Value-driven Approach for Designing Extended Data Warehouses

Nabila Berkani
Ecole nationale Supérieure d'Informatique, BP 68M,
16309, Oued-Smar, Alger, Algérie.
n-berkani@esi.dz

Ladjel Bellatreche
ISAE-ENSMA, France
ladjel.bellatreche@ensma.fr

Selma Khouri
Ecole nationale Supérieure d'Informatique, BP 68M,
16309, Oued-Smar, Alger, Algérie.
s-khouri@esi.dz

Carlos Ordonez
University of Houston, USA
carlos@central.uh.edu

ABSTRACT

In a very short time, the data warehouse ($\mathcal{DW}$) technology has gone through all the phases of a technological product's life: introduction on the market, growth, maturity and decline. Maturity means there is a clearly identified design life cycle plus a race and competition between companies to increase their decision-making power. Decline was signaled by the appearance of Big Data. It is therefore essential to find other challenges that will contribute to the revival of $\mathcal{DW}$ while taking advantage of the V's of Big Data. The arrival of Linked Open Data ($\mathcal{LOD}$) era is an excellent opportunity for both the $\mathcal{DW}$ academia and industry communities. $\mathcal{LOD}$ may bring an additional Value that the sources feeding a $\mathcal{DW}$ typically do not usually succeed to yield. Offering the added value of a $\mathcal{DW}$ is related to a high Variety of sources. In this paper, first, we conceptualize the variety of internal and external sources and study its impact on the ETL phase to ease the value capturing. Secondly, three scenarios for integrating $\mathcal{LOD}$ in the $\mathcal{DW}$ are given. Finally, experiments are conducted to show the effectiveness of our approach.

1 INTRODUCTION

In contrast to traditional database applications, the process of building $\mathcal{DW}$ is a complex, expensive, and time-consuming task. Knowing this risk, companies willing to conduct a $\mathcal{DW}$ project should never start unless managers are convinced that its benefits outweigh the cost, known as Return Of Investment (ROI). Survey studies conducted by analytical companies such as International Data Corporation and SAS clearly conclude that $\mathcal{DW}$ technology provides a good payback, in the sense that the average ROI for a $\mathcal{DW}$ is far above the industry average, confirming the added-value of $\mathcal{DW}$ technology.

With the arrival of Big Data, companies owning a $\mathcal{DW}$ had to change their BI strategy and align it. This alignment comes from facing the V's brought by Big Data (Volume, Variety, Velocity, Veracity). This situation pushes these companies to enhance their $\mathcal{DW}$ environment with Big Data technology, including distributed programming, cloud computing, parallel processing and so on. These technologies mainly focus on managing Volume and Velocity of data, leaving Variety as a second priority. More recently, considering *value-requirements* fixed by a company such as money invested, awed customers, increased sales, etc. has spawned another V: Value.

The $\mathcal{DW}$ value has to be evaluated considering risks [17]. One of the most important risks is the lack of satisfaction of user requirements [9]. This is due to the fact that data sources participating in the $\mathcal{DW}$ construction are not rich enough in terms of *concepts and instances*. This limitation can decrease the value of the target $\mathcal{DW}$ and consequently its ROI, where decisions will be made perhaps without data. This scenario is deeply related to the trade-off between Closed World (CWA) and Open World (OWA) Assumptions. The CWA assumption states what is not known to be true must be false, whereas OWA is the opposite. Therefore, building a $\mathcal{DW}$ only from database sources may penalize its value. To deal with the Value risk, recent studies propose augmenting traditional $\mathcal{DW}$ sources with external Web sources such as $\mathcal{LOD}$ and knowledge graphs [12]. Usually, $\mathcal{LOD}$ store data with high quality, since important efforts in curation, cleaning, entity resolution, etc. are deployed. If a company succeeds in selecting $\mathcal{LOD}$, their integration into $\mathcal{DW}$ will not affect its quality. This is because the traditional ETL processes will be augmented to deal with this new source of data. The price to pay by designers when considering $\mathcal{LOD}$ is managing data Variety. They bring a new format of data usually incompatible with traditional ones. To augment a $\mathcal{DW}$ with $\mathcal{LOD}$, important efforts in conceptualizing and managing data Variety have to be made. This variety concerns these main aspects: (a) the universe of discourse (UOD) of sources, (b) their conceptual formalisms, and (c) their physical implementations.

