Journal for Re Attach Therapy and Developmental Diversities
eISSN: 2589-7799
2023 Dec; 6 (10s2): 1700-1708

Data Ingestion Frameworks for Data Lakes: An Overview

Hamza Elkina[1*], Taher Zaki[1]

[1]Innovation in Mathematics and Intelligent Systems Research Laboratory, Faculty of Applied Sciences, Ibnou Zohr University, Ait Melloul, Morocco

[*]**Corresponding Author:** Hamza Elkina
[*]Innovation in Mathematics and Intelligent Systems Research Laboratory, Faculty of Applied Sciences, Ibnou Zohr University, Ait Melloul, Morocco

Abstract:

Nowadays, information is considered a new capital of organizations as it is considered the basis of decisions made by the pilot committee, incorrect or incomplete information can cause significant losses. In the field of higher education, the emergence of new technologies and the increase of devices and connected users have allowed the creation of new data sources such as software packages, platforms, and social networks. Faced with this amount of information called big data, databases began to show their inability to manage and process this flow, leading to a new data storage technology called data warehousing that has allowed for many years, efficient handling of structured data sources. The need to manage semi-structured and unstructured data types has necessitated the use of new solutions such as data lakes to meet the new challenges imposed. As a new technology, the data lake is still equivocal, due to the incomplete or complicated presentation. For that matter, we will present a comparison between data warehouse and data lake, and discuss big data as well as data lakes as a new data management technology in higher education by presenting its essential components principally the data pipeline rarely cited in the literature. For this reason, a comparative study has been conducted to evaluate the existing data pipeline solutions and propose more valuable ones. In addition, we will introduce the university's data lake ecosystem to disambiguate the data lake concept and its essential components.

Keywords: big data, data lake, data pipeline, higher education, university.

1 INTRODUCTION

The multitude and variety of data sources have created a large amount of data that does not cease to grow with high frequency in all areas, which has led researchers to consider volume, variety, and velocity as basic criteria of big data (Chen et al., 2013). This determines at the same time the three challenges that need to be eliminated in order to take advantage of the collected data and find how to efficiently store a large amount of data that grows rapidly. The issue surfaced after relational databases became incompetent to handle the new generation of data. However, relational databases are still considered the optimal solution to persist transactional data (Smolinski, 2018). Faced with a large amount of data, traditional DBMS have shown their limits in terms of storage, notably in the representation of information extracted which requires the use of data warehouses. Vaisman and Zimányi present data warehouse as a particular database targeted toward decision support with the ability to take data from various sources (Vaisman & Zimányi, 2014). Thanks to the use of tables of facts and the concept of multidimensional, the data are represented in cube format. The cube accepts several operations that allow performing analytical processing for decision-making (Blazic et al., 2017), and permits an efficient representation of the information. The quality of the data stored in the data warehouse is ensured by the ETL (Extract, Transform & Load) process performed before the integration of each data. This operation has made the process slow, complex, and resource-intensive (Herden, 2020). The new data lake concept has enabled more flexible handling of persisted data, this last is based on the ELT (Extract, Load & Transform) process which adopts the idea of schema-on-read as the method of loading data (El Aissi et al., 2022). In other words, the new system stores the data without any modification, so that it can be processed at the time of reading. The use of the ELT process allows the data lake to digest all types of data regardless of their nature, i.e., structured, semi-structured, and unstructured data. To define the difference between the two technologies, a comparison is elaborated in section 3 to detail the ETL and ELT processes insufficiently discussed in the literature (Armbrust et al., 2021; Mukherjee & Kar, 2017; Sreemathy et al., 2020; Wyatt et al., 2009).

The domain of higher education has experienced several changes, the most important is the introduction of distance learning, making it inevitable during and post the Covid-19 pandemic. These changes have contributed to the emergence of new types of data massively generated by the platforms and their users, mainly videos/audio, documents (pdf, docx, pptx...), images as well as data produced by the users' activities (exams/assignments, participation, login time...). As we mentioned in the previous paragraph, the integration of heterogeneous data type in the learning process enforced the use of modern storage technology able to manage and process data to extract information. To our best knowledge, the most

used storage system in the academic field is relational databases and data warehouse. For that, we propose a simple implementation of a data lake to create a master dataset, allowing data processing and extracting sharpened information for decision, the proposition will introduce the academic ecosystem where the data lake is considered as the central storage system.

In this paper, we will investigate the types of data in the academic domain as well as their main sources. The goal is to demonstrate the concept of big data and its 3V aspects in the academic context. In our contribution, we will compare ETL and ELT, which are considered as the most important process for data ingestion. Moreover, we will discuss the implementation of data lake in the academic domain and the added value of creating a master dataset to facilitate data analyzing, further we will inspect the existing data pipelines to propose the most adapted solution for the university ecosystem.

