a appris que les SGBD relationnels continuent d'occuper une place importante et peuvent même cohabiter avec les SGBD NoSQL. Le challenge actuel des $\mathcal{ED}$ est de se positionner dans ce nouvel environnement du Big Data, en tirant profit de ses Vs, pour susciter un regain d'intérêt qui contribuera à leur relance.

L'arrivée de l'ère *Linked Open Data* ($\mathcal{LOD}$) représente une excellente opportunité, car elles sont porteuses de valeur ajoutée, complémentaire à celle extraite à partir des sources de

données internes. Les $\mathcal{LOD}$ se définissent comme un ensemble de principes de conception pour le partage des données lisibles par la machine sur le Web pour une utilisation par les administrations publiques, les entreprises et les citoyens. L'apport en valeur ajoutée de ces données dans les $\mathcal{ED}$s s'accompagne par une augmentation de la variété nécessitant un traitement particulier. Ces données sont en effet connues pour être disparates, fortement évolutives, distribuées, changeantes et autonomes. Leur intégration au moyen d'un processus ETL devient un enjeu majeur qui doit prendre en compte deux principales contraintes : (a) *la variété des sources de données* (par ex. relationnelles, sémantiques, graphes, etc.). Cette variété concerne leur Univers de Discours ($\mathcal{UD}$) et leurs formalismes. (b) La *variété des formats de stockage* existants pour l'$\mathcal{ED}$ pour héberger des types particuliers de données.

Cependant, nous réalisons que pour obtenir de la valeur ajoutée, l'entreprise/organisation doit faire face à la variété. Nous remarquons que dans le contexte actuel des systèmes BI, la valeur et la variété sont traitées à deux niveaux différents : la variété est traitée au niveau technique (lors de la phase d'ETL : Extract-Transform-Load par les concepteurs et administrateurs) et la valeur est traitée au niveau organisationnel (par les décideurs et managers). Nous distinguons deux politiques organisationnelles en fonction du moment où l'entreprise décide de connecter son $\mathcal{ED}$ aux $\mathcal{LOD}$. Plus précisément, ces politiques sont définies comme suit : (a) $\mathcal{ED}$ *rencontre* $\mathcal{LOD}$ et (b) $\mathcal{ED}$ *avant* $\mathcal{LOD}$. Dans la première politique, l'$\mathcal{ED}$ est construit à partir de zéro par une intégration simultanée de sources de données internes et externes, et (b) dans la seconde, nous supposons que l'$\mathcal{ED}$ est déjà opérationnel lorsque l'entreprise décide d'intégrer les $\mathcal{LOD}$, et l'$\mathcal{ED}$ doit continuer à intégrer des données provenant de sources locales et des $\mathcal{LOD}$. Pour remettre ces 2V au même niveau, nous formalisons ce problème qui consiste à augmenter la valeur en considérant la variété des sources comme contrainte : ayant (i) un ensemble de sources internes S_I=$\{Si_1, Si_2, ...Si_m\}$, (ii) un ensemble de sources externes $\mathcal{LOD}$ S_E=$\{Se_1, Se_2, ...Se_m\}$. Chaque source interne et externe possède son propre format $FormatS_i$ et son modèle conceptuel CM_i décrivant son univers du discours. (iii) un ensemble de besoins B à satisfaire. (vi) Un $\mathcal{ED}$ (à définir ou opérationnel) possédant un modèle conceptuel CM_{ED} décrivant son univers du discours et un ou plusieurs formats $Format_{ED}$=$\{f_1, f2, ...fk\}$. Le but est d'augmenter la valeur pour un ou plusieurs secteurs organisationnels O, où la valeur peut être calculée comme suit : Valeur= $\sum_{S_i \in S_I \cup S_E} Poids(S_i, O) * Valeur(S_i)$, tel que le poids de la source peut être estimé pour un secteur organisationnel donné, la valeur nécessite des métriques pour la calculer. Pour notre cas, nous considérons comme métrique principale les concepts multidimensionnels qui permettent d'augmenter le pouvoir analytique de l'$\mathcal{ED}$. Afin de gérer la variété des vocabulaires (univers du discours) et des formats, plusieurs efforts ont été conduits. Concernant l'unification des vocabulaires, les ontologies ont longuement été utilisées pour unifier les concepts des sources. Quant à l'unification des formats, elle se fait selon trois scénarii : (i) l'utilisation d'un schéma ad-hoc, (ii) l'utilisation d'un schéma générique factorisant les schémas sources, ou (iii) l'utilisation d'un schéma pivot suffisamment riche pour représenter les schémas des sources. Il se trouve que l'environnement $\mathcal{LOD}$ représente ce schéma suffisamment riche car il offre une représentation orientée graphe unifiant plusieurs formats des sources (relationnels, XML, sémantiques, etc.). L'environnement $\mathcal{LOD}$ comporte également des ontologies consensuelles permettant la définition des ressources publiées. Ceci fait du $\mathcal{LOD}$ l'environnement 'élu' pour l'unification de la variété des sources (internes et externes). Notre choix facilite également le déploiement de l'$\mathcal{ED}$ selon plusieurs stockages cibles, que nous traitons par une implémenta-

