DECIDE: A New Decisional Big Data Methodology for a Better Data Governance

Mohamed Mehdi Ben Aissa[1](✉), Lilia Sfaxi[2](✉),
and Riadh Robbana[2](✉)

[1] Tunisia Polytechnic School (University of Carthage), Tunis, Tunisia
mehdi_benissa@yahoo.fr
[2] INSAT (University of Carthage), Tunis, Tunisia
lilia.sfaxi@insat.u-carthage.tn,
riadh.robbana@gmail.com

Abstract. Big Data technologies and approaches have an important impact on the organization and governance of the enterprise. With such a high volume of structured & unstructured data, real time and mutualization needs, it is quite complicated to keep a high quality of data by respecting the governance rules and best practices. In addition, new team roles and organization must be applied in order to adapt to the new Big Data decisional constraints. In this direction, we present in this paper an overview of DECIDE, a decisional Big Data Methodology. We focus, particularly, on its team workforce, data quality, storage and governance fundamentals, rules and steps.

Keywords: Big Data Methodology · Analytics · Enterprise systems · Governance · Data quality · Team management · Decision-making

1 Introduction

With the continuous increase of data volume and the need for real time and accurate decision-making, companies rely more and more on Big Data systems and principles to get the best out of their data [1]. These systems are distinguished by the use of federated storage where data from multiple sources are gathered, and where multiple actors are involved in the value creation and consumption. These characteristics, coupled with many shared resources and continuous flows of events and requests, call for a well-defined data governance strategy.

Data Governance is defined as *"a set of policies and procedures adopted in order to manage data in an organization"* [2]. Correct policies are able to protect data from various internal and external risks that can threaten their security and alter their value. At the same time, their application should not be too restrictive that it affects the needed performance and efficiency in value extraction and manipulation [3].

Defining the right data policies in an organization can be of a great impact at the management as well as the technical level. Indeed, an adequate team management policy needs to be defined, to specify the roles and authorizations assigned to each user of the system. The data pipeline needs to be formalized, and storage systems must be consistent and synchronized. All these constraints come with a cost that can be really demotivating for decision makers.

M. Themistocleous et al. (Eds.): EMCIS 2020, LNBIP 402, pp. 63–78, 2020.
https://doi.org/10.1007/978-3-030-63396-7_5

On the other hand, Big Data technologies are evolving, day after day, by offering new features and capacity to deal with the demanding users. However, architects and developers find it increasingly hard to implement adequate solutions for their needs. They try to initiate internal Big Data and decisional projects, which tend to be very complex, with a high level of risk, and the need to involve a big number of actors with various skills. In fact, according to Vidgen et al. [4], the main three management challenges that these companies face in creating value for decision-making are: defining a clear data strategy, finding the right people to conduct a data-driven cultural change, and following information ethics. It is therefore necessary to define the right methodology, that helps organizations address these challenges and determine the desired business and analytics goals [5].

The lack of Big Data decisional methodology with adequate governance rules and fundamentals make it an imperative to define a more comprehensive and adaptable methodology. We refer to this methodology as DECIDE, which stands for **DEC**isional B**I**g **D**ata **ME**thodology.

2 Literature Review

Several decisional methodologies are defined in the literature [6–8]. These methodologies are categorized into three types: **requirement-driven** approaches [9–11], that focus on the business users requirements; **data-driven** approaches [12–14], that focus on the underlying data sources to establish the data warehouse design, and **hybrid** approaches [15–19], that combine the advantages of both data- and requirements-driven approaches, by designing the data warehouse according to available data sources, while taking into consideration the requirements of the business users.

The methodologies cited above apply specifically to classical Business Intelligence applications, where data is provisioned from stable and static sources, the data warehouse is centralized and structured, and ETL(Extract-Transform-Load) jobs are complex and periodic. But these constraints do not always apply to decisional Big Data projects, where data sources can be unstructured or/and generate data flows with a high velocity. Our approach is more exploratory, where the design of the storage systems and global architecture depend strongly on the data sources that include all sorts of real-time and near-real-time events. This is why we qualify our approach as *Event and Data-driven Approach*, as we are adapting to fast-moving and large-scale data sources.

For decisional projects, some teams, such as in D2D CRC (Data to Decisions Cooperative Research Centers) [20], tend to use Agile principles to organize their work, considering these architectural issues like any other development project.

However, agility is difficult to implement in decisional systems, mainly because of the dimensional modeling's rigidity. In fact, the lack of planning and clear architecture can be a hindrance when designing the system, especially when it comes to defining conformed dimensions. Dimensional modeling requires a global view of the users' needs to define the right model for the data warehouse, and it can be problematic to consider the KPIs (Key Performance Indicators) from a limited set of users. But for Big

Data decisional projects, we are not constrained by a frozen structure of the storage system. Agile methods can be very adequate, as the sources change and evolve constantly, and the user's needs can also vary depending on the type and content of these sources. This is why we opt for *agility* as a basis for our methodology.

