Data Lakes: Trends and Perspectives

Franck Ravat[1] and Yan Zhao[1,2(✉)]

[1] Institut de Recherche en Informatique de Toulouse, IRIT-CNRS (UMR 5505), Université Toulouse 1 Capitole, Toulouse, France
{Franck.Ravat,Yan.Zhao}@irit.fr
[2] Centre Hospitalier Universitaire (CHU) de Toulouse, Toulouse, France

Abstract. As a relatively new concept, data lake has neither a standard definition nor an acknowledged architecture. Thus, we study the existing work and propose a complete definition and a generic and extensible architecture of data lake. What's more, we introduce three future research axes in connection with our health-care Information Technology (IT) activities. They are related to (i) metadata management that consists of intra- and inter-metadata, (ii) a unified ecosystem for companies' data warehouses and data lakes and (iii) data lake governance.

Keywords: Data lake · Architecture · Metadata

1 Introduction

In the big data era, a great volume of structured, semi-structured and unstructured data are created much faster than before by smart-phones, social media, connected objects, and other data creators. These data have a great value for companies' Decision Support System (DSS) whose cornerstone is built upon data. Nevertheless, handling heterogeneous and voluminous data is especially challenging for DSS. Nowadays, Data Warehouse (DW) is a commonly used solution in DSS. Data have been extracted, transformed and loaded (ETL processes) according to predefined schemas. DW is popular thanks to its fast response, consistent performance and cross functional analysis. However, according to [4,5], DWs are not adapted for the big data analytics for the following reasons: (i) only predefined requirements can be answered. (ii) some information is lost through ETL processes. And (iii) the cost of a DW can grow exponentially because of the requirements of better performance, the growth of data volume and the complexity of database.

To face the challenges of big data and the deficiencies of DW, Dixon [4] put forward the concept data lake (DL): "If a data warehouse may be a store of bottled water - cleansed and packaged and structured for easy consumption - the data lake is a large body of water in a more natural state." This explication sketches the outline of DL, but it can not be considered as a formal definition. DL is a relatively new concept. Even though there are some so-called DL solutions in the market, there is not a standard definition nor an acknowledged architecture.

https://doi.org/10.1007/978-3-030-27615-7_23

The goal of this prospective/survey paper is twofold. Firstly, we summarize the state of the art work and present a more complete vision of DL concept and a generic architecture. Secondly, we present future research axes by identifying major issues that appeared in our health-care IT activities. The remainder of the paper is organized as follows: Sect. 2 introduces the DL concept, we analyze different definitions and propose our own definition; Sect. 3 discusses DL architectures and introduces a generic and extensible architecture; Sect. 4 describes future research axes that includes metadata management, position of a DL in an information system and data lake governance.

2 Data Lake Concept

2.1 State of the Art

Data lake, as a relatively new concept, is defined in both scientific community and industrial world [3,5,7,15,18,21,25,31]. All the existing definitions respect the idea that a DL is a repository storing raw data in their native format. Yet, different definitions have different emphases. Regarding input, [5] introduces that the input of a DL is the data within an enterprise. Regarding process, [21] emphasizes that there is no process during the ingestion phase and [3,7,21,25] introduce that data will be processed upon usage. Regarding architecture, [5] presents that DLs are based on an architecture with low cost technologies. Regarding governance, metadata management is emphasized in [7,31]. And regarding users, [18] presents that data scientists and statisticians are DL users.

2.2 Dara Lake Definition

Existing definitions have evolved over time from experience feedback. Nevertheless, as mentioned in the previous paragraph, these different definitions are vague, they are not integrated with each other or even contradictory. To be as complete as possible, we propose a definition that includes input, process, output and governance of data lakes.

In the context of big data analytics, user requirements are not clearly defined at the time of the initial design and the implementation of a DL. A data lake is a big data analytics solution that ingests heterogeneously structured raw data from various sources (local or external to the organization) and stores these raw data in their native format, allows to process data according to different requirements and provides accesses of available data to different users (data scientists, data analysts, BI professionals etc.) for statistical analysis, Business Intelligence (BI), Machine Learning (ML) etc., and governs data to insure the data quality, data security and data life-cycle.

3 Data Lake Architecture

To the best of our knowledge, there does not exist an acknowledged DL architecture in literature. Firstly, we present different existing architectures and then propose a generic and extensible architecture.

