Article

Advancing Logistics 4.0 with the implementation of a Big Data Warehouse: a Demonstration Case at the Automotive Industry

Nuno Silva [1,*], Júlio Barros [1], Maribel Y. Santos [1], Carlos Costa [1], Paulo Cortez [1], M. Sameiro Carvalho [1], João N. C. Gonçalves [1]

[1] ALGORITMI Research Centre, University of Minho, Guimarães 4800–058, Portugal
* Correspondence: nuno.silva@dsi.uminho.pt

Abstract: The constant advancements in Information Technology have been the main driver of the Big Data concept's success. With it, new concepts like Industry 4.0 and Logistics 4.0 are rising. Due to the increase in data volume, velocity, and variety, organizations are now looking to their data analytics infrastructures and searching for approaches to improve their decision-making capabilities, in order to enhance their results using new approaches such as Big Data and Machine Learning. The implementation of a Big Data Warehouse can be the first step to improve the organizations' data analysis infrastructure and start retrieving value from the usage of Big Data technologies. Moving to Big Data technologies can provide several opportunities for organizations, such as the capability of analysing an enormous quantity of data from different data sources in an efficient way. However, at the same time, different challenges can arise, including data quality, data management, lack of knowledge within the organization, among others. In this work, we propose an approach that can be adopted in the logistics department of any organization in order to promote the Logistics 4.0 movement, while highlighting the main challenges and opportunities associated with the development and implementation of a Big Data Warehouse in a real demonstration case at a multinational automotive organization.

Keywords: Big Data; Data Warehouse; Logistics 4.0; Industry 4.0; Implementation.

Citation: Lastname, F.; Lastname, F.; Lastname, F. Title. *Electronics* **2021**, *1*, 0. https://doi.org/

Received:
Accepted:
Published:

1. Introduction

The explosion of the Information Technologies area has been the driver that launched new concepts such as Big Data and Industry 4.0 into the spotlights. The concept of Industry 4.0 relies in the digitization of the production systems to provide the capability of producing customized products within a short time and with costs similar to mass production scenarios [1]. This factor has a tremendous impact in the organizations' logistics due to the need of reacting to the sudden changes made by the customers.

Logistics 4.0 can be defined as *"... the logistical system that enables the sustainable satisfaction of individualized customer demands without an increase in costs and supports this development in industry and trade using digital technologies"*[2]. Such initiative is needed to improve the link between the manufacturers and the customers, in order to avoid failures in the manufacturing system [2].

Big Data technologies, with their capability of analysing massive volumes of diverse data flowing at high velocity, has an important role in the implementation of these new concepts (Industry 4.0 and Logistics 4.0) and in the resolution of their main associated challenges [3].

With the implementation of Big Data technologies became possible to perform tasks that involves a massive quantity of data at high speeds such as providing a supply chain control with real-time data, inventory control and management, improving forecasting models, among others [1].

Along with the influence of concepts like Industry 4.0 and Logistics 4.0, the investments in Big Data technologies are being stimulated making them more stable and mature, ready to be implemented inside the organizations and became part of their business.

A vast range of organizations, from diverse types of business, are now trying to evolve their data analyses infrastructures to this new era, advancing their Data Warehouses (DWs) based on a more rigid data model to the new concept of Big Data Warehouses (BDWs) with a more dynamic data model.

This work aims to demonstrate how the implementation of a Big Data Warehouse (BDW) in a logistics context can drive forward the concept of Logistics 4.0 and improve the organization performance. The contributions of this work are: Propose a general approach that can be adopted in the logistics departments of several organizations; Propose a logical and technological architecture that supports the BDW and data analysis; Propose a data model for a logistics BDW; Demonstrate the challenges and opportunities that emerge throughout the development and implementation of a BDW in the logistics department.

A demonstration case will be presented, having been the same developed inside of a multinational automotive organization by taking advantage of their existing data platform.

This work is structured as follow: Section 2 provides the published works related to BDW and their architectures; Section 3 presents the suggested architecture to solve this problem; Section 4 describes the organization reality and the tasks performed to accomplish the goal; Section 5 presents the results accomplished fowled by a discussion where the challenges and opportunities are highlighted; Section 6 shows the final conclusions and future work.

2. Related work

With the implementation of concepts like Industry 4.0 and Logistics 4.0, it becomes important to endow the organizations' data analyses infrastructure with the capability of retrieving, transforming and analysing massive amounts of data at high velocity. Before the establishment of the Big Data concept, organizations had their data analyses infrastructure based in DWs where the data model was rigid and structured in order to provide the best performance when data were inquired.

Nowadays, Big Data technologies, due to their capacity for distributed processing and storage, allow us to have more dynamic data models with less rigid structures, maintaining high performance even with massive volumes of data.

To implement Big Data technologies, we can follow two different approaches: "the lift and shift" and the "rip and replace". "The lift and shift" strategy means that we replace or extend parts of the existing infrastructure with Big Data technology to improve its capabilities and to solve specific problems. This may result in a use case approach instead of a data-driven approach, which can lead to uncoordinated data silos. The "rip and replace" approach means that the existing Data Warehouse (DW) is totally replaced by Big Data technologies [4].

Independently of these two strategies, there are several architectures and technologies, that can be used to implement a BDW. The use of different types of Not Only SQL (NoSQL) databases, such as document-oriented and column-oriented [5] or graph models [6] can be used to store the different types of data in the BDW. In the literature, we can find different architectures that can be used in a BDW, such as the Lambda architecture [7] and the NIST Big Data Reference Architecture (NBDRA) [8]. The Lambda architecture has three layers and unifies, in a single software design pattern, the batch and real-time data processing concerns. The three layers presented in the Lambda architecture are batch processing, real-time computing, and a layer to query the data. This division between batch processing and real-time processing allows differentiating data according to their nature and relevance to the business. In this way, it is possible to immediately process the data that is needed in time, while data that is only needed in the long run can be processed later [7].

The NBDRA is presented by its authors as a common reference that can be implemented using any Big Data technology or service provider. It is divided into the following five components: System orchestrator; Data provider; Big Data application provider; Big Data framework provider; and Data consumer. The system orchestrator is the component that establishes the requirements for all the infrastructure, including, among others, architectural design, business requirements, and governance. The data provider is the component that makes data accessible through different interfaces. The Big Data application provider deals with all the necessary tasks to manipulate data through its life cycle. The Big Data framework provider consists of several services or resources that are used by the Big Data application provider. The data consumer is the entity that will take advantage of all the data processing made by the Big Data system [9]. Using the NBDRA and the Lambda Architecture as a reference, Santos and Costa [9] created an approach to develop BDWs.

Several examples demonstrate the capacity of Big Data technologies for improving the analytical capabilities of organizations. Chou et al [10] propose a system architecture based on Hadoop, Sqoop, Spark, Hive and Impala to analyse data from electrical grids. Sebaa et al [11] present an architecture based on the Hadoop ecosystem and a conceptual model to develop a BDW in the Healthcare field. Santos et al [12] present a demonstration case where it was applied a Big Data architecture and a set of rules to evolve from a traditional DW to a BDW.