tion orientée service. Dans cet article, nous proposons à une entreprise trois scénarios réalistes et complets (correspondants aux politiques organisationnelles citées) pour intégrer les $\mathcal{LOD}$ dans la conception des $\mathcal{ED}$ tout en répondant aux exigences de variété et de valeur ajoutée.

Cet article est structuré comme suit : La section 2 positionne les $\mathcal{LOD}$ dans le paysage des $\mathcal{ED}$. La section 3 décrit l'approche proposée. Une étude expérimentale est présentée à la section 4. La section 5 conclut notre article.

2 Travaux connexes

Les $\mathcal{LOD}$ ont intégré l'environnement $\mathcal{ED}$ où certains travaux ont consolidé les efforts faits dans les $\mathcal{ED}$ issus de sources internes pour l'unification des formalismes Ravat et al. (2017); Baldacci et al. (2017); Deb Nath et al. (2015); Etcheverry et al. (2014), et l'unification des vocabulaires en utilisant des structures ad-hoc telles que des tables de correspondance (utilisant des mesures de similarité) Ravat et al. (2017); Deb Nath et al. (2015) ou en utilisant une ontologie partagée Alberto et al. (2016). Pour l'ensemble de ces travaux, la politique organisationnelle de l'entreprise est considérée comme figée où la plupart des travaux considèrent un seul scénario d'intégration des $\mathcal{LOD}$ dans un $\mathcal{ED}$ opérationnel. Ce scénario contraint les concepteurs à gérer la variété des $\mathcal{LOD}$ selon l'$\mathcal{ED}$ cible (suivant ses contraintes techniques). Ceci se fait soit : (a) *à priori* au niveau du formalisme d'unification par une approche médiateur Ravat et al. (2017) ou une approche d'entreposage où des fragments du $\mathcal{LOD}$ son dupliqués dans l'$\mathcal{ED}$ pour différentes raisons comme la réparation des informations manquantes Alberto et al. (2016) ou l'unification des cubes internes et externes Deb Nath et al. (2015). Peu de travaux traitent en particulier le processus ETL Baldacci et al. (2017); Deb Nath et al. (2015); Kämpgen et al. (2012). (b) *A postériori* lors de l'interrogation de l'$\mathcal{ED}$ (Saad et al. (2013); Matei et al. (2014)). Contrairement aux travaux existants, notre proposition traite plusieurs scénarios liés à différentes politiques organisationnelles, et met l'accent sur l'interaction entre la variété et la valeur.

3 Approche proposée

Nous avons discuté dans l'introduction deux politiques organisationnelles réalistes pour intégrer les $\mathcal{LOD}$ dans la construction d'un $\mathcal{ED}$. Nous proposons dans ce qui suit trois scénarios (illustrés dans la figure 1) et leurs architectures associées permettant de concrétiser ces politiques.

(S1) Intégration des $\mathcal{LOD}$ en série. Ce scénario correspond à une conception *conventionnelle* d'$\mathcal{ED}$. Les $\mathcal{LOD}$ sont considérés comme étant une nouvelle source sémantique à gérer en plus des sources internes. *(S2) Intégration des $\mathcal{LOD}$ en parallèle.* Ce scénario procède à l'intégration des sources internes et $\mathcal{LOD}$, tout en supposant que l'$\mathcal{ED}$ est opérationnel. Il procède à la conception de deux processus ETL définis en parallèle. *(S3) Intégration des $\mathcal{LOD}$ à la demande.* Ce scénario correspond à un ETL à la demande ("orienté requêtes") où les données sont extraites à partir de l'$\mathcal{ED}$ existant et à partir des $\mathcal{LOD}$, ces dernières sont chargées dans l'$\mathcal{ED}$ uniquement lorsqu'elles sont nécessaires à la satisfaction des exigences exprimées en requêtes OLAP (Figure. 1-c).