3.1 State of the Art

Data lake functional architecture has evolved from mono-zone to multi-zone, and it is always presented with technical solutions.

The first vision of DL architecture is a flat architecture with a mono-zone that stores all the raw data in their native format. This architecture, closely tied to the HADOOP environment, enables load heterogeneous and voluminous data with low cost. Nevertheless, it does not allow users to process data and does not record any user operations.

A second vision of DL architecture contains five data ponds [10]. A *Raw data pond* that stores the just ingested data and the data that do not fit in other ponds. *Analog, application and textual data ponds* stores classified data from raw data pond by their characteristics. And *achival data pond* stores the data that are no longer used. This architecture classifies different types of data and achieves useless data, which make data finding faster and data analytics easier. However, the division of different ponds, especially the archival pond can not ensure the availability of all the raw data, contradicts the general recognition of DL which is to ingest all the raw data and process them upon usage.

To overcome these drawbacks, a third vision of architecture with multi-zones is proposed with a more diverse technological environment in the academic and industrial world. The author of [22] presents Amazon Web Services (AWS) DL architecture with four zones: ingestion, storage, processing and govern & secure. Raw data are loaded in the ingestion zone. The ingested raw data are stored in the storage zone. When data are needed, they are processed in the processing zone. The objective of Govern & secure zone is to control data security, data quality, metadata management and data life-cycle. The author of [19] separates the data processing zone into batch-processing and real time processing zones. He also adds a processed data zone to store all the cleansed data. Zaloni's DL architecture [14] separates the processing and storage zones into refined data zone, trusted data zone and discovery sandbox zone. The refined zone allows to integrate and structure data. Trusted data zone stores all the cleansed data. Data for exploratory analysis moves to the discovery sandbox.

As mentioned, a lot of DL architectures are supported with technical solutions. They are not independent of the inherent technical environment. Consequently, none of the existing architectures draws a clear distinction between functionality-related and technology-related components. What's more, the concept of multi-zone architecture is interesting and deserves further investigations. We believe that some zones are essential, while others are optional or can be regrouped. Concerning the essential zones, based on our DL definition, a data lake should be able to ingest raw data, process data upon usage, store processed data, provide access for different uses and govern data.

3.2 Data Lake Functional Architecture

Unlike several proposals, we want to distinguish functional architecture from technical architecture. Because a functional architecture concerns the usage perspective and it can be implemented by different technical solutions. By adopting

to the existing DL architectures and avoiding their shortcomings, we propose a functional DL architecture (see Fig. 1), which contains four essential zones, and each zone, except the govern zone, has a treatment area (dotted rectangle) and a data storage area that stores the result of processes (gray rectangle):

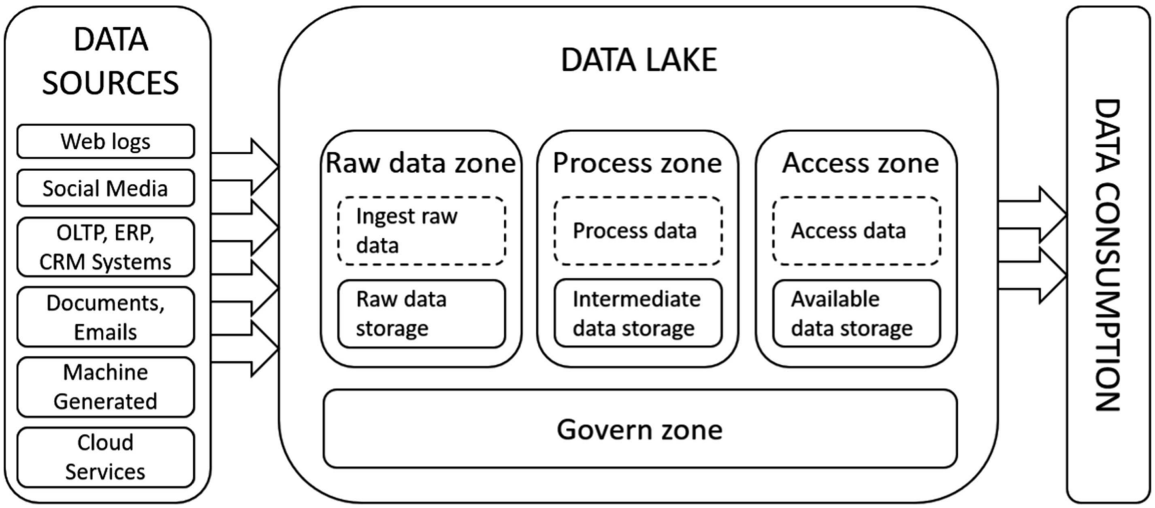

Fig. 1. Data lake functional architecture.