The article is structured as follows. Section 2 discusses the concept of big data in high education and its data type. Section 3 reviews the difference between Data Warehouse and Data Lake, while Section 4 introduces the implementation of data lake in the university's system to handle all data type. In Section 5 we elaborated a comparative study of data pipeline solutions used to load data into the data lake. The 6th section depicts the data lake and its main components (Ingest, Store, Process and Stream) in the high education ecosystem. Finally, a conclusion.

2 BIG DATA IN THE ACADEMIC DOMAIN

The data are considered as the raw material of any treatment to extract useful information, which will be subsequently communicated to the decision-makers to take decisions or to be used as input for another treatment, imposing rigorous handling to provide the correct information as fast as possible. According to the literature, what characterizes information compared to data is its readiness to be exploited directly by the end user, so it can be evaluated as a product characterized by a certain level of quality that directly influences the quality of the decision (Batini & Scannapieco, 2016). In the academic field, we find a large heterogeneous quantity of data, namely structured, semi-structured and unstructured data in most cases. That is due to the diversity of the professions exercised within the university, namely teaching and research as the main activities next to other professions called support professions such as computer science, accounting, HR..., Table 1 presents a non-exhaustive list of data used in the university grouped by type.

Nowadays, the great challenge faced by organizations is to be able to manage all the data generated and extract useful information, knowing that more than 80% of the data generated are unstructured data (Taleb et al., 2018). Which makes the operation more complex since unstructured data requires advanced processing and analysis that uses algorithms, significant resources and time often not negligible to recover the maximum information. What characterizes the unstructured data is the absence of a precise schema or rules defined. Moreover the absence of metadata can make the data incomprehensible (Taleb et al., 2018), yet they are increasingly generated which is explained by the number of publications, images, and videos published on social networks.

Table 1: Examples of data types

Data type	Examples
Structured	Last name, first name, age, address, registration code, … Exam score, number of registrants, …
Semi-structured	CSV, XML, JSON, …
Unstructured	Image, video, PDF, PPTX, DOCX, Streaming, …

As we mentioned earlier, social networks represent today the main source of unstructured data, as they have become ubiquitous in all fields and a strategic tool for organizations. For the university social networks have become a space for interaction and exchange between professors and their students during the learning process, which has improved the motivation and participation rate of students (Hortigüela-Alcalá et al., 2019). Besides social networks, there are other sources of information such as distance learning platforms, digital workplaces, traditional management applications, and connected objects in smart classrooms (Sukare & Al, 2021). The common point between all these sources is the large amount of data generated each moment. Regarding the type, the majority of the generated data are unstructured, we mainly find PDF documents, Word, PowerPoint, images, and videos, which require a flexible storage method, scalable and compatible with the complex processing needed to extract useful information for decision-makers.

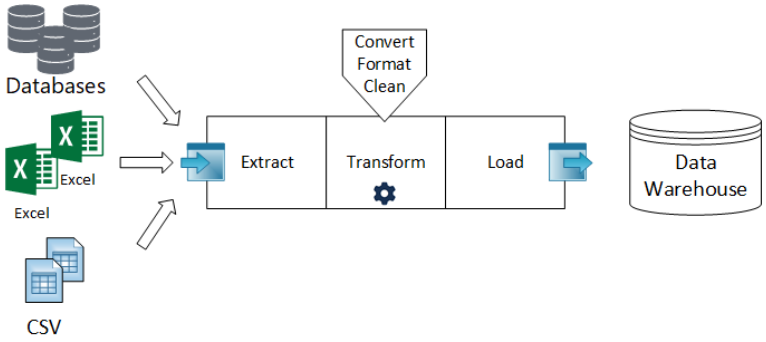

Figure 1: ETL Process for data warehouse.

3 BIG DATA MANAGEMENT TECHNOLOGIES: DATA WAREHOUSE VS. DATA LAKE

To manage big data, several technologies have been developed to answer the main need, "How to efficiently manage a large amount of data?" Better management ensures optimal exploitation and advanced information extraction. The most responded technology is the data warehouse, thanks to the Extract, Transform, and Load process which is considered as a rigorous data integration process, able to guarantee a high data quality. As depicted in Figure 1, firstly, structured data is Extracted from sources (Databases, Spreadsheets, XML), and prepared for the Transformation step where data is converted, formatted and cleaned to match the destination schema in the data warehouse. The last step consists of Loading the new structured data to the data warehouse. According to the author (Shan & Gubin, n.d.), data preparation is a crucial step that determines the validity and accuracy of future decisions. Data cleaning occupies the major part of data preparation and consumes more resources due to the complexity of the algorithms used in different detection and repair techniques: FDs value modification, FDs hypergraph, NADEEF, Unified repair, Sampling Duplicates... (Chu et al., 2016). For the academic domain and according to the literature, the majority treats data warehouse as the first solution to store and exploit big data, for that, several architectures have been proposed to meet the specific needs and include the majority of information (Saggar et al., 2022; Salaki & Ratnam, 2018; Santoso & Yulia, 2017; Williamson, 2018). Unfortunately, the traditional data warehouse was designed to handle only structured data, the need to exploit large semi-structured and unstructured data has pushed researchers to implement NoSQL in the data warehouse (Chevalier et al., 2015; Bicevska & Oditis, 2017) which has allowed extending the supported data and include data with a complex structure such as images and videos.

