Analysis on the Value of Process Support Implementations for Quality Management

Michael Seitz[1] and Stefan Jablonski[2]
[1]PRODATO Integration Technology GmbH, Erlangen, Germany
[2]University of Bayreuth, Chair of Applied Computer Science IV, Bayreuth, Germany
michael.seitz@prodato.de, stefan.jablonski@uni-bayreuth.de

Keywords: Quality Management, Process Support, Value Creation, Maturity Models, Adaptation.

Abstract: Many organizations face competitive pressure to enhance their business process capabilities and to comply with quality management directives. Methods like the Business Process Maturity Model (BPMM) therefore provide assistance by stating concrete requirements and measuring their fulfillment. However, there is still uncertainty about how much technical support (e.g. guidance, documentation) is actually reasonable in order to adapt the business process to a specific maturity level. In this paper an analysis approach is introduced that enables to assess the value of process support implementations for quality management and assists practitioners with the decision-making whether the value is appropriate for the use case. The application of the concept is illustrated using the example of four representative implementations ranging from manual human-controlled to automatic system-controlled support.

1 INTRODUCTION

In order to remain competitive, many organizations continually enhance and optimize their business process maturity. Maturity models like the Business Process Maturity Model (BPMM) (OMG, 2008) therefore serve as orientation as they "have been designed to assess the maturity (i.e. competency, capability) of a selected domain" (Bruin et al., 2005) and derive points for improvements. Maturity levels (ML) describe concrete requirements, e.g., the orientation towards a reference process, but do not provide indication of supporting methods or tools to enact and execute a reference process model. The issue of adequate technical support for process enactment that is managed on basis of the quality requirements of maturity models is rather complex: Reasonable support has to be adapted to current as well as constantly changing requirements of the deployed quality management (QM) standard, i.e. conditions for qualitatively appropriate execution of business processes (for example deliver in time with consistent performance). Moreover, "the proper fit between the tasks in the business processes and information technology/systems must exist" (Trkman, 2010). Since many companies have adopted a value-based approach to manage the deployment of information technology (IT) and decisions on IT are made on basis of the contribution to strategic business objectives (Mauch and Wildemann, 2007), the question arises how the value of process support implementations for QM can be determined and serve as decision guidance. It is obvious: the more tool support is chosen, the more expensive process execution becomes. However, it has to be asked whether this additional tool support has an adequate value with respect to QM. In these matters, decision makers need to rely on more than just "gut feeling".

There already exist some in-depth investigations on the ability of IT for support of specific processes and use cases. Zur Mühlen and Rosemann (2000) outline the economic aspects of workflow-based process monitoring and controlling. Faerber (2010) conceptually designs a process navigation system as implementation for process-oriented QM. Becker et al. (1999) present a structured framework which enables the evaluation of the potential of business processes to be supported by workflow management systems. The listed related work in each case is concentrated on a specific implementation approach. There is still a lack of a general approach considering the whole spectrum of process support including both human- and system-controlled approaches. The key issue about the reasonable degree of process support, i.e. which tasks should be

Seitz M. and Jablonski S.
Analysis on the Value of Process Support Implementations for Quality Management.
DOI: 10.5220/0004775201770186
In *Proceedings of the Third International Symposium on Business Modeling and Software Design (BMSD 2013)*, pages 177-186
ISBN: 978-989-8565-56-3

accomplished fully/semi automated or rather manually, is not yet thoroughly assessed. It is still unclear how the actual value of process support for QM can be measured, independent of the degree of IT assignment and specific implementation approaches.

In this paper the value of process support for QM is investigated. The term "value" should be perceived as the contribution that is made to being compliant to requirements of QM standards. In practical terms, it is evaluated to what extent process management tasks – such as information, coordination or documentation – no longer have to be carried out manually but are handled in whole or part by process support tools. The value is measured by the appropriate fulfillment of maturity level requirements on process support.

Therefore, an adequate analysis approach is introduced and illustrated by the example of four different representative implementations for process support. The objective of the analysis is to get an appreciation of to what extent the implementations penetrate the process – in particular with regard to the degree of IT assignment and the remaining degree of freedom for the users – and thereby support the users in performing their tasks in accordance with requirements of QM standards. The analysis results are intended to provide practitioners a better basis for decision-making about a broad spectrum of suitable implementations.

