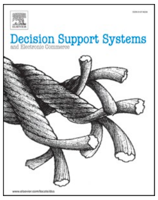
Decision Support Systems 141 (2021) 113450

Contents lists available at ScienceDirect

Decision Support Systems

journal homepage: www.elsevier.com/locate/dss



DMN4DQ: When data quality meets DMN

Álvaro Valencia-Parra [a,*], Luisa Parody [b], Ángel Jesús Varela-Vaca [a], Ismael Caballero [c], María Teresa Gómez-López [a]

[a] *Dpto. Lenguajes y Sistemas Informáticos, Universidad de Sevilla, Sevilla, Spain*
[b] *Dpto. Métodos Cuantitativos, Universidad Loyola Andalucía, Sevilla, Spain*
[c] *Information Systems and Technologies Institute, University of Castilla-La Mancha, Ciudad Real, Spain*

ARTICLE INFO

Keywords:
Data usability
Data quality
Decision model and notation
Data quality rule
Data quality assessment
Data quality measurement

ABSTRACT

To succeed in their business processes, organizations need data that not only attains suitable levels of quality for the task at hand, but that can also be considered as usable for the business. However, many researchers ground the potential usability of the data on its quality. Organizations would benefit from receiving recommendations on the usability of the data before its use. We propose that the recommendation on the usability of the data be supported by a decision process, which includes a context-dependent data-quality assessment based on business rules. Ideally, this recommendation would be generated automatically. Decision Model and Notation (DMN) enables the assessment of data quality based on the evaluation of business rules, and also, provides stakeholders (e.g., data stewards) with sound support for the automation of the whole process of generation of a recommendation regarding usability based on data quality.

The main contribution of the proposal involves designing and enabling both DMN-driven mechanisms and a guiding methodology (DMN4DQ) to support the automatic generation of a decision-based recommendation on the potential usability of a data record in terms of its level of data quality. Furthermore, the validation of the proposal is performed through the application of a real dataset.

1. Introduction

The witnessed changes that Digital Transformation (e.g., Industry 4.0) is introducing in different business processes across various domains have positioned data at the core of operations and strategies [33]. To a certain extent, it can be stated that the role previously played by steam engines in Industry 1.0 is now played by the new and powerful AI-based machines [12]. However, and as happened in those times, the success of these new AI-machines, and therefore, of business processes, largely relies on the quality of the raw material employed, in this case, data. Consequently, the management of the quality of data has become essential in this digital era [20,25].

Given the need for data with adequate levels of quality in such domains, we propose that if organizations could automatically incorporate ways to decide on whether to use or discard records, then business processes would greatly benefit from preventing results that would otherwise produce low levels of data quality. This decision regarding the potential usability of the data could be made after the generation of a recommendation based on the assessment of the quality of the data records.

Since it is generally accepted that the assessment of data quality is context-dependent [3,15], and since we propose that the usability of the data largely depends on the quality of the data, it can therefore be stated that the usability of data is also largely dependent on the context of the use of the data [14,37,44]. This implies modelling the context in which the data is to be used and when a data record is potentially usable.

In order to convert this idea into action, we conducted an investigation to tackle two challenges: (i) how to describe whether a data record is usable for its intended use in a given context; and (ii) how to automate the process of producing a recommendation on the usability of the data for this context.

To deal with the first challenge, we studied how others had already faced the problem of modelling the context, the data, and the rules that describe when a data record is of sufficient quality by identifying and describing various types of business rules for data quality and how the recommendation of the usability of the data could be determined. In

[*] Corresponding author.
E-mail addresses: avalencia@us.es (Á. Valencia-Parra), mlparody@uloyola.es (L. Parody), ajvarela@us.es (Á.J. Varela-Vaca), Ismael.Caballero@uclm.es (I. Caballero), maytegomez@us.es (M.T. Gómez-López).

https://doi.org/10.1016/j.dss.2020.113450
Received 10 June 2020; Received in revised form 15 November 2020; Accepted 15 November 2020
Available online 18 November 2020

order to ensure rigour and repeatability on the process, we decided to incorporate all the elements identified and the necessary steps into a methodology.

The second challenge to address is the necessity to support the generation of the recommendation on the usability of the data, based on its levels of quality in an automatic and technological-agnostic way [35]. This is even more challenging in scenarios where high efficiency is required in terms of computational cost, such as in Internet of Things (IoT) or in the context of CyberPhysical Systems (CPS). As part of this second challenge, and to set our proposal in motion, we suggest the use of a solution that facilitates the description and validation of the business rules employed in the assessment of the level of usability based on data quality, thereby promoting the application of repeatable decisions that can be semantically interoperable with the various technologies through which data quality assessment could be applied. We found that in order to tackle these challenges, it was recommendable to use a decision language that facilitates the description of the business rules so that they could be verified automatically. In this respect, OMG's Decision Model and Notation (DMN) [30] and the FEEL expression language for modelling conditions could prove themselves to be perfect allies in achieving these two challenges. For this reason, DMN is the main pillar of the structure of proposal.

Therefore, the main contribution of the proposal involves *designing and enabling DMN-driven mechanisms to support the automatic generation of a business-based recommendation on the potential usability of a data record in terms of its level of data quality*. To this end, our proposal includes the following actions:

1. Development of the foundations of the proposal through a set of integrated and hierarchical business rules that address the concepts of data quality measurement and data quality assessment for the generation of a data-usability recommendation (Section 3.1).
2. Identification and tailoring of the necessary elements provided by the standard DMN to support our proposal (Section 3.2).
3. Definition of a methodology, called DMN4DQ, to enable data-related users (e.g., stakeholders and data stewards) to drive the process of instantiating the corresponding elements when it comes to producing recommendations for a given dataset in a given context. This includes the definition and implementation of a software architecture supported by commercial implementations of reference (e.g., Camunda DMN) to automate the process of generating the recommendation (Section 4).
4. Validation of the proposal in a case study with real data (Section 5).

The remainder of the paper is organised as follows: Section 6 shows related work; Section 7 analyses threats to the validity of the approach; and finally, Section 8 presents concluding remarks and lessons learned.

2. Foundations

In order to combine DMN and data quality, and before detailing our proposal, it is necessary to revisit certain concepts regarding data quality management and DMN to enable a better understanding of how DMN can be used to describe whether a data record is usable in terms of its level of quality in a given context.

2.1. Data quality management: measurement and assessment

Throughout the literature, the two most widely used definitions of *"data quality"* are based on the notions of *"meeting requirements"* (i.e., a measure of the numbers of defects) given by Crosby, and *"fitness for use"* coined by Juran [38]. From our understanding, these two definitions involve a major difference: while the first definition enables somebody to *measure "how well data is built"* (for instance, by counting the number of times that the data fails to meet stated requirements), the second lets somebody *assess "how usable the data is"* in a given context by comparing the number of defects found (the "measures") with a threshold value representing the appetite for risk of the organisation regarding the reliability of the data in an specific context [8].

Even though the terms "measurement" and "assessment" can sometimes be considered as synonyms, we highlight this difference because it is important for our proposal: the "assessment" requires the "measurement", in the same way that the "generation of a recommendation on the usability" requires the "assessment". Our proposal goes a step beyond, since before determining whether a record is potentially usable, it is necessary to make the most important decision: *While taking into account the impact that using data with inadequate levels of quality can have on the success of the business processes, should the assessed data record be used or discarded in the context of the task at hand?*. If the use of the data is potentially risky for the business, then data stewards may decide: to enhance the data (e.g., data cleansing); to use the unaltered data, thereby assuming a risk; or alternatively, to discard the data record. The main aim of our proposal is therefore to provide business-based recommendations to data stewards to facilitate decision-making on whether to use or discard the data as part of their business activities. Therefore, there is a patent need to manage data quality. The concept of data quality dimension (also called data quality characteristic) lies at the core of data quality management. A data quality dimension can be understood as *a criterion employed to evaluate the quality of data* [28,37,44]. These dimensions or characteristics represent the data quality requirements stated or expected by the various stakeholders involved in the execution of the business processes [45]. A set of data quality dimensions is called a data quality model. Several researchers and practitioners in a variety of contexts have proposed their own data quality models [32,36]. Due to its importance at different stages of the data quality management discipline, we would like to highlight two generic models from among all the existing models: (1) the model proposed by Wang et al. [43] (see Table 1); and (2) the model proposed in ISO 25012 [21] (see Table 2).

The first model has been the most widely used in recent years since it is the most authoritative reference in the field. Moreover, it guides the identification of the specific data quality requirements that are important for a given context. In order to validate the compliance of these requirements, business rules are typically employed in data quality contexts [7,10,34].

The second model, ISO 25012, should not necessarily be understood as an alternative to the proposal of Wang et al. In fact, in conjunction with ISO 25024 [22], it complements their model by providing important indications for the definition of measurements and measurement methods for the data quality dimensions or characteristics. ISO 25012 introduces fifteen data quality characteristics, which are classified into the following three groups: (i) Inherent. The definition for these data quality characteristics is introduced in Table 2; (ii) System dependent. There are various characteristics whose measurement or assessment largely depends on the implementation of the systems in which data is stored, retrieved or processed; (iii) Inherent and system dependent. This group contains some of the previous data quality characteristics whose measurement and/or assessment can be subject to a two-fold interpretation based on the ideas introduced in the two previous groups.

Please, note that Wang et al. (and many other investigations based on this seminal work) use the term "dimension", whereas in the standards,

Table 1
Data quality dimensions by Wang et al. [43].

Data quality category	Data quality dimension
Intrinsic	Accuracy, Objectivity, Believability, Reputation
Accessibility	Access, Security
Contextual	Relevancy, Value-Added, Timeliness, Completeness, Amount of data
Representational	Interpretability, Ease of understanding, Concise representation, and Consistent representation

Table 2
Definition of the inherent data quality characteristics from ISO 25012 [21].

Data quality characteristic	Definition
Accuracy	The degree to which data has attributes that correctly represent the true value of the intended attribute of a concept or event in a specific context of use.
Completeness	The degree to which subject data associated with an entity has values for all expected attributes and related entity instances in a specific context of use.
Consistency	The degree to which data has attributes that are free from contradiction and are coherent with other data in a specific context of use.
Credibility	The degree to which data has attributes that are regarded as true and believable by users in a specific context of use.
Currentness	The degree to which data has attributes that are of the right age in a specific context of use.

the term "characteristic" is preferred. Even though it would be possible to justify the difference, for the sake of the simplicity, let us consider these two terms as synonymous in this manuscript.

As previously stated, measurement and assessment of data quality characteristics is a complex process, which largely depends on the context of use of the data. The context of the data includes: the organisational environment and the description of the business processes in which data is used; the technological architecture supporting the use of the data; and the skills and knowledge of both data stewards and data quality analysts in charge of managing or using the data [8,24]. In order to deal with this complexity, several researchers and practitioners have proposed different methodologies, which include a variety of metrics and/or measurement methods [4,29,34,43].

