

## Multi-Paradigm Process Management

Michael zur Muehlen<sup>1</sup>, Michael Rosemann<sup>2</sup>

<sup>1</sup>Stevens Institute of Technology  
Wesley J. Howe School of Technology Management  
Castle Point on the Hudson  
Hoboken, NJ 07030, USA  
mzurmuehlen@stevens.edu

<sup>2</sup>Centre for Information Technology Innovation  
Queensland University of Technology  
126 Margaret Street  
Brisbane, QLD 4000, Australia  
m.rosemann@qut.edu.au

**Abstract.** Automation and integration of business processes are at the heart of contemporary enterprise systems. In the pursuit of this goal, process automation technology is employed at varying levels of the enterprise information systems architecture. Larger organizations are faced with multiple instances of process management systems, each of which may provide a different paradigm for capturing, representing and executing processes. Still, each of these systems provides unique process support functionality that may not be covered by the other systems. Current systems integration methods focus mainly on the technical connectivity between these disparate systems. They do not address the integration of multiple process modeling methods that may exist in enterprise applications. This paper discusses the method integration necessary to bridge the gap between the high-level process models and the executable workflow definitions. We propose a structured methodology for the systematic design of enterprise processes that takes advantage of the capabilities of different modeling methods, while maintaining a consistent view of enterprise processes across multiple platforms. Using such an approach, business analysts and system engineers can follow a stepwise procedure that will minimize overlap and redundancy in enterprise processes, and maximize integration potential between applications. We call this approach Multi-Paradigm Process Management (MP<sup>2</sup>M).

### 1 Motivation

Process management and automation tools are becoming increasingly specialized. The co-existence of process management systems thus becomes corporate reality, as the different solutions provide specialized support for different types of business processes and/or application scenarios. At the technical level the interoperability between these different tools has to be maintained by system engineers. Standardization efforts in this area have yielded mixed results so far, but the advance of web service standards may soon provide a well-defined platform for application

integration. At the organizational level, however, the integration of process models stemming from the different process management systems remains an unsolved issue. Management applications, such as Business Activity Monitoring applications, rely on the ability of enterprise solutions to provide a consistent and unified view of all processes in an organization.

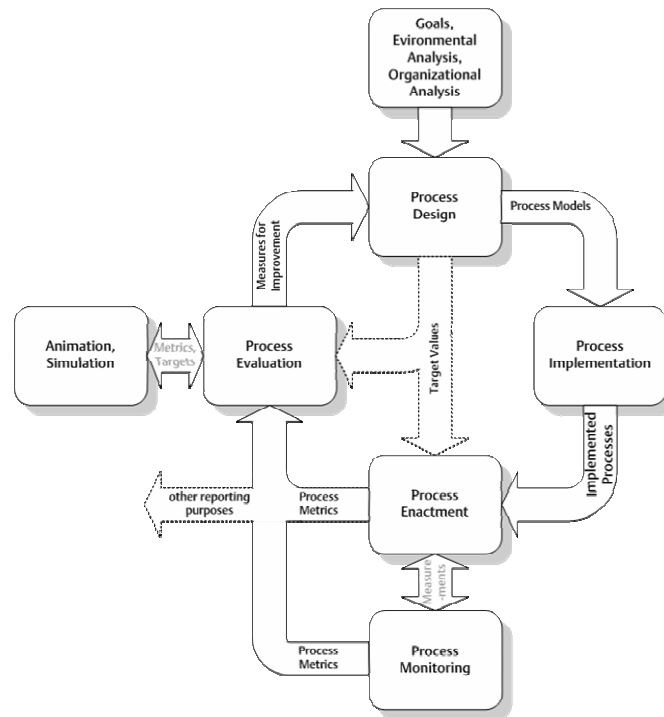
## **2 Process Modeling in the Process Management Life Cycle**

### **2.1 The Process Management Life Cycle**

The development and deployment of process models in an organization follows a life cycle of design, implementation, enactment and evaluation [1, 2]. Figure 1 shows a typical process life cycle (from [3]). After an initial analysis phase, processes are designed using high level modeling languages. They are then transformed into executable workflow specifications (should they lend themselves to automation), and these specifications serve as templates for individual process instances that are coordinated by process automation tools during the process enactment phase. Process monitoring supervises runtime operation and allows for the correction of exceptions during process enactment. Event trails of completed process instances serve as the basis for ex-post analysis in the process controlling phase. The insights gained in this phase serve as guidelines for a revision of the high level models as another iteration of the process management life cycle begins.