In our previous work (Seitz and Jablonski, 2012; Seitz and Jablonski, 2013) we approach the problem of adequate process support from a business point of view without directly referring to QM. In this paper, the IT side is analyzed, especially as regards the selection of specific implementations that comply best with the demanded capabilities. Therefore, in addition to Jablonski (2010) an even more specific classification of process support is performed with respect to each process perspective (e.g. data, organization). The concept is tightly focused on QM so that conclusions can be drawn – on the one hand for the value, i.e. the promoted quality at runtime (guidance during execution) and after runtime (evidence through documentation) and on the other hand for the caused costs (e.g. modeling efforts). In this manner, we want to give a more precise answer to the question of adequate process support.

2 APPROACH

The concept is divided up into three steps. Firstly, the classification framework for the degree of process support according to Seitz and Jablonski (2012, p. 95) is introduced and applied in the context of QM support functions. Secondly, the evaluation instrument for setting the benchmark is presented. Maturity levels thereby serve as measure how valuable process support is. Thirdly, a procedure is suggested that aims for reaching a decision on the question which is the most valuable approach for process support with respect to a specific maturity, i.e. requirements of QM standards.

2.1 Classification

Process support for QM comprises four basic functions (Faerber, 2010, p. 75): information provision, data integration, coordination, and documentation. Depending on the required ML these basic functions can be implemented quite differently. For example, if the goal is to reach a high maturity, it is recommended to coordinate work packages and project staff accurately. Relevant control information should be integrated electronically to be able to collect and analyze key performance indicators systematically. However, a low maturity just demands to achieve the results (anyhow) and therefore allows for the coordination function to be performed rather rudimentarily or not at all. It may be also sufficient to retrieve process instructions or measurement data by hand.

Seitz and Jablonski (2012) introduce an adequate framework for the classification of the degree of process support that is based on the perspective-oriented process model (POPM) (Jablonski, 1994). The five main perspectives of POPM according to Jablonski and Goetz (2008) are: functions (process steps and their purpose), data (used data, e.g. documents, and data flow between the process steps), operations (invoked services and tools), organization (people or machines and their responsibilities) and behavior (control flow). The functional perspective thereby represents the composition ("skeletal structure") of the process on which the other perspectives are built on. This is way the functional perspectives can be excluded from the classification. The framework covers the whole spectrum for both internal and external enactment of process models (under vs. beyond the control of information systems) as well as the range between strict and flexible execution (little or no freedom vs. high degree of freedom and decision making by the users). In the following, the characteristics (perspectives) and the values of the framework are explained using the four basic functions for QM support (see Figure 1).

Information Provision: Users are provided with detailed information across all perspectives about the process steps to be performed, like some kind of handbook or guideline. The more detail of information is supplied the tighter the process execution is restricted. However, this support function is completely separated from the actual process execution; there we call it "external enactment" because it is limited to a passive role by indeed presenting the users all relevant facts of the process but not being able to influence the actual process execution or even the user’s behavior. Regarding the operational perspective, directives on mandatory tools can be set. Concerning the organizational perspective, responsibilities are defined, either at the level of a group of persons acting with a common purpose – maybe a role or department (“non-agent”) – or at the level of individuals (“agent”). The behavioral perspective is covered by a clear textual or visual description of the chronological sequence of the process steps and their dependencies.

Data Integration: A distinction is made between unstructured (e.g. an image) and structured (e.g. a form or a record) data. They are consolidated from different sources in order to make them centrally available in electronic form, e.g. for presentation purpose (data perspective). In doing so, it is also possible to establish application interfaces and to make 3rd party tools more accessible (operational perspective). Those can either be suggested to the users for manual execution – possibly through a launch pad (assisted enactment) – or automatically be invoked and parameterized (automatic execution). The latter option provides less flexibility and is often preceded by a costly development and deployment.

Coordination: Project staff and work packages have to be reconciled and harmonized taking into account restrictions and due dates. Hence, the organizational (task assignment) and the behavioral perspective (temporal and logical sequence of process steps) is concerned. Process support for the coordination function is considered to be system-integrated, because adequate implementations must keep track of the actual course and are in need of feedback about the current process context (internal enactment). It can be performed either flexible or strict: Assigning a task to a role or department provides more flexibility than to a specific person or server process. Accordingly, there arise far more possibilities from a suggested set of suitable process steps than from exactly prescribing the execution order.