Typically, measurement methods that are to be applied first require relational datasets to be profiled for a better understanding of the nature of the data. These profiling processes are often performed in batch activities that include all data records in the dataset and by means of data profiling tools [13,34]. Nevertheless, in order to measure data quality, most authors have developed their own rule-based data quality measurement systems. The foundations of these rule-based data quality measurement systems have been formalised by Bronselaer et al. in [7]. The creation of rule-based data quality measurement systems involves that stakeholders (i.e. data stewards) should conform to the semantics of the data and their context. However, if stakeholders remain unaware of the semantics of the data and the way in which these semantics have been implemented throughout the various data models (conceptual, logical, and physical), then the application of this kind of tool will require an extra effort towards diagnosing the root causes of the low levels of data quality, since the methodology behind these tools fails to contemplate the context in which the data quality process is applied. The instantiating of these frameworks involves the creation of various elements, such as the set of business rules to which data must adhere, and the possible results of the measurement that the capacity function should produce. An example of the instantiating of this conceptualization for the measurement of the level of quality in information systems research is presented in [39], wherein Timmerman and Bronselaer, after reviewing the foundations of rule-based data quality measurement, present a rule-based framework for the measurement and assessment of the quality of data in Information Systems research. As we will explain in Section 3, our proposal, goes beyond the measurement of data quality, involving more stages. Consequently, we have to describe and relate several sets of business rules.

2.2. Decision model and notation

Decision Model and Notation (DMN) is the modelling language and notation standard defined by OMG to describe decision rules [30]. Thus, DMN is a standard approach that facilitates the modelling of repeatable decisions. The decisions can be customised according to the necessities of each organisation or moment, thereby ensuring that the decision models are interchangeable. DMN facilitates the declarative description and formalisation of the decisions with the form "if-then" [19]. Furthermore, since DMN is supported by a set of engines, such as Camunda - DMN Engine[1] and Drools - DMN Engine,[2] we found DMN to be the most suitable concept to come to the data quality assessment to real applicability of it.

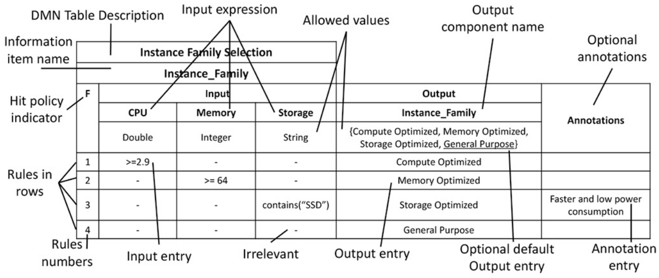
DMN provides a mechanism to define a decision logic model that is understandable by non-expert users (i.e., business data stewards in charge of describing the data quality requirements). In addition, DMN enables the separation of the decision logic from the control-flow logic, thereby centralising the conditional expressions that guide the decisions.

The DMN standard provides two customisable components: the Decision requirement diagram, which enables the definition to be made of the decisions to be taken, of their interrelationships, and of their requirements for decision logic; and the Decision logic, which allows the representation of the required decisions with sufficient details to enable validation and/or automation.

Decision logic is described by means of a *decision table* (see Fig. 1), which includes a set of inputs, decision rules, and output values. In a horizontal representation of the rules (an equivalent vertical representation is also possible), the input and outputs are defined in columns and the rules as rows. Each IF-THEN condition is represented in a row, as a conjunction of basic expressions written in FEEL (Friendly Enough Expression Language) [30]. The output returns the values of the row that is satisfiable according to the input. The example considers three features given as input data (i.e., *CPU*, *Memory*, and *Storage*), which returns a decision as output for the variable *Instance_Family*.

Each condition that appears in the DMN table (such as $>=2.9$) relies on FEEL,[3] an expression language that enables the writing of the conditions for the rules in the DMN tables. FEEL supports several data types as input and output values (e.g., String, Integer, Decimal, Date, Boolean), and implements a set of built-in functions to write more complex conditions on the input values. In addition, this expression language also supports null values and conditions that are always true ('$-$'). Users can modify the behaviour of the built-in functions as well as creating their own functions to better adjust the rules to the nature of their data. This versatility ensures that any of the types of business rules defined in Section 3.1 can be modelled with this technology, and formalised as described in Section 3.2.

The *information item name* is the name of the variable for which the decision table provides the decision logic. The *hit policy indicator* determines how to handle the multiple matches of the rules described in DMN [30]. This indicator takes any of 5 values: *Unique (U):* only one rule can be triggered, and it is not possible that more than one can satisfy a



Instance Family Selection					
Instance_Family					
F	Input			Output	Annotations
	CPU	Memory	Storage	Instance_Family	
	Double	Integer	String	{Compute Optimized, Memory Optimized, Storage Optimized, General Purpose}	
1	>=2.9	-	-	Compute Optimized	
2	-	>= 64	-	Memory Optimized	
3	-	-	contains("SSD")	Storage Optimized	Faster and low power consumption
4	-	-	-	General Purpose	

Fig. 1. Decision table for selecting the *Instance Family* depending on the *CPU*, *Memory*, and *Storage* of the server.

[1] https://camunda.com/products/dmn-engine/
[2] https://www.drools.org/learn/dmn.html
[3] https://docs.camunda.org/manual/7.4/reference/dmn11/feel/

rule for an input tuple; *Any (A):* Multiple rules can be triggered, but they must agree in the output; *Priority (P):* Multiple rules can be triggered, and the output corresponds to the rule that has the highest priority; *First (F):* Multiple rules can be triggered, and the output corresponds to the order of the rows in the decision table; and *Collect (C):* Multiple rules can be triggered, and the output is an aggregation.

In our case, *F* means that although multiple rules can match, only the first hit by rule order is returned. Finally, the possible results of this decision table are "*Compute optimised*", "*Memory optimised*", "*Storage optimised*", and "*General purpose*". Table 3 depicts the results of applying the DMN table presented in Fig. 1.

Decision tables can be combined in a hierarchical way, in that the output of one table can be the input of another further up in the hierarchy. For instance, as shown in Fig. 2, the output of "Instance Family Selection" will be used as input of the decision table "Supplier Selection".

Unlike other alternatives for the verification of business rules, DMN allows users to define their rules in a hierarchical way, thereby maintaining the coherence with the logical structure previously explained. The way in which rules are graphically modelled, its versatility, and the possibility of automating the evaluation of the rules, make DMN the ideal candidate for technically supporting our methodology.

3. Rationale of DMN4DQ

Given the goal of our investigation, in this section, the rationale of our proposal is presented. Firstly, we describe the conceptualization of how to generate the recommendation on the usability of a data record based on its level of quality. Subsequently, an explanation is given on how DMN has been employed to represent and evaluate the required sets of business rules to automate the generation of a recommendation for the use of a data record.

3.1. Business rules for the determination of the usability of the data records

The generation of a recommendation of a business-driven decision on using or discarding a data record can be based on several criteria, but, to our understanding, data quality is the most important criterion because decisions are no better than the data on which they are based [44]. Consequently, to generate this recommendation, it is necessary to assess the level of data quality within a context of use. As stated in Section 2.1, this assessment requires a previous stage of measurement of data quality. Both the assessment and measurement require different data quality dimensions that represent stakeholders' data quality requirements for each task at hand.

It is known that business rules define certain business constraints of an organisation, such as who can execute an action, the order of the activities, and the acceptable thresholds for specific KPIs. In this respect, it is possible to define business rules to address data quality concerns for the proper execution of the processes of an organisation. Hence, business rules for data quality gather the knowledge acquired by an organisation to reflect when a data record can provide value for specific business goals. There exists a consensus that business rules are an effective way to control data quality [2], and the term "Data Quality Rule" has been used in the context of data quality management [27,34].

Our proposal introduces the notion of Business Rules for Data Decisions (BR.DD), that must be defined in order to ascertain the usability of a data record. The types of BR.DD considered in our proposal are the following:

Table 3
Results of applying the DMN table to a set of data records.

ID	CPU	Memory	Storage	Instance Family (DMN Output)
1	2.4	128	500 GB SSD	Memory Optimised
2	3.2	32	1000 GB	Compute Optimised
3	2.7	32	1000 GB	General Purpose
4	3.0	64	250 GB SSD	Compute Optimised

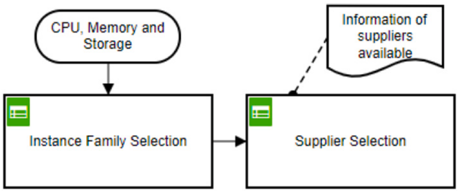


Fig. 2. Decision requirement diagram example.

- The "Business Rules for Data Values" (BR.DV) are those aimed towards evaluating the extent to which a data requirement is met. An example of BR.DV is provided in the following: Given that the length of a String must be longer than 6, if the length of an input String is from 3 to 6 then it returns 'A', else if it is from 6 to 9 then it returns 'B', and 'C' is returned otherwise. Semantically, 'A' is intended to represent the lowest level of fulfilment, and 'C' represents a suitable level of fulfilment, with 'B' representing an average term. We consider that this is not a proper data quality measurement since data quality dimensions have yet to be involved. However, the output of this evaluation will be the input of the specific data quality measurement.
- The "Business Rules for Data Quality Measurement" (BR.DQM) are those rules employed to compute the measurement of the level of quality of each data quality dimension according to the BR.DV. For example, a BR.DQM for the *accuracy* dimension could be stated as follows: A record can be considered as *Dramatically Non-Accurate* if the output of BR.DV.01 is 'A', and *Accurate* if the output of BR.DV.01 is 'B' or 'C'.
- The "Business Rules for Data Quality Assessment" (BR.DQA) are those rules that describe the assessment of the data quality in accordance with a set of BR.DQM by combining the results of the measurement of several data quality dimensions, as indicated by the business. An example of BR.DQA is A record can be considered as: *Usable but assuming High Risk* if it is *Accurate* or *Correct*; *Usable and assuming Low Risk*, if it is *Accurate* and *Correct*; and *non-usable* otherwise.
- The "Business Rules for Data Usability Decision" (BR.DUD) are those rules employed to generate the recommendation about using or discarding the data record for the intended use based on the assessment of its level of data quality. At this point, the organisational risk-appetite of the organisation should be considered with regard to the use of this specific data record. For instance, A record will be *used* if it is *Usable and assuming Low Risk*.

3.2. Tailoring DMN elements to make the decision operative regarding data usability

Regardless of the type of business rules previously described, the formalisation of all of them is the same. The relations between them lies in the semantics derived from our conception on the hierarchy as established in the previous sections, and consequently, we will build and relate the decision table in a hierarchical structure.

3.2.1. Formalisation of data quality rules based on DMN

Based on DMN, the business rules applied in the generation of a recommendation on the potential use of a data record based on data quality concerns are formalised below. The definition of the rules include: (i) a set of input parameters; (ii) a list of if-then conditions; and (iii) the output values for each condition.