These examples demonstrate how Big Data technologies can be used in collaboration with traditional DWs or even replacing them, both aiming to improve the analytical capabilities of the organizations.

3. Propose Architecture for a Logistics 4.0 Big Data Warehouse

In this section, it is presented the logical (3.1) and technological (3.2) architectures that can be used to implement a BDW for the Logistics 4.0 movement.

3.1. Logical Architecture

The main goal of this BDW is to be an analytical repository containing a substantial amount of data, in order to support the daily activities of the logistics decision-makers in the logistics 4.0 era.

Two of the key factors in Logistics 4.0 are the real-time exchange of information between all the actors in the supply chain and the real-time Big Data analytics of vehicles, products and facilities location [3].

The exchange of information between all actors in the supply chain can originate diverse data sources with different types of data that need to be stored and analysed in one central repository in order to be easily accessible by practitioners. The same happens with the real-time Big Data analytics of the diverse supply chain components (vehicles, products, and facilities location). Considering this, the real-time characteristics can be important, nevertheless, it is necessary to adapt to the organizational requirements. Real-time analytics can be a different concept from one organization to other. For example, for one organization, the requirements of real-time can be to have access to data in less than ten seconds, but for other organization, it can be to access the data in less than two minutes. Moreover, some organizations still do not have the need of creating an architecture that takes into consideration the real-time requirements.

In our demonstration case, the organization does not have the requirement of real-time analysis, so the architecture presented in Figure 1 does not incorporate that component. Nevertheless, due to the relevance of real-time in Logistics 4.0, it may be relevant to implement and validate that component in future work.

As can be seen in Figure 1, the logical architecture has the following components:

- Sandbox Storage: where the raw data is stored in a distributed file system before any transformation. This component is divided into two layers: Update Layer and Backup Layer. The Update Layer contains the up-to-date data retrieved from the

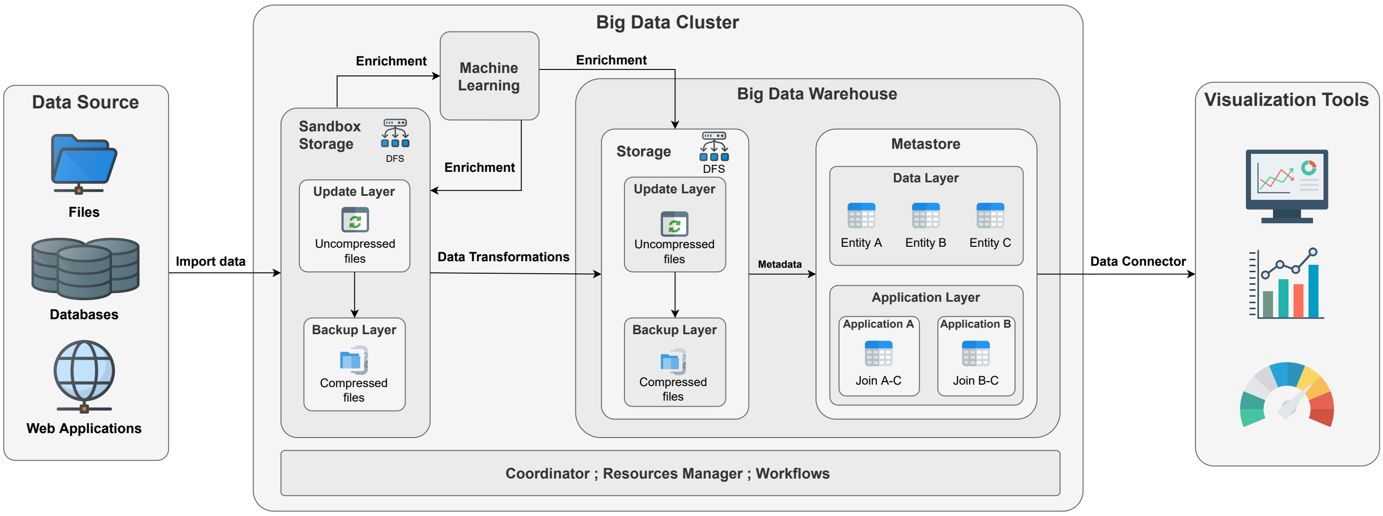

Figure 1. Logical architecture

sources, while the Backup Layer contains compressed outdated data to be used in case of necessity.

- BDW Storage: where data is stored in the distributed file system and accessible using the metastore after being transformed. This component has two layers with the same functionality as the Sandbox Storage layers: i) a layer that provides updated data, ii) and another layer to provide a backup in case of problems with the new data.
- Machine Learning component: uses raw data from the Sandbox storage or clean data from the BDW to create predictions, in order to enrich the data and store it in the Sandbox Storage or in the BDW to provide predictive capabilities for the organization. This component can increase the organization's capabilities to understand and predict changes in their supply chain and be capable to adapt quickly.
- Metastore: provides an interface to access the stored data. This component is divided into two layers: i) the data layer where the data is modelled using a data-driven approach, and; ii) the application layer where we have the necessary materialized objects or views to answer the needs of specific applications. The existence of these two layers provides some advantages. One of these advantages is the capability of creating several abstractions on top of the data layer, providing a simple and fast way to access the data. In this application layer, each application can have its own views or tables (materialized objects), increasing the performance when accessing the data. Moreover, if the organization has different teams working in different applications, if necessary, each team can create the necessary tables or views for their own application, providing higher business agility.
- The Coordinator, Resources Management and Workflows provide functionalities to manage the Big Data Cluster and the data life cycle. The Coordinator and Workflow allow the creation of diverse jobs or tasks that can be submitted in the desired order. The Resource Manager distributes the clusters resources to process the jobs.

Outside the BD Cluster, we can find the data sources that provide the raw data to be used in the BDW and the Visualizations Tools where dashboards are developed to present the results to the users.

3.2. Technological Architecture

Due to the need of analysing big quantities of data in the most efficient way, new technologies that use the power of distributed processing and storage have gained significant attention. Probably the most well-known technology in this context, which can arguably be seen as the originating driver of the Big Data movement, is Apache Hadoop [1], where data can be stored in the Hadoop Distributed File System (HDFS) [13] and then processed using the Map and Reduce [14] programming model. Several other technologies such as Sqoop [2], Hive [15], Spark [16], and Impala [3] [17], among others, are being constantly developed to tackle specific problems in the Big Data ecosystem. These technologies allow the practitioners to retrieve data from the data sources, store it with appropriate metadata and then processing it, in order to provide useful knowledge to the end-users.

Currently, in the Big Data world, the amount of Big Data technologies is overwhelming and sometimes can be difficult to understand and choose the right technology for the right job. For example, for data collection, technologies such as Flume, Kafka, or Talend can be used. For data preparation and enrichment, we can use Spark or Storm. For data storage, Hive with HDFS, NoSQL databases, or Kudu can be used. For machine learning tasks, we can use Spark, H2O, and TensorFlow [18]. For query engines, Impala, Presto, or Drill can be used. For data visualization, tools like Tableau, Power BI, JavaScript can be used [19].