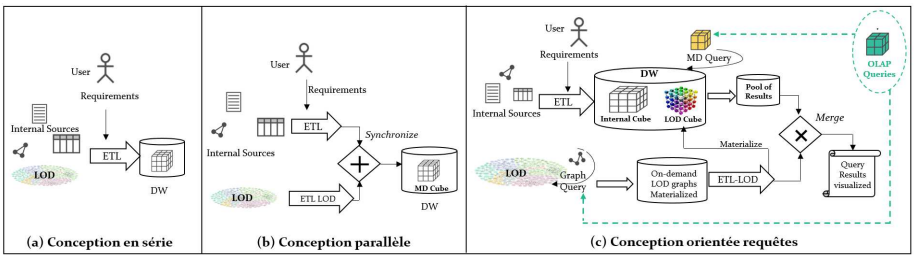


FIG. 1: Scénarios d'intégration des $\mathcal{LOD}$.

Pour la première politique citée (a), les trois scénarios (S1, S2 et S3) peuvent être conduits. Pour la deuxième politique (b), les deux derniers scénarios (S1 et S2) sont les plus pertinents. Même s'il peut être appliqué, le premier scénario n'est pas recommandé car il nécessite une redéfinition complète du processus ETL à chaque apparition de nouveaux besoins nécessitant d'extraire de la valeur à partir des $\mathcal{LOD}$. Nous nous concentrons sur les deux derniers scénarios. Dans ce qui suit, nous décrivons le processus ETL permettant l'alimentation de l'$\mathcal{ED}$ par les sources selon les deux scénarios retenus.

3.1 1^{er} V : Les scénarios d'intégration pour gérer la Variété

L'environnement ETL doit gérer la variété à trois niveaux : (a) opérateurs, (b) activités et (c) workflow. Basé sur le méta-modèle WfMC [1], nous proposons un méta-modèle ETL (Figure 2) pour gérer la variété des sources et des formats cibles. Ce modèle se base sur une représentation des données sous la forme de graphe où le modèle $\mathcal{LOD}$ est considéré comme un modèle pivot 'élu'. La définition d'un modèle pivot permet de rendre générique la représentation ETL et de gérer la variété des sources et formats de stockage cible. Le méta-modèle prend en compte un ensemble de workflows considérés comme une collection globale d'activités ETL et de transitions entre eux. Une transition détermine la séquence d'exécution des activités pour générer un workflow à partir des sources vers l'$\mathcal{ED}$ cible. Les transformations ETL sont réalisées grâce à des opérateurs ETL définis avec une signature d'éléments ayant une représentation de graphe (sous-graphe, nœuds, arrêtes, ..). Nous utilisons les dix opérateurs génériques proposés dans Skoutas et Simitsis (2007) que nous redéfinissons pour la représentation de graphe. Nous classons ces opérateurs en trois catégories : *Opérateurs sources*, *Opérateurs de transformation* et *Opérateurs de stockage*. L'ensemble des opérateurs de chaque catégorie est défini sous forme d'énumérations dans le méta-modèle. Un exemple de redéfinition des opérateurs en utilisant la représentation graphe est donné comme suit :

— $Extract(G, N_j, CS)$: Extrait de G, le nœud N_j qui satisfait les contraintes CS ;
— $Context(G, G_c, Ctx)$: Extrait de G le sous-graphe G_c qui satisfait le context Ctx ;
— ...

Nous décrivons dans ce qui suit l'utilisation de cet environnement ETL selon les deux scénarios retenus.