Let an instance of Business Rules for Data Decision (BR.DD) be a tuple ⟨Inputs, Rules, Outputs⟩ where:

- Inputs: This is a tuple of attributes a_i of type A_i, $\langle a_1{:}A_1, \ldots, a_n{:}A_n \rangle$, where the types permitted are String, Boolean, Integer, Real, and Date. It is represented by means of the Input data of the DMN tables.
- Output: This is a tuple of attributes b_i of type B_i, $\langle b_1{:}B_1, \ldots, b_m{:}B_m \rangle$, where the types permitted are: String, Boolean, and Integer (in a limited and finite domain). It is represented by means of the output data of the DMN tables.
- Rules: This is an ordered list of if-then rules $\langle r_1, \ldots, r_k \rangle$ where r_{i-1} has greater priority than r_i, which corresponds with the Hit Policy indicator (F) of the DMN table. Each r_i corresponds with a tuple of a DMN table, where each r_i has the form $\langle \{Q_1, \ldots, Q_n\}, \{o_1, \ldots, o_n\} \rangle$, where Q_j represents the conditions applied to the attribute a_i expressed in FEEL (if), and o_j represents the resulting *then* expression. Next, the formalisation presented in [9] is detailed:

$$Q ::= \text{“}-\text{”} \mid Term \mid \text{“}not(\text{“ } Term \text{ ”})\text{”} \mid Comparison \mid Interval \mid Q_1,\ Q_2$$

$$Comparison ::= COpTerm$$

$$COp ::= \text{“}={\text{”}} \mid \text{“}<\text{”} \mid \text{“}>\text{”} \mid \text{“}\leq\text{”} \mid \text{“}\geq\text{”} \mid \text{“}\in\text{”}$$

$$Interval ::= (\text{“(”} \mid \text{“[”})\ Term_1\ \text{“..”}\ Term_2\ (\text{“)”} \mid \text{“]”})$$

$$Term ::= v \mid f(Term_1, \ldots, Term_m)$$

For the grammar of FEEL certain remarks are needed: (i) v is a value of the domain and f is a function (e.g., $+$, $-$, round, ceiling, duration, day, etc.); (ii) “– “represents *any value*; (iii) Comparison and Interval are only applicable to numeric types; (iv) “Q_1, Q_2 “represents “$Q_1 \vee Q_2$“; and (v) if an attribute a_i fails to exist, the only condition that it could meet is “$-$“. For a further description of FEEL, please consult [30].

3.2.2. DMN hierarchical structure

As stated in Section 3.1, we have identified different types of BR.DD involved in the process of generating a recommendation on the usability of a data record. As stated, each BR.DD can be described by means of a decision table in DMN, and BR.DDs can be combined according to their semantics to generate a final decision regarding the data record usability. Therefore, we propose a description of all the BR.DDs as DMN tables and a combination thereof in a hierarchical way as shown in Fig. 3. The hierarchy enables: (1) BR.DV (at the top) is evaluated for every data record provided as Input. This data record can be of any of the types defined in the formalisation; (2) for each data quality dimension, a BR.DQM uses the retrieved Outputs of the required BR.DVs as Input (which can be Boolean, String or a bounded range of Integers) in a similar way; (3) a BR.DQA uses the output of different DMN tables related to the measurement of a dimension (BR.DQM) as Input; and (4) BR.DUD (at the bottom of the hierarchy) takes the Outputs of BR.DQA as Input to which it applies its if-then rules.

In addition, the output of the business rules for data decisions must return an output from an established and ordered scale, whereby the best and the worst outputs are indicated, in order to guarantee the monotonicity of the business rule [39].

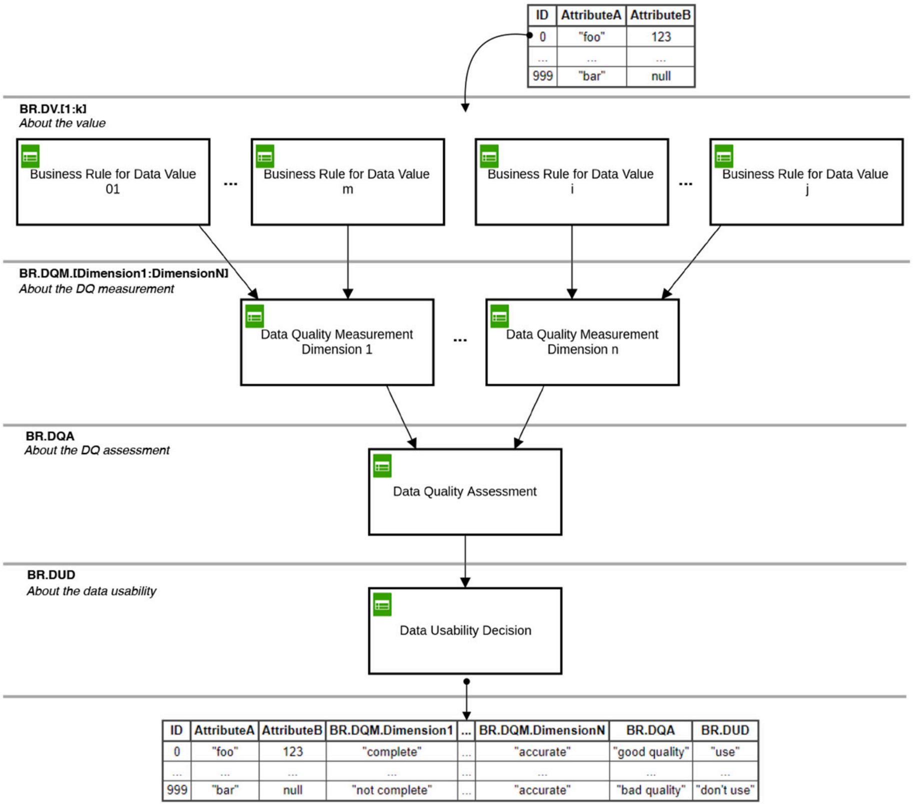
4. DMN4DQ: a methodology to develop a system to generate recommendations on the usability of a data record

In order to systematically instantiate all the DMN elements identified in Section 3.1, we now introduce a methodology called DMN4DQ. DMN4DQ will guide data stewards and stakeholders towards achieving the goal of implementing a system that can be integrated along with the Information Systems supporting the business process. The methodology consists of the following phases: (i) Phase 1. Define Business Rules for Data Decisions and the underlying hierarchy (see Subsection 4.1); (ii) Phase 2. Instantiate the DMN tables of the DMN4DQ hierarchy (see Subsection 4.2); (iii) Phase 3. Deploy, test, and integrate the DMN4DQ hierarchy into the systems needing a recommendation on the potential use of a data record (see Subsection 4.3). Fig. 4 summarises these phases. We provide a detailed description of each phase in the next sections.

4.1. Phase 1. Define business rules for data decisions with the aim of generating a recommendation on data usability

The Definition of the Business Rules for Data Decisions includes the following steps aligned with the types of business rules defined in the previous section:

- Step 1.1. Define Data Context: Describe the context in which the data is used.
- Step 1.2. Describe the Dataset: Describe the dataset, its attributes, and the technological stack that supports the management and use of the data.
- Step 1.3. Define Business Rules for Data Values (BR.DV): Identify the business rules to enable the validation of the data requirements on the data to generate a value representing the extent to which the data requirement is met. All these requirements should be desirably implemented during the design of the data repository [11,28].
- Step 1.4. Select the Data Quality Dimensions that best represent the usability of the data: Identify the combination of relevant data quality dimensions that best represent business requirements for data in the specific context of the use of data, such as completeness, consistency, or any of those dimensions identified by Wang et al. [43] or ISO 25012 [21] as introduced in Section 2.1. In addition, it is necessary to identify the possible output values that can be assigned to the measurement of every data quality dimension. Although stakeholders can define any domain of values for the results of these activities, for the sake of simplicity, we propose employing Likert scales [23]. For example, the data quality dimension of consistency could admit three possible values as a result: “Sufficiently Consistent”, “Insufficiently Consistent”, and “Dramatically Non-consistent”. Additionally, in order to ensure the monotonicity of the rule [39], it is necessary to denote which value represents the highest level of quality and which represents the lowest level of quality. For example, “Sufficiently Consistent” and “Dramatically Non-consistent”, respectively.
- Step 1.5. Define Business Rules for Data Quality Measurement (BR.DQM): Identify, describe, and validate the business rules aimed to measure the chosen data quality dimensions in Step 1.4. This step needs to be broken down into two further steps: (1) to associate specific BR.DV to every data quality dimension considered; and (2) to produce the “Data Quality Measurement Business Rules” (BR.DQM) that consider the data quality requirements stated by the business data stewards for the data in a given context. Depending on the granularity, a BR.DQM can cover one or more attributes and one or more BR.DVs [34].
- Step 1.6. Define Business Rules for Data Quality Assessment (BR.DQA): Identify, describe, and validate the business rules aimed to assess the level of data quality. This step includes the following actions: (i) Identify the relative importance (i.e., weight) of the data quality dimension in the assessment of the data quality of every data record in the context of use; (ii) identify the possible states of the usability of the data (output of BR.DQA). The states that can be enumerated include: “Fully Usable”, “Usable but cleansing recommended”, “Usable with a high risk”, “Not usable”; (iii) produce the “Data Quality Assessment Business Rules” (BR.DQA) that cover the combination or aggregation of the data quality dimensions involved, and by considering their relative importance.
- Step 1.7. Define Business Rules for the generation of a recommendation on the potential usability of Data: Identify and describe the business rules aimed to generate a recommendation on the use of the data (BR.DUD) in the given context of the data. To generate a recommendation on “Using” or “Discarding” the data, it is crucial to take into account the organisational appetite-risk related to data


BR.DV.[1:k]
About the value
Business Rule for Data Value 01
...
Business Rule for Data Value m
Business Rule for Data Value i
...
Business Rule for Data Value j
BR.DQM.[Dimension1:DimensionN]
About the DQ measurement
Data Quality Measurement Dimension 1
...
Data Quality Measurement Dimension n
BR.DQA
About the DQ assessment
Data Quality Assessment
BR.DUD
About the data usability
Data Usability Decision

ID	AttributeA	AttributeB
0	"foo"	123
...	...	...
999	"bar"	null

ID	AttributeA	AttributeB	BR.DQM.Dimension1	...	BR.DQM.DimensionN	BR.DQA	BR.DUD
0	"foo"	123	"complete"	...	"accurate"	"good quality"	"use"
...	...	...	...	...	...	...	...
999	"bar"	null	"not complete"	...	"accurate"	"bad quality"	"don't use"

Fig. 3. Decision table diagram of DMN4DQ.