Due to the organizational requirements and due to the technologies available in the organization depicted in this demonstration case, the technological architecture presented in Figure 2 was used to support this demonstration case. Nerveless this technological architecture can be used inside others organization's logistics departments, assuming the goals and requirements are similar to the ones depicted in this work. In case of distinct requirements, some technologies could be adjusted. Regarding data ingestion from the sources, this work uses Sqoop. Despite the fact that Sqoop can only connect to structured databases [20], due to the fact that, for this demonstration case, the organization's data sources were only SQL databases, there was no need to use another technology to ingest the data. After the data is retrieved from the sources, the same is stored in HDFS, using the Parquet format, which is one of the several formats that can be used to store data in HDFS. Other formats that can be used are, for example, ORC or AVRO [21]. Parquet was chosen not only due to its adequate compatibility with Spark and Impala technology but also due to its read-oriented format and with adequate compression, which will bring advantages when we need to query the data [22]. Moreover, it was necessary to develop a Bash script in order to provide a mechanism to create data backups in the Sandbox Storage and in the BDW.

Spark was the chosen framework due to its data cleansing and transformation capabilities and due to the capability to develop several machine learning models. Spark has the SparkSQL [23] library that allows the use of SQL functions in conjunction with the Spark programming API and complex libraries such as Spark MlLib [24]. Being able to perform all these tasks in one unique framework is a significant advantage, since, in this way, it is not necessary to spend more time using and learning different technologies. Moreover, Spark is compatible with Parquet files and Hive, which will be used to provide the data and metadata to the end-users.

Hive includes the Hive Metastore (the system catalog) where the metadata (schema and other statistics) are stored, allowing proper data exploration and query optimizations [15]. Hive allows the creation of external tables where data is stored in HDFS directories and its life cycle is not managed by Hive [15]. Within Hive, we create two levels of interaction with the data. In the first level, the data is modelled using a data-driven approach where the core entities (such as Needs, Stocks, Products, among others) and other entities like Date and Time are stored. This layer allows ad hoc access to the data from these entities to be used by any team or project. In the second layer, the application layer, a new set of objects (materialized tables or views), oriented to the applications'

[1] https://hadoop.apache.org/
[2] https://sqoop.apache.org/
[3] https://impala.apache.org/

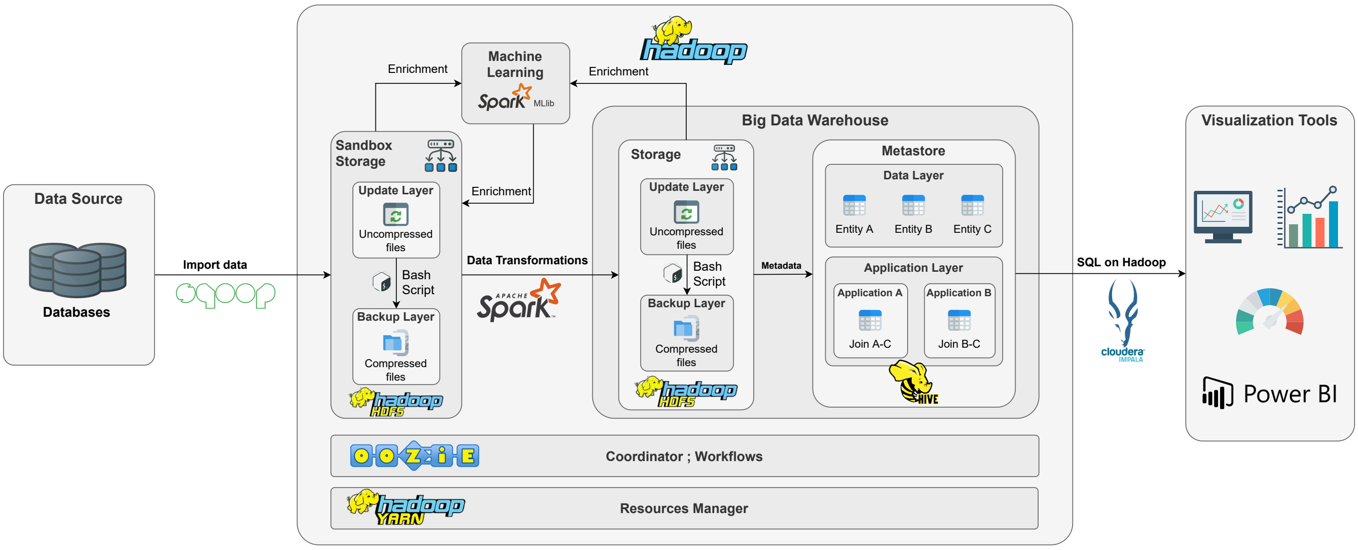

Figure 2. Technological architecture

needs, are created to provide access to the specific data that each application or project needs. This will provide more personalized access to the data that will increase the application performance and higher business agility, thus each team can create their own tables or views as they need.

Impala provides a massively parallel processing (MPP) SQL engine that combines the flexibility and scalability of Hadoop with the familiarity of SQL and has proven to be generally faster than Spark or Hive according to Qin et al [25] and to Bittorf et al [17]. Impala can too be used to query data from HBase and provide a connection to visualization applications, such as Tableau or Power BI, where dashboards can be developed to present to the end-user the knowledge retrieved from the data [17].

This technological architecture supports all the requirements of this project, granting that we can allow the data analysis team to provide knowledge to be used by the end-users, in order to support their decisions and therefore improving the organization's results. Moreover, it can be used in other Logistics 4.0 projects to create a new centralized repository that aggregates different data sources and requires predictive capabilities.

4. Demonstration Case

The application domain addressed in this paper is the Logistics Innovation Department of an automotive factory. In this context, the logistics department handles large volumes of data related to nearly 7000 raw materials from a set of about 400 suppliers spread all over the globe, which impact the production of about 1100 finished products. With regard to internal logistics management, the department is responsible for monitoring and analysing data and material movements referring to approximately 85 daily scheduled deliveries, in order to ensure the supply of material necessary for the proper functioning of about 100 production lines associated with various high-service level customers. In light of the complexity of the organization's supply chain topology, the organization intends to foster the proposal, development and evaluation of Big Data Analytics tools capable of integrating and automating a large part of the logistics processes that, until now, are managed by conventional spreadsheets extracted from classic and parameterizable material requirements planning (MRP) methodologies existing in a given enterprise resource planning (ERP) system.

It is an essential department inside of a production facility and deals on a daily basis with orders, deliveries, delays, production plans, inventory, among other processes. These business processes are crucial to maintain the production lines working and to deliver in time the finished goods to the clients. It is a complex and enormous department with countless business processes.

Due to this complexity, the implementation of a BDW needs to be addressed in an interactive way, choosing one process at a time, looking at the data sources, selecting the appropriated attributes and modelling the data in a data-driven approach that has as a final goal an integrated BDW supporting Logistics 4.0.