On another perspective, Data Mining methods, such as SEMMA, KDD [21] and CRISP-DM [22], can be considered as alternatives for Big Data projects. Contrary to classical BI (Business Intelligence) methodologies, Data Mining methodologies can easily adapt to Big Data projects, thanks to their bottom-up approach helping to apply data discovery to make complex analytics [23]. In fact, their main goal is to understand and discover data, in order to deploy adequate data mining models that will help prediction and decision-making.

Some works [24], tried to adapt these Data Mining methodologies to Big Data constraints. However, these solutions are still very strongly related to the specific case of data mining projects, contrary to our more global solution. On the other hand, pure Data Mining solutions do not take into account all transversal but crucial requirements of Big Data systems, such as DevOps approaches and the choice of the architecture, technologies, tools and roles. In addition, governance is still not the priority of this sort of methodology especially when it comes to data quality, storage design and team workforce considerations.

3 DECIDE and Data Governance Fundamentals

Responsible and effective data management requires attention to three main areas: data governance, which is the overarching framework for maintaining quality; the workforce structure, necessary for operational execution; and the quality elements, metrics and remediation approaches.

Data governance [25] is the pillar of the data management field. It defines a set of rules, processes, roles, policies and standards used to ensure the overall management of the availability, usability, understandability and correctness of the data in the company.

Data governance helps the organization in several aspects [26]: to create a shared understanding of the data, to improve data quality in an efficient and cost effective manner, to represent a clear data mapping in order to locate needed data and have a single version of the truth for business entities, to ensure that the data strategy is compliant with government and corporate regulations, and finally, to continuously improve data management policies by defining proper codes of conduct and best practices.

Figure 1 shows the main activities to encompass in every data governance strategy [26], that we tend to apply at various spots in the overall methodology.

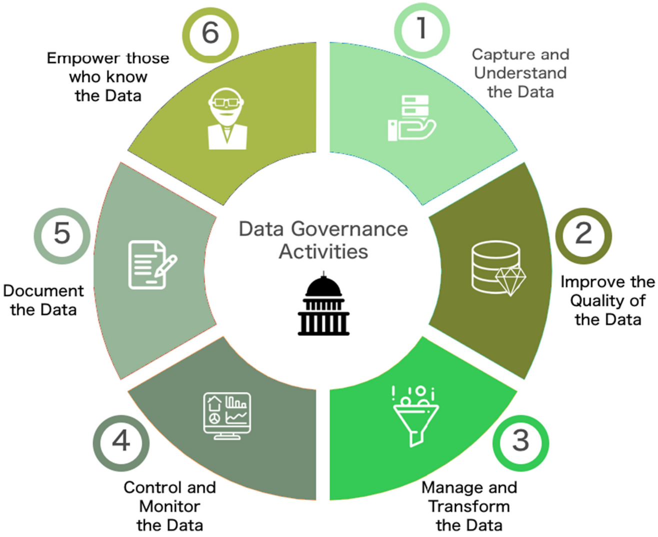
Data Governance Activities
1
Capture and Understand the Data
2
Improve the Quality of the Data
3
Manage and Transform the Data
4
Control and Monitor the Data
5
Document the Data
6
Empower those who know the Data

Fig. 1. Data governance activities

4 Overview of DECIDE

The **DEC**isional **BI**g **D**ata **ME**thodology respects five fundamentals:

- **Agility**: it follows the Agile principles [27].
- **Bottom-up approach**: it is designed independently of specific KPIs.
- **Data and event driven**: it relies on the type and structure of the data sources, and on the various events generated by the environment, to design the data storage.
- **Multi-architectures**: it supports all decisional architectures that respect Big Data constraints.
- **Multi-technologies**: it is designed for any technology or stack.

DECIDE defines four phases: (1) preparation phase, (2) transversal phase, (3) data collection & storage phase, (4) data analysis & presentation phase. Each of these phases is composed of a set of steps, with iterations in the same phase and between phases, showcasing the agility of the methodology. Figure 2 summarizes the main steps and their correlation.

4.1 Preparation Phase

The preparation phase helps to prepare the building blocks of the project. A successful and solid Big Data project should start by defining the initial requirements, considering the target business impact and putting the deployment and automation infrastructure into place.