- *Raw data zone*: all types of data are ingested without processing and stored in their native format. The ingestion can be batch, real-time or hybrid. This zone allows users to find the original version of data for their analytics to facilitate subsequent treatments. The stored raw data format can be different from the source format.
- *Process zone*: in this zone, users can transform data according to their requirements and store all the intermediate data. The data processing includes batch and/or real-time processing. This zone allows users to process data (selection, projection, join, aggregation, etc.) for their data analytics.
- *Access zone*: the access zone stores all the available data for data analytics and provides the access of data. This zone allows self-service data consumption for different analytics (reporting, statistical analysis, business intelligence analysis, machine learning algorithms).
- *Governance zone*: data governance is applied on all the other zones. It is in charge of insuring data security, data quality, data life-cycle, data access and metadata management.

To exemplify our architecture, we propose an example of implementation (Fig. 2). Raw datasets (RD1, RD2) are ingested in data lake and stored in the raw data zone in their native format. Data are processed in the process zone and all the intermediate datasets (PD1, PD2, PD3, PD4) are stored in this area too. All the available data (AD1, AD2, AD3) are stored in the access zone for data consumption.

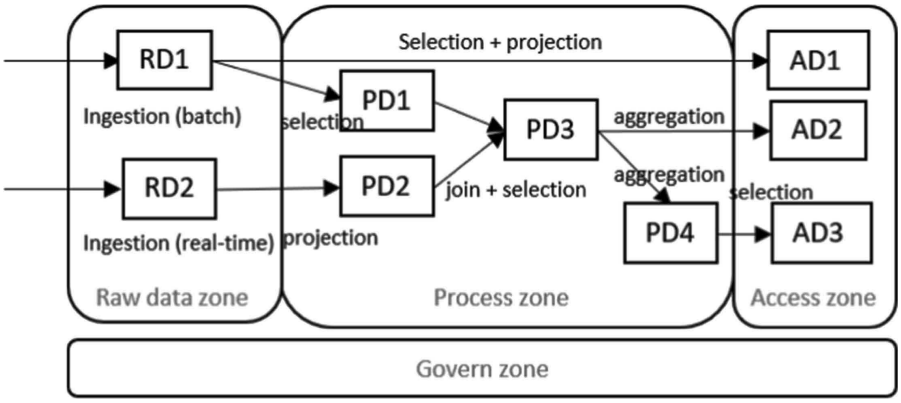

Fig. 2. An implementation of the data lake functional architecture.

4 Future Research Axes

The University Hospital Center (UHC) of Toulouse owns a great amount of data produced by different applications, it can also access to many external data. In order to facilitate data analytics to improve medical treatments, UHC of Toulouse lunched a project of DL to combine data from different individual sources. In this context, we encounter some problems: How to integrate a DL in the existing DSS? How to ensure the quality of data analytics by tracing back to the various transformations of data since the ingestion? Based on the questions that we are facing, we propose some research axes.

4.1 Integration of a Data Lake in an Information System

In a data lake, different users can access and process data for data exploration or statistical analysis for the purposes of decision making. Thus, DLs should be considered as one part of the DSS in enterprises' Information Systems (IS). Nowadays, the commonly used DSS solution is DW. According to the authors of [5,15,18], DLs and DWs are both created for extracting value of data to support decision makings but they also have differences. DWs are data repositories which store cleansed data based on predetermined schema. DLs ingest all types of raw data in their native format with low cost technologies to provide more flexibility and scalability. Regarding the similarities and differences, some questions like how do a DL and DWs work together, will a DL replace DWs need to be answered.

Many papers compared DLs and DWs but only a few papers introduced the impact of a DL for a data management ecosystem. Some authors present DL as a advanced version of DW [14,31]. The author of [5] introduces data lake cloud which is an elastic data storing and computing platform, DWs are constructed based on the data in the data lake. The author of [15,18] introduced that a DL can be fed by DWs and a DL can also be the source of DWs.