Therefore, in this specific case, to start the BDW proposal we analysed the processes that should be considered the core component of this BDW. With the collaboration of key experts in the logistics department, the following processes were selected: Product Inventory, Delivery, Purchase Order, and Needs. This is the first task in the development process presented in Figure 3

These processes will be the main drivers of the analytical objects in the BDW. Besides these objects, other objects will be created, such as a spatial object with information related to countries, Date and Time objects, and complementary analytical objects such as Product, Plant and Vendor. Each one of these processes are supported by one or more tables in the Enterprise Resource Planning (ERP) used by the organization. These different types of objects are explained later in this section.

The understanding and selection of the business processes, together with the understanding and selection of the data sources compose the first activity of the development process (Figure 3) called Data Understanding. In this activity it is necessary to understand the data from the data sources, namely the tables associated to each business process, how they are related, their private and foreign keys, the meaning and possible values of each attribute, among other steps. The second task is to select what tables will be used to develop the BDW.

The next activity is related to the "Data Quality" activity. Data quality is one of the most important tasks in data-related projects. In this case, this activity has significant importance due to the complexity of the data sources and their high number of attributes. For example, some transactional tables have more than 200 attributes, although many of them are not used. In our demonstration case, data quality criteria were defined to verify if an attribute will be used in the BDW. In this specific case, we established that any attribute with more than 90% of empty or nulls values will not be used. This rule was essential to limit the number of used attributes, excluding the ones that have low analytical value. Another rule that was used was to manually verify if the attributes with only one or two distinct values were worth to use. All these rules were defined considering the organizational and decision-making context. The next step was to produce the data quality reports through the execution of several spark jobs that analysed the data extracted from HDFS. The attributes that will be part of the BDW are selected applying the previously defined data quality criteria.

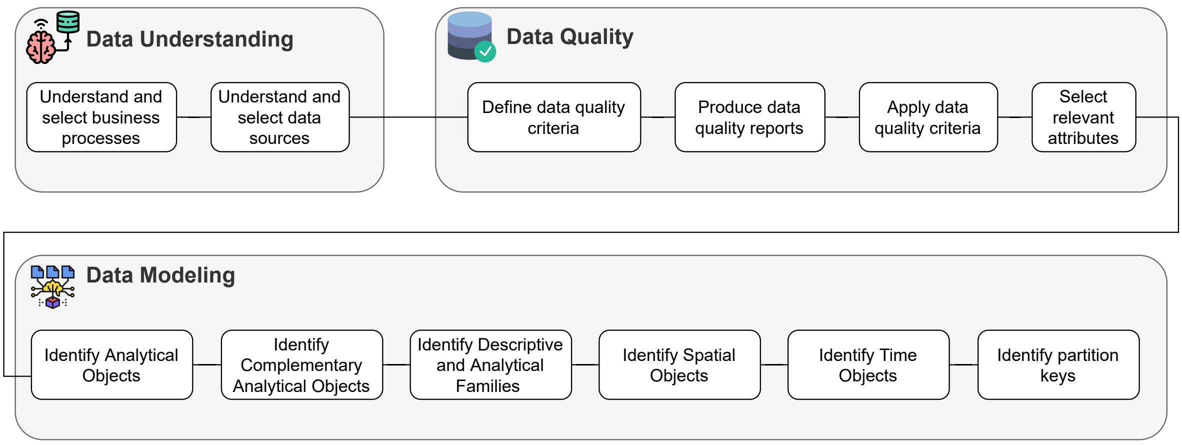

Figure 3. Development process

After the Data Understanding and the Data Quality, it was possible to model the BDW. To do that, the modelling methodology presented by Santos and Costa [9] was applied in order to propose a data model capable of integrate a significant amount of data. The methodology is based on the creation of the following objects: Analytical Objects, Complementary Analytical Objects, Spatial Objects, Time Object, and Date Object.

An Analytical Object is a subject of interest, highly denormalized and that can answer queries by itself avoiding joins with other objects. These objects are directly related to the business processes such as sales or deliveries and should be the firsts to be analysed and identified in order to verify if it is necessary, or not, to create Complementary Analytical Objects. A Complementary Analytical Object is an object that includes attributes usually used or shared by different Analytical Objects and that can be used to complement the analysis of other objects, such as the Analytical Objects. Each object can be divided into two distinct parts, the descriptive and analytical families. These families provide a logical group for the object attributes depending on their type and purpose. The descriptive family group all the attributes that can provide different views of analysis, while the analytical family group the attributes with values to analyse the object. These objects can be integrated with the use of join operations [9]. Figure 4 presents the data model identified with the application of this methodology. Due to privacy concerns, it is only possible to disclose some of the attributes present in the several objects This data model was developed in the logistics context of this specific factory but can be used as starting point for any logistics department of any organization.

The Analytical Objects used in this work are: Product Inventory that has all information about the stocks of each product; Deliveries that has information about when each order is delivered; Purchase Order that has information about how many products are ordered; and Needs that has information about production lines needs.

The Time and Date objects were created from scratch and populated with information related to each one. For example, in the Date object, we created boolean attributes such as week_day, weekend, summer, winter, monday, tuesday, and others. In the Time object, attributes such as lunch-time, in-office, out-office, rush hour, were created. This allowed us to analyse the relevant information and contextualize it in time and date.

The Complementary Analytical Objects had emerged in the data modelling process due to the need of analysing different Analytical Objects using data from the Complementary Analytical Objects. In these objects were stored relevant and specific information that can provide useful information when used together with data from several Analytical Objects. From these objects, we can highlight the following: Plant, Product Valuation, and Vendor.

The object Country is a Spatial Object due to the geographical domain that includes information from the transactional database and from a JSON file (already stored in HDFS) with more information, such as the continent name.

The implementation process presented in Figure 5 starts with the data extraction performed using Sqoop and Oozie Workflows and all data was stored in a HDFS directory called Sandbox. This Sandbox directory allows the storage of all raw data and it is divided into sub-directories where each data source has its own directory and is divided into tables or entities. In this demonstration case, two data sources were used, the transactional database and a JSON file.