Documentation: The compliance with quality requirements has to be proved through documentation. This can be achieved either manually by the users or automatically through IT. The actual process execution is documented paper-based or electronically on different level of detail for each process perspective. For instance, in many cases it is sufficient to simply record that a certain process step has been accomplished (e.g. by presenting the process results), whereas often also additional information such as the executing agent or the applied tools have to be documented.

Process support through information provision (external enactment) is complemented or rather substituted on the one hand by data integration regarding data and operations and on the other hand by coordination regarding organization and behavior (internal enactment). While external enactment just enables to communicate the way the process should be executed, internal enactment also ensures that it will actually be done (within the granted degree of freedom). In the following it is assumed that both data integration and coordination include information provision with regard to the respective process perspectives. Furthermore, with internal enactment by data integration and coordination also the documentation of the related process perspectives is covered more or less automatically.

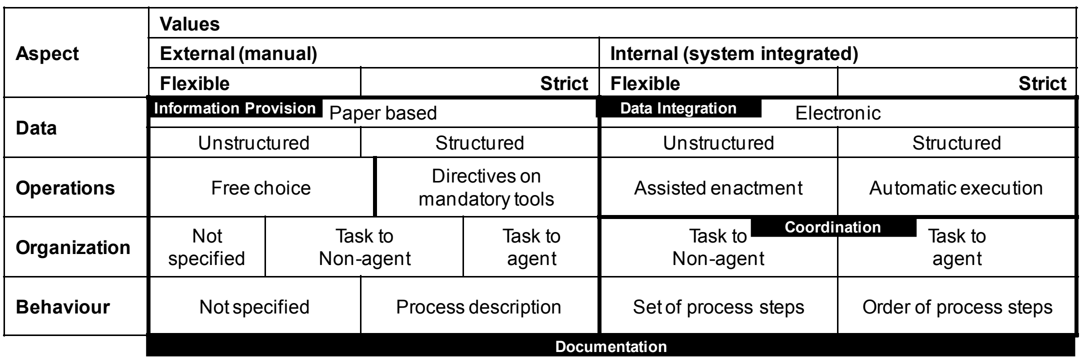

Figure 1: Classification framework.

This simplified example for a single process step of the travel expense report shows how support functions and their interdependencies work:

Information Provision: Via the enterprise intranet it is communicated that the traveler (organization) must make the reimbursement application (data) within three days after returning from the travel (behavior) and send to the travel department by email (operation).

Data Integration: The document template can be downloaded by clicking on the provided link (data). A word processing application is recommended for editing (operation).

Coordination: As soon as the travel is approved (behavior) the traveler is given the task to perform the travel expense report (organization).

Documentation: By means of the email containing the reimbursement document the process step is – more or less automatically – documented. It proves who applied which reimbursement application and when (all perspectives).

Some higher QM standards demand to take corrective actions in case of exceptions and moreover to incorporate improvements in future instances of the process model. In this framework this aspect is not be dealt with explicitly. Even though the investigated support functions the concept is based on do not directly cover continuous improvement they include the collection of relevant data (data integration) and the delegation of further measures (coordination) for improvement and innovation to human agents or 3rd party tools that provide process support through assistance and planning according to Jablonski (1994).

The contribution of the classification framework to the research question is to determine to what extent the QM support functions are implemented. In order to establish a common scale for the evaluation of which value a certain implementation thereby actually creates, in the next section, the quality requirements of the MLs are mapped to the support functions.

2.2 Evaluation

Maturity models "are used as an evaluative and comparative basis for improvement" (Bruin et al. 2005) and therefore are suitable to establish a common scale for the evaluation of the value proposition. The Business Process Maturity Model (BPMM) is chosen for this evaluation because it is focused on all kinds of processes of an organization (Hogrebe and Nüttgens, 2009). In the following, based on Seitz and Jablonski (2012) and Schönig et al. (2012), the quality requirements of business process MLs on the previously introduced support functions are investigated. In turn, it can be determined which ML implementation approaches are able to reach for each support function, and to what extent additional support by other tools or by hand is required.

Initial: ML1 just demands to achieve the process results. It does not place any specific requirements on process support. In this respect all implementation approaches meet ML1 (by definition).