We think DLs should coexist with DWs because they have different objectives and users, a DL cannot simply replace a DW. To the best of our knowledge,

a coexisting ecosystem has not been studied and implemented. To propose a such ecosystem, different research problems are induced. The first problem relates to the functional architecture definition which must determine precisely the information flow between DWs and a DL. If DWs feed a DL, the questions to solve are: where are the extracted data (from a particular DW or DM (Data Mart))? Do the data get into the ingestion zone or the process zone of a DL (because they have been processed in the DW)? Is it the same type of ingestion without data transformation as the ingestion from other data sources? If a DL is the source of a DW, the questions are similar on the sources (data in process zone or access zone), the target (a particular DW or DM) and the transfer process (ETL, ELT or only EL). Once these problems are solved, we must answer the problem on refreshing and updating data. We need to therefore answer the following questions: when is the data transformation done (real time, near real time, batch)? What is the type of refreshment (never, complete, incremental)? Finally, the third issue concerns the technical architecture which ensures the power and reliability of data flows between a DW and a DL).

4.2 Metadata

The main idea of data lake is to ingest raw data without process and process data upon usage. Therefore, data lakes keep all the information and have a good flexibility. Nevertheless, data lakes, which contain a lot of datasets without explicit models or descriptions can easily become invisible, incomprehensible and inaccessible. So that it is mandatory to set up a metadata management system for DL. In fact, the importance of metadata has been emphasized in many papers [1,7,31]. The first research problem that needs to be solved is the content of the metadata.

Data lake metadata, inspirited by DW metadata classification [16,26], are mainly classified in two ways. The first classification has three categories: *technical metadata* for data type, format and data structure, *operational metadata* for process history and *business metadata* for the descriptions of business objective. This classification focus on each single dataset, but not on the relationships between different datasets. Nevertheless, for a DL, datasets relationships are important to help users to find relevant datasets and verify data lineage. The second classification has two categories: *inter-metadata* for the relationships between data and *intra-metadata* for specifying each single dataset [17]. Inter-metadata is classified into dataset containment, provenance, logical cluster and content similarity by the author of [9]. Intra-metadata is classified into data characteristics, definitional, navigational, activity, lineage, rating and assessment [2,6,30]. The second classification is evolved, but it can still be improved. Some sub-categories are not adapted, for instance, the rating sub-category is not adaptive, because a DL can be accessed by different users who have different objective [30]. Moreover, the classification can be extended by some sub-categories, for instance, data sensitivity needs to be verified. Due to the specificity of DL with different zones including both storage and transformation processes, it is

important to include intra- and inter-metadata. Therefore, we propose an metadata classification which contains inter- and intra-metadata and adapted subcategories:

- For *inter-metadata* [28], we propose to integrate *Dataset containment* which means a dataset is contained in another dataset. *Partial overlap* which means that some attributes with corresponding data in different datasets overlap. *Provenance* which means that one dataset is the source of another dataset. *Logical clusters* which means that some datasets are from the same domain (different versions, duplication etc.). And *Content similarity* which means that different datasets share the same attributes.
- For *intra-metadata* [28], we retain data characteristics, definitional, navigational and lineage metadata proposed in [2] and add the access, quality and security metadata.
  - *Data characteristics*: attributes describing a dataset, such as identification name, size and creation date.
  - *Definitional metadata*: datasets' meanings. Structured and unstructured datasets can be described semantically with a textual description or a set of keywords (vocabularies). Definitional metadata help users to understand datasets and make their data exploitation easier.
  - *Navigational metadata*: location information like file paths and database connection URLs.
  - *Lineage metadata*: information concerns data life-cycle. For example, data source, data process history.
  - *Quality metadata*: data consistency and completeness [26] to ensure dataset's reliability.
  - *Security metadata*: data sensitivity and access level. Some datasets may contain sensitive information that can only be access by certain users. Security metadata can support the verification of access.

The second research problem is to define an appropriate solution for metadata management. To the best our knowledge, there isn't a general metadata management system that works on heterogeneous data for the whole data life-cycle in DLs. We only have partial solutions in the literature. Some works concentrate on the detection of relationships between different datasets [1,9,27]. Some other work focus on the extraction of metadata for unstructured data (mostly textual data) [27,29].

