View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by AIS Electronic Library (AISeL)

Association for Information Systems
AIS Electronic Library (AISeL)

UK Academy for Information Systems Conference Proceedings 2010

UK Academy for Information Systems

Spring 3-23-2010

MONITORING DATA PRODUCT QUALITY

Markus Helfert
Dublin City University, Markus.Helfert@computing.dcu.ie

Juan Yao
Dublin City University, jyao@computing.dcu.ie

Follow this and additional works at: http://aisel.aisnet.org/ukais2010

Recommended Citation
Helfert, Markus and Yao, Juan, "MONITORING DATA PRODUCT QUALITY" (2010). *UK Academy for Information Systems Conference Proceedings 2010*. 26.
http://aisel.aisnet.org/ukais2010/26

This material is brought to you by the UK Academy for Information Systems at AIS Electronic Library (AISeL). It has been accepted for inclusion in UK Academy for Information Systems Conference Proceedings 2010 by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

MONITORING DATA PRODUCT QUALITY

-OUTLINING A FRAMEWORK AND DESCRIPTION APPROACH-

Markus Helfert

Dublin City University, Dublin 9, Ireland

Markus.Helfert@computing.dcu.ie

Phone: +353-1-700-8727 Fax: +353-1-700-5442

Fakir Mohammad Zakir Hossain

Dublin City University, Dublin 9, Ireland

Fakir.Hossain@computing.dcu.ie

Abstract

The importance of data quality has been considered for many years and is well recognized among practitioners and researchers. A great deal of work has been done and most of the work to date fall under two main categories. One group of scientists has focused on mathematical and statistical model to work at the database layer to introduce constrain based mechanism to prevent data quality problems. Another group has focused on the management of the process of data generation. While the body of knowledge in the area is vast, the practical application of these approaches is still limited. One particular area which is still rarely considered in improving data quality is the development cycle of information system. Recognising this limitation and aiming to provide a practical-orient approach, we take a process centric view, and focus on preventing deficiencies during the IS design. In this paper we propose a process centric framework for data quality monitoring.

Keywords: Information Quality, Quality Monitoring, Information Manufacturing

1. Introduction

One of the challenges in the advancement in the data quality area is that most of the knowledge is quite hard to implement in a typical small to midsize Information Systems (IS) development. They are quite technical in nature or cost and time prohibitive. Also, most of the researches suggest a strategy that is often quite disjoint from a typical IS development cycle which makes it difficult to adopt. Of the three elements of IS, namely, data, process and rules, last one is often neglected (Kovacic, 2004). Several approaches had been adopted to link data models and business processes (Nelson, Rariden, & Sen, 2008) (Vasilecas & Smaizys, 2006) (Muehlen, Indulska, & Kamp, 2007) (Khan, Kapurubandara, & Chadha, 2004). But most of the approaches have failed to provide an integrated environment for modelling quality business rules linking business process and data models. Often systems are designed without sufficient model to define quality or ensuring its ongoing conformance. Motivated by this challenge, in order to ensure quality of data, we suggest to model quality rules in such a way that data in the IS confirms with its quality requirements throughout its lifecycle. In this paper we follow an approach proposed by (Ballou 1998) and describe an approach to modelling Data Quality Blocks.

To demonstrate various issues addressed in the paper, we illustrate our work in the context of a Hotel Reservation System. In our sample reservation system, booking must be made in the future, i.e. the arrival date must be after the reservation date. Once a booking is made, hotel is notified immediately of the booking. At this stage, the booking will be provisional. However, once the payment is received, the booking will become confirmed and a further email will be sent to the client confirming the booking. Eventually if the booking was cancelled, two cancellation emails will be sent, one to the client and one to the hotel.

The remainder of this paper is structured as follows: In section 2 we summarize related literature. In section 3 we propose a data quality monitoring framework, which will be incorporated in section 4 into the IS development lifecycle. In the final section we evaluate and conclude on our findings. We also highlight the limitations of our work and give some direction for further research.

2. Related Literature

Significant amount of progress has been made aiming to improve data quality. Every aspect of the development of IS has come under scrutiny to improve data quality. Researchers have focused at the design and modelling of IS, database layer, coding and implementation standards, user training and responsibility, ongoing data quality monitoring, etc.