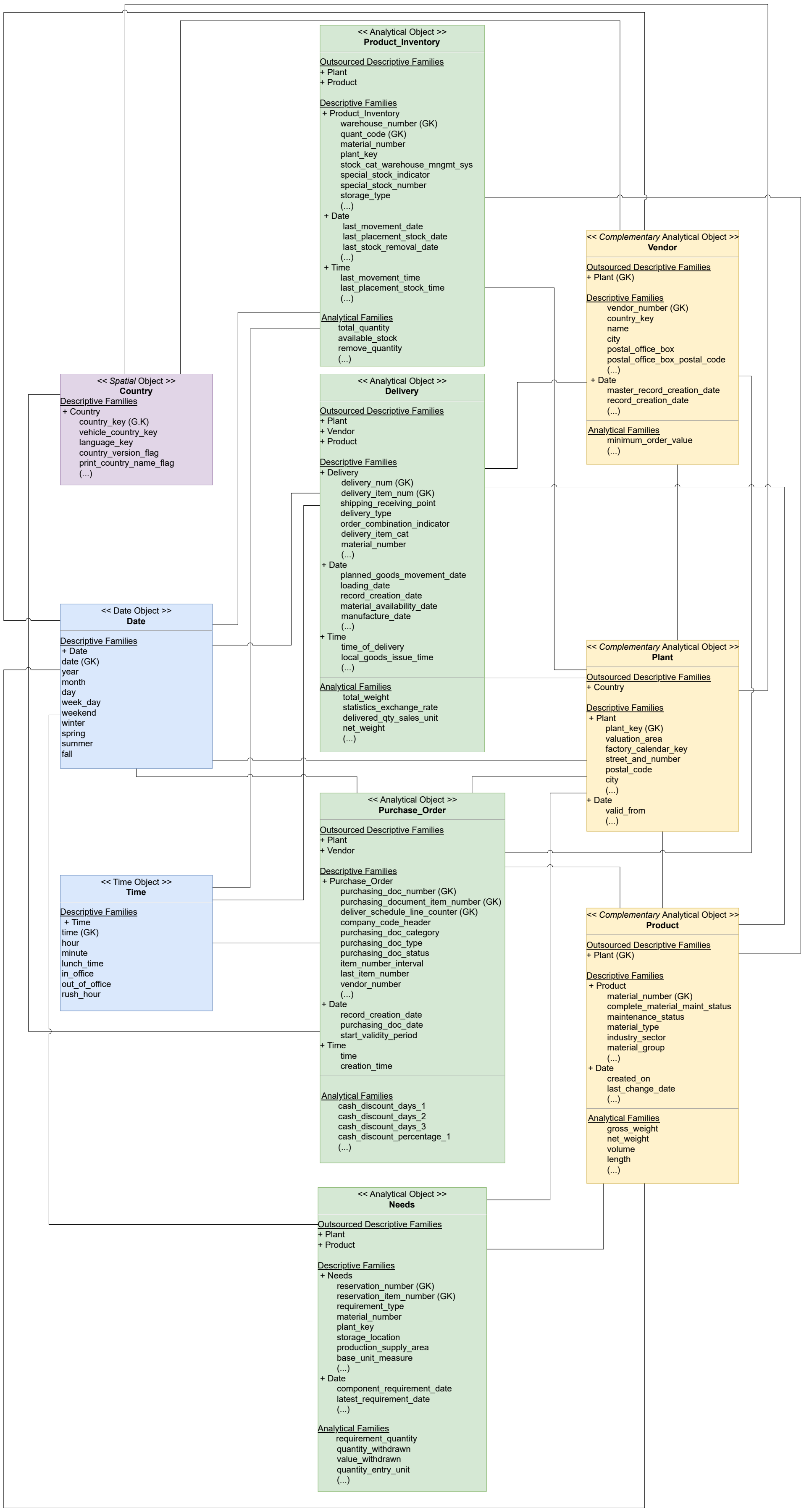

Figure 4. BDW Data Model

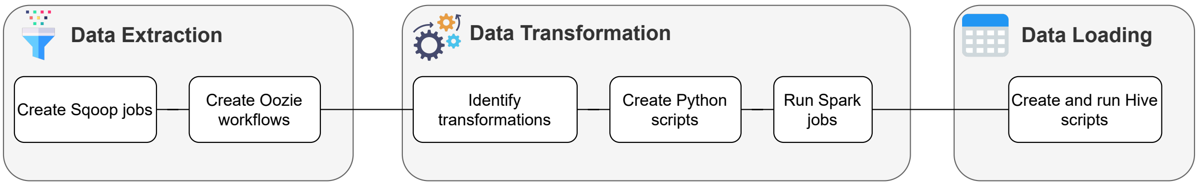

Figure 5. Implementation Process

With all the necessary data stored in HDFS, we can use Spark to perform the data transformation phase, where transformations and partitions keys are identified. Moreover, it is in this phase that the data enrichment can be performed with predictions from the machine learning models.

After the data transformation, the data is stored in the BDW where one table represents one of the objects included in the data model. Moreover, when the size of the object is too large to be used as one unique file, the object is partitioned according to their partition keys in order to improve the performance when querying the data. Furthermore, external Hive tables were created to provide Impala access to data. Impala will be the SQL query engine that allows the connection between Power BI and the data stored in HDFS.

5. Results and Discussion

In this section, we discuss the efficacy and efficiency (5.1) of the BDW implementation. In subsections 5.2 and 5.3 it is presented the challenges and opportunities that arise while data-related projects are developed.

5.1. Efficacy and Efficiency

With the BDW implementation, it was possible to create a data repository that includes several businesses processes of the logistics department. Each process contains data from one or more tables from the transactional database used by the organization.

The data model is dynamic and able to change quickly, in order to include more tables, with more information related to any object that already exists in the BDW or to create new ones. The Time and Date objects can be used with other objects to understand the organization temporal dynamics, such as understand if there are any specific moments in the year where more delays are verified, or even when the suppliers are usually late with the deliveries. Similar reasoning can be used with the objects Plant and Inventory to analyse which plant has more inventory in its storage facilities.

With this work, it is now possible for the practitioners to use raw data extracted from the data sources (using the Sandbox layer) or use data already cleaned and transformed using the BDW layer. This can be achieved using the BDW Hive tables (as an example, Figure 6 shows the Country table view using the HUE interface) or the parquet files stored in the HDFS. They can also create specific materialized objects in the Application Layer in order to decrease the time needed to query the data. This reduces or even avoid the initial development time needed to understand, extract, store, and transform data.

The Machine Learning component can also use data from the different architecture components to provide useful predictions. For example, the available data can be used to predict if some scheduled delivery will be late or not. With this information, the logistics planners can take actions to reduce the impact of this situation. This can be achieved using data from the Sandbox or from the BDW. Machine Learning models can be created with this data using the Spark ML framework. Both the model and the predictions are stored in the HDFS being available for later use and for possible updates in the future. Furthermore, this data is now accessible to the organization through Impala connector and can be used to provide different insights about the organization status, or even in projects that use Machine Learning to predict or classify data to help in the decision making. This means that the time and the necessary knowledge to develop useful dashboards for management is smaller. In Figure 7, a dashboard that analyses

PROPERTIES

Table

External and stored in location

Created by aed1brg on Tue May 25 12:11:44 CEST 2021

STATS

Files 1 Rows 801 Total size 30.75 KB

Data last updated on 05/25/2021 11:11 AM +01:00

SCHEMA

Filter...

Column (17)	Type	Description	Sample	
country_key	string		HU	BD
vehicle_country_key	string		H	BD
language_key	string		H	E
country_version	boolean		true	false
print_country_name	boolean		false	true
iso_code	string		HU	BD
iso_code_3_char	string		HUN	BGD
iso_code_nume_3_c...	string		348	050
eu_member	boolean		true	false
nationality	string		165	460
altern_cntry_key	string		064	666
trde_stat_short_name	string		UNGARN	BANGLA
date_form	string		1	Unknown
country_currency	string		Unknown	BDT
continent_code	string		EU	AS
continent_name	string		Europe	Asia

Figure 6. Country table in Hive

historical and predicted data is present, showing information about deliveries. It is an overview where the historical and predicted delayed or at time deliveries are analysed in several dimensions.