To propose a appropriate solution, different research problems are induced. The first one relates to the way that metadata are stored: what is the conceptual schema of metadata? Which attributes should be recorded? How should we store the metadata (distributed RDBMS, NOSQL DBMS, RDF Stores etc.)? The second problem relates to the data feeding process: how to extract metadata from structured data as well as semi or unstructured data. What is the text analysis engine to detect automatically the keywords from unstructured data?

4.3 Data Lake Governance

A DL ingests and stores various types of data and can be accessed by different users. Without best practices in place to manage it, many issues that concern accessing, querying, and analyzing data can appear [23]. Given this context, data lake governance is required. In the state of the art work we find some partials solutions. A "Just-enough Governance" for DLs is proposed by [8] with data quality policy, data on-boarding policy, metadata management and compliance & audit policy. [11] indicated that the data governance needs to ensure data quality and availability throughout the full data life-cycle. [24] presented data governance with data lineage, data quality, security and data life-cycle management. However, these papers don't integrate the data lake governance through a complete vision.

We firstly propose a definition: the data governance of DL enables policies, standards and practices to be applied to manage data from heterogeneous sources and associated process (transformation and analysis) to ensure an efficient, secure usage, and a reliable quality of the analysis results. We propose to classify the DL data governance into data assets and IT assets [12,32]. Data assets refers to the value or potential value of data, while IT assets refers to technologies. Secondly, we identify some research axes based on the classification:

- Concerning the IT assets, the future researches must integrate the four following points: *data lake principles* concerns the position of the data lake in an information system. *Data lake functional architecture* concerns the evolution of different zones adapted to the enterprises' needs. *Data lake technical infrastructure* concerns the decisions about the technological solutions. *Data lake investment and prioritization* concerns the decisions about how much to invest for a data lake and the distribution for different zones.
- Concerning the data assets, the future research must be related to:
  - *Metadata management* (cf. previous section).
  - *Data quality management*: it influences the reliability of analytics result. In a data lake, data quality should at least be evaluated upon usage. New models must be proposed to make sure that data are valid in different data lake zones to ensure the data quality. Automatic validation [13,20] in the raw data zone, users' comments of analytics in the access zone can be one of the solutions.
  - *Data life-cycle management (DLM)*: it's necessary to define specific workflows to model the life-cycle of all the data that stored in different zones of a data lake and the relationships between different data.
  - *Security/Privacy*: a data lake can be accessed by different users with different privileges. Data sensitivity has to be authenticated, users access has to be managed [24]. New models and systems must be defined to be adapted to the data lake governance.

5 Conclusions

In this prospective/survey paper, we propose a complete definition of data lake and an extensible functional architecture based on 4 zones. Our definition has

the advantages of being more complete than the literature and includes both input and output, different functions as well as users of data lakes. In our data lake architecture, each zone is defined more formally than the literature and is composed of a process layer and a data storage layer.

We also introduce future research axes. In the metadata context, we identify two issues: (i) identifying and modelling intra- and inter-metadata, (ii) implementing these metadata in an adequate data management system. In the information system context, the future work concentrates on the definition of a unified ecosystem in which the enterprise data warehouses and data lakes coexist. Finally, in the data lake governance context, we propose to work in the fields of data assets and IT assets.

References

1. Alserafi, A., Abelló, A., Romero, O., Calders, T.: Towards information profiling: data lake content metadata management. In: 2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW), pp. 178–185. IEEE (2016)
2. Bilalli, B., Abelló, A., Aluja-Banet, T., Wrembel, R.: Towards intelligent data analysis: the metadata challenge. In: Proceedings of the International Conference on Internet of Things and Big Data, Rome, Italy, pp. 331–338 (2016)
3. Campbell, C.: Top five differences between data lakes and data warehouse, January 2015. https://www.blue-granite.com/blog/bid/402596/top-five-differences-between-data-lakes-and-data-warehouses
4. Dixon, J.: Pentaho, Hadoop, and data lakes, October 2010. https://jamesdixon.wordpress.com/2010/10/14/pentaho-hadoop-and-data-lakes/
5. Fang, H.: Managing data lakes in big data era: what's a data lake and why has it became popular in data management ecosystem. In: 2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), pp. 820–824. IEEE (2015)
6. Foshay, N., Mukherjee, A., Taylor, A.: Does data warehouse end-user metadata add value? Commun. ACM **50**(11), 70–77 (2007)
7. Hai, R., Geisler, S., Quix, C.: Constance: an intelligent data lake system. In: Proceedings of the 2016 International Conference on Management of Data, pp. 2097–2100. ACM (2016)
8. Haines, R.: What is just enough governance for the data lake?, February 2015. https://infocus.dellemc.com/rachel---haines/just--enough--governance--data--lake/
9. Halevy, A.Y., et al.: Managing google's data lake: an overview of the goods system. IEEE Data Eng. Bull. **39**(3), 5–14 (2016)
10. Inmon, B.: Data Lake Architecture: Designing the Data Lake and avoiding the garbage dump. Technics publications (2016)
11. Kaluba, K.: Data lake governance - do you need it?, March 2018. https://blogs.sas.com/content/datamanagement/2018/03/27/data-lake-governance/
12. Khatri, V., Brown, C.V.: Designing data governance. Commun. ACM **53**(1), 148 (2010). https://doi.org/10.1145/1629175.1629210. http://portal.acm.org/citation.cfm?doid=1629175.1629210
13. Kwon, O., Lee, N., Shin, B.: Data quality management, data usage experience and acquisition intention of big data analytics. Int. J. Inf. Manag. **34**(3), 387–394 (2014)