In the context of $\mathcal{LOD}$, their schema has the potential for being the chosen model that federates external sources, as it is open, standardized, visualizable, and associated with graph database tools, while maintaining interoperability with semantic databases and ontologies. Moreover, it allows conceptualizing variety at two levels: vocabulary (using ontologies) and formalisms (RDF schema). In this paper, we defend this $\mathcal{LOD}$-augmented scenario and we show that it significantly impacts the ETL environment including its tasks, workflows and operators. With this motivation in mind, we propose three comprehensive scenarios for a company to integrate $\mathcal{LOD}$ into the $\mathcal{DW}$ design, while meeting the Variety and added Value requirements at the conceptual level. These scenarios are distinguished based on ordering events stipulating the time when the company decides to build its $\mathcal{DW}$ and the time when it decides to connect its $\mathcal{DW}$ to a relevant $\mathcal{LOD}$. More precisely, these scenarios follow two main schedules: (a) the $\mathcal{DW}$ meets $\mathcal{LOD}$ and (b) the $\mathcal{DW}$ was designed before $\mathcal{LOD}$. In the first schedule, the $\mathcal{DW}$ is built from scratch by a simultaneous integration of internal and external data sources, whereas in (b), we assume that the $\mathcal{DW}$ was constructed well before the company decided to integrate $\mathcal{LOD}$. Therefore, such $\mathcal{DW}$ needs to continuously integrate data from local sources and $\mathcal{LOD}$. Thanks to conceptual modeling, variety is managed

at ETL level. Value is handled by introducing metrics related to requirement satisfaction.

This paper is structured as follows: Section 2 positions $\mathcal{LOD}$ in the $\mathcal{DW}$ landscape. Section 3 is related to the conceptualization of the variety and its impact on ETL. Section 4 details our scenarios integrating $\mathcal{LOD}$ in the $\mathcal{DW}$ design to add value. An experimental study is conducted in Section 5. Section 6 concludes our paper and outlines future work.

2 RELATED WORK

A couple of recent studies consider $\mathcal{LOD}$ in the process of $\mathcal{DW}$ construction, without emphasizing on value. These studies can be projected on the *conventional* $\mathcal{DW}$ life cycle design that includes: requirements definition, conceptual design, ETL, logical and physical design. Even though the experience has shown that bring requirements in forefront ensure the $\mathcal{DW}$ to be tightly tailored to the users requirements [8], in most studies considering $\mathcal{LOD}$ in $\mathcal{DW}$ design, users' requirements are either ignored or assumed defined. Our work is motivated by the importance of requirements on the $\mathcal{DW}$ system incorporating $\mathcal{LOD}$ in identifying missing concepts and instances required for $\mathcal{DW}$ value augmentation.

$\mathcal{LOD}$ works have come up with new approaches for managing variety of sources, covering only some parts of the life cycle phases, namely : (i) Conceptual level: the unification of the universe of discourse is either ignored, or handled relying on ad-hoc structures such as correspondence tables (using similarity measures) [6, 15] or a shared ontology [1]. (ii) Logical level: most studies highlight multidimensional models and map $\mathcal{LOD}$ sources to this logical format. These models are either generic multidimensional models or ad-hoc models. Other studies privilege $\mathcal{LOD}$ format based on the graph representation, for the target $\mathcal{DW}$ [5–7]. (iii) ETL : variety management process was handled using conventional ETL process that integrate and load external sources to the $\mathcal{DW}$ [5, 6]. Other studies propose an incremental fetching and storing of external sources on-demand, i.e., as they are needed during the analysis process [2, 5]. In these studies, a single scenario for integrating $\mathcal{LOD}$ is assumed, thus variety is treated partially, in the sense that it has already been dealt with in the initial $\mathcal{DW}$. (iv) Physical level: proposed scenario obliges designers to manage variety of $\mathcal{LOD}$ according the physical implementations of $\mathcal{DW}$: at unification formalism level [2, 6, 10] or at querying level ([11, 13, 16].