The image of the process management life cycle is idealized in large parts. In fact, the process management life cycle is broken in many locations. Some examples for gaps in the life cycle are the following:

- Different modeling methods are employed in the process design and process implementation stages. While process design is supported by high-level modeling languages like Event-driven Process Chains, IDEF0 or UML activity diagrams, most execution platforms rely on proprietary process representations or XML process specification formats such as XPDL [4] or BPEL4WS [5]. The translation of models between these representation languages is prone to information loss and semantic ambiguities. In effect, some considerations that led to particular design time models may be lost in the implementation models.
- For the automation of a given business process a multitude of process automation platforms are available. While some of these are separate software components, others are embedded components of large-scale application systems. Each of these platforms has specific strengths and weaknesses that affect the suitability for individual process types. To date, no formal evaluation method has been developed that can help users determine which execution platform is best suited for a given process.



**Fig. 1.** Process Management Life Cycle

- Most audit trail data formats contain only technical information about completed process instances, such as processing and wait time, involved resources and frequencies. What is missing is a link to business information such as the business objects that were being worked on in the course of individual process instances. The creation of this link is crucial for the generation of meaningful evaluations in the process analysis phase.
- The correlation between the insights of an ex-post process review and measures for process improvement is largely unknown. This is partly due to the fact that few enterprises have closed this part of the process management life cycle. Another cause for this gap is the number of external factors that influence process performance. The structure of organizations evolves with a different speed than the processes of an organization. Maintaining a fit between these two factors requires constant refinement of the process models.

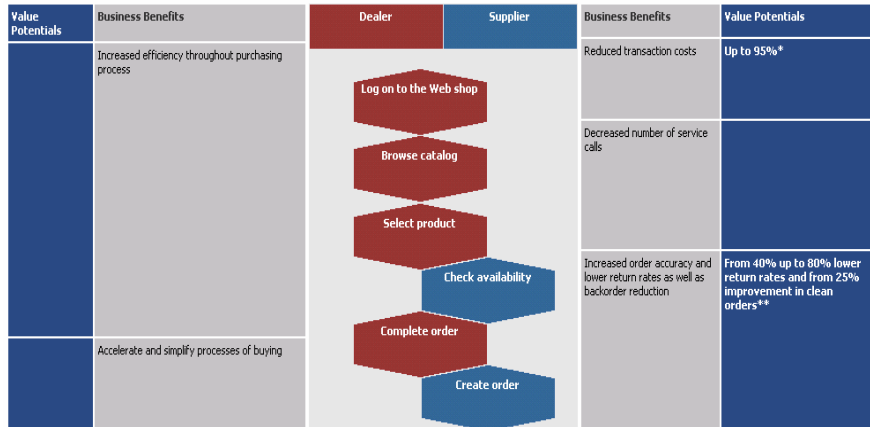
## 2.2 High-Level Process Support

Large-scale information systems, such as enterprise resource planning systems (ERP), provide users with support for high-level processes. They support the customization of system functionality through the tailoring of reference processes and guide users through the customization process using high-level process representations (compare figure 2).

zur Muehlen, M.; Rosemann, M.: Multi-Paradigm Process Management. In: Proceedings of CAiSE'04 Workshops - 5th Workshop on Business Process Modeling, Development and Support (BPMDs 2004). Eds.: Janis Grundspenkis, Marite Kirikova, Riga, Latvia, 2004, pp. 169-175.

**Automotive: CRM Internet Sales for Suppliers (Collaborative)**

This business scenario is designed to show how two business partners - a customer and a vendor - use the Internet to buy respectively sell products or services. The map illustrates the benefits of collaboration. The result is a streamlined Internet Sales process in which both parties save time and money.