Managed: ML2 demands proper results in time. Therefore, the process has to be planned like a project. In order to set up and perform a project schedule, information about organization and behavior have to be provided and the execution has to be coordinated (either flexibly or strictly). Besides the achieved results it must be documented that the project plan was adhered to (organization and behavior).

Standardized: ML3 demands that the process execution follows a reference process. Therefore, all information about the reference process have to be specified across all perspectives. Document templates and input screens (if needed) should be made available centrally and applications or tools to be used should be suggested or prescribed (flexible or strict data integration). Similar to ML2, adequate coordination is necessary. All relevant process perspectives have to be documented properly to provide evidence for being compliant to the reference process.

Predictable: ML4 additionally demands measurable results. Therefore, data for the defined key performance indicators (KPI) have to be collected systematically. In this respect control data, result data and – if needed – data from external sources should be integrated and made available in electronic and structured form for documentation purpose and further statistical analysis (strict data integration and strict documentation of data and operations). Furthermore, corrective action is demanded in case of KPI exceptions. This requirement is covered through the support functions coordination and documentation and can be modeled through definition of respective controlling tasks and appropriate recording of recognized deviations and taken countermeasures.

Innovating: While corrections and improvements in ML4 only affect the currently running instance, ML5 demands (automatic) continuous improvement and innovation of the reference process und future instances. Within the scope of the support functions this paper deals with there are no further requirements through ML5.

The ML requirements on process support are summarized in Table 1. They can be used as a common scale for the evaluation of the value proposition of implementations. In the next section, it is outlined how to decide which implementation is the most appropriate to a particular situation.

2.3 Decision

In this section, a procedure for reaching a decision and finally selecting an adequate implementation for process support is suggested.

The principle objectives for the decision about adequate process support stated by Seitz and Jablonski (2012) therefore indicate a general direction. On the one hand, process support must guide the attainment of the process strategy, e.g. to reach a specific ML. On the other hand, implementations are in need of a process model that is defined properly and completely. Following the principle of utility maximization and cost minimization, the implementation approach must be chosen that fits best the desired process maturity (nothing more and nothing less) and simultaneously requires the least modeling effort and gains the broadest possible acceptance by the users. So the decision process may look as follows:

As a first step – assuming the demanded ML is set by management or other stakeholders – it must be decided to which ML the process support is adapted. Thereby, it is differentiated between the promoted quality at runtime (through information provision, data integration and coordination) and the proven quality afterwards (through adequate documentation). As a rule, it is quite useful to make sure that the aspired ML is properly documented. For example, if ML3 is demanded by the customer, all relevant process perspectives for ML3 should be traceable and comprehensible in the requisite degree of detail. However, at runtime, under certain conditions it may be sufficient to promote a lower quality in favor of lower costs and higher execution flexibility. Depending on the granted freedom, quality is just supported, rather covered or even enforced. It should be worked out to what extent "undefined paths" should be secured or should remain flexible to create necessary space for creativity. For example, although ML3 is demanded process execution is only supported according to ML1 or ML2, because the participants are in need of a certain creative freedom and their scope of actions must not be restricted through standardization. The decision also depends on the risk for process errors and their consequences. Consequently, the tighter process execution must be secured the more the demanded ML should be adapted. Documentation should always be compliant to the demanded ML.

The second step involves the assessment of possible implementation approaches with respect to their supported MLs for each QM support function. Therefore, each approach is first classified based on the framework outlined in Section 2.1 (see also Figure 1) in order to determine "how much" process support is delivered. Then, based on this classification, the approaches are evaluated according to the ML requirements on process support discussed in Section 2.2 (see also Table 1) to find out which ML can be reached.

The highest benefit is realized when information provision, data integration, coordination and documentation are implemented at the best with the required ML. The third step will therefore be to limit possible implementation approaches to the ones

Table 1: Process support requirements of the maturity levels.

	Information Provision	Data Integration	Coordination	Documentation
ML1	None	None	None	None
ML2	Organization and Behavior	None	Organization and Behavior	Organization and Behavior
ML3	All perspectives of the reference process	Data and Operations	Organization and Behavior	All perspectives of the reference process
ML4	All perspectives of the reference process	Data and Operations (strict)	Organization and Behavior	All perspectives of the reference process (thereof Data and Operations strict)
ML5	All perspectives of the reference process	Data and Operations (strict)	Organization and Behavior	All perspectives of the reference process (thereof Data and Operations strict)

providing the best “fit” with both the required runtime and documentation quality.