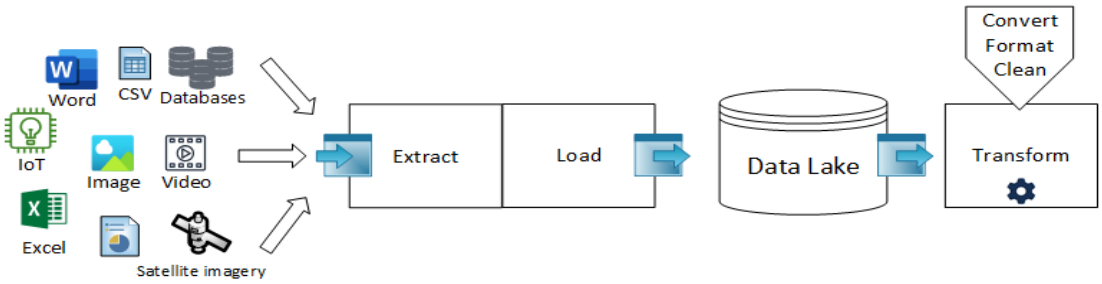

Figure 2: ELT process for Data Lake

Unfortunately, this solution is considered partial, since the ETL process is still used to integrate the data in the new version of data warehouses (Dabbèchi et al., 2021; Yangui et al., 2017). This primordial process has become a bottleneck that complicates the data integration in the data warehouse since it requires a long execution time with more hardware resources allocated for processing, as well as an update after each schema change (Herden, 2020). Finally, the idea of integrating NoSQL systems in the data warehouse has brought several modifications to the ETL process mainly in the Transformation part because the absence of a data schema in NoSQL systems – considered as their strong point – and consequently the absence of schema-on-write.

Data lakes are conceded as a new concept introduced by James DIXON in 2010, the new system offers more flexibility compared to data warehouses (Miloslavskaya & Tolstoy, 2016; Ravat & Zhao, 2019). The integration of data in the data lake is based on the ELT process for Extract, Load and Transform. Compared to ETL, the new process allows storing data retrieved from the sources without any transformation (Ravat & Zhao, 2019). In other words, it goes directly from the extraction to the loading without any transformation. As mentioned in Figure 2, the ELT process can Extract all kinds of data given the absence of the transformation as next step. After extraction, data is directly Loaded to the data lake. Such an agile process guarantees the storing of data in their raw format without loss of information; moreover, this simple process allows fast loading of any type of data whether it is structured, semi-structured or unstructured without any

restriction. The concept of Schema-On-Read leaves the transformation step of the ELT process at the moment of data retrieval. In other words, once the data is stored, the user can retrieve a copy to perform the necessary transformations that meet the needs of his business while keeping the original version in the data lake in order to be exploited by other users in other contexts. The flexibility offered by data lakes allows extending the area of files that can be stored to include files with a complex structure such as images and videos and to offer an efficient solution to capture the broadcast stream regardless of its nature (Giebler et al., 2021).

As a new concept, the data lake system is still considered a big work in progress, several research works have been done but many aspects are not yet addressed such as the integration of data lakes in the ecosystem of organizations, metadata management, and governance (Ravat & Zhao, 2019). The majority of the current researches discuss the metadata management, considered as a complex and important feature to process efficiently the stored data.

