PAIS-DQ: Extending Process-Aware Information Systems to support Data Quality in PAIS life-cycle

Luisa Parody*, María Teresa Gómez-López*, Isabel Bermejo†, Ismael Caballero†, Rafael M. Gasca*, Mario Piattini†
*Dto. Lenguajes y Sistemas Informáticos
Universidad de Sevilla, Seville, Spain
{lparody, maytegomez, gasca}@us.es
†Dto. Tecnologías de la Información y Sistemas
Universidad de Castilla-La Mancha, Ciudad Real, Spain
{Isabel.Bermejo, Ismael.Caballero, mario.piattini}@uclm.es

***Abstract*—The successful execution of a Business Process implies to use data with an adequate level of quality, thereby enabling the output of processes to be obtained in accordance with users requirements. The necessity to be aware of the data quality in the business processes is known, but the problem is how the incorporation of data quality management can affect and increase the complexity of the software development that supports the business process life-cycle. In order to gain advantages that data quality management can provide, organizations need to introduce mechanisms aimed at checking whether data satisfies the established data-quality requirements. Desirably, the implementation, deployment and use of these mechanisms should not interfere into the regular working of the business processes. In order to enable this independence, we propose the PAIS-DQ framework as an extension of the classical Process-Aware Information System (PAIS) proposal. The PAIS-DQ addresses the concerns related to data quality management activities by minimizing the required time for the software developers. In addition, with the aim of guiding developers in the use of PAIS-DQ, a methodology has been also provided to facilitate organizations to deal with complex concerns. The methodology renders our proposal applicable in practice, and has been applied to a case study where a service architecture implementing the standard ISO/IEC 8000-100:2009 parts 100 to 140 is included.**

I. Introduction

Dumas et al. in [1] introduced the concept of Process-Aware Information System (PAIS) for facilitating the specification and enactment of business processes. As a fundamental characteristic, and opposed to data-centric or function-centric information systems, a PAIS separates process logic from application code [2] in four different layers: Presentation, Process, Application and Persistence. In particular, a business process, henceforth referred to as BP, consists of a set of activities that are performed in a coordinated way by an organization. The activities jointly attain one or more business goals that outline the behaviour of the organization [3]. In the PAIS framework, models of business processes are represented in the Process Layer, and the functionality of their activities are implemented and deployed by services located in the Application Layer. In addition, the process can communicate with several stakeholders through the Presentation Layer. In this way, the PAIS framework and BPs are strongly linked.

The BPs under the scope of our research are those which are centred on developing sound data products rather than the sound execution of the processes. When a BP is executed in a PAIS, it implies that the straightforward management of the data is exchanged between activities or that the data is acquired from external resources in order to compose the final product. Consequently, the data that generates and composes the final product of outcome data is considered critical [4], and is essential for the BP [5]. Among other factors, it can be said that the success of an instance of the processes is grounded in the quality of the data used. Therefore, the management of data with the adequate level of quality constitutes a key value for the successful execution of these processes. In order to make organizations aware of the importance of data quality, a data quality management methodology must be implemented to cover the main data and data-quality requirements in the BP. The scope of both kinds of requirements must be described by business experts. On the other hand, Information Technology (IT) people have to implement the corresponding mechanism to satisfy the stated requirements. This is the point where our investigation is addressed. We propose that certain data quality management activities (e.g. assessment) can be both supported by and implemented as part of the PAIS. The main consequence of this hypothesis is that these data quality management activities can be implemented as part of the BP. It also permits the externalization as services and simply the invocation and use of these activities.

The PAIS life-cycle consists of four phases [6], as shown in Figure 1:

(1) The requirements analysis is established, the business processes are identified, reviewed, validated and presented as process models in the *process design* phase.

(2) The designs are developed and configured in a software system in the *system configuration* phase.

(3) During the *process enactment* phase, the process is executed by using the system configuration in the way prescribed by the process model. More specifically, an instance of a BP represents a specific case in the operational business execution of an organization.

(4) Finally, in the *diagnosis* phase, the operational process is analysed to identify problems in order to improve the process, and can even make a diagnosis with the aim of proposing a solution to these problems [7].

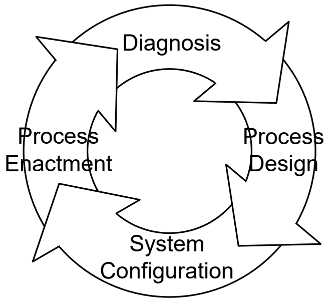


Fig. 1. PAIS Life-cycle [6].

As mentioned earlier, both business experts and IT people (along with Data Quality experts) have to decide how to incorporate the data and the data quality requirements through these four phases as suitable mechanisms and make them available for use. Once this goal is achieved, the next step is to adapt the BPs model to use these new mechanisms without altering their fundamental structure.

Although certain data-quality related studies could be used at the design phase, such as [8], [9], and [10], there is a lack of proposals for their *system configuration* and *process enactment* phases: a necessity which we intend to cover in this paper. Therefore, not only do we propose a theoretical solution, but we also define the steps to obtain an executable data quality-aware BP. To this end, we propose a modification of the traditional PAIS with the aim of supporting and addressing the data quality management in various phases of the PAIS life-cycle. In our proposal, all the data quality activities in charge of the control and enhancement are externalized and gathered in a new Data Quality Layer, called the DQ Layer. In addition to the advantages of making such mechanisms available, the DQ Layer enables the reduction of the time-to-execution for the specific data-quality requirement for each problem, since it maintains the functionality as independent and external to the process itself. To guide the incorporation and usage of this new DQ Layer, a methodology is also provided to help and support both business and data quality experts through the various phases.

The remainder of the paper is organized as follows: Section II proposes a case study which could be successfully applied to our problem. Section III introduces certain concepts related to quality and Data Quality Management (DQM). Section IV presents the PAIS-DQ framework, extended with the DQ Layer to address DQM. Section V shows the steps to transform a BP into a data-quality aware BP. In Section VI, our proposal is applied to the case study shown in Section II. Section VII includes an overview of relevant work and definitions about PAIS, BPs and DQM. Finally, conclusions are drawn and future work is described in Section VIII.

II. Detailing a Case Study

In order to let the readers achieve a better understanding of the benefits of this proposal, an example related to a trip planner is presented. This corresponds to the process aimed at finding and booking the cheapest flight to make a trip according to customer requirements. It is important to realize that our investigation is not directly interested in the quality of the result of the BP, but in the data used to build the product. Increasingly, people use marketplace applications on the web that integrate several results from queries to a variety of flight providers. A possible BP using the PAIS framework is shown in Figure 2. Business Process Model and Notation (BPMN) [11] is used to describe the different activities that constitute the BP, namely:

(i) Firstly, the customer, through the presentation layer, introduces his or her travel requirements.
(ii) Several activities are then executed in parallel to search for the flight that best meets customer requirements. These activities invoke a number of external services to obtain the flight information.
(iii) The best flight is chosen, typically according to the cheapest price.
(iv) The customer is informed of this best flight.