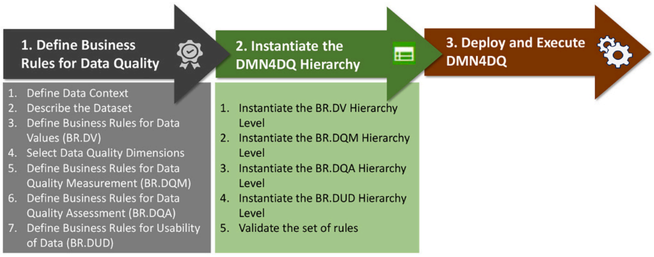

1. Define Business Rules for Data Quality
1. Define Data Context
2. Describe the Dataset
3. Define Business Rules for Data Values (BR.DV)
4. Select Data Quality Dimensions
5. Define Business Rules for Data Quality Measurement (BR.DQM)
6. Define Business Rules for Data Quality Assessment (BR.DQA)
7. Define Business Rules for Usability of Data (BR.DUD)
2. Instantiate the DMN4DQ Hierarchy
1. Instantiate the BR.DV Hierarchy Level
2. Instantiate the BR.DQM Hierarchy Level
3. Instantiate the BR.DQA Hierarchy Level
4. Instantiate the BR.DUD Hierarchy Level
5. Validate the set of rules
3. Deploy and Execute DMN4DQ

Fig. 4. The DN4DQ methodology.

quality when it comes to making a decision: in this respect, business data stewards should analyse the impact of the decision, and find a balance between discarding data or using data with a low level of quality.

4.2. *Phase 2. Instantiate the DMN4DQ integration and hierarchy*

As explained in Section 3.2, each set of BR.DD is represented by a DMN table that must be designed, implemented, and conveniently validated. Since there are four hierarchy levels of business rules (see Fig. 3), it is necessary to carry out the following steps: (1) Instantiate the BR.DV hierarchy level; (2) Instantiate the BR.DQM hierarchy level; (3) Instantiate the BR.DQA hierarchy level; (4) Instantiate the BR.DUD hierarchy level; and (5) Validate the set of rules as stated by D. Calvanese et al. [9].

DMN enables the decision logic to be described in a decision table. In the context of data usability recommendations, the decision table describes the data quality rules introduced by business experts that can be either correct or incorrect. A relevant previous paper [9] provides formal semantics and an algorithm for the detection of overlapping and missing rules. Other solutions can be found in the literature [41,42] but are limited to the Boolean or Enumerate domains.

4.3. *Phase 3. Deploy and execute the instance of DMN4DQ decision requirement diagram*

The last phase of DMN4DQ includes the development, testing, and possible deployment as an external service in a given system using software that supports an implementation of reference, as is the Camunda modeler and engine.[4]

5. Validation of DMN4DQ in a case study

The main purpose of this case study is to demonstrate that DMN4DQ can be used in a real dataset. In this case, it represents a catalogue of servers built on data provided by third parties. To ensure that the data is potentially useful in selling instances of servers in private clouds, it is necessary to analyse the data quality requirements for a decision to be made. Any lack of completeness, accuracy, and consistency of the data in this context might cause distrust among users, such as the inclusion of products in a publicity catalogue that fail to correspond to real products. In the following subsections, we show the most interesting results of this case study. The full case study is available online.[5]

5.1. *Phase 1. Define business rules for recommendations on the usability of the data*

Once the impact of poor-quality data in the business has been studied, the business rules for data decisions can be defined, as explained in Section 4.1.

[4] Camunda Modeler: https://camunda.com/products/modeler/

[5] DMN4Spark. Case Study: http://www.idea.us.es/dmn4dq/

5.1.1. Step 1.1. Define data context

The data, which is to be employed to build the catalogue, is extracted mainly from Amazon Web Services.[6] The data is acquired in CSV format, and each record contains information on a server instance.

5.1.2. Step 1.2. Describe the dataset

At the time of running the case study, the dataset was composed of 1,048,571 records. An extract of the data dictionary describes the dataset is: *Location* is a String attribute identifying the geographical location of the machine. It is represented by the name of the country or region where it is located; *InstanceFamily* is a String that describes the category to which the machine belongs (consistent with the features of the machine); *ClockSpeed* is a String representing the speed of the CPU (a decimal value followed by the String "GHz"); *Memory* is a String that represents the size of the RAM memory (an Integer followed by the String "GiB"); *Storage* is a String that describes the type of storage of the machine; *OperatingSystem* is a String that specifies the operating system of the machine; and *PricePerUnit* is a String representing a numeric value indicating the price of an instance of the machine.

5.1.3. Steps 1.3 and 1.4. Define business rules for data values and identify data quality dimensions

For the sake of simplicity, all the BR.DVs and the data quality dimensions to which the BR.DVs could be assimilated are presented together as follows.

Completeness. The lack of relevant data poses a potential risk in the offered service. For this specific case, we considered that the measurement of the completeness involves several BR.DVs. For the sake of simplicity, the BR.DVs are described by specifying the field to which they apply: *Location* (*BR.DV.01*), *ClockSpeed* (*BR.DV.03*), *Memory* (*BR.DV.05*), *InstanceFamily* (*BR.DV.07*), *OperatingSystem* (*BR.DV.10*), and *PricePerUnit* (*BR.DV.12*). These BR.DVs return one of the following values on a scale in the interval [0,2]: (i) 0, if the value is *null*; (ii) 1, if it is an empty String (except for BR.DV.12, for which *PricePerUnit* should be 0); and (iii) 2, otherwise. Semantic: Having a *null* value is more risky than an empty field since it might lead to misinterpretations of the idea of completeness [18]. In this case, we employ the indices of the Likert scale so that values regarding the completeness can take advantage of the operations enabled in DMN, such as the possibility of employing comparison operators in the measurement phase (see Section 3.2.1).

Accuracy. Inaccurate data might cause negative effects in terms of credibility and technical aspects. For example, if the data syntax fails to follow a specific pattern, it might not be properly processed and might cause problems when being displayed or analysed. We have considered three groups of BR.DVs involved in the measurement of the accuracy in this particular case:

1. Those which bound the value that a String can take (*BR.DV.02*, *BR.DV.08* and *BR.DV.11*, whose input fields are: *Location*, *InstanceFamily*, and *OperatingSystem*, respectively). These BR.DVs return a value on a scale composed of three elements: (i) *Appropriate* in the case where the value is in a set of very acceptable values; (ii) *Sufficiently appropriate* in the case where the value is in a set of fairly acceptable values; and (iii) *Inappropriate* if the value is not present in any set. Semantic: Unexpected values might lead to failures in data analysis processes and to misleading information. For this reason, the list of accepted values is bounded.
2. Those indicating the format which the data must take (*BR.DV.04* and *BR.DV.06*, with these inputs: *ClockSpeed* and *Memory*, respectively). They return *true* if the value matches the expected pattern. Otherwise, they return *false*. Semantic: If these fields fail to match the pattern, certain processes will fail completely.
3. Those bounding a numeric range (*BR.DV.13*, with *PricePerUnit* as input). It returns a value on a scale of three elements: (i) *Realistic* if the value is in the range (0.0, 10,000.0); (ii) *Exaggerated* if it is in the range [10,000.0, 99,999.9]; and (iii) *Unrealistic* in any other case. Semantic: Certain price values might be too high. These cases could be acceptable, but should be carefully analysed.

Consistency. Inconsistent data entails not only a potential risk from the user's point of view, but also legal issues (e.g., advertising a server instance with false characteristics). *BR.DV.09* is the only business rule we considered as necessary to be involved in the measurement of this data quality dimension, with various fields as input: *Memory*, *ClockSpeed*, *Storage*, and *IntanceFamily*. It returns a value within a range of [0,3] with the following semantic: (i) 0 if it satisfies the condition that *Memory* is less than 64 GiB and *InstanceFamily* must not be *Memory Optimised*; (ii) 1 if it satisfies the condition that *ClockSpeed* is less than 2.9 GHz and *InstanceFamily* is not *Compute Optimised*; (iii) 2 if it satisfies the condition that *Storage* does not contain the substring *SSD* and *InstanceFamily* is not *Storage Optimised*; and (iv) 3 in any other case. Semantic: Inconsistencies are more serious when found in the information regarding *Memory*, *ClockSpeed*, and *Storage*, in that order.

The next step prior to defining the BR.DQM is to design the outputs of the measurement. For the sake of simplicity, in this example we will select different Likert scales with all the possible values that might result from the measurement of each data quality dimension. We remark that it is of paramount importance to carefully study the context in which data is to be employed so that these values have a proper semantic.

Regarding the **completeness** dimension, we established the following measurement based on a Likert scale and on the risks associated to missing data: (i) *Suitably Complete* if the information about the server is complete. The record might be used in advertisement campaigns; (ii) *Sufficiently Complete* in the case where there is a minimal subset of attributes which are complete, and hence, the record can be shown in the catalogue; and (iii) *Not Complete* if the record cannot be included in the catalogue due to the lack of important attributes for sale.

Regarding the **accuracy** dimension, the management team established the following measurement levels. As in the previous case, these have been defined according to the risks associated to inaccurate data, and are based on a numerical scale: (i) 100 if the information about this record is accurate, and hence, it could be employed for advertisement campaigns; (ii) 70 in the case where there is a minimal subset of attributes which are sufficiently accurate, and hence the record could be listed in the catalogue; and (iii) 50 if values and ranges are sufficiently accurate although certain formats remain inaccurate; and (iv) 0 if there is a lack of accurate technical data, which renders this record unsuitable for listing in the catalogue.

Finally, regarding the **consistency** dimension, the following levels are defined. Again, the measurement is based on the risks associated to inconsistent data, as well as on a Likert scale: (i) *Consistent* if attributes derived from technical features are consistent between them, and hence the record could be listed in the catalogue and employed for advertisement campaigns, and (ii) *Inconsistent* if derived attributes are not consistent with technical features, the tuple must not be listed in the catalogue.

In order to simplify the proposal, other dimensions have been omitted, although their inclusion would require little effort. For example, we could have included the following BR.DV related to the timeliness dimension: the timestamp must have been generated a maximum of 15 min before the moment at which data quality measurement is performed. Its corresponding BR.DQM would set the record as *Timely* if it fulfils that BR.DV, otherwise, it would be set as *Not timely*. This BR.DV might be implemented by creating a custom function named "*current_timestamp()*", which returns the current timestamp (i.e., the timestamp at which the rule is evaluated). It would then be verified that the difference between the current timestamp and the stored timestamp is less or equal to 15 min.

[6] Dataset employed in the case study: https://www.kaggle.com/akashsarda/aws-ec2-pricing-data/version/1

5.1.4. Step 1.5. Define business rules for data quality measurement (BR.DQM)

The BR.DQM for the measurement of the **completeness** dimension (BR.DQM.Completeness) includes the following conditions: 1. A record is considered as *Suitably Complete* when the output of BR.DV.01, BR.DV.03, BR.DV.05, BR.DV.07, BR.DV.10, and BR.DV.12 are greater than or equal to 2. 2. A record is considered as *Sufficiently Complete* when BR.DV.03 and BR.DV.05 are greater than equal to 2, and BR.DV.12 is greater than or equal to 1. 3. A record is considered as *Not Complete* in any other case.