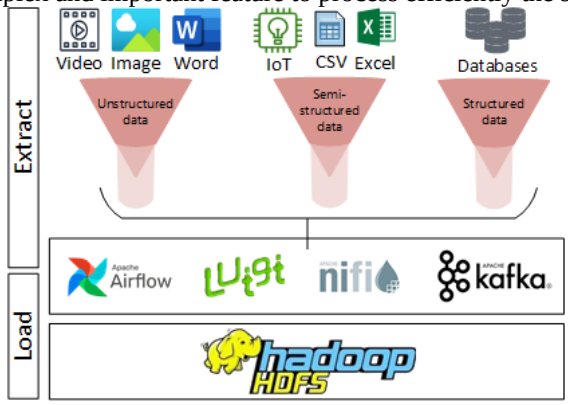

Figure 3: Data Lake architecture

4 DATA LAKE FOR UNIVERSITY

The need to store data of different types and sizes while keeping them accessible for future use has imposed the use of a flexible, secure, and above all expandable storage system without being costly. For the university, data management is an essential operation, whether for administrative or scientific operations in research laboratories. The proposed architecture will allow an opening on the various internal and external data sources through a data collection system that allows automation of the extraction of data from operating systems, database management systems, and different platforms and applications, to save them in the data lake in their raw format. For the storage system, and according to the literature (Fang, 2015; Giebler et al., 2019; Munirathinam et al., 2019; Sawadogo & Darmont, 2021), the solution proposed by Hadoop via its distributed file system HDFS is considered the most implemented and most suitable solution for data storage in data lakes since it allows an easy and fast extension of the storage support, which is a key advantage for the data lake. Moreover, its management of files divided into blocks and duplicated on several Data Nodes ensures the continuous availability of the data in the Master Dataset, the consistency of the data, and the distribution according to the CAP (Triangle Consistent-Available-Partition) theorem. For the extraction and loading of data in the data lake, we have noticed that the subject is treated in a limited and superficial way (Rooney et al., 2019; Sawadogo & Darmont, 2021; Zhao et al., 2021) despite the importance of this layer responsible for the ingestion of data in the data lake. Compared to data warehouses, the extraction of data from their sources and recording them is more complicated for data lakes, since the types of data processed include in addition to structured data (Spreadsheets, Databases) semi-structured and unstructured data (Videos, Images, APIs, streams ...). Which requires a tool capable of communicating with multiple systems and capturing the different data flows as streaming. In order to find the optimal solution, we have elaborated a comparative study between the different data ingestion tools that will be used next to the Hadoop system. As mentioned in Figure 3, data pipeline solution is present in the two important phases, Extract and Load, and considered as the interface between the data lake's component of storage and the external systems. In addition, an efficient data pipeline must be able to collect all data types and support continuous data streaming.

5 DATA PIPELINES: A COMPARATIVE STUDY

Data pipelines act as data carriers from their sources to the data lake, a process that integrates several challenges from connecting to the source to extracting and routing the retrieved data, and requires transformations depending on the use case for the ETL process. In the case of data lakes, the data extraction tool saves the data in its raw format without the necessity to change data's structure – ELT – since the transformation is processed according to the need, which makes the task less complicated and less costly in terms of time and hardware resources. In this section, we will elaborate a comparison between the most used open-source solutions as data pipelines. The objective of this comparison is to find the most suitable and efficient solution to retrieve the different types of data from their various sources. Furthermore, the

comparison aims to determine the level of openness of the solutions to accept future modifications in order to optimize and enrich the ELT process. After a study of the existing, we found several solutions, the majority of them were commercial or closed source, which allowed us to make the first filter and keep only the most used free solutions with public source code. The solutions subject to comparison are Apache AirFlow, Luigi, Apache nifi, and Apache Kafka.
Table 2 gives a general view of data pipelines before detailing each solution, the table shows the community that develops the solution, a very important point as a widely used solution guarantees continuous development and improvement. The language used for the development gives an idea of the possibility of introducing improvements that help to better exploit the data that passes through before arriving at the destination, using the advantages offered by artificial intelligence. On the other hand, the selected tools all have a web interface to visualize the tasks and the programmed operations except for Apache Kafka. Our study is based mainly on the official documentation of each solution.

Table 2: Data Pipelines - General comparison.

Data Pipeline	Developed by	Open source	Language	Multiplatform	Web interface	Data streaming
Apache AirFlow	Apache Software Foundation (Airbnb)	Yes	Python	Yes	Yes	No
Luigi	Spotify	Yes	Python	Yes	Yes	No
Apache nifi	Apache Software Foundation (NSA)	Yes	Java	Yes	Yes	Yes
Apache Kafka	Apache Software Foundation (LinkedIn)	Yes	Java & Scala	Yes	No	Yes

5.1 Apache AirFlow

According to the official documentation, the main feature offered by Apache AirFlow is its ability to create processes organized in DAG (Directed Acyclic Graph) (*Apache Airflow*, 2015/2022). According to graph theory the use of DAGs allows avoiding loops, therefore it will eliminate the situation of infinite execution loops which can be considered as a failure point for the Extract-Load process (Singh, 2019). Through a selection of operators like the "Python operator" and the "Bash operator", Apache Airflow can extract data from several sources and route it to its destinations. For each operation, Apache AirFlow stores a set of information considered as metadata that can be used to manage the versions of each processed data. On the other hand, Apache AirFlow is unable to stream data that has already been retrieved, which can make data analysis and processing more difficult in the case of large data sets.

5.2 Luigi

Luigi is considered the most HDFS-related pipeline since it offers several features dedicated to Hadoop, yet it offers other connectors to extract data from other sources such as databases using "Tasks" ordered as Dependency Graphs. According to Luigi's documentation, it is possible to run Machine Learning algorithms on the data passing through the pipeline (*Spotify/Luigi*, 2012/2022), which is considered an advantage in the case of data classification before integrating them into the data lake.