Intégration des $\mathcal{LOD}$ en parallèle. Ce scénario procède à l'intégration des sources internes et $\mathcal{LOD}$, tout en supposant que l'$\mathcal{ED}$ est opérationnel. Le processus ETL de $\mathcal{LOD}$

1. http ://www.wfmc.org/

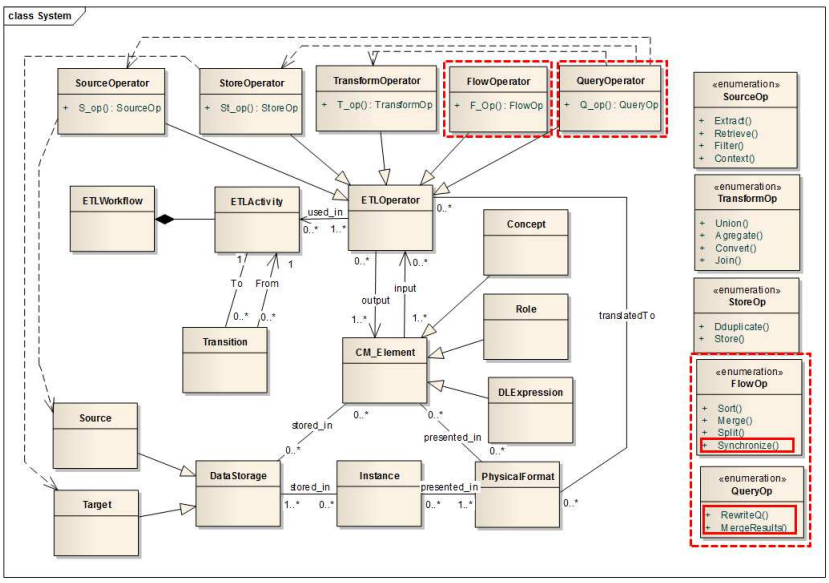


FIG. 2: Le méta-modèle de workflow ETL.

(ETL-$\mathcal{LOD}$) est généré puis synchronisé avec le processus ETL initial (ETL-$\mathcal{ED}$) défini à partir des sources internes (Fig. 1-b). Cette proposition décrit la *réaction* des parties du processus ETL affectées par un changement de flux. Ce scénario nécessite la consolidation de deux workflow (internes et externes) pour maintenir l'entrepôt cible à jour. Nous formalisons le problème de la consolidation en introduisant l'opérateur *synchronize* dans le méta modèle ETL proposé (illustré par des points rouges dans la Figure 2). Cette opération correspond à une synchronisation entre deux workflows : (i) le *Workflow actuel* : ETL workflow qui satisfait les *n* exigences en cours à l'instant *t*, et (ii) le *Nouveau Workflow* : ETL workflow qui satisfait les exigences à venir à l'instant *t+1*.

L'opérateur *Synchronize* correspond à une série d'opérations fréquemment rencontrées dans la gestion de workflow van Der Aalst et al. (2003) : (i) AND-Join : identifier les deux flux ETL, appliquer les opérations de verrouillage potentielles et effectuer l'opération de jointure entre les concepts, (ii) OR-Join : correspond à une opération de fusion de concepts et propriétés effectuée à l'aide de l'opérateur Merge et (iii) Clean : effectue un nettoyage des données, vérifie les valeurs nulles et supprime les données en double avant de les charger dans l'$\mathcal{ED}$. L'opérateur *Synchronize* est défini en utilisant la représentation de graphe comme suit : $Synchronize(G, G_i, G_j, CS)$: Synchroniser deux sous-graphes G_i et G_j en fonction de certains critères CS (AND-JOIN/OR-JOIN).

Intégration des $\mathcal{LOD}$ à la demande. Ce scénario correspond à un ETL à la demande ("orienté requêtes") où les données des $\mathcal{LOD}$ sont extraites puis chargées dans l'$\mathcal{ED}$ uniquement lorsque ces données sont nécessaires à la satisfaction des exigences exprimées en requêtes OLAP (Figure. 1-c). Ce scénario nécessite d'abord la réécriture des requêtes OLAP sur les $\mathcal{LOD}$ dans le but d'extraire le fragment répondant aux exigences (requêtes OLAP). Puis d'appliquer les transformations nécessaires sur le processus ETL dédié aux $\mathcal{LOD}$ en utilisant