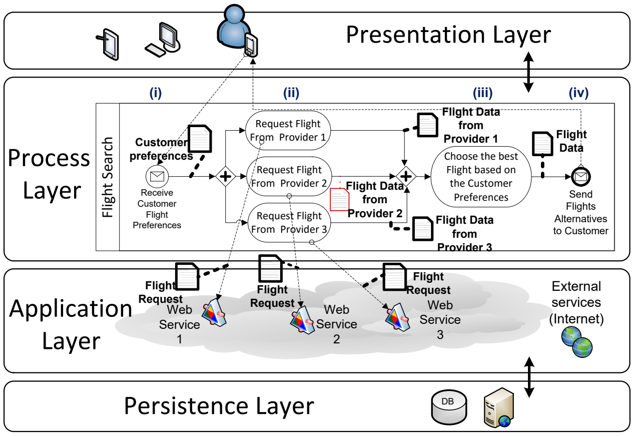

Presentation Layer
Process Layer
Flight Search
(i)
(ii)
(iii)
(iv)
Customer preferences
Receive Customer Flight Preferences
Request Flight From Provider 1
Request Flight From Provider 2
Request Flight From Provider 3
Flight Data from Provider 1
Flight Data from Provider 2
Flight Data from Provider 3
Choose the best Flight based on the Customer Preferences
Flight Data
Send Flights Alternatives to Customer
Application Layer
Flight Request
Web Service 1
Flight Request
Web Service 2
Flight Request
Web Service 3
External services (Internet)
Persistence Layer
DB

Fig. 2. Illustrative Example: Flight Search process.

In particular, the process model described in the Process Layer includes:

- The activities (*"Request Flight From Provider 1"*, *"Request Flight From Provider 2"*, ...);
- The data generated that flows through the process and is exchanged with the Application Layer (highlighted in bold, *"Customer preferences"*, *"Flight Request"*, ...);
- The control-flow (*AND* gateway in the example).

At the same time, in the Application Layer, the external services invoked by the activities are deployed and ready to be called (*"Web Service 1"*, *"Web Service 2"*, ...).

In each step of the process, a set of data is created, queried and updated. Special attention should be paid to the data obtained from external resources (step (ii) of the example), and to the decisions taken based on the just-received data (step (iii) of the example). In other words, it is paramount to observe the levels of quality of the data of interest in steps (ii) and (iii). In our case, data-quality concerns are addressed for the set of data used by the activities. In a particular case, the data whose levels of quality should be borne in mind due to its importance, are on the one hand the *Customer Preferences*, which is represented by *departingFrom*, *goingTo*, *departDate* and *returnDate*; and on the other hand, the *Flight Request* is represented by *flightNumber*, *carrier*, *departureTime*, *arrivalTime*, *priceFlight*, *checkedLuggagePrice* and *creditCardCharge*.

The question are: what does the quality in theses objects mean? how can the quality of data be preserved or improved in a BP? In the following sections both aspects are studied.

III. Data Quality Management in a Nutshell

For a better understanding of the scope of our proposal and to better ascertain how data quality can affect the implementation of a BP, let us firstly delve into the concepts of data quality management, and of suitable mechanisms which can be implemented in the DQ Layer.

In spite of the widest usage of the classic definition of **Data Quality** (DQ) as *fitness for use* [5], in this paper we rather prefer the definition of *meeting requirements* [12]. It brings certain advantages from the point of view of implementing the various mechanisms of the DQ Layer. The main advantage is that this definition involves the barely used concept of *internal data quality* vs *external data quality* as a way in which a set of data satisfies the stated requirements. This concept forces differences to be highlighted between: (i) what defines the data (as a product); and (ii) what factors are specific to the assessment of the quality. The reason for such distinction is to better identify and separate certain aspects of the quality of *"data product"*, aspects that may or not fit with what defines the data. The data is defined by means of certain features (i.e. *any of the components of the piece of data: name, attribute, value, data type*) that are set up through the design process in accordance with the data requirements. Therefore, in the case when data fails to satisfy the requirements, this will be due to the fact that corresponding features are not properly designed or used.

When the assessment of the level of quality of a piece of data is required, the specification of a number of **DQ requirements** against which DQ is to be judged become necessary. These DQ requirements can be of two types: **High-Level Data-Quality Requirement** and **Low-Level Data-Quality Requirements** [8]. For the definition of high-level DQ requirements, some criteria, commonly known as DQ requirements [9], which represent generic concepts, should be identified in accordance with what customers need to know. Some example can be found when customers could state that their data should be accurate, or complete, or on time. On the other hand, low-level DQ requirements address the degree of accuracy required, or the required level of completion, or how long it could be delayed and still remain usable. To this end, the DQ characteristics that are observable on data (e.g. those appearing in ISO 25012 [13]) must be specified. Commonly, DQ characteristics (what it should be observed in the data) can be mapped 1:1 to DQ dimensions (i.e. that customers need to measure). To effectively measure a DQ dimension on an item of data, for each of the DQ characteristics involved, certain measurable attributes have to be identified.

Given that the definition of a quality requirement being met is related to the idea of zero defects, a customer must specify whether a data is defective or not by expressing some acceptance criteria that should be part of the low-level DQ requirement. For example, customers can decide that their data is defective with regard to completeness, when the ratio of incomplete records is lower than 90%. Thus, in order to assess the level of quality of a piece of data or of a dataset, the DQ experts should define explicit and customized measurement procedures for each DQ dimension that take into account the corresponding measurable attributes. These measures will later be used against the low-level DQ requirement to decide whether the data is defective or not.

In the case when data could be considered defective, then some enhancement according to the DQ policies defined by the organization is required. This process first involves the identification of: the cause of the systematic production of defective data; the root causes (e.g. the collection process is failing, or the database was not appropriately designed); the specific feature of data which contributes towards the defect; and how the defect could be fixed. As a consequence, existing data requirements must be revised or even new requirements should be generated which specifically address the reparation of errors. This could be covered by the DQ management function of the organization. In our proposal, we consider that this task is implemented in the DQ Layer of the PAIS. Therefore, in order to fix a defect (enhance an item of data), the corresponding service of the DQ Layer should be invoked to change a specific feature; commonly the change of the value of the data.

IV. PAIS-DQ Description

In order to enrich the BP models with DQ aspects, it is necessary to consider the following points as the main rational for our investigation: (i) the idea of considering data as an outcome product of the process [5], which enables the application of basic quality principles to the data; (ii) the need to consider the quality of data in the process design phase; and (iii) the increment of the complexity of the model due to the inclusion of the quality and more control-flow tasks.

We propose PAIS-DQ as a way to simplify the execution of a number of the low-level DQ management activities. These activities articulated as part of the high-level DQ management activities are defined by the organization in order to ensure the overall level of quality of the data used in their running BPs. The simplification includes the possibility of externalizing the services to enable their reusability at different points of the BP. Therefore, the objective is the attainment of advantages of having services externalized. It provides the possibility to have the set of reusable mechanisms that are aligned to DQ strategy that can be included in a BP in an easier way.

The set of low-level activities includes *measurement*, *assessment*, and *enhancement* of data. These activities should be considered as atomic operations for the high-level DQ management activities of DQ control and/or DQ assurance. In other words, if an organization is willing to grant its BPs DQ awareness at design level [8], then the PAIS-DQ offers the capacity to include the necessary components at an implementation level.