Regarding the measurement of the **accuracy** dimension (BR.DQM.Accuracy), the following conditions are defined: 1. The accuracy will have a value of 100 (*accurate*) when BR.DV.04 and BR.DV.06 are met; BR.DV.02, BR.DV.08 and BR.DV.11 are *Appropriate*, and BR.DV.13 is *Realistic*. 2. The accuracy will have a value of 70 *(sufficiently accurate)* when it meets BR.DV.04 and BR.DV.06; BR.DV.02, BR.DV.08 and BR.DV.11 are either *Appropriate* or *Sufficiently Appropriate*; and BR.DV.13 is either *Realistic* or *Exaggerated*. These records could be listed in the catalogue. 3. The accuracy will have a value of 50 when the conditions of BR.DQM.05 are met except for BR.DV.04 and BR.DV.06, which might be *false* and for BR.DV.02, which might be *Inappropriate*. 4. The accuracy will take a value of 0 otherwise.

Finally, the conditions for the business rule of the measurement of **consistency** dimension (BR.DQM.Consistency) are: 1. A record can be considered as *Consistent* when BR.DV.09 is greater than or equal to 3. 2. A record can be considered as *Inconsistent* when fails to meet BR.DV.09.

5.1.5. Step 1.6. Define business rules for data quality assessment (BR.DQA)

The output levels for the assessment have been defined as follows: (i) *Suitable or Sound Quality*. This level represents those records that are *Suitably Complete*, *Very Accurate*, and *Suitably Consistent*. The recommendations associated to these records can be to *"include them in the catalogue", "use them in advertisement campaigns"*; (ii) *Sufficient Quality*. This level represents those records that have a sufficient level of quality for them to be listed in the catalogue, although they cannot be used for advertisement campaigns to prevent risk. These records must be *Consistent* and can neither be *Not Complete* nor *Inaccurate*; and lastly, (iii) *Non-usable*. A record is *Non-usable* when it is *Not Complete*, *Inaccurate*, or *Inconsistent*. Non-usable records must not be listed in the catalogue.

The Business Rule for Data Quality Assessment is then modelled with the following conditions: 1. A record has *Suitable Quality* when its BR.DQM.Completeness is *Suitably Complete*, its BR.DQM.Consistency is *Consistent*, and its BR.DQM.Accuracy takes a value of 100. 2. A record has *Sufficient Quality* when it is *Consistent*, is not *Not Complete*, and its BR.DQM.Accuracy is greater than or equal to 70. 3. A record has *Bad Quality* when it is *Consistent*, it is not *Not Complete*, and its BR.DQM.Accuracy is greater than or equal to 50. 4. A record is *Non-usable* when it is *Not Complete*, *Inconsistent*, or its BR.DQM.Accuracy is less than 50.

5.1.6. Step 1.7. Define business rules for usability of data (BR.DUD)

This step consists of deciding the level of quality that each record from the dataset must fulfil in order to be employed in the catalogue. The decision to be made concerns whether or not to include each single record in the catalogue of server instances. According to the way in which the BR.DQA has been modelled, a record might be listed in the catalogue if its level of quality is *suitable* or *sufficient*. Thus, the conditions of the business rule for user decision-making are: 1. A record will be listed in the catalogue only when the BR.DQA is either of *suitable* or *sufficient quality*. 2. A record will not be listed in the catalogue when its BR.DQA is classified as *Non-usable*.

5.2. Phase 2. Design, implement, and validate the DMN tables

At this point, every business rule for data decisions has been modelled. Each level of the hierarchy presented in Fig. 3 must be implemented and integrated. Fig. 5 depicts the DMN hierarchy of this example. The steps followed in this example are described in the following sections.

5.2.1. Step 2.1. Instantiate the BR.DV hierarchy level

One table for each BR.DD explained in Subsection 5.1.3 must be created. Inputs are expected to be the attributes from the dataset. The output is a numeric value in the interval [0, 2] that indicates whether or not the attribute(s) fulfil(s) the conditions. The order of priority of the conditions is established when the business rule is defined, since the order of priority is in the order in which the conditions are defined. In the DMN table, each condition appears in a row in the same order in which they are defined, and the Hit Policy indicator is established as *F* (see Section 2.2). In this case study, there are 13 BR.DV, and 13 DMN tables must be created. Due to limitations on the length of the paper, only two of these BR.DVs are shown, although the reader can find the full list in the web presented by the authors.[7]

The first is BR.DV.04, depicted in Fig. 6. It is composed of three rows. The first row checks whether the input String matches the required pattern. If so, it returns the value 2. If it is an empty String, it returns the value 1, and 0 otherwise.

The second table is BR.DV.09, shown in Fig. 6. This has four inputs and four rows (if-then conditions). The three top rows are intended to verify whether the attributes *Memory*, *ClockSpeed* and *Storage* are inconsistent with the *InstanceFamily* attribute. These conditions were described in Section 5.1.3. Conditions in rows 1 and 2 verify whether the attributes *Memory* and *ClockSpeed* are less than 64 and 2.9, respectively. This is implemented by means of FEEL built-in functions. The condition is modelled by splitting the String in terms of its white spaces, then taking the expected numeric part and comparing the resulting numbers.

5.2.2. Step 2.2. Instantiate the BR.DQM hierarchy level

The DMN tables are built as described in Section 3.2. In this case, the inputs are BR.DQM are the output of the BR.DV. The measurement of each dimension is defined in a DMN table where each condition yields one value per dimension. Fig. 7 shows the DMN tables for the three defined dimensions.

5.2.3. Step 2.3. Instantiate the BR.DQA hierarchy level

Fig. 7 depicts how BR.DQA is modelled. In this case, the table inputs are the output of the business rules for data quality measurement. Each row specifies the conditions which must be accomplished for each assessment value.

5.2.4. Step 2.4. Instantiate the BR.DUD hierarchy level

Fig. 7 depicts the modelling of the BR.DUD. The input is the result of the BR.DQA.

5.2.5. Step 2.5. Validate the set of rules

DMN tables may be validated [9]. We propose the use of two tools to validate the DMN tables: dmn-js,[8] which verifies a table by checking possible missing and overlapping rules; and dmn-check,[9] which checks duplicate rules, conflicting rules, shadowed rules, types of expressions, correct use of enumerations, and correctly connected requirement graphs.

5.3. Phase 3. Deploy, test, integrate, and execute the tables obtained by applying DMN4DQ

We developed a tool, called *dmn4spark*,[10] which takes a DMN file and

[7] DMN4DQ: http://www.idea.us.es/dmn4dq/

[8] dmn-js: http://dmn.cs.ut.ee.

[9] dmn-check: https://github.com/red6/dmn-check#validations.

[10] dmn4spark: https://github.com/IDEA-Research-Group/dmn4spark

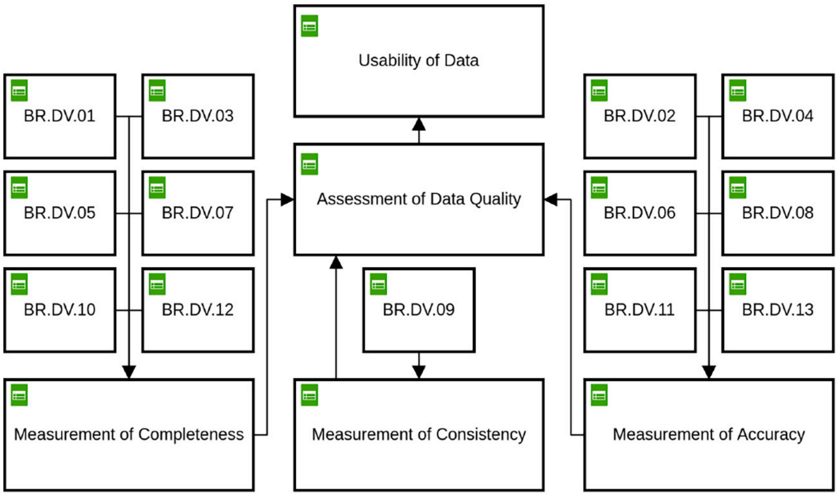
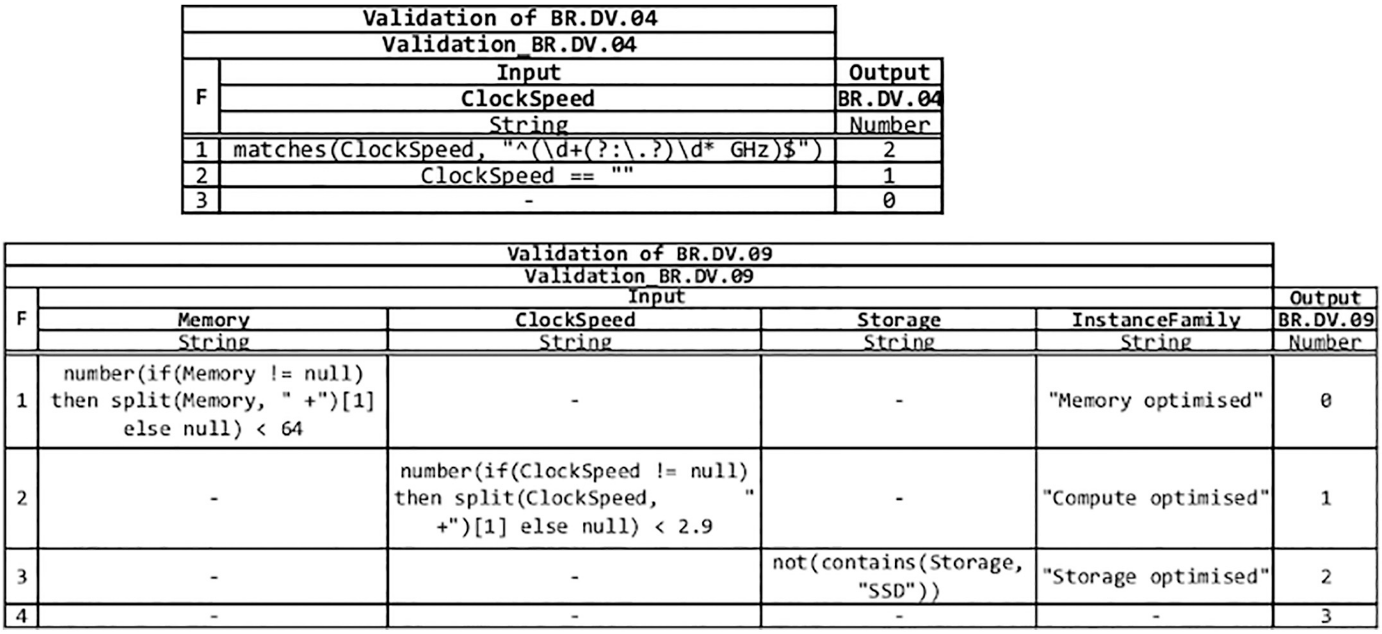


Fig. 5. DMN4DQ - Decision table diagram of the case study.