la classe *Transform-Operator* du méta-modèle proposé (Figure 2). Le résultat de ce processus ETL est matérialisé en utilisant classe *Store-Operator*. Les résultats de cette opération sont d'abord intégrés dans un cube de données dédié à l'analyse des données $\mathcal{LOD}$ puis *fusionnés* avec les résultats des requêtes OLAP exécutées sur l'$\mathcal{ED}$, pour enfin être affichés à l'utilisateur final (Figure 1-c). Cette opération de fusion nécessite l'extension du méta-modèle de workflow ETL par la classe *Query-Operator* (Figure 2). Conceptuellement, cette opération est illustrée dans le méta-modèle ETL en liant la classe *Query operator* aux classes : *Source-Operator* pour extraire les fragments de $\mathcal{LOD}$, *Store-Operator* pour déployer le résultat de l'ETL $\mathcal{LOD}$ sur l'$\mathcal{ED}$ et *Transform-Operator* afin de gérer les transformations requises par le processus ETL dédié aux $\mathcal{LOD}$ (par exemple, opérations d'agrégation et de jointure). Nous avons également enrichi la classe *Query-Operator* par les méthodes *Rewrite_Query* et *MergeResult_Queries* permettant l'unification des résultats obtenus et la gestion des différentes opérations d'interrogation mentionnées. La signature de l'opérateur de fusion est défini en utilisant la représentation graphe comme suit :
- $Merge(G, G_i, G_j)$: fusionne les sous-graphes G_i et G_j en un seul graphe G.

Déploiement de l'$\mathcal{ED}$. Le déploiement de la solution se base sur le méta-modèle proposé où la classe *Store operator* offre l'opérateur ETL *Store* permettant le changement des données dans l'$\mathcal{ED}$ cible. La signature de l'opérateur Store est définie en utilisant la représentation graphe comme suit : $Store(G_T, I)$: chargement des instances notées par le nœud I dans le graphe cible G_T. La variété étant présente également au niveau cible, un $\mathcal{ED}$ peut être déployé en utilisant différents modèles hybrides horizontaux, verticaux, NoSQL, etc. Dans le but de fournir un déploiement souple et adéquat, nous proposons un déploiement orienté service : un ETL en tant que service (ETLaaS) et un stockage physique en tant que service (PSaaS). Le W3C définit un service Web comme étant un logiciel conçu pour prendre en charge l'interaction inter-machines sur un réseau. Plusieurs protocoles peuvent être utilisés, les plus répandus sont : SOAP [2] et REST [3]. Notre choix s'est porté sur le protocole REST et le format JSON (JavaScript Object Notation). REST est considéré comme un style architectural relativement léger, reposant sur des méthodes HTTP (POST, GET, PUT ou DELETE). De plus, notre solution traite les données RDF en provenance du web où la plupart des applications exposent leurs services en tant qu'API Web RESTful, principalement en raison de sa simplicité et de sa facilité d'implémentation.

3.2 2^{eme} V : Augmentation de la Valeur

Dans la section 1, nous avons formalisé la valeur comme suit :
$Valeur = \sum_{S_i \in S_I \cup S_E} Poids(S_i, O) * Valeur(S_i)$, tel que le poids de la source peut être estimé pour un secteur organisationnel donné, et la valeur nécessite des métriques pour la calculer. Le taux de valeur de chaque source $Valeur(S_i)$ est calculé en termes de taux de concepts multidimensionnels $Valeur(S_{iMD})$ et d'instances intégrées (dénotée par $Valeur(S_{iI})$) et de taux de besoins satisfaits $Valeur(S_{iB})$ suite à l'intégration de la source. Ces métriques sont choisies car elles augmentent le pouvoir analytique et décisionnel de l'$\mathcal{ED}$. Dans nos expérimentations, nous considérons le poids des sources (internes - externes) équivalents afin d'obtenir un taux de valeur représentatif de chaque. La formule de la valeur peut ainsi être précisée

2. https ://www.w3.org/TR/soap/
3. https ://www.w3. org/2001/sw/wiki/REST

comme suit :

$$Valeur(S_{iMD}) = \frac{Nbre\_Concepts(S_i)}{NbreTotal\_Concepts(ED)} \quad (1)$$

où $Nbre\_Concepts(S_i)$ et $NbreTotal\_Concepts_ED$ représentent respectivement le nombre de concepts MD obtenus de la source $_i$ et le nombre total de concepts MD de $\mathcal{ED}$.

$$Valeur(S_{iI}) = \frac{NbreInstancesInt(S_i)}{NbreTotalIns(ED)} \quad (2)$$

où $NbreInstancesInt(S_i)$ et $NbreTotalIns(ED)$ représentent le nombre d'instances intégrées la source S_i et le nombre total d'instances de l'entrepôt. Concernant la satisfaction des besoins, nous évaluons la valeur ajoutée en utilisant la valeur de chaque source comme suit :

$$Valeur(S_{iB}) = \frac{NbreReponsesBes(S_i)}{NbreReponsesReq(ED)} \quad (3)$$

où $NbreReponsesBes(S_i)$ et $NbreReponsesReq(ED)$ décrit le nombre de réponses aux requêtes (correspondant à un besoin) et le l nombre de réponses aux requêtes sur l'entrepôt.