This section describes how the PAIS-DQ is structured. Along with the extension of PAIS, we also consider that, in order to support business and DQ experts and to enable PAIS-DQ to be used, some methodological guidelines must be provided in PAIS-DQ-HOW (see Section V).

A. PAIS-DQ Architecture

We propose extending the classic PAIS framework [2] with the so-called DQ Layer, combined with the Presentation, Process, and Application Layers (see Figure 3).

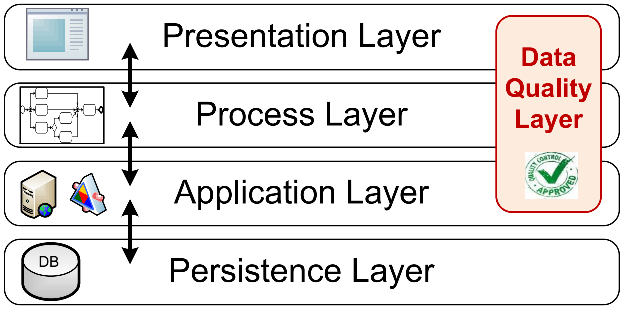


Fig. 3. Framework for Data Quality Management.

BP models designed in the Process Layer establish communication with the customer through the Presentation Layer and run the services implemented in the Application Layer, since it is possible to combine several services within the same process. The DQ Layer is in charge of providing, in an external and independent way, the necessary functionalities and mechanisms to manage the level of quality of the data that flows through the process in each instance. To this end, not only the Process Layer is affected by the DQ Layer. Firstly, certain mechanisms must be provided to enable the customer to establish the required DQ level. These mechanisms must be provided by the DQ Layer at the Presentation Layer. At the same time, in the process model of the BP under study, it must be specified which data is to be controlled and/or assured.

Therefore, the DQ Layer must provide the necessary mechanisms to the Process Layer to locate, specify and define the data involved in the low-level DQ management activities. And finally, the services in charge of the DQ functionalities (i.e. activities) are implemented, deployed and located by the DQ Layer in the Application Layer.

Table I, shown at the beginning of the next page, summarizes the various needs in each of the four PAIS Layers: DQ Layer Necessities, What needs to work; DQ Capabilities, What is provided; and DQ Layer Responsible, Who is in charge of configuring these capabilities.

On the other hand, once data used in the BP is analysed and/or modified in the DQ Layer, the information about the level of DQ should be included in the BP model. For this reason, a new type of data is included in order to make the BP "data quality aware". This new data type is called *Data-Quality Items* (DQ Items).

Hence, *DQ Items* constitutes the set of data that contains the information about the level of quality of a specific item of data and/or a specific dataset, and is related to a set of DQ dimensions. Therefore, these *DQ Items* should point to other types of data, since they could include additional information related to the level of quality. In our case study, the piece of data called *arrivalTime* can be enriched with the *DQ item* corresponding to its *accuracy* information (e.g. level required, moment in the BP where it must be controlled and/or assured, and value obtained). How this information should be included in the data flow is detailed in Section V.

B. Data Quality Management Activities

As explained earlier, the high-level DQ management activities (control and/or assurance) permit to use some low-level DQ management activities (measurement, assessment, decision-taking, and enhancement). In this way, companies establish their DQ necessities through these high-level DQ management activities, which, in turn, imply the development and implementation of certain low-level DQ management activities. The relationship between these activities is shown in Table II.

TABLE II. ACTIONS DEPENDING ON DQ REQUIREMENTS

High-Level Activity Low-Level Activity	Control	Assurance
Measurement	Yes	Yes
Assessment	Yes	Yes
Decision-Taking	Yes	Yes
Enhancement	No	Yes

How details concerning these DQ management activities can affect the BP, and how PAIS-DQ support this influence are introduced in the following subsections.

1) How High-Level DQ Management can affect the BP Model: When an organization decides to make its BP "data quality aware", then the high-level DQ management activities (control and/or assurance) should be included into the BP. This decision implies the modification of the process model.

Implementing DQ control involves inclusion of the low-level DQ management activities of measuring, assessing and/or enhancing data. To this end, business experts, together with the DQ experts, must decide whether or not the BP model should be changed, and if necessary, how to introduce these new activities. For example, a BP model should not be changed if the consequences of including decisions related to DQ only imply the modification of the conditions associated to the branches of the flow, or if the consequences only affect one activity. Specifically, Figure 4 shows how the conditions to take the different outgoing branches of an exclusive gateway are enriched with the assessment results.

On the other hand, on implementing DQ assurance in order to ensure a specific level of DQ in a BP, both types of expert should study and evaluate how, at a certain point in the process, a set of DQ requirements must be implemented. For example, a level of DQ can be assured by enhancing a set of data, or possibly, by going back to a point at which the level of DQ was acceptable, such as shown in Figure 5, where if the DQ level of the flight data from the provider is assessed as "not

TABLE I. DQ LAYER IN PAIS-DQ

PAIS Layer	DQ Layer Necessities	DQ Layer Capabilities	DQ Layer Responsible
Presentation Layer	Required DQ level by customer	Data defining required DQ level	Business and DQ Expert
Process Layer	BP model + DQ level + required Low-level DQ management activity	BPMN enriched with DQ requirements	Business and DQ Expert
Application Layer	DQ Services	DQ functionality	DQ Expert
Persistence Layer	-	-	Business Expert

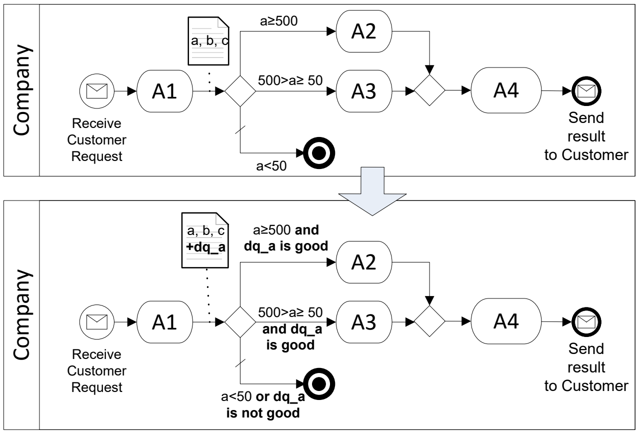


Fig. 4. Data Quality Control by including new decision rules in a gateway.

good"[1], then the provider is asked for flight information again. In this case, new branches, gateways and even an end event, are included in the BP (marked in bold in Figure 5).