14. LaPlante, A., Sharma, B.: Architecting Data Lakes. O'Reilly Media, Sebastopol (2014)
15. Llave, M.R.: Data lakes in business intelligence: reporting from the trenches. Procedia Comput. Sci. **138**, 516–524 (2018)
16. Lopez Pino, J.L.: Metadata in business intelligence, January 2014. https://www.slideshare.net/jlpino/metadata-in-business-intelligence
17. Maccioni, A., Torlone, R.: Crossing the finish line faster when paddling the data lake with kayak. Proc. VLDB Endow. **10**(12), 1853–1856 (2017)
18. Madera, C., Laurent, A.: The next information architecture evolution: the data lake wave. In: Proceedings of the 8th International Conference on Management of Digital EcoSystems, pp. 174–180. ACM (2016)
19. Menon, P.: Demystifying data lake architecture, July 2017. https://medium.com/@rpradeepmenon/demystifying-data-lake-architecture-30cf4ac8aa07
20. Merino, J., Caballero, I., Rivas, B., Serrano, M., Piattini, M.: A data quality in use model for big data. Future Gener. Comput. Syst. **63**, 123–130 (2016)
21. Miloslavskaya, N., Tolstoy, A.: Big data, fast data and data lake concepts. Procedia Comput. Sci. **88**, 300–305 (2016)
22. Nadipalli, R.: Effective Business Intelligence with QuickSight. Packt Publishing Ltd., Birmingham (2017)
23. O'Leary, D.E.: Embedding AI and crowdsourcing in the big data lake. IEEE Intell. Syst. **29**(5), 70–73 (2014)
24. Patel, P., Greg, W., Diaz, A.: Data lake governance best practices, April 2017. https://dzone.com/articles/data-lake-governance-best-practices
25. Piatetsky-Shapiro, G.: Data lake vs data warehouse: key differences, September 2015. https://www.kdnuggets.com/2015/09/data-lake-vs-data-warehouse-key-differences.html
26. Ponniah, P.: Data Warehousing Fundamentals: a Comprehensive Guide for IT Professionals. Wiley, Hoboken (2004)
27. Quix, C., Hai, R., Vatov, I.: Metadata extraction and management in data lakes with gemms. Complex Syst. Inf. Model. Q. **9**, 67–83 (2016)
28. Ravat, F., Zhao, Y.: Metadata management for data lakes. In: East European Conference on Advances in Databases and Information Systems. Springer (2019)
29. Sawadogo, P., Kibata, T., Darmont, J.: Metadata management for textual documents in data lakes. In: 21st International Conference on Enterprise Information Systems (ICEIS 2019) (2019)
30. Varga, J., Romero, O., Pedersen, T.B., Thomsen, C.: Towards next generation BI systems: the analytical metadata challenge. In: Bellatreche, L., Mohania, M.K. (eds.) DaWaK 2014. LNCS, vol. 8646, pp. 89–101. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10160-6_9
31. Walker, C., Alrehamy, H.: Personal data lake with data gravity pull. In: 2015 IEEE Fifth International Conference on Big Data and Cloud Computing, pp. 160–167. IEEE (2015)
32. Weill, P., Ross, J.W.: IT Governance: How Top Performers Manage IT Decision Rights for Superior Results. Harvard Business Press, Boston (2004)