5.3 Apache nifi

In the field of scientific research, Apache nifi is often used and quoted in several papers when the subject of the data pipeline is discussed (Dehury et al., 2020, 2022; Poojara et al., 2022). According to the documentation of the Apache nifi, the solution offers a rich environment of components (Processor, Flow Controller, Connection...) which ensures the processes ETL and ELT from diverse sources of data to various destinations. Thanks to the Content Repository component, which is a temporary storage space. Apache nifi offers the possibility to perform the necessary analysis and processing before routing the data to the destination. In the case of a data lake as a destination, it will allow enriching the metadata collected on a file before storing it.

5.4 Apache Kafka

The Apache Kafka solution is composed of several sub-projects organized in the form of APIs (Producer API, Consumer API, Streams API, Connect API, and Admin API) (*Apache Kafka*, n.d.; Martín et al., 2022) which facilitates interaction with each part. Apache Kafka allows managing the flow direction from the "Producer" to the "Consumer", in other words, in the case of a data lake, it allows routing a file from the source to the data lake and makes it accessible later in streaming for any processing or analysis. For scalability, it is possible to add other Brokers to build a Kafka Cluster which will allow supporting more flows whether it is incoming or outgoing.

5.5 Discussion

Choosing a data pipeline solution is strongly tied to the application context, it depends principally on the needed functionalities to avoid the overuse of resources. According to the comparison and documentation, Apache Airflow can be considered as the simplest data pipeline, requiring a simple configuration and basic python programing knowledge to create DAGs. In addition, Apache Airflow provides a web interface to make the management of data ingestion tasks simple. Luigi is based on python class to define tasks and uses dependency graph to schedule them. For Luigi, the web interface is still simple with limited features compared to Apache Airflow. Unfortunately, both of the solutions do not support data streaming evaluated as an essential feature for data lake implementation. Apache nifi and Apache Kafka provide advanced features that make the data ingestion more efficient, and allow the execution of complex operations to process transited data before forwarding it to the data lake.

6 DATA LAKE ECOSYSTEM FOR THE UNIVERSITY

In this section, we will try to draw a general ecosystem of the data lake in high education. As we mentioned in section 2, new technologies are the new tool for learning and sharing knowledge specially during and post COVID-19 pandemic. In addition, the use of learning management systems has allowed the emergence of new types of data not managed by classic databases or data warehouses as cited in the section 4. Furthermore, smart classes and their connected objects produce continued streaming of various data (structured, semi-structured, and unstructured). Faced with this heterogeneous data in the university, and to ensure better governance, the establishment of a data lake has become a necessity to manage all the data in one storage system and implement the concept of master dataset. The proposition below of data lake present a simple implementation which integrates the main features described by James DIXON and considered the basic goals. (1) Answer to the maximum proposed questions, and (2) increase visibility of data by storing the raw format ("Pentaho, Hadoop, and Data Lakes," 2010). Our implementation can be considered as a basic-version candidate for adding more complex operation as data analysis based on artificial intelligence. The intention is to contradict the concept where data lake is treated as Hadoop HDFS technology. As depicted in figure 4, data lake can extract data from various sources (platforms, databases, operating systems…) by using the data pipeline component and forward the data to the storage component. As we mentioned before, HDFS is the most adapted storage system due to its high scalability and the ability to store voluminous files. Based on the comparison between data pipeline solutions discussed in section 5, we found that Apache nifi and Apache Kafka can be suitable solutions to use as a data pipeline next to HDFS as the storage system, both solutions allow us to conduct data from different sources to the data lake and prepare the stored data for streaming. For the other solutions, the absence of a streaming module will force the use of an extra component to ensure data share functionality, or transfer all files to a local storage space before processing operation. The advantage of Apache nifi compared to Apache Kafka is the integration of a web interface, which makes the creation and monitoring of tasks much easier. By this implementation, the data lake becomes the main storage system in the university and the unique data source for data analyst and other applications allowed to consume data, this concept will facilitate data processing and ensure more information quality for the pilot committee to guarantee the best decisions. Based on the master dataset principle, once the data. is loaded into the data lake, modification is no longer allowed but it can be considered as a new version to assure data integrity, which is considered as an important key for any storage system. For data processing, the most important advantage of using the data lake is the continuous data availability in its original format, which allows data analysts to make advanced exploitation and extract the maximum information. Figure 4 represents a simple implementation of data lake where the indexation is managed elementary by the Hadoop Distributed File System, which makes this conception inefficient in the case of heavy or complex query. For that the integration of metadata management becomes an urgency to avoid the transformation of the data lake into a data swamp. In addition, the integration of metadata management will allow fast processing without consuming time and resources.