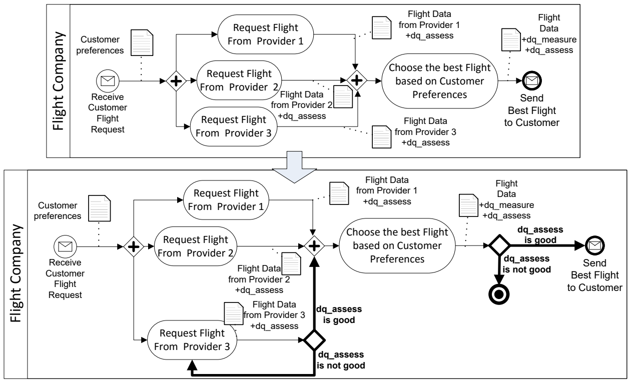

Flight Company
Customer preferences
Receive Customer Flight Request
Request Flight From Provider 1
Request Flight From Provider 2
Request Flight From Provider 3
Flight Data from Provider 1 +dq_assess
Flight Data from Provider 2 +dq_assess
Flight Data from Provider 3 +dq_assess
Choose the best Flight based on Customer Preferences
Flight Data +dq_measure +dq_assess
Send Best Flight to Customer
Flight Company
Customer preferences
Receive Customer Flight Request
Request Flight From Provider 1
Request Flight From Provider 2
Request Flight From Provider 3
Flight Data from Provider 1 +dq_assess
Flight Data from Provider 2 +dq_assess
Flight Data from Provider 3 +dq_assess
dq_assess is good
dq_assess is not good
Choose the best Flight based on Customer Preferences
Flight Data +dq_measure +dq_assess
dq_assess is good
dq_assess is not good
Send Best Flight to Customer

Fig. 5. Data Quality Assurance by inclusion of new branches.

2) How Low-Level DQ Management Activities can affect the BP Model: The implementation of low-level DQ management activities brings several consequences to the BPs:

- **Leverage of Internal Performance:** This strategy proposes the implementation of some of the previously stated low-level DQ management activities inside the BPs. This commonly implies the inclusion of new activities within the BP model by using some of the programmable languages that exist in a Business Process Management System (BPMS). In [14], the authors show how these types of activities can be included in the BP as new activities. Specifically, a new activity is included for the overall assurance of the level of quality for a DQ dimension required.
- **Leverage of External Performance:** External services are responsible for providing the low-level DQ management functionality. Therefore, the activities that need to measure, assess and/or enhance the level of quality of some data must call the external service that contains the necessary DQ functionalities. This external enactment can be invoked from:
  - ○ One of the existing activities in the BP.
  - ○ A new activity created for the invocation, as proposed in [8], [14], and [10].

In this work, externalization into the DQ Layer is proposed of as many of the low-level DQ management activities as possible, as shown in Figure 6.

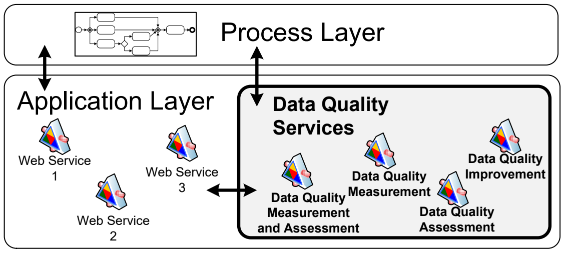


Fig. 6. Data Quality Layer services.

C. Data Quality Layer Functionalities

In our proposal, we strive to optimize the implementation of the DQ management activities from the point of view of the development of the software systems that support the BP. It is assumed that the DQ Layer can minimize the degree of modification required of the BP process model and the corresponding software system(s) that support the BP. Consequently, the more these functionalities can be externalized, the less the software system(s) will need to be modified. It is therefore intended, by means of the DQ Layer, to enable the reduction of the implementation time for the specific DQ requirement for each problem. The various activities of the BP process must

[1] The possible values for the assessment, such as the "not good" value, are established by the DQ expert through a classification (for example, "very good", "good", "not good" and "not very good"), the result of the assessment matches with one of these possible values (see subsection IV-C for further details).

interchange the *DQ items* with the corresponding activities of the DQ Layer associated with measurement, assessment, and/or enhancement. Consequently, the functionalities to be included as part of DQ Layer are determined by the low-level DQ management activities required to render the BP data quality aware. Further details are provided:

- DQ Measurement: the quantification of the level of data quality, following the principles of *meeting requirements*, needs the set of data requirements that data must meet. These data requirements are related with a set of DQ metrics that are relevant for the business. Both, the business expert and the DQ expert must reach an agreement on how to define the corresponding measurement methods for these metrics, and how to implement them in the DQ Layer. The implementation has to be parameterized in order to enable multiple measurements for various items of data of the BP.
  A good way to increment the trustworthiness of data involve adding some extra information *to the DQ items* about the results of the measurement by an authoritative party. Specifically, and when the BP has to interchange data with an external entity (e.g. a provider), the certification could be carried out by an external, independent and authoritative entity in charge of guaranteeing a level of DQ, following the requirements described by ISO 8000-100 [15].

- DQ Assessment: certain decisions concerning the convenience of using the data have to be taken. The decision is to be made on the basis of the results obtained from DQ Measurement being compared to an objective threshold, stated by those business people who really know the processes, and the conditions in which the BP can run smoothly and seamlessly. These conditions commonly represent the context for the assessment of the data. Again, communication with the DQ Layer should be parameterized to enable the various invocations that could be necessary throughout the BP.

- DQ Enhancement: the necessary data requirements to transform, adapt, and/or change the data in order to better meet a specific level of DQ must be defined and communicated to the DQ Layer. This is necessary in order to let functionalities make the corresponding changes to the pieces of data, so that requirements can be met. In our case study, an example of enhancement can be the transformation of the format of some data, e.g. the date as "31/02/1985" (DD/MM/YYYY) instead of "02/31/1985" (MM/DD/YYYY). Although this example could be suited to discuss within schema matching techniques, is regarded as a clear example of enhancement [16]. Further examples include: the identification of spelling mistakes in the names cities and countries or in dates, e.g. when the customer types *"Lndon"* instead of *"London"*. The improvements must be carried out based on the specific DQ dimensions under study, and on some patterns or sources that make these improvements, such as *Wordnet* [17], which is a lexical database of English that can syntactically improve a word. Therefore, the set of requirements or patterns that this data must meet have to be defined.

The deployment and usage of the various functionalities could be performed by taking into account the technologies that best fit the BPs. For example, web-service technology has been demonstrated to be highly useful [6].

V. PAIS-DQ-HOW: Methodology to use PAIS-DQ

A methodology is proposed in order to ease the utilization of PAIS-DQ and to provide some structured guides that enable IT people to use the corresponding DQ Layer. Compared to certain proposals in the literature, such as BPiDQ by [8], our proposed methodology, called PAIS-DQ-HOW, goes beyond the design phases of BPs. In fact, not only does it cover the Design Process phase of the PAIS life-cycle, but it also completes the rest of PAIS life-cycle phases to facilitate the inclusion of DQ management activities into the BP.

In addition, PAIS-DQ-HOW covers:

- The design of the high-level DQ management activities in the organization.
- The design of the low-level DQ management activities.
- A full description of the DQ items that should be exchanged.
- The design and deployment of the DQ Layer.
- How low-level DQ activities are included in the BPs by using the DQ Layer to achieve the goals of the high-level DQ activities.

Therefore, how these inclusions affect the various PAIS-DQ Layers is described and specified through the PAIS life-cycle phases.

A. *Business Process Design*

BP models contain all the information about the activities, the data-flow, and the sequence-flow located in the Process Layer. In addition, the necessary services in charge of the functionality of several activities are situated and described in the Application Layer. BPs can be modelled in any Business Process Management System, or can also form the input of the methodology. This step can be skipped if the BP already exists and if the objective is to make this BP data quality aware.