Validation of BR.DV.04

Validation_BR.DV.04

F	Input	Output
	ClockSpeed	BR.DV.04
	String	Number
1	matches(ClockSpeed, "^(\d+(?:\.?)\d* GHz)$")	2
2	ClockSpeed == ""	1
3	-	0

Validation of BR.DV.09

Validation_BR.DV.09

F	Input				Output
	Memory	ClockSpeed	Storage	InstanceFamily	BR.DV.09
	String	String	String	String	Number
1	number(if(Memory != null) then split(Memory, " +")[1] else null) < 64	-	-	"Memory optimised"	0
2	-	number(if(ClockSpeed != null) then split(ClockSpeed, " +")[1] else null) < 2.9	-	"Compute optimised"	1
3	-	-	not(contains(Storage, "SSD"))	"Storage optimised"	2
4	-	-	-	-	3

Fig. 6. DMN tables for BR.DV.04 and BR.DV.09.

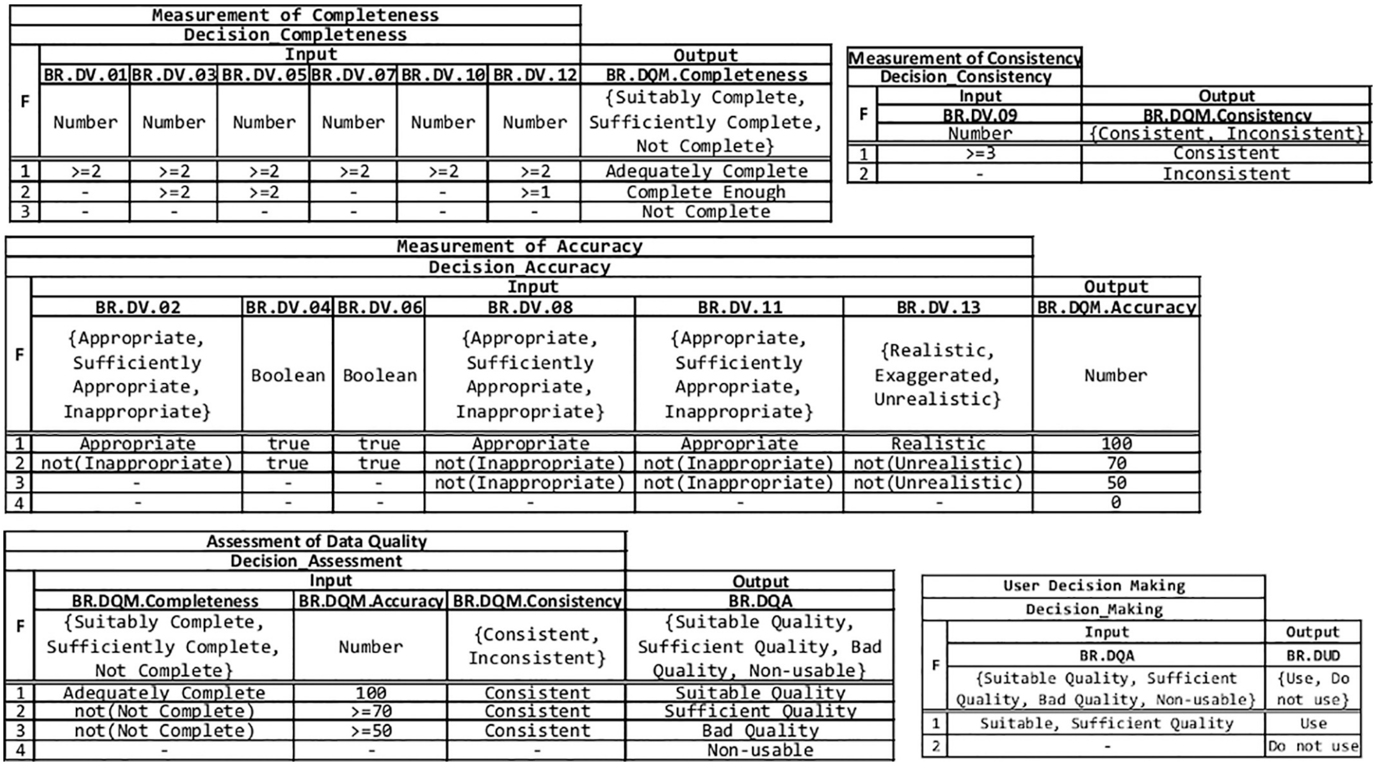
a dataset as inputs, and evaluates all the DMN tables for each record of the whole dataset. This tool is based on Apache Spark,[11] a distributed computing framework. In this way, users can obtain a recommendation for the usability of each data record of the dataset in a given context in Big Data scenarios. One of the main advantages of Apache Spark is the fact that it abstracts users from defining data models, since it is able to infer the schema of semi-structured datasets. In addition, this tool offers the possibility of using external plugins for the structuring of datasets by means of data transformation techniques [40]. Once the corresponding DMN file is defined, it must be uploaded to HDFS or a web server reachable by the cluster on which the application will be run. The steps to follow to use this tool are summarised in Fig. 8.

We employed this implementation to compute the results for the dataset of the case study in order to generate a recommendation on the potential usability of each of the 1,048,571 records. The results are depicted in Tables 4, 5, 6, 7, and 8. These show, for each DMN table, the number of tuples and the percentage thereof which fulfil each possible result of the business rules.

5.4. Conclusion about the execution of the case study

Regarding the results obtained, several conclusions can be drawn: (i) For almost half of the records the recommendation to discard them has been generated. This means that the potential risk of not having filtered out the data which fails to meet minimum standards of quality could have been much higher, since the existence of defects in the definition of half of the server instances would have strongly deteriorated the quality of the services offered, and consequently the reputation of the Company. If the organisation wants to increase the number of usable records, then the quality of the data must be improved; (ii) The main root cause of low-quality data is the lack of accuracy, given that around 37% of the records have an accuracy in the range of [0, 50]. These records might should therefore be analysed in order to find the root cause of the inaccuracy; and (iii) intermediate cases such as *sufficiently complete*, *accurate*, and *sufficient quality* are not very common.

[11] Apache Spark: http://spark.apache.org/



Measurement of Completeness

Decision_Completeness

F	Input						Output
	BR.DV.01	BR.DV.03	BR.DV.05	BR.DV.07	BR.DV.10	BR.DV.12	BR.DQM.Completeness
	Number	Number	Number	Number	Number	Number	{Suitably Complete, Sufficiently Complete, Not Complete}
1	>=2	>=2	>=2	>=2	>=2	>=2	Adequately Complete
2	-	>=2	>=2	-	-	>=1	Complete Enough
3	-	-	-	-	-	-	Not Complete

Measurement of Consistency

Decision_Consistency

F	Input	Output
	BR.DV.09	BR.DQM.Consistency
	Number	{Consistent, Inconsistent}
1	>=3	Consistent
2	-	Inconsistent

Measurement of Accuracy

Decision_Accuracy

F	Input						Output
	BR.DV.02	BR.DV.04	BR.DV.06	BR.DV.08	BR.DV.11	BR.DV.13	BR.DQM.Accuracy
	{Appropriate, Sufficiently Appropriate, Inappropriate}	Boolean	Boolean	{Appropriate, Sufficiently Appropriate, Inappropriate}	{Appropriate, Sufficiently Appropriate, Inappropriate}	{Realistic, Exaggerated, Unrealistic}	Number
1	Appropriate	true	true	Appropriate	Appropriate	Realistic	100
2	not(Inappropriate)	true	true	not(Inappropriate)	not(Inappropriate)	not(Unrealistic)	70
3	-	-	-	not(Inappropriate)	not(Inappropriate)	not(Unrealistic)	50
4	-	-	-	-	-	-	0

Assessment of Data Quality

Decision_Assessment

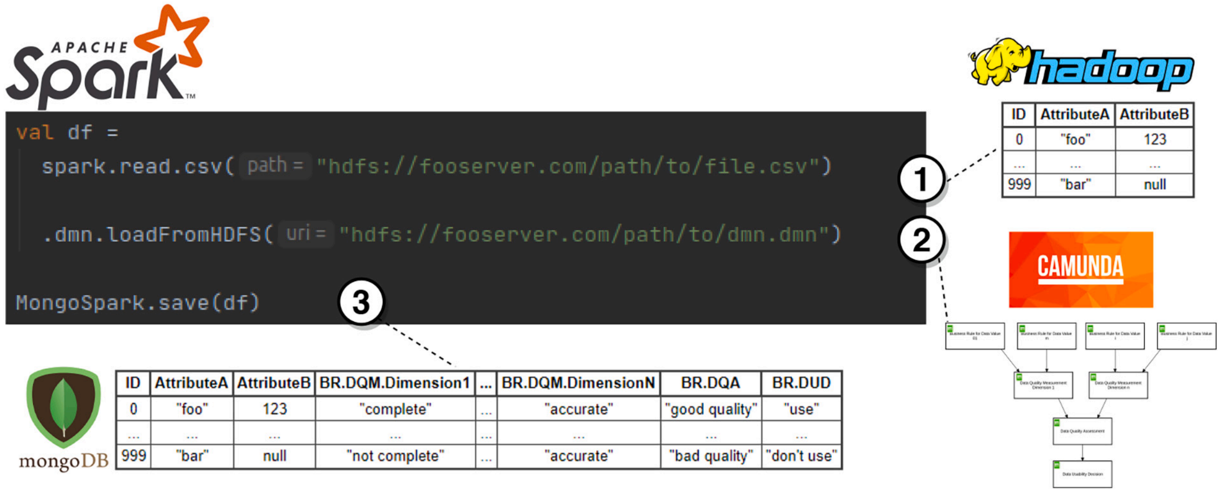
F	Input			Output
	BR.DQM.Completeness	BR.DQM.Accuracy	BR.DQM.Consistency	BR.DQA
	{Suitably Complete, Sufficiently Complete, Not Complete}	Number	{Consistent, Inconsistent}	{Suitable Quality, Sufficient Quality, Bad Quality, Non-usable}
1	Adequately Complete	100	Consistent	Suitable Quality
2	not(Not Complete)	>=70	Consistent	Sufficient Quality
3	not(Not Complete)	>=50	Consistent	Bad Quality
4	-	-	-	Non-usable

User Decision Making

Decision_Making

F	Input	Output
	BR.DQA	BR.DUD
	{Suitable Quality, Sufficient Quality, Bad Quality, Non-usable}	{Use, Do not use}
1	Suitable, Sufficient Quality	Use
2	-	Do not use

Fig. 7. DMN tables for the Completeness, Accuracy, and Consistency dimensions; the Assessment and the Data Usability Decision.



```
val df =
  spark.read.csv( path = "hdfs://fooserver.com/path/to/file.csv")

  .dmn.loadFromHDFS( uri = "hdfs://fooserver.com/path/to/dmn.dmn")

MongoSpark.save(df)
```

ID	AttributeA	AttributeB
0	"foo"	123
...	...	...
999	"bar"	null

ID	AttributeA	AttributeB	BR.DQM.Dimension1	...	BR.DQM.DimensionN	BR.DQA	BR.DUD
0	"foo"	123	"complete"	...	"accurate"	"good quality"	"use"
...	...	...	...	...	...	...	...
999	"bar"	null	"not complete"	...	"accurate"	"bad quality"	"don't use"

APACHE Spark
hadoop
1
2
3
CAMUNDA
Business Rule for Data Value 01
Business Rule for Data Value m
Business Rule for Data Value i
Business Rule for Data Value j
Data Quality Measurement Dimension 1
Data Quality Measurement Dimension n
Data Quality Assessment
Data Usability Decision
mongoDB

Fig. 8. Steps to follow for using our tool *dmn4spark.*. In this example, the dataset and the DMN file are stored in HDFS, and the results are dumped in MongoDB.