Some of the earliest work to address data quality focused at the database layer of IS. Researchers have focused at the database layer to prevent data inconsistency and corruption by introducing data constrains (static and dynamic), transaction management and other measures (Brock, 2000) (Vianu, 1983) (McCune & Henschen, 1989). While data quality related problem has been reduced, it still remains a significant issue (Wang, Kon, & Madnick, 1993).

Researchers from Business and Management background followed the database community in the 80's to focus on how to control the data in the IS to improve the quality situation (Scannapieco, Missier, & Batini, 2005). In contrast to traditional approaches, Information Quality (IQ) researchers proposed a novel perspective on IS and regarded IS as information manufacturing system (IMS) (Wang 1998). He argued "*To increase productivity, organizations must manage information as they manage products*." In order to treat information as a product, understating consumer's need, well defined production process, establishing the Total Data Quality Management (TDQM) lifecycle and appoint of an IP manager is considered to be essential (Wang, Lee, Pipino & Strong, 1998).

Problem of defining, measuring and improving data quality became more prominent by the computer scientists in the beginning of '90s (Scannapieco, Missier, & Batini, 2005). What data quality means must be understood in order to manage it. Traditionally data quality has been broken down into various quality dimensions that represent a single aspect of the quality. Various approaches have been taken into defining the quality dimensions (Wang & Strong, 1996). The most elaborated study was undertaken by Wang and Strong using this approach (Wang & Strong, 1996). Not just the dimensions of

quality, but understanding what quality dimensions are important to the user of a given IS is fundamental to the design of the Quality Block (Wang, Kon, & Madnick, 1993).

Modelling the IS plays a vital role in quality of data. Modelling must describe all information in relation to IS in accurate and consistence manner (Pham, Helfert, & Duncan, 2007). Various approaches have been adopted over the years to model IS (Ballou, Wang, Pazer, & Tayi, 1998) (Ballou & Pazer, 1985) (Shankaranarayanan, Wang, & Ziad, 2000). Waterfall methods, IASDO, agile are just a few to name.

Following the IMS approach, IQ researchers introduced a model to systematically track aspects of data product quality facilitated by information manufacturing analysis matrix which was developed by Ballou (Ballou, Wang, Pazer, & Tayi, 1998). They represented IS by various manufacturing blocks as such as Data Vendor Block, Processing Block, Data Storage Block, Quality Block and Customer Block. This model was further expanded by introducing Decision Block, Business Boundary Block and Organization Boundary Block by Shankaranarayanan (Shankaranarayanan, Wang, & Ziad, 2000). While establishing various block of the IMS had been very helpful, detail design methodology has not be set out for each block.

Various methods have already attempted to describe IS. Information Product (IP) Map aims to identify data requirement and then model data and process together to ensure conformance. Input, output and description of process are addressed by Data Flow Diagram (DFD). But it lacks the ability to describe the organizational aspect of IS. Event Driven Process Chain (EPC) is also useful but fails to model interrelation of all constructs (Pham, Helfert, & Duncan, 2007). The challenges with the models are that they fail to accurately and completely contain sufficient quality related information required by IS to conform to required quality.

3. Introducing the Data Quality Monitoring Framework (DQMF)

Objective of data quality monitoring framework is to develop a comprehensive monitoring system that is independent from the information manufacturing system. The key is to develop a process independent monitoring system that will continuously monitor data to ensure various aspect of data quality. In our example described above, if

confirmed bookings could be continuously monitored to ensure notification to hotels, this problem could be detected much earlier and rectified with no impact to client.

Building form the discussion above and addressing some of the limitations, in our effort to model a data quality block, we emphasis on monitoring as an aspect of ensuring quality. Lack of ongoing monitoring is also a contributor to lack of data quality. A breakdown in IS process or inappropriate use of the IS result in inconsistent data which are not usually discovered until at a much later date.

In the context of our example, if an email failed to be sent from our sample reservation system to the hotel confirming the booking, client might show up at the hotel without actually a room secured for the client. Without effective ongoing monitoring, this will only come to light after the client has arrived at the hotel. Quality block of IS should be self enforcing quality complier. However, an IS independent quality conformance monitor would naturally generate far better result. Developing a parallel system just to monitor data can also be time and cost prohibitive. Our aim in modelling quality block is also to develop data quality rules in such a way so that it can be feed to an independent data quality monitor.