B. *Data Quality Layer and System Configuration*

Once the process is modelled in the Process Design phase, the business expert together with the DQ expert must configure the process (System Configuration phase) and decide which DQ management activities must be considered in the BP. Both experts must then also decide the data affected, the BP changes, and the specific implementation.

1) Configuration of High-Level DQ Management Activities: The organization, taking into account its specific needs, should identify which activities (control and/or assurance) should be implemented to obtain greater benefits. This also implies the selection of those parts of the BP that are susceptible to requiring special attention for the data used.

As part of the control and/or assurance, not only does data requiring control and/or assurance have to be identified, but also the DQ dimensions representing the criteria that should be covered. Developers should therefore specify the DQ requirements for each item of data through the specification of the following information:

- ***Data*:** the data that should be controlled and/or assured.
- ***DQ Requirements*:** the DQ characteristics that are going to be included in the DQ analysis, (e.g. completeness, accuracy, and credibility). These DQ characteristics, along with their corresponding values, are included as new DQ items into the data-flow.
- ***Who measures*:** who is in charge of the measurement of the level of DQ.
- ***Who assesses*:** who is in charge of the assessment of the level of DQ.
- ***Who enhances*:** who is in charge of the enhancement (improvement) of the level of DQ.

As a result, several *DQ items* are identified and must be fully described. Furthermore, their usage has to be standardized across the BP.

In order to communicate to the final customers the state of the data that they are about to use, these DQ items are exchanged between the layers of the PAIS. The way to calculate the value for these DQ items in the BP is performed through the corresponding low-level DQ management activities. These activities establish various algorithms and mechanisms for the calculation of the value for each DQ item as explained below.

2) Configuration of Low-Level DQ Management Activities: In order to conduct the design of the high-level DQ management activities, organizations should coordinate the required low-level DQ management activities.

There are several possible configurations for the low-level DQ management activities. Therefore, along with the identification of the activities, in this stage, certain further elements must be defined:

1) The set of low-level DQ management activities that are part of the high-level DQ management activities.
2) The interface for each activity. As part of this interface, the corresponding input and output data (parameters) that enable customization to different scenarios should also be defined. Examples of these parameters include the data requirements related to certain DQ dimensions, such as the measurement related to completeness, or the assessment related to completeness.
3) The corresponding algorithm/mechanism for the measurement method, assessment method and/or enhancement methods. For example: for measurement, the corresponding measurement methods (the algorithms) have to be defined; for enhancement, the corresponding enhancement of data (including possible sources) should be identified.

Once the low-level DQ management activities are designed, the corresponding implementation must be developed and deployed. In this case, the DQ expert, in representation of the entity (organization or department) in charge of performing the DQ aspects, must develop the necessary implementation in order to make these low-level DQ management activities available.

3) Changes to the BP to make it DQ aware: Therefore, depending on the high-level DQ management activities, the set of changes to apply to the BP varies. On the one hand, the data-flow is always modified to include DQ items. On the other hand, the experts should act accordingly on the control and/or assurance necessities, as explained previously, (e.g. externalizing the measurement, including a decision rule in a gateway, and including a new activity to improve a set of data). In other words, the adaptation or redesign of the BP includes the low-level DQ management activities that form the high-level DQ management activities planned by the organization.

C. Business Process Execution

Once all the parts of the extended model have been defined, the executable process can be performed. When an instance reaches an activity connected with an external service in charge of the DQ aspect, then the values for the data of the instance and the requirements and thresholds (if necessary) are sent to this service. Otherwise, when an instance reaches an activity in charge of the DQ aspects, then the values for the data, the requirements and the thresholds (if necessary) are taken from the incoming sequence-flow.

The synopsis of which PAIS-DQ Layer is affected in each step of the methodology, what is used, what is obtained, and who is responsible is detailed in Table III.

VI. PAIS-DQ Applied to the Case Study

In order to illustrate the usage of both the methodology and the DQ PAIS, in this section an explanation is given as to how to render the case study DQ-aware. To this end, the three steps of our aforementioned methodology are applied and detailed in the following subsections.

A. BP Design: Flight Search Process

The BP used here is the presented in Section II as a case study. This process can be modelled in any Business Process Management System, such as Intalio^{TM} [18], Activiti^{TM} [19], and Bonita Open Solution^{TM} [20]. In our case, Bonita Open Solution^{TM} is applied to design and execute the BPs, as shown in Figure 7, since it is an open-source application with a free distribution, and is commonly used in the private market.

B. DQ Layer and System Configuration

In our case study, we are specifically interested in implementing a **DQ control** plan, since we consider that customers can be aware of the level of quality of the data they are

TABLE III. PAIS-DQ-HOW METHODOLOGY

Methodology Stage	PAIS-DQ Layer	What is used	What is obtained	Responsible
1. Process Design	PAIS	PAIS	BP Model	Business Expert
2. DQ Layer and Sys.	Presentation, Process and Application Layers	PAIS-DQ	BP Model + DQ	Business, IT and DQ Expert
2.1. High Level.	Presentation and Process Layers	BP Model	High Level Activities + DQ Items	Business and DQ Expert
2.2. Low Level.	Application Layers	High Level Activities	Low Level Activities Design and Implementation	DQ Expert
2.3. Changes to BP.	Process Layer	BP Model + DQ Requirements	BP Model with DQ	Business and IT Expert
3. BP Execution	PAIS-DQ	PAIS-DQ	Executable BP	IT Expert

using when booking flights. Therefore, the DQ Layer and the System configuration is focused on providing and developing the necessary mechanisms to control the level of DQ in the Flight Search Process. Therefore, the aim of this stage is to make a BP become DQ-aware through the configuration of its DQ items, required low-level DQ activities and implementation of these low-level DQ activities.

1) Configuration to Control Data Quality Levels: Regarding the data that must be controlled, we focus on that corresponding to the outcome data of the activities *Request Flight from Provider 1*, *Request Flight from Provider 2*, and *Request Flight from Provider 3*. These items of data in the case study are the result of the data exchange between these three activities and the related external services. Table IV provides details of the information needed, given by the DQ Experts, for some of these items of data. In this case study, no item of data is enhanced since it is not required and therefore this information is omitted.

2) Data Quality Items: The DQ dimensions chosen were **completeness** and **accuracy**. For these DQ dimensions, it was decided to include this information into that which should be communicated to the customer. This information is given by the DQ items. These DQ items must be managed as part of the operation of the DQ Layer, and must be exchanged from and to the DQ Layer and the corresponding activities in our example. See the next subsection for a more in-depth description.

3) Low-Level DQ Activities to Control DQ Configuration: On the other hand, since we have chosen a DQ control plan, and taking into account the set of activities in Table II, the low-level DQ management activities that should be implemented and deployed to the DQ Layer are the measurement, assessment, and decision on what to do. Therefore, an explanation is given below on how these DQ requirements are carried out through a BPMS.

As part of this process, details are shown in Figure 7 on how to measure the level of quality of the data produced by the activity *Request Flight from Provider 1* (first step of the business process).