Table 4
Ratio of results for the measurement of completeness dimension.

BR.DQM.Completeness	#	%
Suitably Complete	839,990	80.11
Sufficiently Complete	888	0.08
Not Complete	207,692	19.81

Summarising, the generation of a recommendation on the usability of the data helps both to automate the data quality assessment and the detection of the reason why the data fails to satisfy the business rules defined. Therefore, an in-depth study into the quality of those records which have been considered non-usable should be carried out.

Table 5
Ratio of results for the measurement of accuracy dimension.

BR.DQM.Accuracy	#	%
100	629,106	60.00
70	25,022	2.39
50	392,714	37.45
0	1728	0.16

6. Related work

Organizations today are aware of the importance of ascertaining the levels of the quality of data. The necessity to generate recommendations on the use of the data records based on some business restrictions with

Table 6
Ratio of results for the measurement of the consistency dimension.

BR.DQM.Consistency	#	%
Consistent	909,565	86.74
Inconsistent	139,005	13.26

Table 7
Ratio of the results for data quality assessment.

BR.DQA	#	%
Suitable	520,271	49.62
Sufficient Quality	25,022	2,39
Bad Quality	185,862	17.73
Non-usable	317,415	30.27

Table 8
Ratio of generated recommendations.

BR.DUD	#	%
Use	545,296	52.00
Do not use	503,277	48.00

regard to the measured or assessed level of data quality (i.e., the appetite for risk involved in using data with inadequate levels of quality) has been studied previously [14,29,36,45]. However, DMN4DQ goes one step further in that it is a holistic solution where the processes of measurement, assessment (these two typically considered as synonyms), and generation of a recommendation of the use of data are integrated and adequately related by incorporating the business needs. Furthermore, we have tailored the OMG's international standard DMN to support the automation of the required actions to generate the recommendation on the potential use of data grounding our proposal on the concept of decision rules. To model the decision rules about data can be described, we ground our proposals on previous works aimed to formalise the data quality rules [27,34], expressed through some business rules [2] that data should meet. Other proposals, as [39], reflect that the discovering and definition of business rules - expressed by regular expressions, representing functional dependencies, by using control digits or employing association analysis - constitute the cornerstone of any data quality management initiative [1,13,17,34]. However, it is important to highlight that our work is not about discovering and defining business rules, but to combine them to generate automated business-based recommendation. In this sense, we encourage to read and to use the works describing traditional types of integrity constraints for data quality management, such as functional dependencies (FDs), and their extension conditional functional dependencies (CFDs) [5,16] or even the Fellegi-Holt method that automatically "corrects" data that fail some predefined requirements [6]. On the other hand, there exist generic approaches to define business rules but not used in the context of data quality as used in the paper. For example, SBVR [31] facilitates the definition of vocabularies and rules, but it is not decision-oriented. As said, to make operational the integration of the different parts, and to facilitate the modelling of the decision rules, instead of proposing a new one language, we propose the application of DMN, the OMG's standard, which includes the FEELs. FEELs increments the easiness and feasibility of the writing of the rules - and the agnostic-technological implementation. Some other authors have proposed their own frameworks to automatically measure the levels of data quality, just to name a few, let us bring the works done by Liu et al. who introduces in [26] a semantic-aware data quality assessment for image big data; or the work by [3] who propose a methodology to build a data quality adapter module selecting the best configuration for the data quality assessment in big data. However, to the best of our knowledge, DMN4DQ is the first solution that integrates every type of decisions needed to judge about the usability of a data record in the same framework, being our contribution the tailoring of such mechanisms to support a holistic solution.

7. Limitations of the proposal

In this section, we analyse the potential limitations of the proposals. Firstly, our approach is thought to be applied record by record (e.g. acting on a given tuple). Consequently, the definition of the rules is thought to describe business restrictions applying to every record, not to the whole dataset. However, the generalisation would not be difficult by including some logic aimed at computing global measurement on the whole datasets, which was initially out of the scope of our investigation. And secondly, the main issues of validation according to [46] are of internal, external and conclusion validity. 1. *Internal validity* refers to the trustworthiness of the result. In this respect, our work can be limited to three lines: (a) the assessment and measurement processes are database and data type agnostic, but are carried out over each independent tuple; (b) the type of business rules is limited to the support currently provided by the DMN specification and the FEEL language; and (c) the assessment and measurement of complex dimensions could require additional effort to construct auxiliary and extra functions in order to obviate complex attributes and rules. 2. *External validity* refers to the generalisation and the potential interest in the approach. To encourage the validation, usability, and generalisation of our approach: (a) we have provided a methodology; (b) we have provided a tool; and (c) a step-by-step case study is given and results of the tool are analysed. Therefore, researchers or practitioners who wish to use, replicate, or extend our approach are welcome to do so. 3. *Conclusion validity* refers to the rigorousness in the relationship established between the research questions raised and the findings obtained. We have striven to overcome this limitation by providing all the resources employed in the paper, namely, the tool and the data used in pursuit of repeatability and replicability of the findings established.

8. Conclusions

The usability of the data largely depends on the data quality, and on the context where the data is used. In this paper, we have presented a methodology that integrates different types of business rules for data decisions, holistically tackling the data-usability recommendation. DMN4DQ provides a hierarchy to integrate decision rules about data values, measurements of various dimensions, assessment through the aggregation of dimensions, and the data usability. Derived from the necessity to make decisions regarding the data usability, we rely on the OMG standard for decisions, DMN, as a suitable mechanism to model and automate the generation of the recommendations on the usability of the data in a specific context, since it coherently and comprehensively enables the description and evaluation of the business rules regarding data decisions. Moreover, DMN facilitates the transformation of the knowledge held by business experts into a formal model. Thanks to the use of DMN, the automation of the evaluation of the level of data quality ceases to be a solely theoretical contribution and becomes real technology that is applicable to real scenarios. Furthermore, we have developed a tool that supports the methodology validated with a real dataset.

Author statement

All the authors are responsible for the concept of the paper, the results presented and the writing. All the authors have approved the final content of the manuscript. No potential conflict of interest was reported by the authors.

Acknowledgements

This work has been partially funded by the Ministry of Science and

Technology of Spain via ECLIPSE (RTI2018-094283-B-C33 and RTI2018-094283-B-C31) projects; the Junta de Andalucíavia the COPERNICA and METAMORFOSIS projects; the European Fund (ERDF/FEDER); the Junta de Comunidades de Castilla-La Mancha via GEMA: Generation and Evaluation of Models for dAta quality (Ref.: SBPLY/17/180501/000293), and by the Universidad de Sevilla with *VI Plan Propio de Investigación y Transferencia* (VI PPIT-US).

References

[1] Z. Abedjan, L. Golab, F. Naumann, Profiling relational data: a survey, VLDB J. 24 (2015) 557–581.

[2] P. Alpar, S. Winkelsträter, Assessment of data quality in accounting data with association rules, Expert Syst. Appl. 41 (2014) 2259–2268.

[3] D. Ardagna, C. Cappiello, W. Samá, M. Vitali, Context-aware data quality assessment for big data, Futur. Gener. Comput. Syst. 89 (2018) 548–562, https://doi.org/10.1016/j.future.2018.07.014.

[4] C. Batini, C. Cappiello, C. Francalanci, A. Maurino, Methodologies for data quality assessment and improvement, ACM Comput. Surv. (CSUR) 41 (2009) 1–52.

[5] P. Bohannon, W. Fan, F. Geerts, X. Jia, A. Kementsietsidis, Conditional functional dependencies for data cleaning, in: 2007 IEEE 23rd International Conference on Data Engineering, 2007, pp. 746–755.

[6] A. Boskovitz, R. Goré, M. Hegland, A logical formalisation of the fellegi-holt method of data cleaning, in: M. Berthold, H.J. Lenz, E. Bradley, R. Kruse, C. Borgelt (Eds.), Advances in Intelligent Data Analysis V, Springer Berlin Heidelberg, Berlin, Heidelberg, 2003, pp. 554–565.

[7] A. Bronselaer, R. De Mol, G. De Tré, A measure-theoretic foundation for data quality, IEEE Trans. Fuzzy Syst. 26 (2017) 627–639.

[8] I. Caballero, E. Verbo, C. Calero, M. Piattini, A Data Quality Measurement Information Model Based on iso/iec 15939, ICIQ, Cambridge, MA, 2007, pp. 393–408.

[9] D. Calvanese, M. Dumas, Ü. Laurson, F.M. Maggi, M. Montali, I. Teinemaa, Semantics and analysis of dmn decision tables, in: M. La Rosa, P. Loos, O. Pastor (Eds.), Business Process Management, Springer International Publishing, 2016, pp. 217–233.

[10] F. Chiang, R.J. Miller, Discovering data quality rules, in: Proceedings of the VLDB Endowment 1, 2008, pp. 1166–1177.

[11] E.F. Codd, Extending the database relational model to capture more meaning, ACM Trans. Database Syst. 4 (1979) 397–434, https://doi.org/10.1145/320107.320109.

[12] T. Davenport, J. Harris, Competing on Analytics: Updated, with a New Introduction: The New Science of Winning, Harvard Business Press, 2017.

[13] L. Ehrlinger, E. Rusz, W. Wöß, A Survey of Data Quality Measurement and Monitoring Tools, 2019 arXiv preprint arXiv:1907.08138.

[14] A. Even, G. Shankaranarayanan, Utility-driven assessment of data quality, in: ACM SIGMIS Database: The DATABASE for Advances in Information Systems 38, 2007, pp. 75–93.

[15] A. Even, G. Shankaranarayanan, P.D. Berger, Evaluating a model for cost-effective data quality management in a real-world crm setting, Decis. Support. Syst. 50 (2010) 152–163.

[16] W. Fan, Data quality: Theory and practice, in: H. Gao, L. Lim, W. Wang, C. Li, L. Chen (Eds.), Web-Age Information Management. WAIM 2012. Lecture Notes in Computer Science vol. 7418, Springer, Berlin, Heidelberg, 2012, pp. 548–553.