In a fourth step, the costs are evaluated for each selectable implementation. It is appropriate to reflect costs by the efforts for process model engineering. Those can be measured in view of two dimensions: One major factor the costs depend on is how complete the process model must be. The degree of completeness is related to the scope, especially which process perspectives must be modeled, and the detail for each perspective (e.g. task assignment to non-agents vs. agents). Another major factor is the needed degree of formalization, in particular what proportion of the process model must be interpretable by technical means with regard to system-controlled execution. Furthermore, modeling costs may vary according to the used modeling paradigm. While “for an imperative model, every possible path must be foreseen […] and encoded explicitly”, “in declarative modeling, on the contrary, only undesired paths and constellations are excluded so that all remaining paths are potentially allowed and do not have to be [defined] individually” (Schönig et al., 2012). Finally, extraordinary costs have to be considered when the implementation does not entirely promote the desired maturity and some support functions have to be carried out manually in order to be compliant to the desired ML nevertheless. In turn, there probably are also unnecessary costs when the implementation “overfulfills” the desired maturity. To sum up, on the one hand, there are investigated modeling costs that rise with the increasing demand of completeness (strictness of execution) and the growing degree of formalization (use of IT), and, on the other hand, follow-up costs for insufficient or exceeding support due to deviations from the target maturity.

In the last step, the cost-benefit-ratio is determined and the implementation promising the most reasonable ratio is chosen. Thereby, it can be considered if it is useful to take a loss of quality fit in favor of lower costs. For example, in contrast to WfMS a paper-based support tool indeed does not reach high maturity runtime support but can be implemented at significantly lower costs.

3 ILLUSTRATION

In the following, the application of the concept presented in the previous chapter is illustrated. Firstly, existing implementations which are generally accepted and recognized as enactment approaches for process management as described by Schönig et al. (2012) are introduced and classified according to their degree of process support. Secondly, it is evaluated which quality requirements they promote. Finally, the decision-making is illustrated using the example of the CL as potential implementation for standardized process execution according to ML3.

3.1 Classification

In the following, different implementation approaches are described and classified according to the previously introduced framework in Section 2.1. The selected approaches can be considered to be representative, because according to Jablonski (2010) they cover the whole spectrum of process usage from human-controlled to system-controlled (degree of IT assignment) and from flexible to strict execution support (degree of freedom).

Wallpaper (WP): The WP approach provides various possibilities to present processes visually and to depict compressed information (Information Provision) with both low (flexible) and high (strict) detail. It uses the process model “as it is, e.g. printed out as wallpaper, outlined on a flip chart or published online as process graphic in wiki” (Seitz and Jablonski, 2013). It is one strength of the WP to outline the process flow precisely and to strictly state the behavior, even though “the process itself happens completely offline” (external enactment). This is why data integration, coordination and documentation are not supported.

Checklist (CL): “A checklist comprises the main process steps including documents that must be produced and agents that are responsible to perform the corresponding process” (Jablonski, 2010). With the process steps being serialized the process behavior can only be specified roughly by the arrangement so that the actual execution order remains flexible. Depending on its implementation suitable or mandatory tools can be additionally stated. Similar to the wallpaper approach the checklist is enacted externally and therefore cannot support data integration or coordination. The documentation support is designed to collect the executing agents (organization), the sequence of execution (behavior, e.g. via timestamps) and optionally a statement of used applications (operations).

Process Navigation System (PNS): This approach is intended to support flexible, human-centric processes. It suggests suitable actions and tools and refers to restrictions, but never enforces them (Schönig et al., 2012). Hence, the PNS supports flexible coordination by recommending a set of process steps (flexible behavior), normally in interaction with human agents (strict organization). Standardized data interfaces and automatic execution of 3rd party tools are not intended, whereas adequate document and application links are contextually provided (flexible data integration). It is therefore perceived as a decision support system. The feedback of the users about the actual process flow is made available electronically as structured data and utilized to document the order of the executed process steps (strict behavior), the performing agent (strict organization) and the usage of applications and tools (flexible operations).