The top right of the dashboard shows the number of products that belongs to each category (A, B or C). This product classification demonstrates how important is each product for the organization. Products classified with A mean that this product is expensive for the organization and normally with more lead time, for example, electronic screens. The B category is for products less expensive, and the C category is for cheap products such as bolts. The impact on delays for products classified with A is superior to the products classified with B and C. The graph shows that are a bigger number of deliveries of C classification products demonstrating that this type of product has more frequent deliveries. So, if for some reason there is a shortage in stock of this product type, the organization will be able to solve that problem rapidly.

The two graphs in the lower-left corner of the dashboard compare the on-time deliveries and the delayed deliveries analysed by the season year. Each one compares the historical data and the predictions made by the machine learning algorithm. The left one shows that the predictions followed the trend of the historical date. The right one shows that is predicted an increase in delays in Autumn. With this information, the organization can prepare mitigation actions to decrease the impact of the delays.

The middle graphs compare the delayed deliveries and on-time deliveries by transportation mode. For example, we can see that the predictions (centre lower graph) show a general increase in the percentage of on-time deliveries.

The right side graphs compare the historical data with delays and the predictions. Bigger the circle means that are more deliveries from that country that arrive with delays.

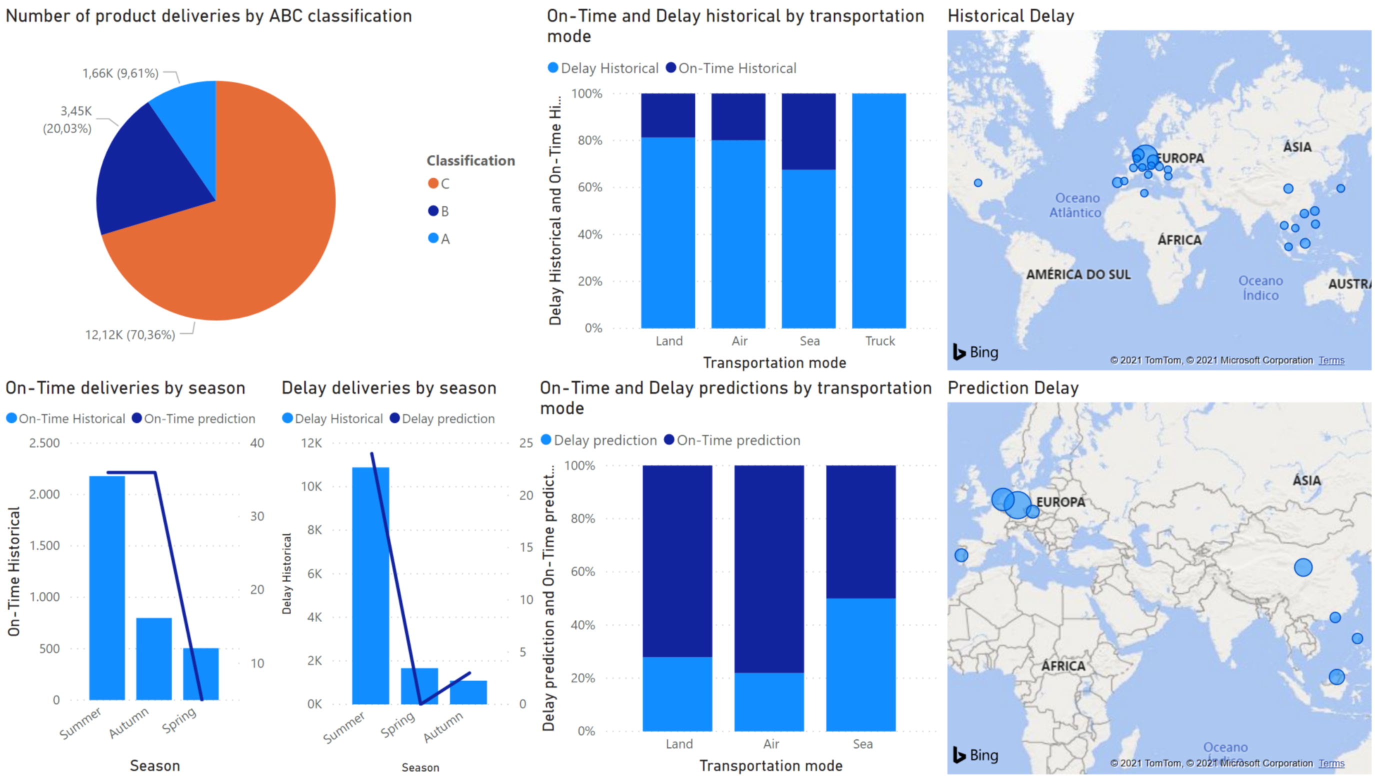

Figure 7. Dashboard with historical and predicted data related with deliveries

We can see that are more delays from products shipped by European countries. The same is visible on the predictions.

These results are based on a portion of the historical data provided by the organization. In future work is necessary to verify if the predictions comply with reality and probably improve the model quality with more data.

5.2. Challenges

The implementation of new technology inside the organization's logistics department can be difficult and rises diverse types of challenges. These challenges can be related to the technology itself, with the lack of knowledge to develop the project, with the organizational culture, with the time and the cost to develop the project, among others. When that technology will use or rely on the provided transactional data to be successful, a new type of challenges related to data emerge.

Moreover, if the organization has a large dimension, can be extremely difficult to get the necessary knowledge to understand the different business processes inside the logistics department and the data generated by them. For example, if we are inside of a multinational organization, with diverse divisions, spread by multiple countries, with a complex transactional database, the data understanding will be one of the most challenging steps in the project.

The following list provides the identification and brief characterization of the most relevant challenges that can be faced while developing Big Data projects.

1. Data and technological challenges
    - Data Understanding
      Understanding the data that is stored in the transactional database is usually a challenge, even worse when the organization is a multinational with a considerable dimension. Transactional databases are complex systems, with misleading tables and attributes names. The existing documentation about the data source is usually sparse, not given enough insights about the data.

Several logistics concepts need to be known, such as safety stock, safety time, delivery time, procurement, among others, in order to better understand the data and their relationships.

- Poor or missing raw data
  When an organization starts a project that will use the raw data generated by the daily business, it is necessary to identify if the necessary data is being generated and stored in the transactional system and its overall quality. Sometimes the project goals can not be achieved due to the lack of data or data with quality. In complex ERP systems is possible to verify that many attributes are not used by the organization. For example in logistics, knowing where an order is in transit to its destination can be very useful to predict if it will be on time, or not, and to make decisions about how to avoid stops in the production line.
- Different values in different data sources for the same attribute
  Due to the large and complex transactional system, is fairly common to find the same attribute in different tables, related to the same entity, but with different values. Understand why this happens and understand the type of situations that motivate this type of behaviour can be difficult.
- Technological infrastructure
  The adequate technological infrastructure is essential to stable a project development. In an organization, the technological infrastructures can be based on outdated technology or the technological infrastructure can change during the project lifetime. This will lead to a project adaptation to the existing technologies or their evolution as the infrastructure change.