The framework consists of three core components. Data Quality Monitor (DQM), Data Product Markup Language (DPML) and Information Quality Markup Language. First we will introduce each element briefly. We will use the example of a hotel reservation system to elaborate on the components. Finally, we will describe how it works all together to develop the framework.

3.1. Data Quality Monitor (DQM)

The principle of the DQM is based on the model proposed by Shankaranarayanan (Shankaranarayanan, Wang, & Ziad, 2000). The data quality monitor is an application that accepts data product quality rules as its input and continuously monitors data product to ensure that it meets the agreed quality as defined. When designing the quality block, IP MAP/Business Process Modelling Notation (BPMN) can be supplemented by metadata about each manufacturing block.

Objective of the DQM is not to intervene in the process, but merely to monitor the data products to see if meets the quality requirement of the product relevant to the stage of its production. If the product fails to meet the requirement, it will report the inconsistency in accordance with agreed protocol to facilitate immediate intervention for corrective measures.

3.2. Data Product Markup Language (DPML)

A key element of our DQMF is DPML. In order to be effective quality controller, Information System models must describe sufficiently and accurately static, dynamic and organizational aspect of IS.

In a traditional manufacturing assembly line, as a product reaches various stages of its development, it can be inspected to ensure that it has met the requirement to be achieved at the relevant stage of the production. This is possible because a product in traditional sense will be predefined to achieve certain quality criteria that will be developed as part of designing the product.

For our framework to work, we treat data as a product of information manufacturing system. At the design phase, we must then define the quality criteria that a data must meet at various stages of its production. In order to achieve this objective, we developed a Data Product Markup Language as an IP Unified Modelling Language (UML) based data product definition language. By using UML we can build on previous work to create visualized mapping of the data processes (Ballou, Wang, Pazer, & Tayi, 1998). Furthermore, UML/BPMN is widely accepted because it can be exported to code directly by cutting down on development time. This was further developed by IPMAP which extended a systematic method of representing the processes involved in manufacturing of IP (Shankaranarayanan, Wang, & Ziad, 2000). Flow of data at various stages is also visualized by IPMAP. However, it lacks the ability to bridge various process and information product (Pham, Helfert, & Duncan, 2007). There is also a need to, as described in the next section, to export the quality rules for automated execution. Hence we also base DPML on BPMN. We extend this model to model an integrated approach to define data quality requirements and business processes together.

In order to demonstrate the application of DPML we refer back to our example. Let us assume that BOOKING is a data product that will be produced by the hotel reservation system. One of our assumptions, independent of its state, reservation must be made before the guest's arrival date. This particular aspect of the BOOKING can be described as below in Figure 1.

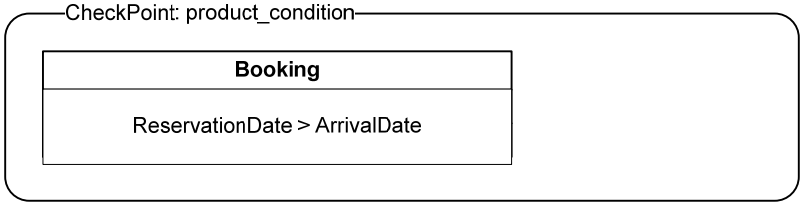

Figure 1: DPML for Booking Reservation Date Condition

Outer border is more relevant to the framework and for the DQM to know what to inspect. But the inner part captures the quality (accuracy) criteria the product, BOOKING, must meet at all times. In this manner a condition that the product must meet can be defined.

Let's examine a second aspect of our example and consider the BOOKING product as it passes through various stages in production. We can easily track and record all emails sent for a given booking. Let's assume that all of these emails are also stored in the database in a table called MailTracker. This stage based product criteria can be described in the figure 2.