Contrary to existing studies, our approach proposes three main contributions: (i) a conceptualization of variety in the presence of internal and external $\mathcal{LOD}$ sources, (ii) it proposes different scenarios inspired from the organizational level of a company that decide to incorporate $\mathcal{LOD}$ within its internal sources. (iii) Our approach analyzes then the impact of these variety scenarios on the Value.

3 VARIETY AND VALUE MANAGEMENT

3.1 First V: Variety

In this section, we provide a conceptualization of the ETL environment covering the three scenarios proposed. The ETL processes includes: (a) operators, (b) activities and (c) work-flow. By the means of the WfMC[1], we propose a metamodel to handle the variety of internal and external sources (Figure 1). An ETL workflow is the global collection of ETL activities and transitions between them. A transition determines the execution sequence of activities to generate a data flow from sources to the target $\mathcal{DW}$. The ETL activities are defined at the conceptual level in order to manage $\mathcal{UOD}$. ETL operators manage instances that are stored in the sources according to a defined format (relational, semantic, etc). The graph model of $\mathcal{LOD}$ is used as a pivot model, and internal sources are mapped to this format. A mapping effort from internal sources within pivot model is needed. ETL operators are redefined using the graph format of $\mathcal{LOD}$, that we consider as the elected pivot model. ETL process allow extracting the instances, transform them and load them (*ETLOperator* Class). We used the ten generic operators proposed in [18] that we classify into three groups: *Source operators*, *Transform operators* and *Store operators*. In the next Section, we extend these operators to deal with our scenarios. The set of operators of each group is defined as enumerations in the model. An example of redefinition of operators on $\mathcal{LOD}$ graph format is given:
$Extract(G, N_j, CS)$: extracts, from G, the node N_j satisfying constraint CS;
$Context(G, G_c, Ctx)$: extracts from G a sub-graph G_c that satisfy the context defined in Ctx.

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

3.2 Second V: Value

The construction of value-based $\mathcal{DW}$ is formalized as follows: given: (i) a set of internal sources $S_I = \{Si_1, Si_2, ...Si_m\}$, (ii) a set of external $\mathcal{LOD}$ sources : $S_E = \{Se_1, Se_2, ...Se_m\}$. Each internal and external source has its own format $FormatS_j$ and its conceptual model CM_j describing its universe of speech. (iii) a set of requirements G to be satisfied. (vi) A $\mathcal{DW}$ (to be defined or operational) with its conceptual model CM_{DW} describing its $\mathcal{UOD}$ and one or more formats $Format_{DW}=\{f_1, f_2, ...f_k\}$. (v) The value-requirement fixed by the company. The added value of the target warehouse ($Value$) regarding a given value-requirement can be calculated as follows:

$$Value = \sum_{S_i \ in S_I \cup S_E} Weight(S_i) * Value(S_i) \quad (1)$$

where $weight(S_i)$ describes the weight of each source and it can be estimated for a given organizational sector.