2. Organizational challenges
   - Access to data and to a technological infrastructure
     One of the first tasks in projects of this nature is to get access to data and to the infrastructure that will be used to process and store it. This is a task that needs to be done at the beginning of the project and where the organizations' policies can interfere in a negative way. This can not be an obstacle or take a long time to overcome.
   - Understand the business processes
     Commonly, large organizations have many and complex business processes, with diverse rules, exceptions and paths, which can be difficult to understand. Moreover, the documentation about the business processes can be insufficient, creating another obstacle in this type of project. In the logistics area, where daily interactions with the suppliers and their systems exist, where processes are complex in order to achieve better results in the production line, and where concepts such as just in time production are being implemented, the documentations has a relevant impact when new projects start to be developed.
3. Project team challenges
   - Lack of knowledge in the used technologies
     As Big Data is a recent concept, there is a lack of human resources with experience in the technologies used to support this concept. Building a team without any experience in Big Data can lead to several problems in the project. Moreover, when adding specific requirements of a complex area like logistics, more difficult is to get multidisciplinary teams with knowledge in both areas.
   - Lack of sufficient human resources
     To develop such a complex project, the project team needs an adequate number of human resources. The lack of sufficient human resources can cause delays in project development. Teams with a high number of elements can be prejudicial

to the project too, but very small teams lead to a lack of different backgrounds and points of view that can hinder the project.

The challenges enumerated in this section are some of the biggest challenges that a team can encounter while develop and implement a BDW inside of an organization with a considerable size. The challenges can cause delays in the project milestones and they should be taken into account when the project is planned. Most of them can be mitigated with simple actions such as grant early access to all necessary resources and develop the necessary documentation in all projects..

5.3. Opportunities

When an organization go through a technological change such as the creation of a BDW, some opportunities emerge. Indeed, we can say that each challenge can be transformed into one opportunity. Therefore, we will take the challenges provided in section 5.2 and transform them into opportunities.

1. Data and technological opportunities
    - Improve documentation
      Very often, documentation is treated as the less important part of the project. The time and effort put in the documentation development are lower than required, leading to poor documentation. With the development of a new project, the poor documentation of the previous one becomes evident. The effort that needs to be done to understand the previous project can be reused to improve the documentation and, therefore, decrease the time and effort needed for the next ones.
    - Improve data quality
      Data quality is essential to the development of these data-based of projects. As we need to perform data quality tasks, this can be used to detect and report data problems that can be fixed in the near future. This can be useful not only for this project but even for past and future projects.
    - Technological infrastructure
      A new project that requires new technology can be an excellent driver to improve the technological infrastructure existent in the organization. These changes can include, for example, updating the existent technologies or the implementation of new ones.
2. Organizational opportunities
    - Improve internal processes
      With the implementation of new technology, some internal processes will be analysed and can be improved. Moreover, processes can use the newly available technology to improve their performance.
    - Improve business processes documentation
      Many analytical teams do not know the business processes and they need to found the right person to ask. Often, if they ask the same question to different persons, they will get different answers. Properly document the business processes can be a key way to improve the business understanding not only inside the analytical teams but for the organization in general.
3. Project team opportunities
    - Creation of a team specialized in Big Data technologies
      Research projects can have a tremendous impact on organizations, not only by the obtained results but also by the improved capabilities of human resources. In this specific case, the creation of one team specialized in Big Data technologies can boost more projects, more efficiently, and with more efficacy.
    - Improve workers knowledge in logistics processes
      Human resources with more business knowledge can bring their knowledge to other projects and have a positive impact on them. This can be verified

not only in new ones but also in the maintenance and improvement of other ongoing projects.

- Improve workers knowledge about data sources
  Data analytics projects always depend on the data source. Knowledge about them is essential for a good start and a proper development of the project. It is crucial to have in the project team, at least, one specialized resource in the data sources, helping the development team to understand the data.

Besides the enumerated opportunities, other opportunities can arise with the creation and implementation of a BDW in a logistics department. For example, new projects can be initiated and use the BDW as their data source, providing integrated and consolidated data for their timely development. Other departments can use data in the BDW to improve their predictions and their decision making needs.

6. Conclusions and Future Work

This paper presented the proposal and implementation of a BDW into a logistics department of an automotive factory. The implementation of the BDW is the starting point to push the concept of Logistics 4.0 in this facility, improving the analytical capabilities and supporting the decision-making process in the logistics department.

Through this work, we presented the logical and technological architecture that support the implementation of the BDW that includes several logistics processes. Moreover, we presented the proposed BDW data model. The BDW data model is a key element to get insights about the current state of the organization and to support the logistics planners' decisions in an efficient way. The logical and technological architecture, as well as the data model can be used as starting a point to develop and implement a BDW in similar logistics departments.

As we advance, we faced several challenges and opportunities in the BDW development and implementation. One of the most difficult challenges was to understand the several logistics processes and how the data of these processes is stored in the transactional system. Finding the right data to support the proposed system was a difficult and time-consuming task. Nevertheless, the most important thing is to be aware of the challenges and implement mitigation plans in order to solve them, or at least decrease their impact on the project final results. Other challenges that can be faced in this area are related to the technologies and the available infrastructure used by the organization. Sometimes the technological infrastructure is changing during the project what can lead to several project changes. Moreover, the available infrastructure can include outdated technologies or be short in resources when used by several teams at the same time.

In the opportunities field, there are several points that can be addressed to improve the organization, the logistics department, and the next projects. But these opportunities need to be addressed in new projects with a well-defined goal and scope, due to the new challenges that these projects will rise. Organizations need to promote a culture of continuous improvement to face these opportunities.

As future work, the BDW implementation can be improved by automatizing the data extraction, transforming, and enrichment pipelines to increase the performance and decrease the human intervention. Moreover, the data model can be extended by adding new objects (complementary or analytical) in order to enlarge their scope or improving the existent ones by adding new data to the already existing objects. Furthermore, more machine learning models can be created and integrated into the existing BDW to enrich the data and provide predictions to help the logistics planners. Also, the implementation of a real-time layer should be taken into consideration.

Author Contributions: Conceptualization, Nuno Silva and Julio Barros, Writing - original draft, Nuno Silva and João N.C. Gonçalves; Writing - review & editing, Maribel Y. Santos and Carlos Costa; Investigation, Nuno Silva and Julio Barros; Software, Nuno Silva and Julio Barros; Supervision, Maribel Y. Santos, Carlos Costa, Paulo Cortez and M. Sameiro Carvalho.

Funding: This work has been supported by **FCT** – Fundação para a Ciência e Tecnologia within the R&D Units Project Scope: UIDB/00319/2020, the doctoral scholarship grant: PD/BDE/142895/2018 and PD/BDE/142900/2018.