Initial Requirements Definition. The initial requirements' definition is the result of the first encounters between the development team and the clients. It is done at the

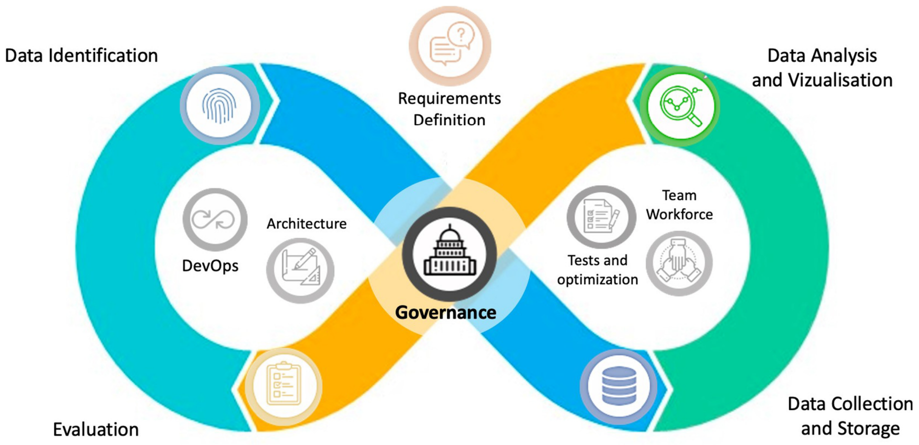

Fig. 2. DECIDE steps

beginning of the project to identify the needs that triggered the implementation of such a system. There are two types of initial requirements:

- **Business requirements**: Where the client states the global needs of the application, and the necessary Business Success Indicators (BSI), to be extracted and represented.
- **Quality requirements**: The definition of the quality requirements justifies many choices, such as the architecture, required tests, or technologies. Several quality attributes can be taken into consideration, such as availability, consistency, security, monitoring, etc.

Agile Approach. Once the requirements and success criteria are defined, the team should produce a first project plan. This first version will be split into different sprints with equal duration, going from one to four weeks. A sprint can include one or many DECIDE steps depending on the context: number of data sources, complexity of the implementation, ecosystem, complexity of the analysis, etc. This part of DECIDE was mainly inspired from the SCRUM methodology [28].

DevOps Approach. Big Data and DevOps are becoming a very close team in big projects. While agility is about adaptation to change and communication, DevOps makes sure that the operations constraints (hardware and deployment environment) are taken into consideration from the beginning at the same level as the functional requirements.

4.2 Transversal Phase

The transversal phase represents recurring actions and operations that occur throughout the execution of the project.

Data Governance. As presented in Fig. 2, Governance is the heartbeat of DECIDE, and not only a step or phase to take into consideration. From an practical point of view, every operation of DECIDE regarding the data life-cycle (extraction, transformation, storage, analysis or linkage) should be recorded and monitored. This enables the construction of the data lineage and the verification of its integrity and confidentiality. A data governance tool should be set up, and used regularly at every step. We recommend implementing an automatic enrichment of this tool during data life-cycle.

Architecture Choice. Choosing the right architecture helps to ensure that the quality requirements defined in the beginning of the project are respected. It depends on several metrics, such as the data and processing characteristics, the quality attributes and the restitution type. The choice of the architecture is not a one-step operation. It is done all over the development phases, as every step of the methodology will contribute in refining your idea of the adequate architecture.

Tests and Optimization. Testing is a critical operation that should occur not only at the end of the project, but also at the end of every step of the methodology. It will determine whether this step needs another iteration, or if we can proceed to the next one. Iterations can be used to optimize code, architectural choices or performance. DECIDE stands out compared to other bottom-up methodologies such as CRISP-DM [22], where the evaluation takes place only at the end of the cycle, instead of at each step, which impacts the agility of the solution and the early detection of faults and design problems.

4.3 Data Collection and Storage Phase

This first phase's purpose is to prepare the environment for the data analysis. This preparation helps mainly in designing the storage system depending on the data to be collected. The design of a system following a bottom-up approach always starts with a data identification step, where various data sources are defined, for immediate use or in anticipation of future needs.

Data Identification. Data can come from many varied physical sources (mainframes, database servers, distributed systems, etc.) that can be of various types (production bases, logs, archived data, etc.). We first need to identify these sources, define their type (streaming, batch or mixed), and reliability. For the latter, a source reliability rating can be used, like for example the *Intelligence source and information reliability rating system* [29], which rates sources from A "*completely reliable*" to E "*unreliable*". From each identified data source, we list the datasets to be collected by describing their characteristics, such as structuring, type, data sensitivity, frequency and throughput, etc. Cleansing and/or enrichment of datasets improves their quality score and prepares the analytics steps. This is a first data scrubbing phase, done individually for each dataset, and can contain operations like identifying and removing duplicate records. In a Big Data approach, as the cost of storage is very low, it's recommended to keep the different versions of the datasets before and after every data cleansing operation.

Data Storage. During this phase, information about the new datasets are added to the governance tool. All storage layers (data lakes and data stores), and relations between them are defined. Access rules are updated. For example, initial data owners, that had all types of rights over their data, now see these rights reduced to read-only, as only the collection processes can insert or update any existing data. Data owners can also grant read access to other users over their data, to enable value extraction.

Data Collection. In a modern decisional architecture and especially for streaming analytics, fresh data must be collected fast and with a high frequency without impacting the data sources. For each data source and its adequate storage, you have to define the collection processes while taking into consideration the transformation type (pre-load or post-load), the pattern (distributed or centralized) and the execution engine (streaming, batch processes or Change Data Capture paradigm).