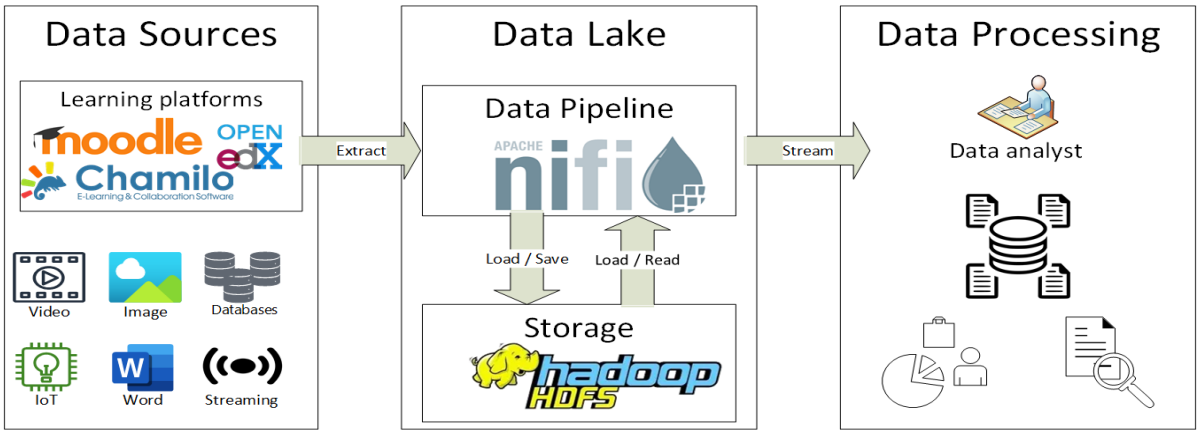

Figure 4: Data Lake ecosystem for university

7 CONCLUSION

Data lake becomes more and more used as centralized storage. Due to its scalability and flexibility compared to a data warehouse, a data lake can carry the master dataset and group all heterogenous data from various sources, in their raw format. Handling big data raise the necessity to manage and index stored data. For this reason, various research activities have been carried out on metadata extraction and metadata management, to avoid converting the data lake into a worthless data swamp. The new concept of the data lake and mainly the ELT process has allowed managing new types of data such as videos and images, which gave birth to new needs to better exploit these data that were not digested by the old systems such as databases or classic data warehouses, even for NoSQL.

In this paper, we exhibited the emergence of big data in the academic field, due to the use of new technologies to share knowledge, next to new connected objects. We showed that data lake can be the adequate technology for the university to store all kinds of data. In addition, we have elaborated a comparative study between data pipeline solutions to propose a suitable tool next to the Hadoop HDFS file system. As we mentioned in the previous section, the goal of proposing data lake with its basic components is to give a simple description, and to point out that data lake cannot be equivalent to a storage system only. As we mentioned in the previous paragraph, metadata management is still an important component to include and an interesting line of research to improve data processing in data lakes.