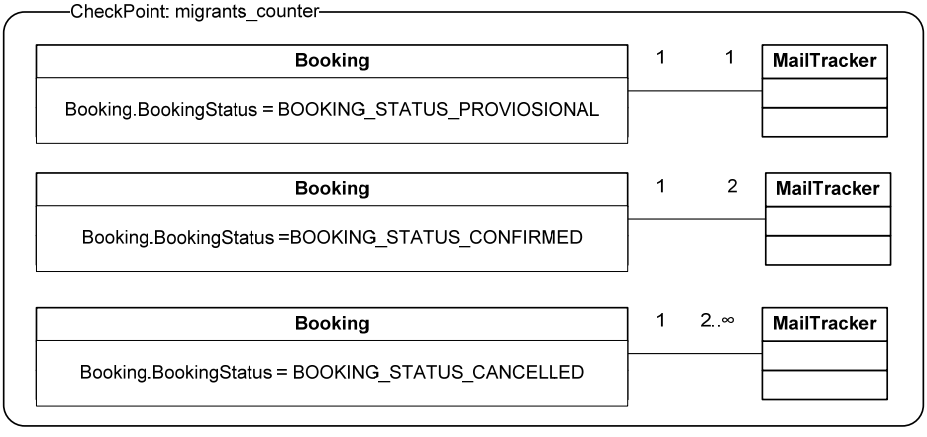

Figure 2: DPML for Booking Email Business Rules

3.3. Information Quality Markup Language (IQML)

Once we are able to model data product using DPML, as described above, we need to translate it into an executable code that can be processed by automated software. Otherwise, for each system a separate monitoring tool have to be developed. This is likely to make it cost and time prohibitive. This is why there needs to be the ability to convert this DPML into and XML based rules that can be accepted by the monitoring tool.

Information Quality Markup Language (IQML) is an XML based data product definition language. The purpose and nature of IQML is identical to that of DPML. Difference is that while DPML is UML based, IQML is XML based. IQML is either auto generated from DPML or generated independent of it. It is merely a means to facilitate data product definitions to be consumed by the Data Quality Monitor.

We will revisit the two examples we used to describe DPML. We will represent the same data product definition using IQML. IQML equivalent of representing the rule about arrival date and reservation date could be represented as below in figure 3.

```
<dq:quatily_check_point>
    <dq:check_type>product_condition</dq:check_type>
    <dq:quality_dimension>accuracy</dq:quality_dimension>
    <dq:condition_test>Booking.ReservationDate lt
Booking.ArrivalDate</dq:condition_test>
  </dq:quatily_check_point>
 </dq:data_product>
```

Figure 3: IQML for Booking Reservation Date Condition

The rule about the email confirmations at various stages of BOOKING can be represented in IQML as below in figure 4:

```
<dq:quatily_check_point>
    <dq:check_type>migrants_counter</dq:check_type>
    <dq:quality_dimension>Completeness</dq:quality_dimension>
    <dq:foreign_table>MailTracker</dq:foreign_table>
    <dq:chose>
        <dq:when test="Booking.BookingStatusID eq
BOOKING_STATUS_PROVISIONAL">
            <dq:number_of_migrants>1</dq:number_of_migrants>
        </dq:when>
        <dq:when test="Booking.BookingStatusID gt
BOOKING_STATUS_CONFIRMED">
            <dq:number_of_migrants>2</dq:number_of_migrants>
        </dq:when>
        <dq:when test="Booking.BookingStatusID eq
BOOKING_STATUS_CANCELLED">
            <dq:number_of_migrants>gt 2</dq:number_of_migrants>
        </dq:when>
    </dq:chose>
</dq:quatily_check_point>
```

Figure 4: IQML for Booking Email Business Rules

The relationship between DPML, IQML and DQM is pictured in figure 5.