4.4 Data Analysis and Presentation Phase

This phase aims to design, implement, test, automate, optimize, present and manage the required analysis. The data collected during the first phase is raw, with a low-density level. This means that it is difficult to exploit it directly or extract information from it natively. In this direction, the next steps allow the end users to explore existing data by building KPIs [30], graphs and dashboards via analytic processes.

Data Analysis. Data users must collect the needs of the end users and decision makers. Each needed analysis is described, along with its attributes, pipeline, the frequency and latency of its operations, its type (OLAP, complex non-linear analysis, machine learning or customized), distribution and scalability, and application pattern (mono- or multi-threading, map-reduce, multi-agent paradigm, etc.). All analytical jobs and flows should be traced in the data governance tool including all the details: input, output, intermediate datasets, workflow and other metadata.

Data Visualization. The aim of data visualization is to restore the results of the processing layer via reports, graphs, dashboards, etc. Data restoration tools and technologies need to use ergonomic and modern features, such as auto-refresh, maps and geo-data, in order to improve as much as possible the user experience. Several choices need to be made, such as the visualization types (reports, graphs, dashboards, etc.), their refresh frequency, whether it is on-demand or pre-built, and other specifications such as their security constraints or multi-tenancy.

Evaluation: Global Optimization and Testing. End-to-end testing and optimization is the step where a complete use case is automatically run on the system, and where KPIs' compliance is evaluated. The key indicators that are considered are: performance, scalability, charge and resilience. This phase will ensure the integration of the different layers, look for the optimization issues and perform the integration, consistency and global performance tests. There is also a possibility to add other data sources and other analysis thanks to the layers' independence. This phase is essential in order to validate the global functional and technical specifications, while taking into consideration other constraints, like: cost, pooling resources, global configuration tuning, etc.

DECIDE is data-centric, and defines the necessary governance activities throughout the entire data life-cycle.

According to Margaret Rouse from TechTarget [31], two of the four pillars of Data Governance are defining the owners and custodians of the data (*Data Team Workforce*), and defining the level of completeness and consistency of the data (*Data Storage* and *Data Quality*). We show hereafter how we apply these concepts in DECIDE.

5 Data Governance Focus in DECIDE

5.1 Storage Design for Better Big Data Governance

Let's imagine new business opportunities that can be implemented via new decisional features, such as customer behavior analytics based on navigation logs and metrics. The problem with classical methodologies is that: (1) unstructured data like logs can't be managed and designed using a multi-dimensional approach, (2) once the data is collected, it is difficult to dynamically add new real time data sources, such as logs in this example, and (3) even if (1) and (2) were possible, the data schema must be modified, and we may have to redesign the warehouse (facts and dimension tables) in order to implement new analytics related to new business needs, which is very costly and can cause a non-negligible down-time of the system.

In addition, the biggest challenge when we have to deal with big data storage remains the governance rules that we have to respect in order to facilitate the usage and accessibility of the different datasets.

Indeed, the storage design must reflect the different storage layers and stores identified and to be implemented for persisting the data, their relationships and the data routing process between them.

In order to overcome these challenges, we propose a storage design to be used in the Data Storage Step (Sect. 4.3). This design aims to keep the archived and time-variant data by keeping a high governance layer. This design is decomposed into four main logical layers or stores [32] (Fig. 3).

- **Integrated raw data lake**: where the data is stored from the sources as it is, conserving their structure, relations and possible redundancies. The main objective of this store is to keep a repository containing all the company's data, without alteration.
- **Integrated refined data lake**: where a denormalized and integrated version of the data is created, after an extraction phase, from the raw data lake.
- **Big Data marts**: which represent a set of dedicated, single subject, schemaon-read, structured and decisional stores. They represent logical views of the data lake, where data having the same function and purpose, and destined to the same users, are consolidated.
- **Serving stores**: which are used to persist the results of the analysis phase. These stores are the only ones made accessible to the decision makers via visualization tools.

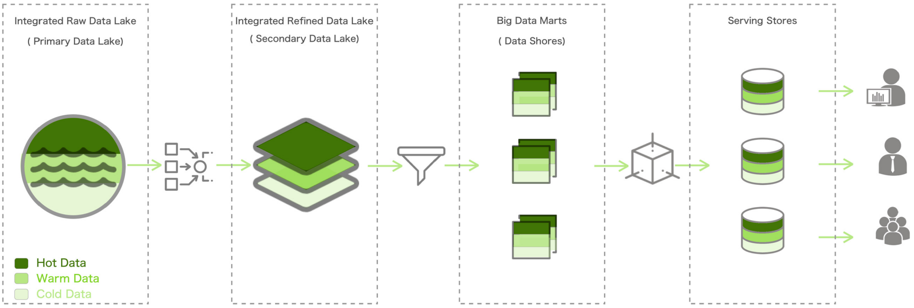

Fig. 3. Big data stores

In this phase, the serving stores can be defined in a preliminary way, using the global information about the types of analysis needed for our requirements. They can later on be changed or refined after the analysis step, in an Agile fashion.