Workflow Management System (WfMS): Traditional WfMS strictly execute the specified workflow logic and thereby communicate with human users and IT applications (Schönig et al., 2012). Due to rigid runtime control functions this approach can be classified as strict coordination support. Documents, databases and 3rd party tools can be connected via pre-defined interfaces (strict data integration). “Most WfMS log data on cases and tasks executed” (van der Aalst, 2004). Besides (unstructured) result data like documents there are also collected (structured) control data about the interaction with human users and external systems that can be used as documentation and further analysis. For this reason, documentation support of WfMS covers all perspectives in high detail (strict).

The classification of the implementation approaches for each support function is summarized in Figure 2, Figure 3, Figure 4 and Figure 5. The figures show the relevant detail from the classification framework for each support function (see Figure 1). On the X axis the support function is drilled down on the associated process perspectives. The Y axis differentiates between flexible and strict execution. Furthermore, the enactment type (external and/or internal) is labeled. The figures illustrate “how much” process support is delivered by the investigated approaches. In the next section it is shown how this classification combined with the ML requirements on process support is used to determine which MLs the approaches are able to reach.

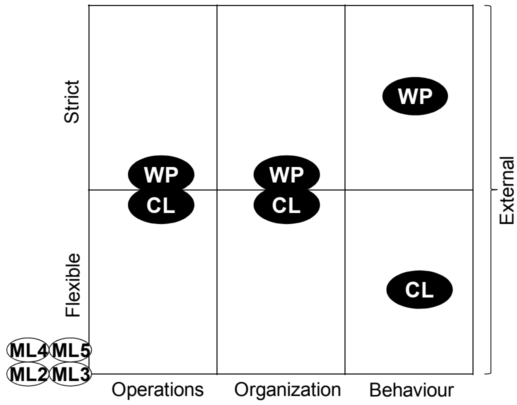

Figure 2: Information Provision.

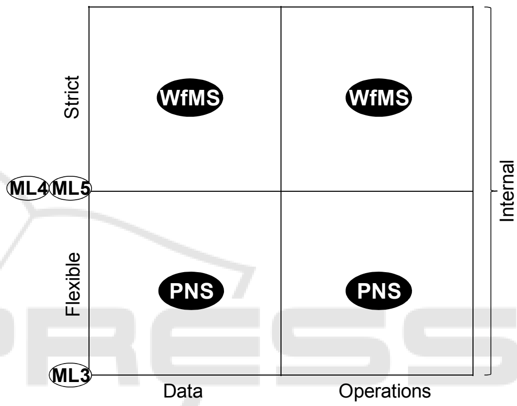

Figure 3: Data Integration.

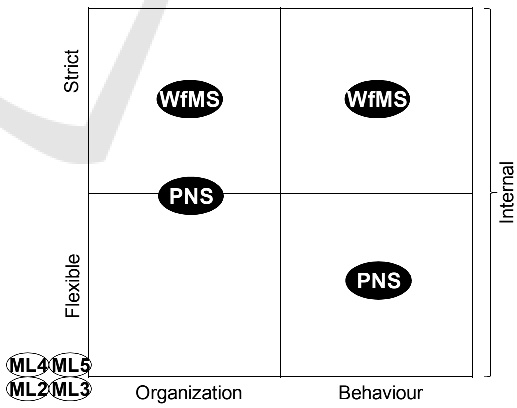

Figure 4: Coordination.

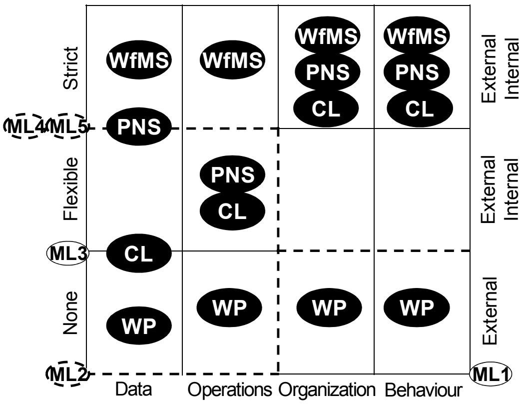

Figure 5: Documentation.

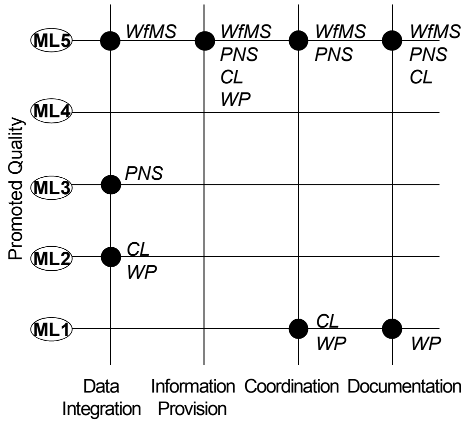

Figure 6: Promoted Quality (Value).