REFERENCES

1. *Apache Airflow*. (2022). [Python]. The Apache Software Foundation. https://github.com/apache/airflow (Original work published 2015)
2. *Apache Kafka*. (n.d.). Retrieved October 7, 2022, from https://kafka.apache.org/documentation/
3. Armbrust, M., Ghodsi, A., Xin, R., & Zaharia, M. (2021). *Lakehouse: A New Generation of Open Platforms that Unify Data Warehousing and Advanced Analytics*. 8.
4. Batini, C., & Scannapieco, M. (2016). *Data and Information Quality*. https://doi.org/10.1007/978-3-319-24106-7
5. Bicevska, Z., & Oditis, I. (2017). Towards NoSQL-based Data Warehouse Solutions. *Procedia Computer Science*, *104*, 104–111. https://doi.org/10.1016/j.procs.2017.01.080
6. Blazic, G., Poscic, P., & Jaksic, D. (2017). Data warehouse architecture classification. *2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2017 - Proceedings*, 1491–1495. https://doi.org/10.23919/MIPRO.2017.7973657
7. Chen, J., Chen, Y., Du, X., Li, C., Lu, J., Zhao, S., & Zhou, X. (2013). Big data challenge: A data management perspective. *Frontiers of Computer Science 2013 7:2*, *7*(2), 157–164. https://doi.org/10.1007/S11704-013-3903-7
8. Chevalier, M., El Malki, M., Kopliku, A., Teste, O., & Tournier, R. (2015). How Can We Implement a Multidimensional Data Warehouse Using NoSQL? In S. Hammoudi, L. Maciaszek, E. Teniente, O. Camp, & J. Cordeiro (Eds.), *Enterprise Information Systems* (pp. 108–130). Springer International Publishing. https://doi.org/10.1007/978-3-319-29133-8_6
9. Chu, X., Ilyas, I. F., Krishnan, S., & Wang, J. (2016). Data Cleaning: Overview and Emerging Challenges. *Proceedings of the 2016 International Conference on Management of Data*, 2201–2206. https://doi.org/10.1145/2882903.2912574
10. Dabbèchi, H., Haddar, N. Z., Elghazel, H., & Haddar, K. (2021). Social Media Data Integration: From Data Lake to NoSQL Data Warehouse. In A. Abraham, V. Piuri, N. Gandhi, P. Siarry, A. Kaklauskas, & A. Madureira (Eds.), *Intelligent Systems Design and Applications* (pp. 701–710). Springer International Publishing. https://doi.org/10.1007/978-3-030-71187-0_64
11. Dehury, C. K., Jakovits, P., Srirama, S. N., Giotis, G., & Garg, G. (2022). TOSCAdata: Modeling data pipeline applications in TOSCA. *Journal of Systems and Software*, *186*, 111164. https://doi.org/10.1016/j.jss.2021.111164
12. Dehury, C. K., Srirama, S. N., & Chhetri, T. R. (2020). CCoDaMiC: A framework for Coherent Coordination of Data Migration and Computation platforms. *Future Generation Computer Systems*, *109*, 1–16. https://doi.org/10.1016/j.future.2020.03.029
13. El Aissi, M. E. M., Benjelloun, S., Loukili, Y., Lakhrissi, Y., Boushaki, A. E., Chougrad, H., & Elhaj Ben Ali, S. (2022). Data Lake Versus Data Warehouse Architecture: A Comparative Study. *Lecture Notes in Electrical Engineering*, *745*, 201–210. https://doi.org/10.1007/978-981-33-6893-4_19
14. Fang, H. (2015). Managing data lakes in big data era: What's a data lake and why has it became popular in data management ecosystem. *2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*, 820–824. https://doi.org/10.1109/CYBER.2015.7288049
15. Giebler, C., Gröger, C., Hoos, E., Eichler, R., Schwarz, H., & Mitschang, B. (2021). *The Data Lake Architecture Framework*. Gesellschaft für Informatik, Bonn. https://doi.org/10.18420/btw2021-19
16. Giebler, C., Gröger, C., Hoos, E., Schwarz, H., & Mitschang, B. (2019). Leveraging the Data Lake: Current State and Challenges. In C. Ordonez, I.-Y. Song, G. Anderst-Kotsis, A. M. Tjoa, & I. Khalil (Eds.), *Big Data Analytics and Knowledge Discovery* (pp. 179–188). Springer International Publishing. https://doi.org/10.1007/978-3-030-27520-4_13