This design aims to keep and trace the different data versions from edge to visualization without any data loss. Data is mutualized, which helps to share the different layers between teams and departments for a better governance.

The architecture of the refined data lake is composed of a set of entities (tables or collections). These entities must be independent and completely noncorrelated, respecting the principles of Big Databases: replication of attributes is thus favored over any dependency relation. The usual normal forms are no longer respected, which makes our refined database *denormalized*. The refined data lake respects these characteristics:

- All the entities should be independent (no physical relationships, no possibility of automatic joins).
- An entity should encapsulate all the dimensions of a specific measure.
- Attributes are never filtered or deleted.
- The new data models are referenced in the data governance tool.

The advantage of this design is that the transformations can be executed via the ELT (*Extract-Load-Transform*) pattern thanks to the Big Data technologies and their high locality level[1] independently of the data sources.

It is important to underline that the raw and refined data lakes are two logical structures of the same global data lake, and should be located in the same physical infrastructure. We notice a physical overhead of non-used or duplicated data between stores during the first implementations. On the other hand, important benefits can be gained from the storage mutualization compared with the data duplication in other approaches. The recommended use of open source Big Data technologies also contributes in costs decrease, thanks to their reduced storage costs and scalability.

[1] High locality level: data storage and processing are localized on the same nodes.

5.2 Data Quality

The most important component when it comes to Data Governance is the Data Quality. It is indeed one of the biggest challenges for all companies dealing with a huge amount of data coming from various sources: how to make sure that you are dealing with quality data, while having a data management process that is at the same time *thorough*, *fast* and *cheap*? The thing is, you can't. You can only pick two of these constraints, or try to balance between them all by making compromises. You can choose to let go of some quality metrics in order to save some time and/or money. But to do this, you must, first, define these metrics, and second, prioritize them. In this context, there are no miraculous formulas applicable to all cases when it comes to defining data quality. Each company, depending on their field, size, and available data and technology, has to define the criteria they want to focus on when calculating the data quality score of their data sets. When defining this score, they need to look for the relevant quality metrics they want to focus on. For instance, as defined by the DAMA UK Working Group [33], these metrics can be defined as (but not limited to) *Completeness*, *Uniqueness*, *Timeliness*, *Validity*, *Accuracy* and *Consistency*.

DQS: A New Data Quality Evaluation Metric. We define a formula to compute the data quality of a dataset, called DQS (*Data Quality Score*) [34–36]. We propose the following process:

1. Choose the target quality metrics $M_1, M_2, \ldots, M_n$
2. For each quality metric M_j:
   a) Associate a weight w_j to the metric, depending on its importance for the company's quality strategy.
   b) For every dimension D_i ($i \in [1..m]$) of the dataset which value is relevant to the chosen metric, define quality rules to be respected in order to conform to the quality metric M_j.
   c) Compute the score of every record regarding the defined rule.
   d) The average of the scores of all records of a dimension D_i is considered to be the score S_i of the dimension.
   e) The quality score S_j of the chosen metric is the average of all the dimension scores of the dataset:

$$S_j = \frac{\sum_{i=1}^{m} S_i}{m} \tag{1}$$

3. The quality score *DQS* of the data set is the weighted average of all metrics' quality scores:

$$DQS = \frac{\sum_{j=1}^{n} w_j S_j}{\sum_{j=1}^{n} w_j} \tag{2}$$

This metric can be computed in the Data Identification step (Sect. 4.3).

Applying the DQS to an eCommerce Example. Let's take the example of a simplified e-commerce decisional platform. We want to compute the quality score of a dataset imported from an operational database and composed of three tables, as presented in Fig. 4: *Customer*, *Product* and *Purchase*. We set our target metrics as being the *Freshness* (M_1) and *Completeness* (M_2) of the dataset. We give more importance in our quality strategy to *Completeness* of the information in the dataset rather than to the *Freshness* of the purchases, this is why we associate to the metric *Freshness* the weight $w_1 = 1$, while the weight $w_2 = 2$ is associated to *Completeness*.