**Fig. 2.** High-level Process Definition Tool

Figure 2 shows a collaborative process model of an Internet sales process as it can be found in an ERP solution. Other examples of high-level process models are those found in the MIT process handbook [6]. These processes are rarely deployed as is, but in most cases they are adjusted to meet the context of the organization implementing the enterprise system. One of the factors that can determine the feasible degree of customization is the responsiveness of an organization to change. In a fortune 100 pharmaceutical enterprise, an ERP solution was introduced in the regional offices in Mexico and Europe. While the solution in Mexico was deployed as-is, and workers had to adjust to the changed business processes, the solution in Europe was heavily customized, so existing work habits need not be changed. One project manager explained this difference with the resistance of the European work force to changes in their processes.

**2.3 Execution-level Process Support**

Business process automation and workflow management systems provide technical support for the automated coordination of business processes at the execution level. For this purpose, high-level process models need to be enhanced with runtime-relevant information, such as interface specifications of invoked application systems, rules for the assignment of activities to resources, and transactional boundaries to ensure recoverability of the workflows should a system failure occur. The specification of processes at this level of abstraction has little to do with the high-level process models described in the previous section. In fact, many contemporary workflow specification languages do not have a graphical notation, but are designed as XML schemas in order to be machine readable and interpretable (examples are XPDL, BPML, and BPEL4WS). A notation for these processes has been proposed in form of the Business Process Modeling Notation (BPMN) [7], but the mapping of this

notation to individual process grammars is not yet finalized. While BPMN is based on the BPEL meta model, the mapping of this meta model to the BPEL meta model (or XPDL, for example), has not yet been demonstrated.

While the level of detail necessary to specify an executable process in many cases is beyond the capabilities of a high-level process design, certain measures can be taken during the process design phase to ensure an easier transformation of the high-level models into executable specifications. These measures include:

- The capturing of workflow-relevant data (i.e. data that determines the control flow of individual process instances at decision points in the process) and the sources of this data in the design phase
- Determining task assignment policies and documenting them in the process model
- Determining communication and autonomy requirements of process participants
- Determining the overall suitability of a process for automation purposes. Not all processes are likely candidates for automation, due to a lack of structure, frequency, or IT support.

#### **2.4 Specifying the Gap between high- and execution-level Models**

There clearly exists a gap between the high-level process models used in the early stages of the process management life cycle, and the detailed execution models of the implementation and enactment stages. This gap manifests itself in the following points:

- Lack of an appropriate language that covers both high-level process design and low-level process execution.
- Lack of appropriate conversion between languages of different stages of the process management life cycle.
- Lack of guidance for the conversion of high-level process models into low-level executable models. This guidance includes support for the selection of workflow-suitable processes; advice, which attributes need to be maintained in order to ease the transition to an executable model, and general advice on the capabilities of different automation platforms.

### **3 A Framework for Multi-Paradigm Process Management**

We argue that process research focuses on well defined phases of the process management life cycle (e.g. design or implementation). What is lacking is a detailed analysis of the transition between these phases. Figure 3 shows a framework for such an analysis. We distinguish between three different levels of process management. At the Selection and Configuration level, high-level enterprise models, reference process models and solution maps are used by business analysts to create an enterprise-specific configuration of the more general process models that are supplied with current enterprise systems.

These models are then transferred into an enterprise-wide process model management layer. At this level, business analysts and process managers receive guidance in choosing the right implementation platform for individual process

components. These alternatives can be either of the system-to-system, people-to-system, or people-to-people workflow solutions, an external automation system, or the manual execution of process parts (if no automation is desired or feasible).

Through export filters, relevant parts of the overall process model are exported for further refinement in the chosen execution platforms. At the implementation level, workflow engineers can refine the process fragments, while at the same time they are able to view the overall context of the process fragment they are working on.