3.2 Evaluation

In Figure 2, Figure 3, Figure 4 and Figure 5 the MLs are located as discussed in Section 2.2 and summarized in Table 1. An implementation approach is considered to reach a specific ML for a particular support function if it is classified above the ML line.

Principally, all implementation approaches fulfill ML1. As for ML2, the deployment of the CL and the WP should be accompanied by appropriate coordination (e.g. manually by a project leader).

Both the WfMS and the PNS approach support ML3 per se, while as for the CL and the WP – along with coordination – there is also a lack of central and standardized access to process data and 3rd party tools.

ML4 is only covered by WfMS, whereas PNS could be extended by adequate interfaces in order to not only establish ML4 through coordination and documentation but also promote ML4 quality through enabling access to all required data source.

In Figure 6 the evaluation is summarized. It shows the assignments of support function and ML for each implementation approach. ML1 is reached by each approach. ML2 is fulfilled by WfMS, PNS and CL (CL in case of additional coordination support only). ML3 is implemented by WfMS and PNS. ML4 and ML5 are only reached by WfMS.

Whether the value is actually appropriate for a specific process case depends on the desired ML of the process and the cost-benefit ratio of the possible implementation approaches. An exemplary decision is outlined in the next section.

3.3 Decision

In the preceding section the value proposition of the implementation approaches was discussed, in particular how the support functions for QM are fulfilled and which MLs can be reached. Now, in this section, it is illustrated how the evaluation results can be used in order to decide whether an approach is appropriate to a particular situation. The evaluation matrix in Figure 6 serves as decision support for the identification of adequate process support for a specific ML. Therefore, the deviations (both gaps and exceedings) of the accomplished maturity in comparison with the required maturity are analyzed, as to whether they result in additional costs. An example for ML3 and the CL approach is depicted in Figure 7. There are following deviations:

Gap 1 arises from the lack of a central provision of document templates according to the reference model. One solution would be to place the CL items directly into the header of the resulting documents. The templates could be published electronically as download in the enterprise intranet or printed out and handed over as paper-based forms. Gap 2 is due to the missing coordination function of the CL approach. It can be closed by appointing a project manager that allocates tasks according to the CL and monitors that the process flow ranges in the course of the reference model. As a consequence, there arise additional costs for the provision of the document template and project management in order to “secure” process execution against mistakes.

Exceeding 1 and Exceeding 2 can be compensated by adequate configuration of the CL. Information should only be stated and documented if it is actually needed in order to meet ML3. For example, if the reference process does not prescribe a specific order of process steps it is not worth to inquire date and time of finished tasks when filling in the CL. Hence, unnecessary costs can be avoided.

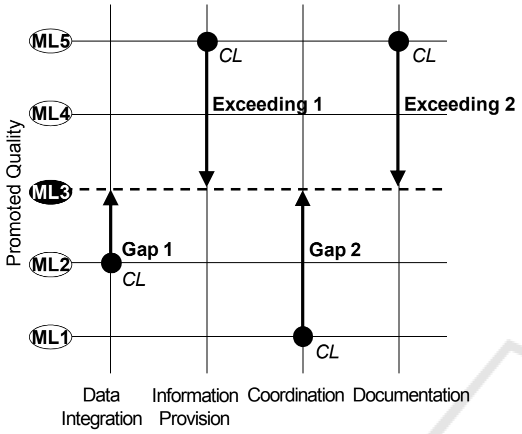

Figure 7: Gap/Exceeding Analysis (Example).

In summary, it can be seen that the CL approach – provided that coordination support at runtime can be neglected for the respective use case or substituted by manual project management – is indeed appropriate for ML3, as it enables sufficient flexibility in modeling information provision and documentation support.

4 CONCLUSION AND OUTLOOK

In this paper an approach for the analysis on the value of process support implementations for QM was introduced. On the basis of a classification framework it was shown how to determine the degree of process support differentiating between execution and documentation assistance (i.e. information provision, data integration, coordination and documentation). With the help of the BPMM requirements on process support a comparative basis for the selection of adequate implementations was created and applied to the Wallpaper, Checklist, Process Navigation System and Workflow Management System, which are four representative implementations. Finally, a procedure for decision-making was suggested and some cost aspects were discussed. The illustration of the concept and the decision revealed that the Checklist is definitely appropriate to support process execution and documentation according to ML3 (standardized).