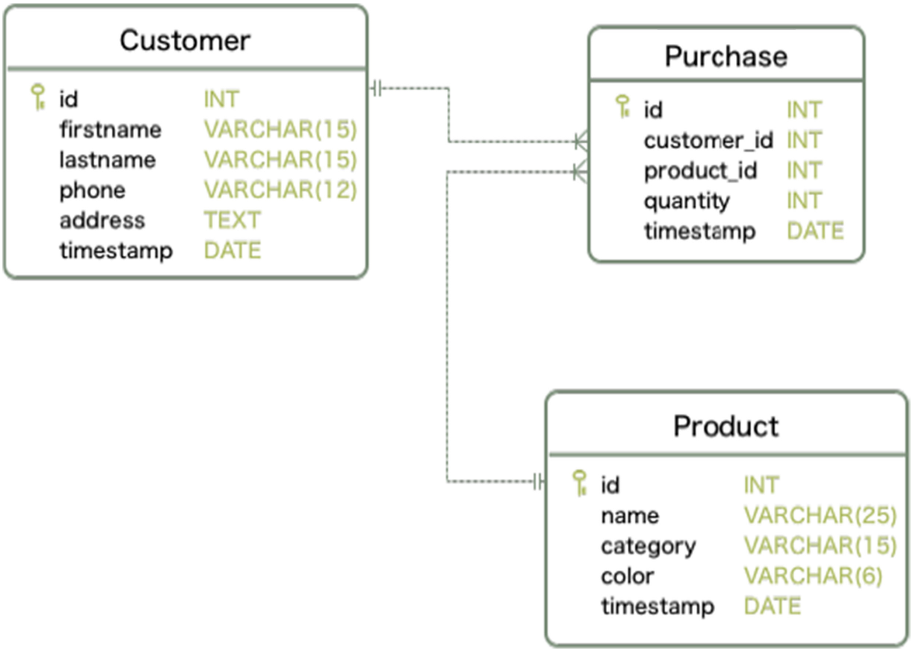

Fig. 4. eCommerce example dataset

For the first metric *Freshness*, we define the following rule: "*a purchase is considered fresh if it was stored less than one year ago*". In this case, the dimension we are concerned with is the *Purchase*, and we focus in particular on its attribute *timestamp*.

We choose to apply a score *1* to any record stored less than a year ago, and *0* otherwise. If we suppose that in our dataset of 50.000 purchases, 10.000 were done the preceding year, then the quality score S_1 for the *Freshness* metric is:

$$S_1 = \frac{10.000}{50.000} = 0,2 \tag{3}$$

As for the metric *completeness*, the quality rule defined in this case is: "*for every table in the dataset, a record is considered complete if all its information are filled*". As stated in the rule, all the dimensions (*Product, Customer* and *Purchase*) with all their attributes have to be considered. We will apply in this case a score of *1* to every record where all the attributes are defined, *0.5* for records where only one attribute is missing, and *0* if more than one attribute is missing. In this case, let's suppose that among the 50.000 purchases, 20.000 customers and 120 products that we have on our dataset, 34.000 have the score *1*, 30.100 have the score *0.5* and 6.020 have the score *0*. In this case, the quality score S_2 of our dataset for the metric *Completeness* is:

$$S_2 = \frac{34.000 * 1 + 30.100 * 0,5 + 6.020 * 0}{70.120} = 0,7 \tag{4}$$

Finally, the overall quality score of our dataset is estimated to:

$$DQS = \frac{w_1 * S_1 + w_2 * S_2}{w_1 + w_2} = \frac{1 * 0,2 + 2 * 0,7}{3} = 0,33 \tag{5}$$

Quality Improvement Scenarios. In order to improve the quality of this dataset, many cleansing and enrichment processes can be defined. We can for example add, as a metric, the number of clicks on the company's website per product and customer, which will considerably increase the amount of recent data (thus improving the *Freshness* metric), while adding a new attribute that can give more insight about the interest of the customers in the displayed products. A new extraction process must then be defined and the data quality score must be computed again.

As part of the data governance strategy, the computation of the quality score should be done regularly, stored periodically as important metadata, and adequate measures should be taken by the company in order to improve the quality of the internal data. If the considered data sources are external and have a poor-quality score, the data managers should either look for better data sources, or proceed to a cleansing and homogenization phase that can help to improve their quality. A quality score threshold should be defined depending on the needs and expectations of the decision makers.

The cost of this operation can vary depending on the type of dataset and the business context. But, compared to a classical decisional approach, we notice a low additional effort mainly in the first step when the data is identified. This is a part of each data owner's responsibility, who has to keep it updated and to define potential cleansing and enrichment processes in order to improve its data quality score (DQS). Regarding costs and overheads, DECIDE can present a small overhead for the setup and definition of the DQS, but the extraction and needed calculation cost will be the same compared to any other classical approach or methodology. On the other hand, and thanks to this part of DECIDE, the data governance strategy will make it possible to share the data, which will facilitate its usage for more efficient analysis. All team members will have the same data vision and can improve its quality in a collaborative manner. This is very important, especially when we have to deal with external and/or unstructured data sources for better analytics precision. Despite the light overhead of the quality metric's set up, its benefits are clear and will show their results during the DECIDE steps and in the business process.

5.3 Data Team Workforce

Just like any other asset, data needs protection and safeguards, with varying degrees of control and well-defined roles and privileges. It is thus mandatory to define all the actors that are involved in the creation, update, deletion, monitoring, transformation and tractability of the data in its environment: we call it *the data team workforce*.