In our work, the value-requirement concerns the $\mathcal{DW}$ requirement satisfaction which is strongly related to multidimensional concepts and instances provided by sources. Therefore, we propose three value metrics associated to each source S_i: $ValueReq$, $ValueMD$ and $ValueInst$ that are defined as follows, note that these equations measure the percentage of value added from external sources in terms of MD concepts and requirements to be met (rate needs of table 1). :

$$ValueReq(S_i) = \frac{NumberReponsesReq(S_i)}{NumberReponsesReq(DW)} \quad (2)$$

where $NumberReponsesReq(S_i)$ indicates the number of results of the queries expressing the initial requirements on the source S_i and $NumberReponsesReq(DW)$ represents the number of results of the queries expressing the requirements on the target $\mathcal{DW}$.

$$ValueMD(S_i) = \frac{Number\_Concepts(S_i)}{TotalNumber\_Concepts(DW)} \quad (3)$$

where $Number\_Concepts(S_i)$ is the number of multidimensional concepts of $\mathcal{DW}$ schema by integrating the source $_i$ and

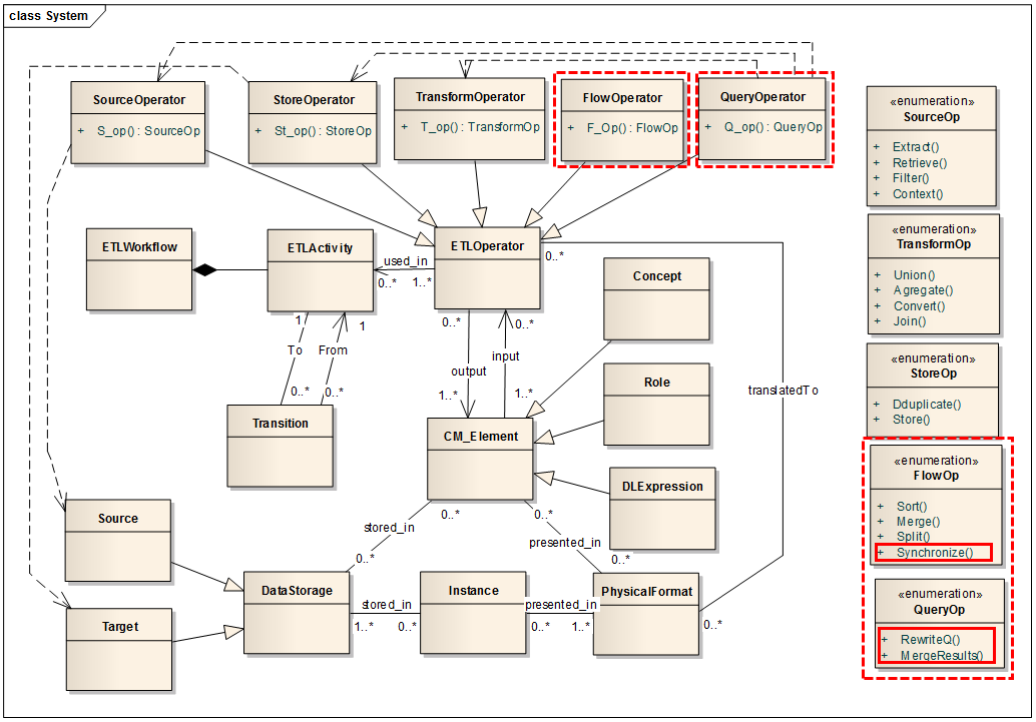


Figure 1: ETL workflow MetaModel.

$TotalNumber\_Concepts(DW)$) describes the total number of multidimensional concepts of $\mathcal{DW}$.

$$ValueInst(S_i) = \frac{NumberInstancesInt(S_i)}{TotalIns(DW)} \quad (4)$$

where $NumberInstancesInt(S_i)$ and $TotalIns(DW)$ represent the number of instances of $\mathcal{DW}$ by integrating the source S_i and the total number of instances of the $\mathcal{DW}$.

These metrics could be enriched by considering both completeness and consistency. Several metrics to measure quality of referential integrity exists [14]. We could revisit them to measure Value, combined with other Vs.