[17] W. Fan, Data quality: from theory to practice, ACM SIGMOD Rec. 44 (2015) 7–18.

[18] W. Fan, J. Li, S. Ma, N. Tang, W. Yu, Towards certain fixes with editing rules and master data, VLDB J. 21 (2012) 213–238.

[19] K. Figl, J. Mendling, G. Tokdemir, J. Vanthienen, What we know and what we do not know about DMN, Enterpr. Model. Inform. Syst. Architect. 13 (2) (2018) 1–16.

[20] B. Glavic, Big data provenance: Challenges and implications for benchmarking, in: Specifying Big Data Benchmarks - First Workshop, WBDB 2012, San Jose, CA, USA, May 8-9, 2012, and Second Workshop, WBDB 2012, Pune, India, December 17-18, 2012, Revised Selected Papers, 2012, pp. 72–80, https://doi.org/10.1007/978-3-642-53974-9_7.

[21] ISO-25012, Iso/iec 25012: Software Engineering-Software Product Quality Requirements and Evaluation (square)-Data Quality Model, 2008.

[22] ISO-25024, Iso/iec 25024:2015 Systems and Software Engineering — Systems and Software Quality Requirements and Evaluation (square) — Measurement of Data Quality, 2015.

[23] A. Joshi, S. Kale, S. Chandel, D.K. Pal, Likert scale: explored and explained, Br. J. Appl. Sci. Technol. 7 (2015) 396.

[24] J. Ladley, Data Governance: How to Design, Deploy, and Sustain an Effective Data Governance Program, Academic Press, 2019.

[25] S. Lee, B. Ludäscher, B. Glavic, PUG: a framework and practical implementation for why and why-not provenance, VLDB J. 28 (2019) 47–71.

[26] Y. Liu, Y. Wang, K. Zhou, Y. Yang, Y. Liu, Semantic-aware data quality assessment for image big data, Futur. Gener. Comput. Syst. 102 (2020) 53–65.

[27] D. Loshin, 17 - data quality and business rules in practice, in: D. Loshin (Ed.), Enterprise Knowledge Management, Academic Press, San Diego. The Morgan Kaufmann Series in Data Management Systems, 2001, pp. 425–461, https://doi.org/10.1016/B978-012455840-3.50017-0.

[28] D. Loshin, The practitioner's Guide to Data Quality Improvement, Elsevier, 2010.

[29] J. Merino, I. Caballero, B. Rivas, M.A. Serrano, M. Piattini, A data quality in use model for big data, Future Generation Comp. Syst. 63 (2016) 123–130, https://doi.org/10.1016/j.future.2015.11.024.

[30] OMG, Decision Model and Notation (DMN), Version 1.2, URL, https://www.omg.org/spec/DMN, 2019.

[31] OMG, Semantics Of Business Vocabulary And Rules, Version 1.5, URL, https://www.omg.org/spec/SBVR/About-SBVR/, 2019.

[32] B. Otto, Y.W. Lee, I. Caballero, Information and data quality in business networking: a key concept for enterprises in its early stages of development, Electron. Mark. 21 (2011) 83.

[33] J.M. Pérez-Álvarez, A. Maté, M.T. Gómez-López, J. Trujillo, Tactical business-process-decision support based on kpis monitoring and validation, Comput. Ind. 102 (2018) 23–39, https://doi.org/10.1016/j.compind.2018.08.001.

[34] L. Sebastian-Coleman, Measuring Data Quality for Ongoing Improvement: A Data Quality Assessment Framework, Newnes, 2012.

[35] K. Sha, S. Zeadally, Data quality challenges in cyber-physical systems, J. Data Inform. Qual. (JDIQ) 6 (2015) 1–4.

[36] G. Shankaranarayanan, Y. Cai, Supporting data quality management in decision-making, Decis. Support. Syst. 42 (2006) 302–317, https://doi.org/10.1016/j.dss.2004.12.006.

[37] V.C. Storey, R.M. Dewan, M. Freimer, Data quality: setting organizational policies, Decis. Support. Syst. 54 (2012) 434–442, https://doi.org/10.1016/j.dss.2012.06.004.

[38] J.G. Suarez, in: Philip B. Crosby, W. Edwards Deming, Joseph M. Juran (Eds.), Three Experts on Quality Management, Total Quality Leadership Office, Arlington VA, 1992. Technical Report.

[39] Y. Timmerman, A. Bronselaer, Measuring data quality in information systems research, Decis. Support. Syst. 126 (2019) 113138, https://doi.org/10.1016/j.dss.2019.113138.

[40] Á. Valencia-Parra, Á.J. Varela-Vaca, M.T. Gómez-López, P. Ceravolo, Chamaleon: Framework to Improve Data Wrangling with Complex Data, in: 40th International Conference on Information Systems, ICIS 2019, Association for Information Systems, 2019. URL, https://aisel.aisnet.org/icis2019/data_science/data_science/16.

[41] J. Vanthienen, E. Dries, Illustration of a decision table tool for specifying and implementing knowledge based systems, Int. J. Artif. Intell.Tools 3 (1994) 267–288.

[42] J. Vanthienen, C. Mues, A. Aerts, An illustration of verification and validation in the modelling phase of KBS development, Data Knowl. Eng. 27 (1998) 337–352, https://doi.org/10.1016/S0169-023X(98)80003-7.

[43] R.Y. Wang, A product perspective on total data quality management, Commun. ACM 41 (1998) 58–65.

[44] R.Y. Wang, M.P. Reddy, H.B. Kon, Toward quality data: an attribute-based approach, Decis. Support. Syst. 13 (1995) 349–372, https://doi.org/10.1016/0167-9236(93)E0050-N (information technologies and systems).

[45] S. Watts, G. Shankaranarayanan, A. Even, Data quality assessment in context: a cognitive perspective, Decis. Support. Syst. 48 (2009) 202–211, https://doi.org/10.1016/j.dss.2009.07.012.

[46] C. Wohlin, P. Runeson, M. Höst, M.C. Ohlsson, B. Regnell, Experimentation in software engineering, Springer. (2012), https://doi.org/10.1007/978-3-642-29044-2.

Álvaro Valencia-Parra (PhD. Student) Universidad de Sevilla, Dpto. Lenguajes y sistemas informáticos – Spain. Álvaro Valencia-Parra obtained his B.S degree in Software Engineering at the University of Seville in 2017. In 2019, he graduated with honors from the University of Seville with a M.Sc. degree in Computer Engineering. Currently, he is a PhD student. His research areas include the improvement of different activities in the Big Data Pipeline, such as data transformation, data quality, and data analysis. The scenarios he is facing up are mainly focused on the process mining paradigm. Hence, his goal is to improve the way in which final users deal with data preparation and specific scenarios in which configuring a Big Data Pipeline might be tricky. For this purpose, he is working in the improvement of these processes by designing Domain-Specific Languages, user interfaces, and semi-automatic approaches in order to assist users in these tasks. He has participated in prestigious congresses such as the BPM Industry Forum or the International Conference on Information Systems (ICIS).

Luisa Parody (Associate Professor), Universidad Loyola Andalucía, Sevilla, Spain. Luisa Parody studied computer engineering (including a minor in systems engineering) at the Universidad de Sevilla (Spain) and graduated with honors in July 2009. She then earned an M.Sc.degree in software engineering and technology(2010) and obtained her international PhD with honors at the Universidad Sevilla (2014). Since 2018, she has been working as an associate professor in Dto. Método Cuantitativos at the Universidad Loyola. She belongs to the IDEA Research Group and has participated in several private and public research projects and has published several high-impact papers.

Ángel Jesús Varela-Vaca, (Assistant Professor) Universidad de Sevilla, Dpto. Lenguajes y sistemas informáticos – Spain. Angel J. Varela-Vaca received the B.S. degree in Computer Engineering at the University of Seville (Spain) and graduated in July 2008. M.Sc. on Software Engineering and Technology (2009) and obtained his PhD with honors at the University of Seville (2013). Angel is currently working as Assistant Professor at Languages and System Informatics Department at the Universidad Sevilla and belongs to the Idea Research Group. Angel has and leaded various private projects and participated in several public research projects and he has published several impact papers. He was nominated as a member of Program Committees such as ISD 2016, BPM Workshops 2017, SIMPDA 2018. He has been reviewer for international journals such as Journal of Supercomputing, International Journal of Management Science and Engineering Management Multimedia Tools and Applications, Human-Centric Computational and Information Sciences, Mathematical Methods in Applied Sciences among others.

Ismael Caballero, (Associate Professor) Universidad de Castilla-La Mancha, Dpto. Tecnologías y Sistemas de Información. ISMAEL CABALLERO received the M.Sc. and Ph.D. degrees in computer science from the University of Castilla-La Mancha, Spain, in 2004, where he works as Associate Professor with the Information Systems and Technologies Department. In 2017, he cofounded the spinoff DQTeam where he serves as Training Head. He has been researching on data quality management and data governance, since 1998, coauthoring several books, conference and journal articles. He teaches data quality management and data governance foundations in many universities and companies. He holds several professional certifications: CISA certification by ISACA, since 2016, CDO-1 certification by UALR-MIT, since 2017. He is currently a member of ISO TC184/SC4 working as Project Editor for several parts of ISO 8000-60 series development project. He led the project of ISO 8000-62SMAEL CABALLERO received the M.Sc. and Ph.D. degrees in computer science from the University of Castilla-La Mancha, Spain, in 2004, where he works as Associate Professor with the Information Systems and Technologies Department. In 2017, he cofounded the spinoff DQTeam where he serves as Training Head. He has been researching on data quality management and data governance, since 1998, coauthoring several books, conference and journal articles. He teaches data quality management and data governance foundations in many universities and companies. He holds several professional certifications: CISA certification by ISACA, since 2016, CDO-1 certification by UALR-MIT, since 2017. He is currently a member of ISO TC184/SC4 working as Project Editor for several parts of ISO 8000-60 series development project. He led the project of ISO 8000-62.

María Teresa Gómez-López, (Associate Professor) Universidad de Sevilla, Dpto. Lenguajes y sistemas informáticos – Spain. María Teresa Gómez-López is a Lecturer at the University of Seville and the head of the IDEA Research Group. Her research areas include Business Processes and Data management, and how to improve the business process models including better decisions and enriching the model with Data Perspectives. She has led several private and public research projects and has published several impact papers, among others in Information and Software Technology, Information Systems, Information & Software Technology, or Data & Knowledge Engineering. She was nominated as a member of Program Committees, such as ER, BPM, EDOC, ISD or CAISE Doctoral Consortium. She has been reviewing for international journals, such as International Journal of Data and Information Quality, Journal of Systems and Software, Artificial Intelligence in Medicine; Business & Information Systems Engineering Journal or Information Science. She has been invited as a keynote speaker in various occasions, such as at the Workshop on Data & Artifact Centric BPM, International Workshop on Decision Mining & Modelling for Business Processes BPM, the X National Conference of BPM, the 28th IBIMA Conference.