The team is typically composed of the following actors [37]:

- *Data governance council*: in charge of the strategic view of the data governance plan,
- *Data governance board*: in charge of the tactical view,
- *Data owner(s)*: in charge of defining the requirements and ensuring the data quality and accessibility,
- *Data manager*: usually in the IT department, in charge of implementing the requirements of the data owner and of the management of the infrastructure,
- *Data steward*: in charge of defining the rules and planning the access and delivery of data,
- *Data user*: who has access to the reliable data.

The distribution of the team workforce throughout the different phases and steps of the methodology is represented in Fig. 5. A scaling system from 0 to 10 is defined, representing the implication of every role in each step of the process. 0 means *absent*, while 10 says *essential*.

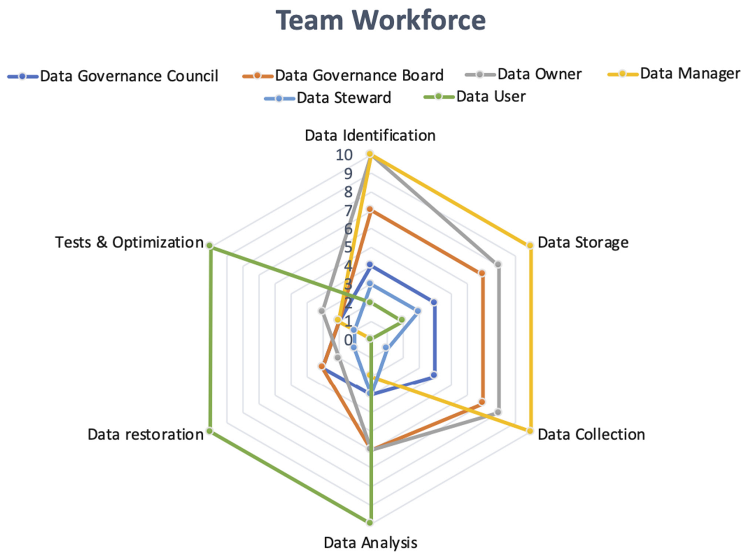

Fig. 5. Team workforce

6 Conclusion

We present in this article DECIDE, a Big Data Decisional Methodology that puts the focus on Data Governance. We recon that DECIDE can be more expensive than classical decisional methodologies when we have to deal with small projects with only one team or less than six members in the same team. However, the medium- and long-term benefits fully cover this overhead thanks to the gain in terms of time to market and refactoring costs in the case of rapidly evolving needs. The governance rules and fundamentals of DECIDE help to keep a solid link between the different teams and their members. In addition, governance shows also its benefits with data mining and exploration use cases, as all lineage and metadata can be accessible for all team members from any place. Indeed, DECIDE enforces a collaborative approach and helps data reuse and analysis.

It is still a real hardship to convince business stakeholders and project managers to apply a big data methodology in their projects, as their main focus is to increase their Return on Investment (RoI). In fact, such methodologies may seem expensive, but their repercussions can be huge in terms of new opportunities, integration of new business needs, taking full advantage of the available data and the team, and reduction of the overall time to market. It is then necessary to find a way to quantify these profits. We intend in a future work to define a method to estimate the overhead and the actual return on investment.

References

1. Koscielniak, H., Puto, A.: Big data in decision making processes of enterprises. Procedia Comput. Sci. **65**, 1052–1058 (2015). International Conference on Communications, management, and Information technology (ICCMIT'2015)
2. Al-Badi, A., Tarhini, A., Khan, A.I.: Exploring big data governance frameworks. Procedia Comput. Sci. **141**, 271–277 (2018). The 9th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (EUSPN-2018)/The 8th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare (ICTH-2018)/Affiliated Workshops
3. Tallon, P.P.: Corporate governance of big data: Perspectives on value, risk, and cost. Computer **46**(6), 32–38 (2013)
4. Vidgen, R., Shaw, S., Grant, D.B.: Management challenges in creating value from business analytics. Eur. J. Oper. Res. **261**(2), 626–639 (2017)
5. Saltz, J.S.: The need for new processes, methodologies and tools to support big data teams and improve big data project effectiveness. In: Proceedings - 2015 IEEE International Conference on Big Data, IEEE Big Data 2015, pp. 2066–2071 (2015)
6. Abai, N.H.Z., Yahaya, J.H., Deraman, A.: User requirement analysis in data warehouse design: a review. Procedia Technol. **11**(ICEEI), 801–806 (2013)
7. Romero, O., Abello, A.: A survey of multidimensional modeling methodologies. Int. J. Data Warehouse. Min. **5**(2), 1–23 (2009)
8. Selma, K., Ily`es, B., Ladjel, B., Eric, S., Stephane, J., Michael, B.: Ontology-based structured web data warehouses for sustainable interoperability: requirement modeling, design methodology and tool. Comput. Ind. **63**(8), 799–812 (2012)