4 OUR SCENARIOS

Three main scenarios reflecting the policies of a company for incorporating $\mathcal{LOD}$ are proposed :

4.1 Serial Design

It is feasible when a company decides to build its $\mathcal{DW}$ by considering both internal and external sources from scratch. This scenario follows a *conventional* $\mathcal{DW}$ design. $\mathcal{LOD}$ is considered as a new semantic source to manage in addition to internal sources. An ETL process is defined considering all the sources. This scenario is not realistic because it requires redefinition of the whole ETL process each time new requirements are needed for extracting value from $\mathcal{LOD}$.

4.2 Parallel design

This scenario assumes that the target $\mathcal{DW}$ is operational and keeps integrating data from internal sources and $\mathcal{LOD}$. The ETL process from $\mathcal{LOD}$ is generated and then synchronized with the initial ETL process from internal sources before the loading step. This proposal describes the *reaction* of the parts of the ETL process affected by a flow change. It requires the consolidation of two parallel ETL flows (internal and external) which is needed to keep the target warehouse up to date. We formalize the problem of consolidation, by introducing the main design operation *synchronize*. It corresponds to a synchronization between: (i) *Current flow*: existing ETL flows satisfying the n current information requirements at t time and (ii) *New flow*: ETL flow satisfying the upcoming requirement at (t+1) time. Synchronize corresponds to a workflow pattern and corresponds to a series of operations commonly encountered in workflow management: (i) AND-Join: identify both ETL flows, apply potential deadlocks and perform the join operation between concepts, (ii) OR-Join: corresponds to a merger operation of concepts and properties done using Merge operator and (iii) Clean: performs a data cleaning, checks null values and deletes duplicate data before loading in the target $\mathcal{DW}$. Synchronize operator is defined using the graph format as follows:

- $Synchronize(G, G_i, G_j, CS)$: Synchronize two sub-graphs G_i and G_j based on some criteria CS (AND-JOIN/OR-JOIN).

4.3 Query-driven Design

This scenario corresponds to the on-demand ETL to feed the target $\mathcal{DW}$. Here, data are incrementally fetched from existing $\mathcal{DW}$ and $\mathcal{LOD}$ (in case where it is necessary), then loaded into the $\mathcal{DW}$ only when they are needed to answer some cubes queries. This scenario requires rewriting the cubes queries on $\mathcal{LOD}$, extract required fragments of LOD (using *Context* operator), apply the transformations required (using the *Class Transform-Operator*) by mean on an ETL process dedicated to $\mathcal{LOD}$ and then materialize (using *Store* operator) the resulting graphs in case they are required later. The results are first integrated in a data cube reserved for $\mathcal{LOD}$ data analysis, and final results of queries are *merged* with the results of cubes queries executed on

the internal $\mathcal{DW}$ in order to display the query result to the end user.

This requires the extension of the ETL workflow meta model by Query operator class (illustrated by red dots in Fig. 1). The process of this scenario is conceptually illustrated in ETL meta model by linking *Query operator* Class to the classes: *Source Operator* (for extracting $\mathcal{LOD}$ of queries), *Store Operator* (for materializing $\mathcal{LOD}$) and *Transform Operator* in order to handle the transformations required by the ETL process dedicated to $\mathcal{LOD}$ (e.g., aggregation and join operations). We also added the methods *Rewrite_Query* and *MergeResult_Queries* (unify the results obtained) to the *Query operator* Class to manage the different querying operations mentioned. Merge operator is defined using the graph format as follows:
- $Merge(G, G_i, G_j)$: merges two sub-graphs G_i and G_j into one graph G.

5 EXPERIMENTAL STUDY

In this section, we carry out a set of experiments to show the effectiveness of our proposal in terms of managing the variety and augmenting the value of the final $\mathcal{DW}$.