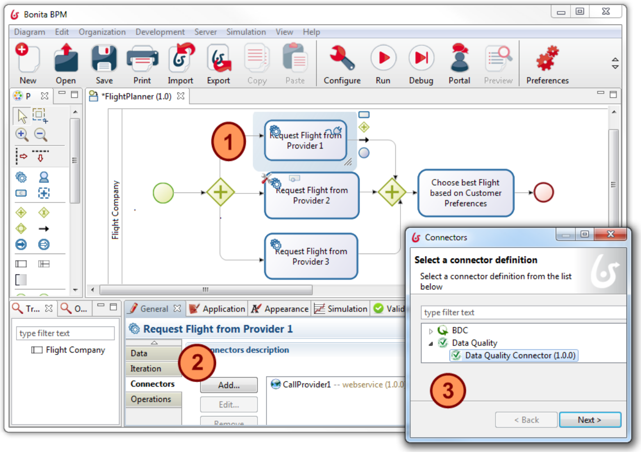
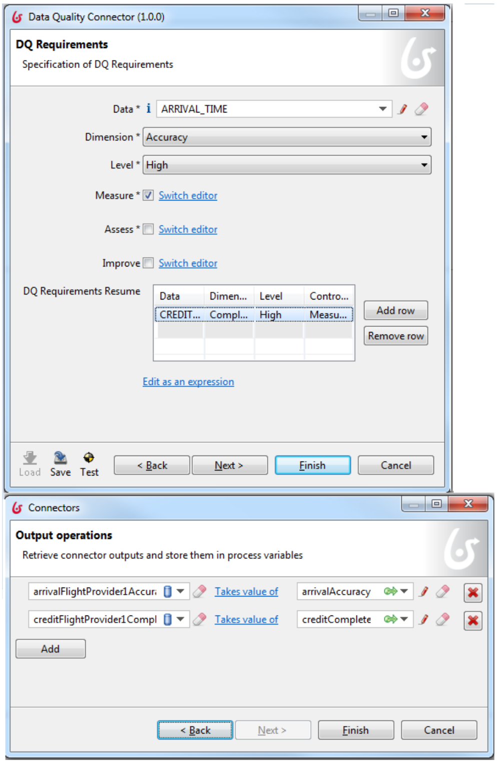

Bonita BPM
Diagram Edit Organization Development Server Simulation View Help
New Open Save Print Import Export Copy Paste Configure Run Debug Portal Preview Preferences
*FlightPlanner (1.0)
Flight Company
1
Request Flight from Provider 1
Request Flight from Provider 2
Request Flight from Provider 3
Choose best Flight based on Customer Preferences
type filter text
Flight Company
General Application Appearance Simulation
Request Flight from Provider 1
Data
Iteration
Connectors
Operations
2
Add...
Edit...
CallProvider1 -- webservice
Connectors
Select a connector definition
Select a connector definition from the list below
type filter text
BDC
Data Quality
Data Quality Connector (1.0.0)
3
< Back
Next >

Fig. 7. Add a *Data Quality Connector* to an activity with Bonita Open Solution^{TM}.

Since the DQ measurement is externalized and located in the DQ Layer, one way to measure the level of quality of the

TABLE IV. DATA QUALITY REQUIREMENTS FOR *Request Flight from Provider 1* ACTIVITY OUTPUT

Data	**Dimension**	**Level**	**Control/ Assurance**	**Who measures**	**Who evaluates**	**Who enhances**
arrivalTime	Accuracy	High	Control	I8K	I8K	-
checkedLuggagePrice	Accuracy	High	Control	I8K	I8K	-
creditCardCharge	Completeness	High	Control	I8K	I8K	-
...	...	...	...	...	...	...


Data Quality Connector (1.0.0)

DQ Requirements

Specification of DQ Requirements

Data * ARRIVAL_TIME

Dimension * Accuracy

Level * High

Measure * ☑ Switch editor

Assess * □ Switch editor

Improve □ Switch editor

DQ Requirements Resume

Data	Dimen...	Level	Contro...
CREDIT...	Compl...	High	Measu...

Connectors

Output operations

Retrieve connector outputs and store them in process variables

arrivalFlightProvider1Accur: Takes value of arrivalAccuracy

creditFlightProvider1Compl Takes value of creditComplete

Fig. 8. *Data Quality Connector* configuration for the activity *Request Flight From Provider 1*.

data returned by a provider involves the addition of a *Data Quality Connector* to this activity (second step). A *connector* is the way in which Bonita Open Solution^{TM} links an activity with the service or application that executes a functionality, such as the DQ measurement.

The *Data Quality Connector* has been designed with the aim of specifying the DQ requirements in order to send the data values at run-time to the software in charge of the DQ measurement, assessment, and/or improvement. In other words, it connects activities in the Process Layer to the DQ Layer to obtain the results of DQ functionalities. In addition, the *Data Quality Connector* also permits connector outputs to be retrieved and to store them in the process variables. As shown in Figure 8, the DQ requirements detailed in Table IV are indicated through the wizard provided and then the result obtained by the service can be stored in the DQ items of the BP. In the example, the DQ items are: *arrivalFlightProvider1Accuracy* and *creditFlightProvider1Completenness*.

4) Low-Level DQ Activities to DQ Control Implementation: It is possible to find various solutions that implement the necessary functionalities to support the operation of the DQ Layer. The implementation must support several combinations of the aspects described in the previous section. In this section, the opportunity of adding certain information concerning the certification of DQ levels is also considered in order to increment the reliability of the data. This information is to be included into the DQ items. Once the information of the certification of the data is conveniently supplied, the process can be adapted so that it may be managed, and decisions, based on such information (control and/or assurance), may be made in execution time.

Specifically, we have developed a service architecture, named I8K and previously introduced in [21], which satisfies the requirements for the DQ certification schema given by the ISO 8000-1x0:2009 family [15]. These requirements consist in the incorporation of certain information on the data being exchanged between the Process and the Application Layer. As part of this information, the certification of the DQ is included (see Figure 9).

More specifically, the standard supports the certification of only those two DQ dimensions chosen: accuracy in ISO 8000-130 [16], and completeness in ISO 8000-140 [22].To the best of our knowledge, there is currently only one public usable implementations of standard ISO 8000-1x0:2009: that developed by ECCMA, which is available at [23] under payment. However, the I8K implementation strives to satisfy all of the requirements established in the various parts of the family of standards. This motivated our decision to carry out our own implementation.