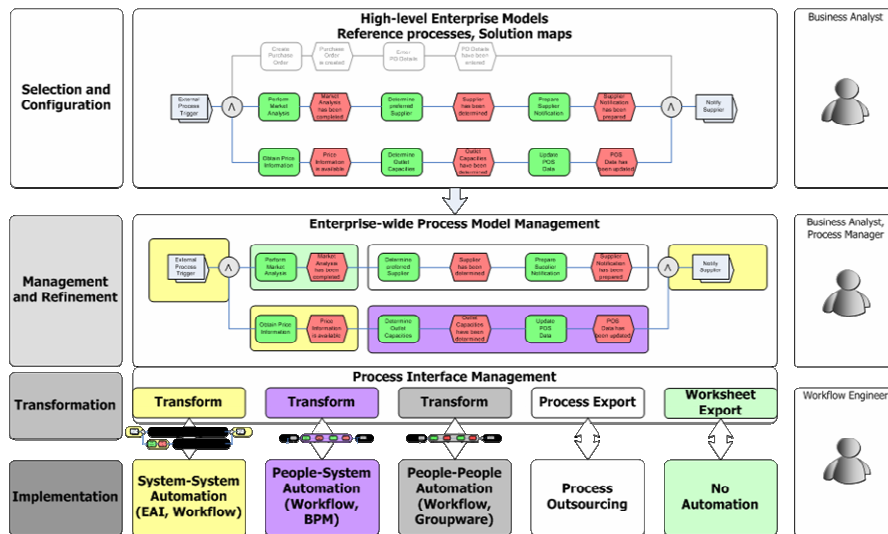


Fig. 3. Hierarchical Levels of Process Management Support

The top-down transformation of models is not the only way the proposed framework can be used. Changes at the execution level require a propagation of these changes to the higher levels of the framework, in order to keep design and implementation models in sync.

## 4 Summary

We have discussed gaps in the process management life cycle that stem from the transition between design and implementation phases in process management projects. The lack of integrated modeling languages, transformation rules, and the missing mapping of process support tool capabilities to process model properties make the automation of business processes a time-consuming and error-prone endeavor. We have presented a framework that can serve as a guideline for future research to ease the transition between design and implementation phase. Our future work focuses on providing a mapping between high-level process design models, and low-level process execution models, so business analysts can capture information that is relevant for workflow engineers.

zur Muehlen, M.; Rosemann, M.: Multi-Paradigm Process Management. In: Proceedings of CAiSE'04 Workshops - 5th Workshop on Business Process Modeling, Development and Support (BPMDS 2004). Eds.: Janis Grundspenkis, Marite Kirikova, Riga, Latvia, 2004, pp. 169-175.

## References

- [1] Heilmann, H.: Die Integration der Aufbauorganisation in Workflow-Management-Systeme. In: H. Heilmann, L. J. Heinrich, and F. Roithmayr, (eds.): Information Engineering. Oldenbourg (1996) 147-165
- [2] Galler, J., Scheer, A.-W.: Workflow-Projekte: Vom Geschäftsprozeßmodell zur unternehmensspezifischen Workflow-Anwendung. Information Management 10 (1995) 20-27
- [3] zur Muehlen, M.: Workflow-based Process Controlling. Foundation, Design, and Implementation of Workflow-driven Process Information Systems. Logos, Berlin (2004)
- [4] WfMC: Workflow Process Definition Interface - XML Process Definition Language. Document Status - 1.0 Final Draft. Workflow Management Coalition, Document Number WFMC-TC-1025 Lighthouse Point (FL) (2002)
- [5] Curbera, F., Goland, Y., Klein, J., Leymann, F., Roller, D., Thatte, S., Weerawarana, S.: Business Process Execution Language for Web Services, Version 1.0. BEA, IBM, Microsoft, (2002)
- [6] Malone, T. W., Crowston, K., Lee, J., Pentland, B.: Tools for Inventing Organizations: Toward A Handbook of Organizational Processes. In: Proc. of the 2nd IEEE Workshop on Enabling Technologies Infrastructure for Collaborative Enterprises (1993)
- [7] White, S.: Business Process Modeling Notation. BPMI.org, Working Draft (1.0) San Jose, CA (2003)