Experimental Setup. Let us consider the *Film Academy Awards* organizations from four countries considered as internal data sources to be integrated as follows: *French Cesar awards* (12, 123 004 triples), *Deutscher Filmpreis* (8 96 962 triples), *India IIFA Awards* (15, 3,9 x 10^5 triples) and *USA Oscar* (19, 2,5 x 10^6 record sets). Let us assume that these organizations collaborate to globally analyze the cinematography industry. We considered a set of (15) analytical requirements (eg. the popularity of an actor/actress by year). The first three data sources are implemented on Oracle semantic DBMS using N-Quads format, while the fourth one on a traditional (Relational) Oracle DBMS. Our external resource corresponds to a fragment of DBpedia extracted using the context operator applied to *Movies*. The obtained fragment contains around 7,9 x 10^6 graph Quads. Our evaluations of the ETL processes according to the different scenarios were performed on a laptop computer (HP Elite-Book 840 G3) with an Intel(R) CoreTM i7-6500U CPU 2.59 GHZ and 8 GB of RAM and a 1 TB hard disk. We use Windows10 64bits.

Variety Evaluation. The purpose of this evaluation is to study the impact of the efforts in conceptualizing variety in the obtained warehouse. In the first experiment, we compare the impact of our elected graph model against a pivot metamodel proposed in [3] (called graph property pivot model) in terms of concepts, attributes, relationships and instances. To conduct this experiment, we considered our three scenarios. Figures 2a and 2b summarize the obtained results by averaging the number of elements per scenario. It clearly shows that our elected graph model captures more elements than the pivot metamodel. This is because all elements satisfying the requirements of the $\mathcal{LOD}$ fragment are 100% materialized in the warehouse.

Added Value Evaluation. The second experiment was conducted to measure the value captured by the obtained $\mathcal{DW}$. We use two criteria representing the rate of multidimensional concepts and integrated data and the rate of satisfied requirements during the integration of $\mathcal{LOD}$. Figure 3a illustrates the obtained results. The consideration of $\mathcal{LOD}$ fragment increases the number of multidimensional concepts for the three scenarios. By comparing the three scenarios, we figure out that they are almost equivalent.

To evaluate the second criterion, we have formulated our user requirements in the form of cubes queries executed once on the target $\mathcal{DW}$. The execution of the cubes queries was carried out in four time stages (*t0*, *t1*, *t2* and *t3*) during the integration process. The time *t0* corresponds to the time of considering $\mathcal{LOD}$ in addition to internal sources. Figure 3b describes the obtained results that demonstrate that before taking into account $\mathcal{LOD}$ in the integration process, the user requirements that are satisfied by internal data sources represent ~ 65%. Once $\mathcal{LOD}$ integration process has begun, we remark that this rate increases considerably until reaching a maximum rate of 96%. We also noticed that the third scenario (Query driven design) gives the best result and meets the user needs faster than the other scenarios. This can be explained by the fact that this scenario focuses on integrating data that correspond to specific queries reflecting user needs.

Table 1 extends the above results and demonstrates the value added by considering $\mathcal{LOD}$ in the design of $\mathcal{DW}$. A comparison is given between our proposal considering the three scenarios and previous work [4], on the basis of some criteria identified during the experimentation. These results clearly indicate that the consideration of $\mathcal{LOD}$ data offers a value-added in terms of the final number of dimensions (Dim) and measures (Meas), the size of the target $\mathcal{DW}$ and rates of satisfied requirements than classical approaches.

Scenarios	Dim/Meas	Rate needs	Input Size	Response time
Internal Sources	6/1	60%	550 x 10^3	1.1
Serial Design	10/7	80%	7.9 x 10^6	3.2
Parallel Design	11/8	84%	3.1 x 10^6	2.6
Query driven design	12/8	96%	2.9 x 10^6	1.7

Table 1: Value added by considering $\mathcal{LOD}$

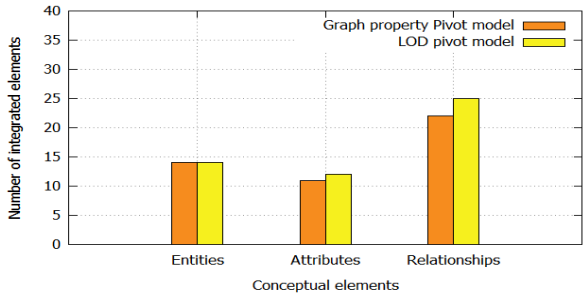
.