4 Expérimentations

Nous présentons un ensemble d'évaluations pour montrer l'efficacité de notre proposition. Nous considérons le scénario où une organisation gouvernementale souhaite construire un $\mathcal{ED}$ pour analyser l'état de la recherche scientifique dans les universités de différents pays. Pour ce faire, les universités doivent fusionner leurs bases de données pour construire un $\mathcal{ED}$ qui doit répondre à certaines exigences identifiées telles que : (a) permettre aux enseignants de consulter les relevés de note des étudiants qui ont obtenu le master, (b) la popularité d'une université par année (événements Wiki), (c) obtenir la liste des publications dans une conférence (par ex. EDA) par année (disponible sur les bases dblp, ...), (d) le nom de l'auteur et la date de publication d'un livre, (e) le budget consacré aux conférences par pays, (f) les thématiques de recherche d'actualité. Supposons que quatre universités de différents pays participent à cette opération où chaque université est considérée comme étant une source de données : la France (S_{FR}), les Etats Unis (S_{USA}), l'Allemagne (S_{AL}) et l'Algérie (S_{AG}). La particularité de ces sources est qu'elles sont dérivées du référentiel relatif aux universités (LUMB [4]). Le choix du déploiement de l'entrepôt a été défini pour Oracle relationnel et graphe. En considérant uniquement les sources ci-dessus, les besoins exprimés ne sont pas entièrement satisfaits, par conséquent, l'appel à des ressources externes telles que *DBpedia* ou *Yago* [5] devient pertinent pour ajouter de la valeur à l'entrepôt cible. Pour illustrer la variété des sources, nous considérons trois catégories très répandues pour les $\mathcal{ED}$: les sources relationnelles, les sources sémantiques et les sources $\mathcal{LOD}$ caractérisées par une représentation orientée graphe.

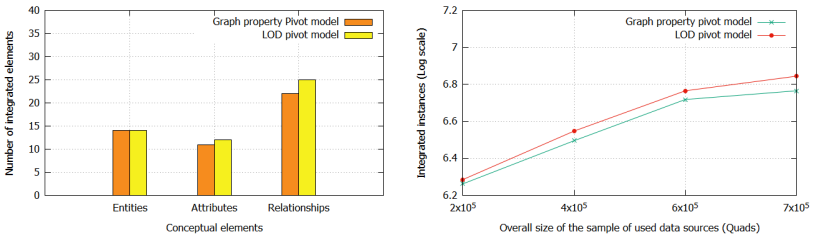
Environnement de l'expérimentation. Considérons les quatre sources internes alimentées comme suit : S_{FR} (12, 123 004), S_{AG} (8 96 962), S_{AL} (15, 3,9 x 10^5) et S_{USA} (19, 2,5 x 10^6). Afin d'évaluer l'impact de la variété des sources, les trois premières sources de données

4. http ://swat.cse.lehigh.edu/projects/lubm/
5. www.yago-knowledge.org/

sont implémentées sur le SGBD sémantique Oracle en utilisant le format N-Quads, tandis que la quatrième sur un SGBD Oracle relationnel. Notre ressource externe correspond à un fragment de DBpedia extrait à l'aide de l'opérateur de contexte appliqué sur le sujet *University*. Le fragment obtenu contient environ 7,9 x 10 ^{6} Quads.

Nos évaluations ont été réalisées sur un ordinateur portable (HP Elite-Book 840 G3) avec un processeur Intel(R) CoreTM i7-6500U 2,59 GHz et 8 Go de RAM et un disque dur de 1 To. Nous utilisons Windows10 64bits.