9. Kimball, R., Reeves, L., Ross, M., Thornthwaite, W.: The Data Warehouse Lifecycle Toolkit: Expert Methods for Designing, Developing, and Deploying Data Warehouses. Wiley, Chichester (1998)
10. Husemann, B., Lechtenborger, J., Vossen, G.: Conceptual data warehouse design. In: 2nd International Workshop on Design and Management of Data Warehouses, Stockholm, Sweden. CEUR-WS.org (2000)
11. Winter, R., Strauch, B.: A method for demand-driven information requirements analysis in data warehousing projects. In: Proceedings of the 36th Annual Hawaii International Conference on System Sciences, 9 pp. IEEE (2003)
12. Moody, D., Kortink, M.: From enterprise models to dimensional models: a methodology for data warehouse and data mart design. In: Proceedings of 2nd International Workshop on Design and Management of Data Warehouses, Stockholm, Sweden. CEUR-WS.org (2000)
13. Jensen, M.R., Holmgren, T., Pedersen, T.B.: Discovering Multidimensional Structure in Relational Data, pp. 138–148. Springer, Berlin (2004)
14. Song, I.Y., Khare, R., Dai, B.: SAMSTAR: a semi-automated lexical method for generating star schemas from an entity-relationship diagram. In: Proceedings of the ACM Tenth International Workshop on Data Warehousing and OLAP, pp. 9–16 (2007)
15. Golfarelli, M., Rizzi, S.: A methodological framework for data warehouse design. In: Proceedings of the 1st ACM International Workshop on Data Warehousing and OLAP - DOLAP ’98, pp. 3–9. ACM Press, New York (1998)
16. Boehnlein, M., Ulbrichvom Ende, A.: Deriving initial data warehouse structures from the conceptual data models of the underlying operational information systems. In: Proceedings of the 2nd ACM International Workshop on Data Warehousing and OLAP - DOLAP ’99, pp. 15–21 (1999)
17. Bonifati, A., Cattaneo, F., Ceri, S., Fuggetta, A., Paraboschi, S.: Designing data marts for data warehouses. ACM Trans. Softw. Eng. Methodol. **10**(4), 452–483 (2001)
18. Cabibbo, L., Torlone, R.: A logical approach to multidimensional databases. In: Schek, H.-J., Alonso, G., Saltor, F., Ramos, I. (eds.) EDBT 1998. LNCS, vol. 1377, pp. 183–197. Springer, Heidelberg (1998). https://doi.org/10.1007/BFb0100985
19. Romero, O., Abello, A.: Multidimensional design by examples. In: 8th International Conference on Data Warehousing and Knowledge Discovery, pp. 85–94 (2006)
20. D2D CRC: Agile Methodologies for Big Data Projects (2018). https://www.d2dcrc.com.au/article-content/agile-methodologies-for-big-data-projects
21. Fayyad, U., Stolorz, P.: Data mining and KDD: promise and challenges. Future Gener. Comput. Syst. **13**(2–3), 99–115 (1997)
22. Shearer, C., et al.: The CRISP-DM model: the new blueprint for data mining. J. Data Warehous. **5**(4), 13–22 (2000)
23. Provost, F., Fawcett, T.: Data science and its relationship to big data and data driven decision making. Big Data **1**(1), 51–59 2013
24. Angée, S., Lozano, S., Montoya-Munera, E., Ospina Arango, J., Tabares, M.: In: 13th International Conference on Towards an Improved ASUM-DM Process Methodology for Cross-Disciplinary Multiorganization Big Data & Analytics Projects, KMO 2018, pp. 613–624, Zilina, Slovakia, 6–10 August 2018, Proceedings (2018)
25. Khatri, V., Brown, C.V.: Designing data governance. Commun. ACM **53**(1), 148–152 (2010)
26. Talend Team: What is Data Governance (and Do I Need It)?
27. Beedle, M. et al.: Manifesto for Agile Software Development (2001)
28. Sims, C., Johnson, H.L.: Scrum: A Breathtakingly Brief and Agile Introduction. Dymax, Torrington (2012)
29. United States Army: Human Intelligence Collector Operations. Volume 2–22.3 (2006)

30. TDWI: TDWI BI BENCHMARK REPORT: Organizational and Performance Metrics for Business Intelligence Teams. Technical report (2010)
31. Rouse, M.: Data Governance (DG) (2017)
32. Fowler, M.: Data Lake (2015)
33. Sattler, K.U.: Data Quality Dimensions. Encyclopedia of Database Systems, pp. 1–5 (2016)
34. ShellBlack: Formulas to Create Data Quality Lead Score
35. Evans, P.: Scaling and assessment of data quality. Acta Crystallogr. Sect. D Biol. Crystallogr. **62**(1), 72–82 (2006)
36. White, C.H., Gonzalez, L.R.: The Data Quality Equation—A Pragmatic Approach to Data Integrity (2015)
37. BARC: Data Governance: Definition, Challenges & Best Practices. Technical report, Business Application Research Center (2018)