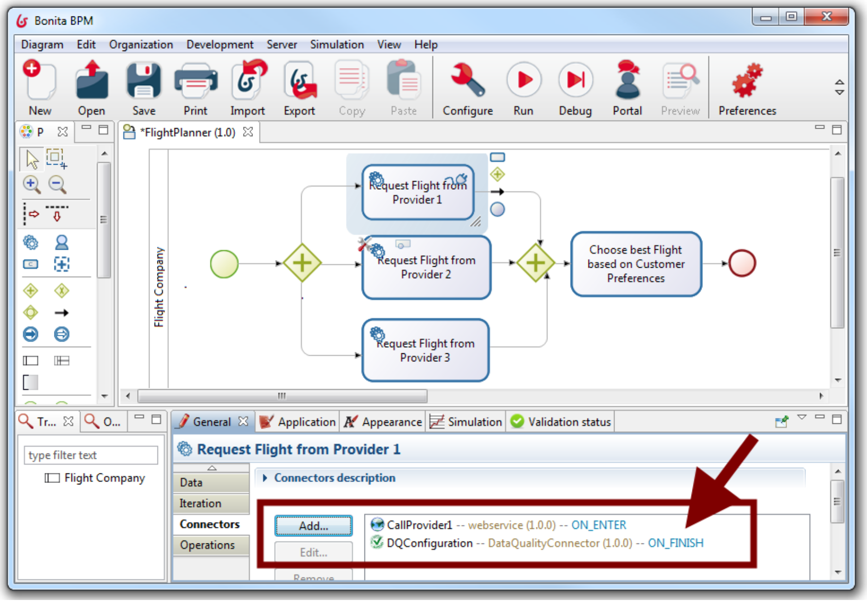
This I8K therefore provides our DQ Layer and supports the low-level DQ management activities needed for our case study.

5) Flight Search Process Changes: While focusing on the case study, let us explain one of the changes to be made: as part of the DQ control, the business and DQ experts must adapt the *Choose the best Flight based on Customer Preferences* activity since they are responsible for verifying and deciding how the decision can affect these DQ levels. For example, if a customer establishes, as a DQ requirement, that the landing time (arrivalTime) has to be accurate, then it is expected that if any of the activities returns a flight with a landing time without specifying whether it is "a.m." or "p.m.", then this flight is not considered by the activity *Choose the best Flight based on Customer Preferences*, and therefore, is not offered to the customer.

On the other hand, for the activity *Request Flight From Provider 1*, it was decided to include the measurement of the DQ. To implement the measurement, two connectors to the activity were required: the first, which connects with the provider; and the second, which connects with the entity in charge of the DQ measurement, such as shown in Figure 10. This implementation is similar for the remainder of the

```
<completeness-event event-type="certify140" organization-ref="
    I8K" date="2013-05-21T23:49:58.213+02:00">100 %</completeness
    -event>
```

Fig. 9. Fragment corresponding to the completeness dimension given by I8K Architecture.


Bonita BPM
Diagram Edit Organization Development Server Simulation View Help
New Open Save Print Import Export Copy Paste Configure Run Debug Portal Preview Preferences
*FlightPlanner (1.0)
Flight Company
Request Flight from Provider 1
Request Flight from Provider 2
Request Flight from Provider 3
Choose best Flight based on Customer Preferences
General Application Appearance Simulation Validation status
type filter text
Flight Company
Request Flight from Provider 1
Data
Iteration
Connectors
Operations
Connectors description
Add...
Edit...
Remove

CallProvider1 -- webservice (1.0.0) -- ON_ENTER
DQConfiguration -- DataQualityConnector (1.0.0) -- ON_FINISH

Fig. 10. Connectors in Bonita Open Solution^{TM} for the activity *Request Flight From Provider 1*.

activities.

C. BP Execution of the Case Study

Once all the parts of the BP have been defined, the execution process can be performed using the execution engine of Bonita Open Solution^{TM}. The selection of the best flight now takes into account the DQ items containing information about the completeness and the accuracy, and that information can be used accordingly.

D. Evaluation

In order to test the proposal with the case study, we have execute the process to find the cheapest flights from different locations, in two scenarios, (1) without asking data having adequate levels of quality and (2) asking data with the adequate level of quality. For example, we obtain in a test that the cheapest flight in the first scenario had a cost of 290 , whereas in the second case, we obtained a flight with a cost of 320.40 . In our tests, both results were offered to various users, who rejected the first offer, because of the need of trusting in data was more important that having cheaper flights. This led us to check how companies can lose customers due to inadequate levels of quality. Although data representing the flights could have not adequate level of quality, the flights per se would be correct, and the user could become totally satisfied with the flight, but users do not want to risk their money. Anyway, it is also important to highlight that if DQ management is not implemented, then the requests to DQ services are not done, which means that the cost in communications is not increased. It is necessary to highlight that nowadays the cost of communications between services in the same server is almost imperceptible to customer thanks to the new technologies. Furthermore, in spite of the increase of the cost in communication, customers satisfaction will also be increased since received data will be better than without the adequate level of data quality. On the other hand, in order to avoid infinite loops when a required DQ level is unreachable, various procedures can be followed. For example, if after a number of requests the required level of data quality is not given by a provider, then: (i) this provider is disregarded; (ii) the solution found is given to the customer but pointing out which level of data quality has been reached; or (iii) the customer is asked to lower the level of quality in order to obtain a satisfied solution. The organization should decide which procedure follows.

VII. Related Work

Once the underlying concepts about PAIS and DQM have been studied, our purpose is to include DQM activities into PAIS in a transversal way. This enables BP to take advantages of using the various activities of DQ management. This implies the capacity to ascertain the level of quality of the data that flows through a process without devaluating the process itself. It means that it would be possible to implement the DQ management activities, along with any activities or decision processes, without the need for implementing a specific solution to manage the DQ in each case. Therefore, an increase in the number of the layers is implied (Presentation, Process, Application and Persistence) with any other layers that have specific goals. In [24], Jablonski and Bussler identified other important perspectives for PAIS: causality, integrity and failure recovery, history, security and quality. The quality perspective is related to the *"establishment of a control mechanism to determine whether a process instance has been executed in an efficient manner or not"*. However, they fail to define a DQ perspective related to the level of quality that data should reach. In [25], [26], and [27], Gómez-López et al. the authors present an extension of the PAIS framework, where an analysis of the correctness of the data stored in the Persistence Layer is proposed in order to diagnose the incorrect data according to the Business Rules of the process, but this extension is not related to the DQ aspects.

On the other hand, there are various studies that specifically focus on how to design quality-aware BPs. In [28], Heravizadeh et al. identify which DQ dimensions should be analysed in a BP. Cappiello and Pernici, in [29], describe a methodology for integrating some concerns related to DQ management, specifically, on how BPs should react when errors due to poor DQ occur during the enhancement of the Web Services. However, they focused their research on the detection and correction of errors of data exchanged by the services and found at runtime. In addition, they also analysed the correct way of working for an activity based on this data. The main difference to our work is that we study how to better address the various ways to measure, assess, control, improve and assure the DQ level in a BP that is supported by a PAIS, and not just the possible errors of this data. In [30], Martin et al. reports on integrating DQ consideration into business process management into process modelling but lacks in explaining how to include it in the execution part. On their part, Rodríguez et al., in [10], propose a BPMN extension to model several DQ aspects. Nevertheless, their focus is on a descriptive approach rather than an analytical approach. Following on from that work, Caro et al. [14], and Cappiello et al. [8] tackle the problem of how the BP is affected by the management of DQ by defining a methodology called BPiDQ to consider DQ issues in the BP modelling phase; however, they fail to consider how this BP is affected at runtime when the model is executed in a BPMS.

Furthermore, after conducting a systematic literature review, it is found that none of the proposals encountered specifically address this DQ perspective in the PAIS Framework and through any of the PAIS life-cycle stages. In addition, we demonstrate this proposal with an implementation of these ideas in a commercial BPMS.