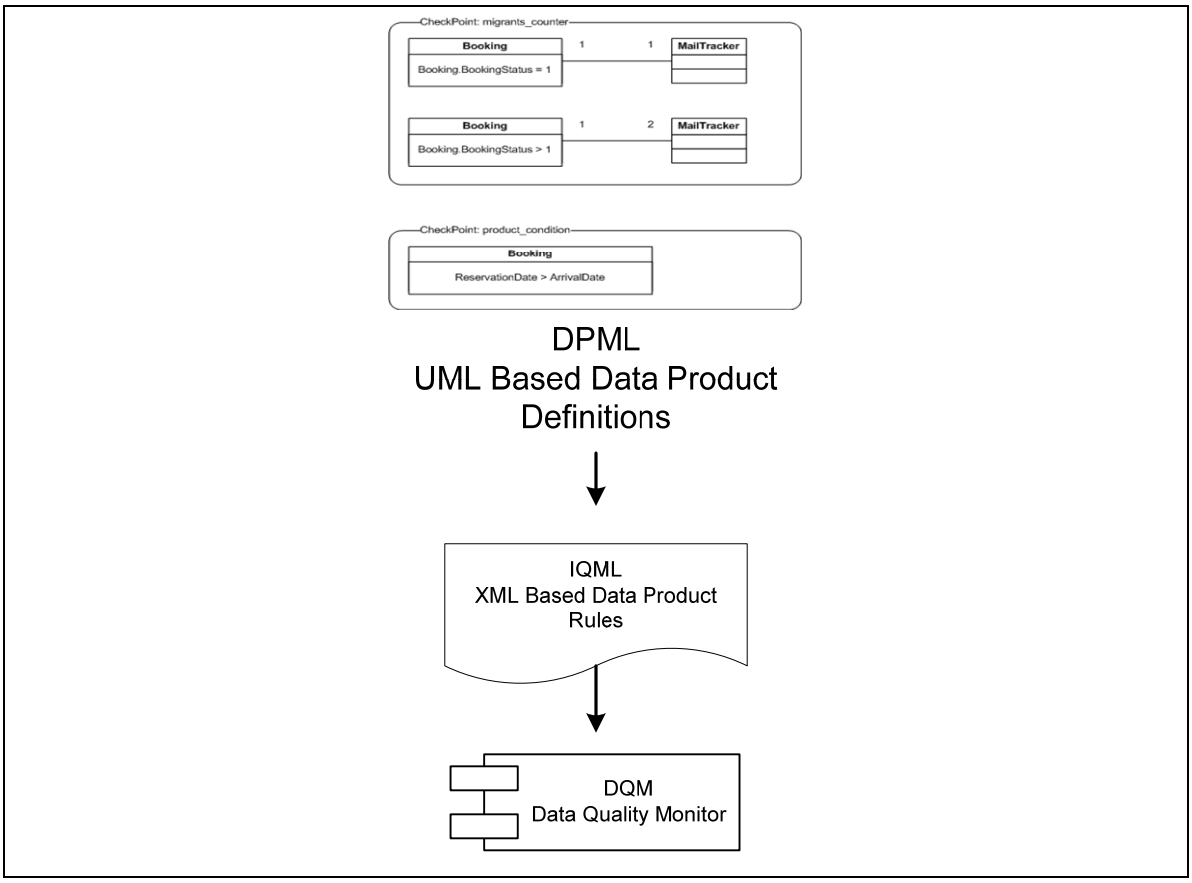

Figure 5: Quality Rules Development Process

4. Practical Integration into Development Framework

As discussed previously, various frameworks for data quality monitoring and measuring have already been developed. However, as discussed above, despite the large number of research one of main drawback is the limited practical application or adoption for most of them. The framework proposed in this paper aims to achieve an integrated framework for quality aspect of IS that can be easily integrated in day to day practice. In a traditional system development cycle, represented below in figure 6, business requirement is documented using UML. A database schema is developed based on the requirements.

Businesses processed are then materialized in an application which work in connection with a data base backend developed based on the schema developed earlier.