The evaluation of scenario presented in the example was made by users who evaluated *80%* of concepts and instances in total. They judged that 97% of the concepts and 91% of the instances in the sample are correct and 96% of requirements are satisfied. Additionally, adding $\mathcal{LOD}$ only takes 1% of additional time (1-3 secs), which we may consider as fast (cf. Table 1).

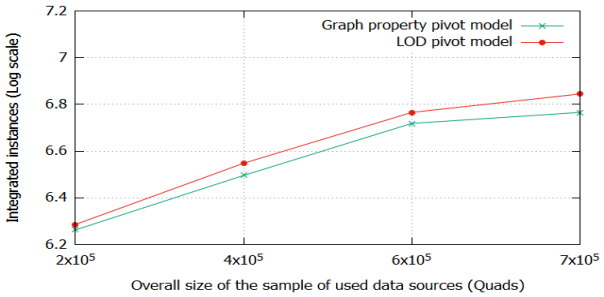
6 CONCLUSION

In this paper, we show that the $\mathcal{DW}$ technology is still alive and the advances brought by Big Data and the explosion of $\mathcal{LOD}$ contribute to its renaissance. We consider two V's related to value and variety. $\mathcal{LOD}$ are viewed as external sources that contribute to increasing the variety of the whole sources. We formalized the ETL environment and proposed a pivot meta model ETL workflow. Based on modeling and meta-modeling efforts, the variety is nicely managed. Regarding value of the target $\mathcal{DW}$, we have defined metrics associated to initial requirement satisfaction. Our scenarios for integrating $\mathcal{LOD}$ into the $\mathcal{DW}$ design have been proposed. They are: (i) $\mathcal{LOD}$ and internal sources are physically materialized in the $\mathcal{DW}$ and (ii) both $\mathcal{LOD}$ and $\mathcal{DW}$ query results are merged. Our validation showed that adding $\mathcal{LOD}$ increases the value of the target warehouse represented by the initial user satisfaction. We also realized that the incorporation of the $\mathcal{LOD}$ is inexpensive in terms of development and fast thanks to our variety management.

We are currently working on the development of a CASE tool that deals with internal and external sources. Another issues concerns the risk of integrating the $\mathcal{LOD}$ in a $\mathcal{DW}$ and detecting

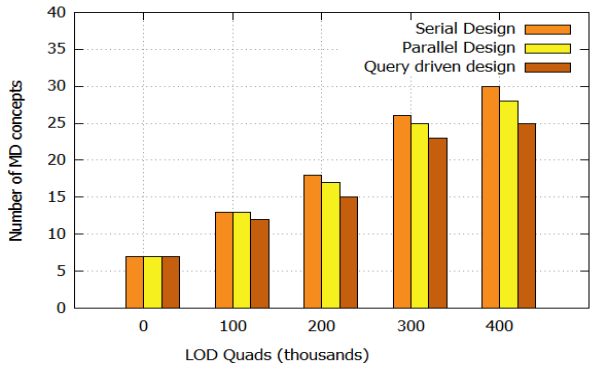


(a) Conceptual elements integrated using Graph LOD and Graph Property.