The analysis approach can be enhanced in several respects. Besides modeling efforts also "soft" factors like user acceptance and general conditions like political constraints can be additionally considered for the cost assessment. Furthermore, it could be deliberated about whether the scope of supportive functions should be extended by, e.g., incorporation of improvements into the running process (in terms of exception handling and individual optimization) and into the reference model (in terms of adaptation and global optimization). Currently, the analysis approach is designed for a snapshot. Future work could be concentrated on how to take a more dynamic, prospective view on the evaluation (e.g. long-term planned maturity).

REFERENCES

Becker, J., Uthmann, C. v., Zur Mühlen, M., Rosemann, M. (1999). *Identifying the Workflow Potential of Business Processes*. In Proceedings of the 32nd Hawaii International Conference on System Sciences.

Bruin, T. de, Rosemann, M., Freeze, R., Kulkarni, U. (2005). *Understanding the Main Phases of Developing a Maturity Assessment Model*. In Proceedings of the 16th Australasian Conference on Information Systems (ACIS), Sydney.

Faerber, M. (2010). *Prozessorientiertes Qualitätsmanagement. Ein Konzept zur Implementierung*. Wiesbaden: Gabler.

Hogrebe, F., Nüttgens, M. (2009). *Business Process Maturity Model (BPMM): Konzeption, Anwendung und Nutzenpotenziale*. HMD – Praxis der Wirtschaftsinformatik, 266, 17-25.

Jablonski, S. (1994). *MOBILE: A Modular Workflow Model and Architecture*. In Proceedings of the 4th International Working Conference on Dynamic Modeling and Information Systems, Noordwijkerhout.

Jablonski, S. (2010). *Do We Really Know How to Support Processes? Considerations and Reconstruction.* In Engels, G., Lewerentz, C., Schäfer, W., Schürr, A., Westfechtel, B. (Eds.), Graph Transformations and Model-Driven Engineering. Essays Dedicated to Manfred Nagl on the Occasion of his 65th Birthday (pp. 393-410). Berlin/Heidelberg: Springer.

Jablonski, S., Goetz, M. (2008). *Perspective Oriented Business Process Visualization*. Business Process Management Workshops – BPM 2007 International Workshops, BPI, BPD, CBP, ProHealth, RefMod, semantics4ws, Brisbane, Australia, September 24, 2007, Revised Selected Papers. Lecture Notes in Computer Science (Vol. 4928, pp. 144-155).

Mauch, C., Wildemann, H. (2007). *Wettbewerbsfaktor IT.*

Wege zur erfolgreichen IT-Gestaltung; Ergebnisse einer empirischen Untersuchung. München: TCW, Transfer-Centrum.
Object Management Group (OMG) (2008). *Business Process Maturity Model (BPMM), Version 1.0 (2008)*. Retrieved from: http://www.omg.org/spec/BPMM/1.0/ [17 March 2013].
Schönig, S., Seitz, M., Piesche, C., Zeising, M., Jablonski, S. (2012). *Process Observation as Support for Evolutionary Process Engineering*. International Journal on Advances in Systems and Measurements, 5(4), 188-202.
Seitz, M., Jablonski, S. (2012). *Evolutionäres Prozess-Engineering: Zum angemessenen Grad an Prozessunterstützung*. HMD – Praxis der Wirtschaftsinformatik, 287, 93-102.
Seitz, M., Jablonski, S. (2013). *Evolutionary Process Engineering: User Guide and Case Study for Adequate Process Support*. In Proceedings of the 3rd International Conference on Business Intelligence and Technology, Valencia. (forthcoming)
Trkman, P. (2010). *The critical success factors of business process management*. International Journal of Information Management, 30(2), 125-134.
van der Aalst, W.M.P. (2004). *Business Process Management: A Personal View*. Business Process Management Journal, 10(2), 135-139.
Zur Mühlen, M., Rosemann, M. (2000). *Workflow-based Process Monitoring and Controlling – Technical and Organizational Issues*. In Proceedings of the 33rd Hawaii International Conference on System Sciences.