VIII. Conclusions and Future Work

This paper proposes an extension of the Process-Aware Information System framework, called PAIS-DQ, with the aim of including data-quality aspects into business processes. The control and/or assurance of data quality should not be given by the activities that shape the process, since the level of data quality has to be guaranteed by an external and independent entity. PAIS-DQ includes a Data Quality Layer in charge of incorporating these quality aspects into the BP, thereby avoiding modification of the BP model itself with data quality aspects, and maintaining a separation between the BP model and how the quality level is obtained. Thanks to this traversal layer, it is possible to execute a BP instance which not only covers the preferences of customers, but also surpasses their expectations with regards to data quality. In addition, a methodology to apply PAIS-DQ is proposed. Data quality awareness is incorporated by means of the inclusion of these data quality aspects into a BP model in a commercial BPMS using DQ Layer.

The usage of the methodology and the DQ Layer have been illustrated by applying them to a case study, in which we have shown how to introduce several data-quality activities to control the levels of completeness and accuracy of the data. In our case study, I8K, which meets the requirements of ISO 8000-100 to ISO 8000-140, has been used as the implementation of the DQ Layer, to provide the support to the low-level data quality activities.

As future work we plan to incorporate some of these modification following the DMN [31] standard to facilitate the incorporation of decision according to data quality level.

Acknowledgement

This work has been partially funded by the research projects TDiaCO-BPMS (TIN2015-63502) and GEODAS-BC (TIN2012-37493-C03-01), financed by the Ministry of Science and Technology of Spain and by the Ministry of Economy and Competitiveness of Spain, and the European Regional Development Fund (ERDF/FEDER), respectively; and by the *V Research Plan* of the University of Seville (VPPI-US). Thanks to Lesley Burridge for revision of the English version of the manuscript.

References

[1] M. Dumas, W. M. van der Aalst, and A. H. ter Hofstede, *Process-aware Information Systems: Bridging People and Software Through Process Technology*. New York, NY, USA: John Wiley & Sons, Inc., 2005.

[2] B. Weber, S. W. Sadiq, and M. Reichert, "Beyond rigidity - dynamic process lifecycle support," *Computer Science - R&D*, vol. 23, no. 2, pp. 47–65, 2009.

[3] M. Weske, *Business Process Management: Concepts, Languages, Architectures*. Springer, 2007.

[4] L. Parody, M. T. Gómez López, and R. Martínez Gasca, "Hybrid business process modeling for the optimization of outcome data," *Information and Software Technology*, vol. 70, pp. 140 – 154, 2016.

[5] R. Y. Wang, "A product perspective on total data quality management," *Communications of the ACM*, vol. 2, no. 41, pp. 58–65, 1998.

[6] W. M. P. van der Aalst, A. H. M. ter Hofstede, and M. Weske, "Business process management: A survey," in *Business Process Management*, ser. Lecture Notes in Computer Science, W. M. P. van der Aalst, A. H. M. ter Hofstede, and M. Weske, Eds., vol. 2678. Springer, 2003, pp. 1–12.

[7] M. T. Gómez López, L. Parody, R. Martínez Gasca, and S. Rinderle-Ma, "Prognosing the compliance of declarative business processes using event trace robustness," in *On the Move to Meaningful Internet Systems: OTM 2014 Conferences - Confederated International Conferences: CoopIS, and ODBASE 2014, Amantea, Italy, October 27-31, 2014, Proceedings*, 2014, pp. 327–344.

[8] C. Cappiello, A. Caro, A. Rodríguez, and I. Caballero, "An approach to design business processes addressing data quality issues," in *ECIS*, 2013, p. 216.

[9] L. Pipino, Y. Lee, and R. Wang, "Data quality assessment," *Communications of the ACM*, vol. 45, no. 4, pp. 211–218, Apr. 2002. [Online]. Available: http://doi.acm.org/10.1145/505248.506010

[10] A. Rodríguez, A. Caro, C. Cappiello, and I. Caballero, "A bpmn extension for including data quality requirements in business process modeling," in *BPMN*, 2012, pp. 116–125.

[11] OMG, "Business Process Model and Notation (BPMN), Version 2.0," Object Management Group Standard, 2011.

[12] P. B. Crosby, *Quality is free*. McGraw-Hill, 1979.

[13] ISO, "Iso/iec 25012: Software engineering-software product quality requirements and evaluation (square)-data quality model," 2008.

[14] A. Caro, A. Rodríguez, C. Cappiello, and I. Caballero, "Designing business processes able to satisfy data quality requirements," in *ICIQ*, 2012, pp. 31–45.

[15] ISO, "ISO/DIS 8000-100: Master Data: Exchange of characteristic data: Overview," ISO, 2011.

[16] ——, "ISO/DIS 8000-130: Master Data: Exchange of characteristic data: Accuracy," ISO, 2011.

[17] Princeton University. (2014) WordNet: A lexical database for English. Accedded on the 24th of February of 2014. [Online]. Available: http://wordnet.princeton.edu/

[18] I. Community, "Intalio," http://www.intalio.com/, 2012.

[19] A. Team, "Activiti BPM Platform," http://www.activiti.org/, 2012, accedded on the 24th of February of 2014.

[20] B. Community, "Bonita Open Solution," http://www.bonitasoft.org//, 2012, accedded on the 24th of February of 2014.

[21] I. Caballero, I. Bermejo, L. Parody, M. T. Gómez López, R. Martínez Gasca, and M. Piattini, "I8k: An implementation of iso 8000-1x0," in *ICIQ*, 2013, pp. 356 – 370.

[22] ISO, "ISO/DIS 8000-140: Master Data: Exchange of characteristic data: Completeness," ISO, 2011.

[23] P. Benson and M. Hildebrand, "Managing blind: A data quality and data governance vade mecum," Bethlehem (Pensylvania): ECCMA, 2012.

[24] S. Jablonski and C. Bussler, *Workflow management - modeling concepts, architecture and implementation.* International Thomson, 1996.

[25] M. T. Gómez López and R. Martínez Gasca, "Run-time auditing for business processes data using constraints," in *Business Process Management Workshops*, 2010, pp. 146–157.

[26] M. T. Gómez López, R. Martínez Gasca, and J. M. Pérez-Álvarez, "Compliance validation and diagnosis of business data constraints in business processes at runtime," *Inf. Syst.*, vol. 48, pp. 26–43, 2015.

[27] M. T. Gómez-López and R. Martínez Gasca, "Fault diagnosis in databases for business processes," in *21st International Workshop on Principles of Diagnosis, 2010*, 2010.

[28] M. Heravizadeh, J. Mendling, and M. Rosemann, "Dimensions of business processes quality (qobp)," in *Business Process Management Workshops*, ser. Lecture Notes in Business Information Processing. Springer Berlin Heidelberg, 2008, vol. 17, pp. 80–91.

[29] C. Cappiello and B. Pernici, "A methodology for information quality management in self-healing web services," in *ICIQ*, 2006, pp. 18–29.

[30] M. H. Ofner, B. Otto, and H. sterle, "Integrating a data quality perspective into business process management," *Business Process Management Journal*, vol. 18, no. 6, pp. 1036–1067, 2012.

[31] OMG, "Decision Model and Notation," Object Management Group Standard, 2015.