Acknowledgments: This work has been designed using resources made by: Those Icons, Pixel perfect, DinosoftLabs, Becris, Smashicons, and Freepik from www.flaticon.com.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Panetto, H.; Iung, B.; Ivanov, D.; Weichhart, G.; Wang, X. Challenges for the cyber-physical manufacturing enterprises of the future. *Annual Reviews in Control* **2019**, *47*, 200–213.
2. Winkelhaus, S.; Grosse, E.H. Logistics 4.0: a systematic review towards a new logistics system. *International Journal of Production Research* **2020**, *58*, 18–43.
3. Strandhagen, J.O.; Vallandingham, L.R.; Fragapane, G.; Strandhagen, J.W.; Stangeland, A.B.H.; Sharma, N. Logistics 4.0 and emerging sustainable business models. *Advances in Manufacturing* **2017**, *5*, 359–369.
4. Costa, C.; Santos, M.Y. Evaluating several design patterns and trends in big data warehousing systems. International Conference on Advanced Information Systems Engineering. Springer, 2018, pp. 459–473.
5. Chevalier, M.; Malki, M.E.; Kopliku, A.; Teste, O.; Tournier, R. Implementing Multidimensional Data Warehouses into NoSQL. 17th International Conference on Enterprise Information Systems (ICEIS 2015) held in conjunction with ENASE 2015 and GISTAM 2015. INSTICC - Institute for Systems and Technologies of Information, Control and Communication, 2015, pp. 172–183.
6. Gröger, C.; Schwarz, H.; Mitschang, B. The Deep Data Warehouse: Link-Based Integration and Enrichment of Warehouse Data and Unstructured Content. 2014 IEEE 18th International Enterprise Distributed Object Computing Conference, 2014, pp. 210–217. doi:10.1109/EDOC.2014.36.
7. Kiran, M.; Murphy, P.; Monga, I.; Dugan, J.; Baveja, S.S. Lambda architecture for cost-effective batch and speed big data processing. *Proceedings - 2015 IEEE International Conference on Big Data, IEEE Big Data 2015* **2015**, pp. 2785–2792. doi:10.1109/BigData.2015.7364082.
8. NBD-PWG. NIST Big Data Interoperability Framework: Volume 6, Reference Architecture. Technical Report NIST SP 1500-6, National Institute of Standards and Technology, 2015.
9. Santos, M.Y.; Costa, C., Big Data: Concepts, Warehousing, and Analytics. In *Big Data: Concepts, Warehousing, and Analytics*; River Publishers, 2020; pp. 1–284.
10. Chou, S.; Yang, C.; Jiang, F.; Chang, C. The Implementation of a Data-Accessing Platform Built from Big Data Warehouse of Electric Loads. 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC), 2018, Vol. 02, pp. 87–92. doi:10.1109/COMPSAC.2018.10208.
11. Sebaa, A.; Chikh, F.; Nouicer, A.; Tari, A. Medical Big Data Warehouse: Architecture and System Design, a Case Study: Improving Healthcare Resources Distribution. *Journal of medical systems* **2018**, *42*, 59. doi:10.1007/s10916-018-0894-9.
12. Santos, M.Y.; Martinho, B.; Costa, C. Modelling and implementing big data warehouses for decision support. *Journal of Management Analytics* **2017**, *4*, 111–129. doi:10.1080/23270012.2017.1304292.
13. Shvachko, K.; Kuang, H.; Radia, S.; Chansler, R. The Hadoop Distributed File System. 2010 IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST), 2010, pp. 1–10. doi:10.1109/MSST.2010.5496972.
14. Dean, J.; Ghemawat, S. MapReduce: Simplified Data Processing on Large Clusters. *Commun. ACM* **2008**, *51*, 107–113. doi:10.1145/1327452.1327492.
15. Thusoo, A.; Sarma, J.S.; Jain, N.; Shao, Z.; Chakka, P.; Anthony, S.; Liu, H.; Wyckoff, P.; Murthy, R. Hive: a warehousing solution over a map-reduce framework. *Proceedings of the VLDB Endowment* **2009**, *2*, 1626–1629.
16. Spark, A. Apache spark. *Retrieved January* **2018**, *17*, 2018.
17. Bittorf, M.; Bobrovytsky, T.; Erickson, C.; Hecht, M.G.D.; Kuff, M.; Leblang, D.K.A.; Robinson, N.; Rus, D.R.S.; Wanderman, J.; Yoder, M.M. Impala: A modern, open-source sql engine for hadoop. Proceedings of the 7th biennial conference on innovative data systems research, 2015.
18. L'Heureux, A.; Grolinger, K.; Elyamany, H.F.; Capretz, M.A.M. Machine Learning With Big Data: Challenges and Approaches. *IEEE Access* **2017**, *5*, 7776–7797. doi:10.1109/ACCESS.2017.2696365.
19. Costa, C.; Andrade, C.; Santos, M.Y., Big Data Warehouses for Smart Industries. In *Encyclopedia of Big Data Technologies*; Springer International Publishing: Cham, 2018; pp. 1–11. doi:10.1007/978-3-319-63962-8_204-1.
20. Aravinth, S.; Begam, A.H.; Shanmugapriyaa, S.; Sowmya, S.; Arun, E. An efficient HADOOP frameworks SQOOP and ambari for big data processing. *International Journal for Innovative Research in Science and Technology* **2015**, *1*, 252–255.
21. Ivanov, T.; Pergolesi, M. The impact of columnar file formats on SQL-on-hadoop engine performance: A study on ORC and Parquet. *Concurrency and Computation: Practice and Experience* **2020**, *32*, e5523.
22. Baranowski, Z.; Grzybek, M.; Canali, L.; Garcia, D.L.; Surdy, K. Scale out databases for CERN use cases. J. Phys. Conf. Ser, 2015, Vol. 664, pp. 042–002.

23. Armbrust, M.; Xin, R.S.; Lian, C.; Huai, Y.; Liu, D.; Bradley, J.K.; Meng, X.; Kaftan, T.; Franklin, M.J.; Ghodsi, A.; Zaharia, M. Spark SQL: Relational Data Processing in Spark. Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data; Association for Computing Machinery: New York, NY, USA, 2015; SIGMOD '15, p. 1383–1394. doi:10.1145/2723372.2742797.
24. Meng, X.; Bradley, J.; Yavuz, B.; Sparks, E.; Venkataraman, S.; Liu, D.; Freeman, J.; Tsai, D.; Amde, M.; Owen, S.; Xin, D.; Xin, R.; Franklin, M.J.; Zadeh, R.; Zaharia, M.; Talwalkar, A. MLlib: Machine Learning in Apache Spark. *J. Mach. Learn. Res.* **2016**, *17*, 1235–1241.
25. Qin, X.; Chen, Y.; Chen, J.; Li, S.; Liu, J.; Zhang, H. The Performance of SQL-on-Hadoop Systems - An Experimental Study. 2017 IEEE International Congress on Big Data (BigData Congress), 2017, pp. 464–471. doi:10.1109/BigDataCongress.2017.68.