17. Herden, O. (2020). Architectural Patterns for Integrating Data Lakes into Data Warehouse Architectures. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *12581 LNCS*, 12–27. https://doi.org/10.1007/978-3-030-66665-1_2
18. Hortigüela-Alcalá, D., Sánchez-Santamaría, J., Pérez-Pueyo, Á., & Abella-García, V. (2019). Social networks to promote motivation and learning in higher education from the students' perspective. *Innovations in Education and Teaching International*, *56*(4), 412–422. https://doi.org/10.1080/14703297.2019.1579665
19. Martín, C., Langendoerfer, P., Zarrin, P. S., Díaz, M., & Rubio, B. (2022). Kafka-ML: Connecting the data stream with ML/AI frameworks. *Future Generation Computer Systems*, *126*, 15–33. https://doi.org/10.1016/j.future.2021.07.037
20. Miloslavskaya, N., & Tolstoy, A. (2016). Big Data, Fast Data and Data Lake Concepts. *Procedia Computer Science*, *88*, 300–305. https://doi.org/10.1016/j.procs.2016.07.439
21. Mukherjee, R., & Kar, P. (2017). A Comparative Review of Data Warehousing ETL Tools with New Trends and Industry Insight. *2017 IEEE 7th International Advance Computing Conference (IACC)*, 943–948. https://doi.org/10.1109/IACC.2017.0192
22. Munirathinam, S., Sun, S., Rosin, J., Sirigibathina, H., & Chinthakindi, A. (2019). Design and Implementation of Manufacturing Data Lake in Hadoop. *2019 IEEE International Conference on Smart Manufacturing, Industrial & Logistics Engineering (SMILE)*, 19–23. https://doi.org/10.1109/SMILE45626.2019.8965302
23. Pentaho, Hadoop, and Data Lakes. (2010, October 14). *James Dixon's Blog*. https://jamesdixon.wordpress.com/2010/10/14/pentaho-hadoop-and-data-lakes/
24. Poojara, S. R., Dehury, C. K., Jakovits, P., & Srirama, S. N. (2022). Serverless data pipeline approaches for IoT data in fog and cloud computing. *Future Generation Computer Systems*, *130*, 91–105. https://doi.org/10.1016/j.future.2021.12.012
25. Ravat, F., & Zhao, Y. (2019). Data Lakes: Trends and Perspectives. In S. Hartmann, J. Küng, S. Chakravarthy, G. Anderst-Kotsis, A. M. Tjoa, & I. Khalil (Eds.), *Database and Expert Systems Applications* (pp. 304–313). Springer International Publishing. https://doi.org/10.1007/978-3-030-27615-7_23
26. Rooney, S., Bauer, D., Garcés-Erice, L., Urbanetz, P., Froese, F., & Tomic, S. (2019). Experiences with Managing Data Ingestion into a Corporate Datalake. *2019 IEEE 5th International Conference on Collaboration and Internet Computing (CIC)*, 101–109. https://doi.org/10.1109/CIC48465.2019.00021
27. Saggar, S., Bitoni, C., Khurana, I., & Alhawat, R. (2022). *Data Warehouse with Big Data Technology for Higher Education* (SSRN Scholarly Paper No. 4128707). https://doi.org/10.2139/ssrn.4128707
28. Salaki, R. J., & Ratnam, K. A. (2018). Agile Analytics: Applying in the Development of Data Warehouse for Business Intelligence System in Higher Education. In Á. Rocha, H. Adeli, L. P. Reis, & S. Costanzo (Eds.), *Trends and Advances in Information Systems and Technologies* (pp. 1038–1048). Springer International Publishing. https://doi.org/10.1007/978-3-319-77703-0_101
29. Santoso, L. W. & Yulia. (2017). Data Warehouse with Big Data Technology for Higher Education. *Procedia Computer Science*, *124*, 93–99. https://doi.org/10.1016/J.PROCS.2017.12.134
30. Sawadogo, P., & Darmont, J. (2021). On data lake architectures and metadata management. *Journal of Intelligent Information Systems*, *56*(1), 97–120. https://doi.org/10.1007/s10844-020-00608-7
31. Shan, H., & Gubin, E. (n.d.). *DATA CLEANING FOR DATA ANALYSIS*. 2.
32. Singh, P. (2019). Airflow. In P. Singh (Ed.), *Learn PySpark: Build Python-based Machine Learning and Deep Learning Models* (pp. 67–84). Apress. https://doi.org/10.1007/978-1-4842-4961-1_4
33. Smolinski, M. (2018). Impact of Storage Space Configuration on Transaction Processing Performance for Relational Database in PostgreSQL. In S. Kozielski, D. Mrozek, P. Kasprowski, B. Małysiak-Mrozek, & D. Kostrzewa (Eds.), *Beyond Databases, Architectures and Structures. Facing the Challenges of Data Proliferation and Growing Variety* (pp. 157–167). Springer International Publishing. https://doi.org/10.1007/978-3-319-99987-6_12
34. *Spotify/luigi*. (2022). [Python]. Spotify. https://github.com/spotify/luigi (Original work published 2012)
35. Sreemathy, J., Joseph V., I., Nisha, S., Prabha I., C., & Priya R.M., G. (2020). Data Integration in ETL Using TALEND. *2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)*, 1444–1448. https://doi.org/10.1109/ICACCS48705.2020.9074186
36. Sukare, N., & Al, E. (2021). Smart Classroom Environment using IoT in advanced and lebanese French university Education. *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, *12*(7), Article 7. https://doi.org/10.17762/turcomat.v12i7.3395
37. Taleb, I., Serhani, M. A., & Dssouli, R. (2018). Big Data Quality Assessment Model for Unstructured Data. *2018 International Conference on Innovations in Information Technology (IIT)*, 69–74. https://doi.org/10.1109/INNOVATIONS.2018.8605945
38. Vaisman, A., & Zimányi, E. (2014). *Data Warehouse Systems*. https://link.springer.com/book/10.1007/978-3-642-54655-6

39. Williamson, B. (2018). The hidden architecture of higher education: Building a big data infrastructure for the 'smarter university.' *International Journal of Educational Technology in Higher Education*, *15*(1), 12. https://doi.org/10.1186/s41239-018-0094-1
40. Wyatt, L., Caufield, B., & Pol, D. (2009). Principles for an ETL Benchmark. In R. Nambiar & M. Poess (Eds.), *Performance Evaluation and Benchmarking* (Vol. 5895, pp. 183–198). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-10424-4_14
41. Yangui, R., Nabli, A., & Gargouri, F. (2017). ETL Based Framework for NoSQL Warehousing. In M. Themistocleous & V. Morabito (Eds.), *Information Systems* (pp. 40–53). Springer International Publishing. https://doi.org/10.1007/978-3-319-65930-5_4
42. Zhao, Y., Megdiche, I., & Ravat, F. (2021). *Data Lake Ingestion Management* (arXiv:2107.02885). arXiv. https://doi.org/10.48550/arXiv.2107.02885