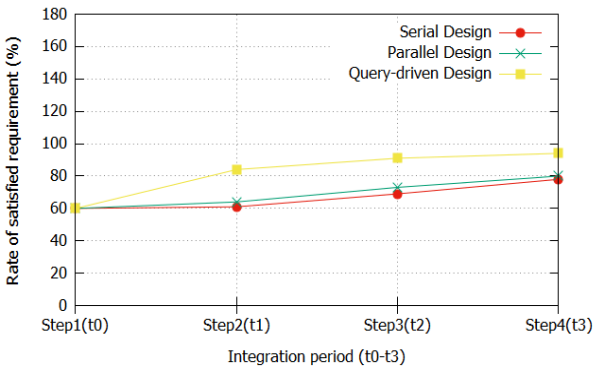


(b) Instances integrated using Graph LOD and Graph Property.

Figure 2: Comparison between $\mathcal{LOD}$ and Graph property pivot models



(a) Number of multidimensional concepts vs. integrated $\mathcal{LOD}$.



(b) Rate of satisfied requirement during the integration of $\mathcal{LOD}$.

Figure 3: The rate of satisfied requirements during the integration of $\mathcal{LOD}$

and repairing inconsistency and incompleteness in external data sources by considering other V's: Veracity .

REFERENCES

[1] A. Abelló Gamazo, E. Gallinucci, M. Golfarelli, S. Rizzi Bach, and O. Romero Moral. Towards exploratory olap on linked data. In *SEBD*, pages 86–93, 2016.

[2] L. Baldacci, M. Golfarelli, S. Graziani, and S. Rizzi. Qetl: An approach to on-demand etl from non-owned data sources. *DKE*, 112:17–37, Nov 2017.

[3] N. Berkani and L. Bellatreche. A variety-sensitive ETL processes. In *DEXA (2)*, pages 201–216, 2017.

[4] N. Berkani, L. Bellatreche, and S. Khouri. Towards a conceptualization of ETL and physical storage of semantic data warehouses as a service. *Cluster Computing*, 16(4):915–931, 2013.

[5] A. Berro, I. Megdiche, and O. Teste. Graph-based ETL processes for warehousing statistical open data. In *ICEIS 2015*, pages 271–278, 2015.

[6] R. P. Deb Nath, K. Hose, and T. B. Pedersen. Towards a programmable semantic extract-transform-load framework for semantic data warehouses. In *DOLAP*, pages 15–24, 2015.

[7] L. Etcheverry, A. Vaisman, and E. Zimányi. Modeling and querying data warehouses on the semantic web using qb4olap. In *DaWAK*, pages 45–56, 2014.

[8] P. Giorgini, S. Rizzi, and M. Garzetti. Goal-oriented requirement analysis for data warehouse design. In *ACM DOLAP*, pages 47–56, 2005.

[9] M. Golfarelli and S. Rizzi. A survey on temporal data warehousing. *IJDWM*, 5(1):1–17, 2009.

[10] B. Kämpgen and A. Harth. Transforming statistical linked data for use in OLAP systems. In *I-SEMANTICS*, pages 33–40, 2011.

[11] B. Kämpgen, S. O'Riain, and A. Harth. Interacting with statistical linked data via OLAP operations. In *ESWC*, pages 87–101, 2012.

[12] N. Konstantinou and et al. The VADA architecture for cost-effective data wrangling. In *SIGMOD*, pages 1599–1602, 2017.

[13] A. Matei, K. Chao, and N. Godwin. OLAP for multidimensional semantic web databases. In *BIRTE*, pages 81–96, 2014.

[14] C. Ordonez and J. García-García. Referential integrity quality metrics. *Decision Support Systems*, 44(2):495–508, 2008.

[15] F. Ravat, J. Song, and O. Teste. Designing multidimensional cubes from warehoused data and linked open data. In *RCIS*, pages 1–12, 2016.

[16] R. Saad, O. Teste, and C. Trojahn. Olap manipulations on rdf data following a constellation model. In *Workshop on Semantic Statistics*, 2013.

[17] T. P. Sales and et al. The common ontology of value and risk. In *ER*, pages 121–135, 2018.

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