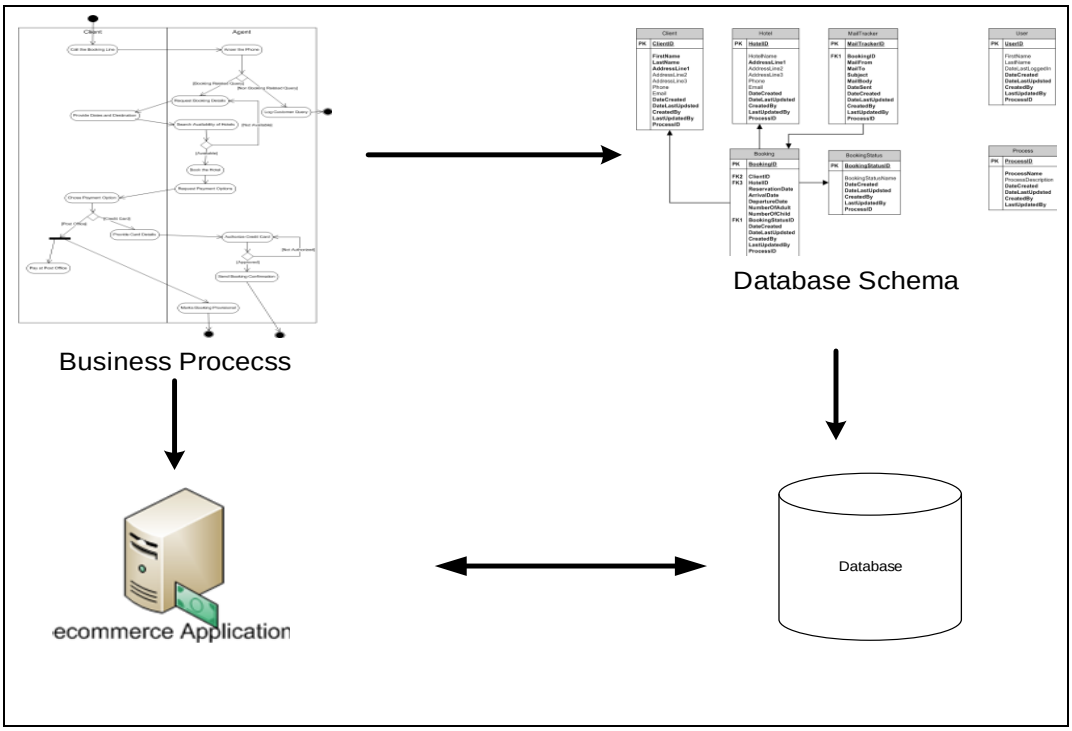

Figure 6: Traditional System Development Cycle

In this new approach, as shown in figure 7 below, we are proposing that along with business process requirements, data products are also defined by using DPML. This DPML then can be converted to IQML and feed to the DQM to monitor the data products. This is practical because no additional software is needed to be developed for the monitoring purpose. Any subsequent change to the business requirement can also be reflected by making change in the data product Meta model into the DQML and the change will automatically be picked up by the DQM via IQML. Here is how the proposed framework might be represented.

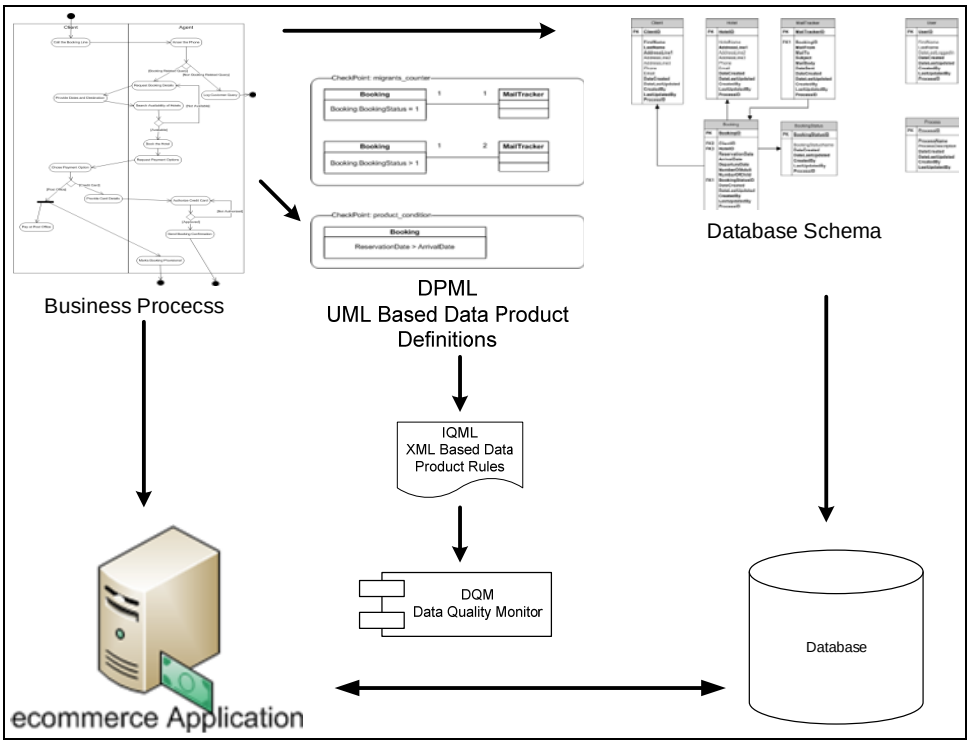

Figure 7: Quality Development Lifecycle

5. Evaluation and Conclusion

Despite best efforts of IS developers, any system is viable to have errors in the system and thus can produce data products of low quality. Errors are often realized at a much later stage at a high cost to the business. One of the main benefits of our DQM approach is that it will not only detect the deficiencies in the data product at its earliest stage, it will also identify the process responsible for the issue. This will expedite the corrective measure. It will also minimize the damage to the business.