Evaluation de la variété. Le but de cette évaluation est d'étudier l'impact des efforts de conceptualisation sur la gestion de la variété dans l'entrepôt obtenu. Dans la première expérience, nous comparons le choix du format 'élu' qui est le format graphe du $\mathcal{LOD}$ utilisé avec le format générique proposé dans Berkani et Bellatreche (2017) (appelé modèle Graph Property). La comparaison est faite en fonction des entités, des attributs, des relations et des instances, tout en considérant les trois scénarios. Les figures 3a et 3b décrivent les résultats obtenus. Ces derniers montrent clairement que notre modèle $\mathcal{LOD}$ orienté graphe capture plus d'éléments que le modèle générique. Nous remarquons une perte d'informations lors de l'utilisation du modèle générique dûe au mappings définis. La perte d'informations est moindre lors de l'utilisation du modèle 'élu' d'une des sources car l'information issue de cette source élue (i.e. fragment $\mathcal{LOD}$) est complètement préservée.



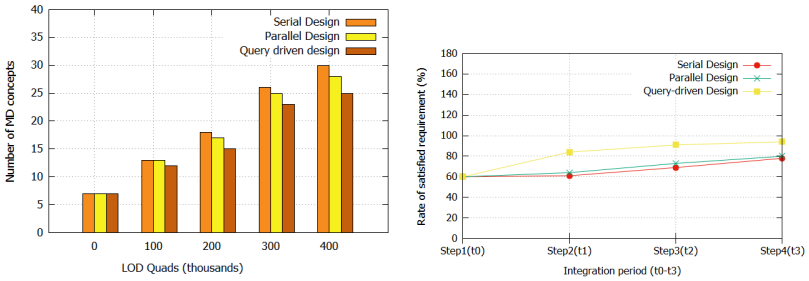
(a) Eléments conceptuels intégrés. (b) Instances intégrées.

FIG. 3: Comparison entre les modèles pivot $\mathcal{LOD}$ Vs générique $\mathcal{LOD}$ et Graph property

Évaluation de la valeur ajoutée. La seconde évaluation a été menée pour mesurer la valeur capturée par l'intégration des sources internes et des sources externes $\mathcal{LOD}$. Pour le premier critère à savoir taux de concepts multidimensionnels et d'instances intégrées, la figure 4a illustre les résultats obtenus, où le nombre de concepts et d'instances intégrés à partir des $\mathcal{LOD}$ est assez important. Il apparait clairement que la prise en compte du fragment $\mathcal{LOD}$ augmente le nombre de concepts multidimensionnels pour les trois scénarios. En comparant les trois scénarios, nous constatons qu'ils sont presque équivalents.

Pour évaluer le second critère se rapportant à la satisfaction des besoins, nous avons formulé nos besoins utilisateurs sous la forme de requêtes OLAP exécutées sur l'$\mathcal{ED}$. L'exécution des requêtes OLAP a été effectuée en quatre étapes (t_0, t_1, t_2 et t_3) durant le processus d'intégration. Avant l'instant t_0, nous avons considéré uniquement les sources internes. L'instant t_0 correspond au moment de considération des $\mathcal{LOD}$. La figure 4b décrit les résultats obtenus. Les besoins des utilisateurs satisfaits par les sources de données internes représentent $\sim$ 65%. Une fois les $\mathcal{LOD}$ intégrées, ce taux augmente considérablement jusqu'à atteindre un

taux maximum de 96%. Nous avons également remarqué que le troisième scénario (ETL à la demande) engendre le meilleur résultat et répond aux besoins des utilisateurs plus rapidement que les autres scénarios. Cela peut s'expliquer par le fait que ce scénario se concentre sur l'intégration des données relatives aux besoins des utilisateurs exprimés par des requêtes OLAP.



(a) Nombre de concepts multidimensionnels vs. $\mathcal{LOD}$ intégrées.

(b) Taux de besoins satisfaits durant le processus d'intégration des $\mathcal{LOD}$.

FIG. 4: La valeur ajoutée par l'intégration des $\mathcal{LOD}$

Le tableau 1 détaille les résultats ci-dessus et montre la valeur ajoutée des sources selon les formules données.

Scenarios	Dimensions/Mesures	Valeur(Si)$_{MD}$	Valeur(Si)$_{B}$	Valeur(Si)$_{I}$	Valeur(Instances)	Temps de réponse
Sources internes	6/1	$\sim$ 31%	60%	550K	$\sim$ 10%	1.1
Conception en série	10/7	71%	80%	7,9 x 10^6	94%	3.2
Conception parallèle	11/8	73%	84%	3,1 x 10^6	85%	2.6
Conception à la demande	12/8	74%	96%	2,9 x 10^6	84%	1.7

TAB. 1: Valeur ajoutée par l'intégration des sources internes - $\mathcal{LOD}$

.