Another benefit of the DQM is that it can be used as a separate testing mechanism independent of the information manufacturing process. It can be used initially when the IMS is developed to ensure that the data products produced by the system confirms with quality requirements specification of the product design. Monitor can also determine if any subsequent change made to the product design has followed through to the IMS.

A summary of our approach has been compared with other leading approaches in Table 1.

Method	**Pros & Cons**
BPMN	• Process Centric • Lack data product aspect • Lack data quality measurement elements
IP Map	• Lack integrated business process mapping • Focus on data product and quality assurance aspects
DPML and IQML	Superimposes IP Mapping over BPMN to create a unique data/process modelling environment allowing • Process modelling • Data product modelling • IP quality definition & measurement modelling • Model dynamic constrains • Incorporated within IS development lifecycle • Executable automated monitoring

Table 1: Evaluation of various DQ modelling approaches

Some of the limitation of this approach is that this will be difficult to apply on existing systems, since they may not have sufficient process required for the framework defined. Adopting this will also require a particular mindset to cater for quality related information to be stored. For the universal DQM to work, IS will have to be developed in particular predefined confinements which might limit the flexibility IS developers require. A solution to this approach, once an organizational level coding standard is adopted, a DQM can be customized for the specific organization. More research is needed to developing a comprehensive meta model for the quality and other IS blocks to offer the benefit of the DQM, yet offering reasonable flexibility to the IS engineers. Often the literature in data quality fields is too complicated or abstract that it can hardly be used in everyday development. We expect the major contribution to be the practical aspect of the DQM.

References:

Ballou, D. P., & Pazer, H. L. (1985). Modeling Data and Process Quality in Multi-Input, Multi-Output Information Systems. *Management Science , 31* (2).

Ballou, D. P., Wang, R. Y., Pazer, H., & Tayi, G. K. (1998). Modeling Information Manufacturing Systems to Determine Information Product Quality. *Management Science, 44* (4).

Brock, E. d. (2000). A general treatment of dynamic integrity constraints. *Data & Knowledge Engineering* (32), 223-246.

Khan, K. M., Kapurubandara, M., & Chadha, U. (2004). Incorporating business requirements and constraints in database conceptual models. *Conferences in Research and Practice in Information Technology Series*, 59 - 64.

Kovacic, A. (2004). Business renovation: business rules (still) the missing link. *Business Process Management* , 158-170.

McCUNE, W. W., & HENSCHEN, L. J. (1989). Maintaining State Constraints in Relational Databases: A Proof Theoretic Basis. *Journal of the Association for Computing Machinery* , 46-68.

Muehlen, M. z., Indulska, M., & Kamp, G. (2007). Business process and business rule modeling languages for compliance management: a representational analysis. *ACM International Conference Proceeding*, 127-132.

Nelson, M. L., Rariden, R. L., & Sen, R. (2008). A Lifecycle Approach toward Business Rules Management. *41st Hawaii International Conference on System Sciences.*

Scannapieco, M., Missier, P., & Batini, C. (2005). Data Quality at a Glance. *Datenbank-Spektrum* , 6-14.

Shankaranarayanan, G., Wang, R. Y., & Ziad, M. (2000). M. IP-Map: Representing the manufacture of an information product. *Proceedings of the 2000 Conference on Information Quality.*

Pham, T. T., Helfert, M., & Duncan, H. (2007). The IASDO Model for Information Manufacturing System Modelling. *International Journal of Information Quality* , 5-21.

Vasilecas, O., & Smaizys, A. (2006). The framework: an approach to support business rule based data analysis. *International Baltic Conference*, (pp. 141 - 147).

Vianu, V. (1983, September). Dynamic Constraints and Database Evolution. *ACM* , 389-399.

Wang, R. W., & Strong, D. M. (1996). Beyond Accuracy: What Data Quality Means to Data Consumers. *Journal of Management Information Systems , 12* (4), 5-33.

Wang, R. Y. (1998). A Product Perspective on Total Data Quality Management. *Communications of the ACM , 41* (2).

Wang, R. Y., Kon, H. B., & Madnick, S. E. (1993). Data Quality Requirement Analysis and Modeling. *Ninth International Conference of Data Engineering* .

Wang, R. Y., Lee, Y. W., Pipino, L. L., & Strong, D. M. (1998). Manage Your Information as a Product. *Sloan Management Review* .