5 Conclusion

Dans cet article, nous avons identifié la forte interaction entre deux V importants apportés par l'ère Big Data qui sont la valeur et la variété. Pour obtenir plus de valeur, les concepteurs d'$\mathcal{ED}$ doivent considérer plus de sources avec une grande variété. Un autre aspect important qui doit être considéré lors de la construction d'un entrepôt est la variété de son format de stockage. Traiter l'interaction entre les 2V a nécessité plusieurs efforts : (a) de formalisation du problème consistant à augmenter la valeur en traitant les contraintes de variété des sources (concernant leurs vocabulaires et leurs formats). Plusieurs métriques de valeur sont proposées. Cette formalisation permet d'entrevoir la valeur et variété au même niveau. (b) De conceptualisation de l'environnement ETL selon plusieurs scénarios organisationnels. Un méta-modèle ETL ainsi que le processus ETL associé sont fournis, considérant l'environnement du $\mathcal{LOD}$ comme pivot (vocabulaire et format). (c) De déploiement orienté service permettant de traiter

la variété du format de stockage cible. Nos expérimentations montrent l'efficacité de notre proposition. Nous travaillons actuellement sur la prise en compte de la variété des plateformes de déploiement, en considérant des polystores, ainsi que la considération de nouvelles contraintes liés aux autres V comme la vélocité et le volume.

Références

Alberto, A., G. Enrico, G. Matteo, R. Stefano, et R. Oscar (2016). Towards exploratory olap on linked data. In *SEBD*, pp. 86–93.

Baldacci, L., M. Golfarelli, S. Graziani, et S. Rizzi (2017). Qetl : An approach to on-demand etl from non-owned data sources. *DKE 112*, 17–37.

Berkani, N. et L. Bellatreche (2017). A variety-sensitive ETL processes. In *DEXA*, pp. 201–216.

Deb Nath, R. P., K. Hose, et T. B. Pedersen (2015). Towards a programmable semantic extract-transform-load framework for semantic data warehouses. In *DOLAP*, pp. 15–24.

Etcheverry, L., A. Vaisman, et E. ZimÃ¡nyi (2014). Modeling and querying data warehouses on the semantic web using qb4olap. In *DaWAK*, pp. 45–56.

Kämpgen, B., S. O'Riain, et A. Harth (2012). Interacting with statistical linked data via OLAP operations. In *ESWC (Satellite Events)*, pp. 87–101.

Matei, A., K. Chao, et N. Godwin (2014). OLAP for multidimensional semantic web databases. In *BIRTE*, pp. 81–96.

Ravat, F., J. Song, et O. Teste (2017). Vers un modèle unifié de données entreposées et de données ouvertes liées. concepts et expérimentations. *ISI 22*(2), 35–67.

Saad, R., O. Teste, et C. Trojahn (2013). Olap manipulations on rdf data following a constellation model. In *1st International Workshop on Semantic Statistics*.

Skoutas, D. et A. Simitsis (2007). Ontology-based conceptual design of ETL processes for both structured and semi-structured data. *Semantic Web 3*(4), 1–24.

van Der Aalst, W. M., A. H. Ter Hofstede, B. Kiepuszewski, et A. P. Barros (2003). Workflow patterns. *Distributed and parallel databases 14*(1), 5–51.

Summary

Nowadays, we are living in an open and connected world, where small, medium and large companies are looking for better data analysis. The data warehouses ($\mathcal{DW}$) is an example of these systems, which have recently experienced a drop in speed with the appearance of massive data and their different V (Volume, Variety, Velocity, Value, etc.). The advent of the Linked Open Data era ($\mathcal{LOD}$) is a great opportunity for $\mathcal{DW}$, because they add value. Taking this Value into account in the design of $\mathcal{DW}$ has to deal with the problem of the Variety of data sources and the Variety of the target storage formats of the $\mathcal{DW}$. In this paper, we present a new $\mathcal{DW}$ value and variety driven design approach, where different integration scenarios of $\mathcal{LOD}$ impacting ETL processes are investigated. Finally, we expose experiments validating our proposals.