Extending BPM Environments of Your Choice with Performance Related Decision Support

Mathias Fritzsche[1], Michael Picht[2], Wasif Gilani[1], Ivor Spence[3], John Brown[3], and Peter Kilpatrick[3]

[1] SAP Research CEC Belfast, United Kingdom
mathias.fritzsche@sap.com, wasif.gilani@sap.com
[2] SAP Product & Technology Unit Suite Core, Germany
michael.picht@sap.com
[3] Queen's University Belfast, United Kingdom
i.spence@qub.ac.uk, tj.Brown@qub.ac.uk, p.kilpatrick@qub.ac.uk

Abstract. What-if Simulations have been identified as one solution for business performance related decision support. Such support is especially useful in cases where it can be automatically generated out of Business Process Management (BPM) Environments from the existing business process models and performance parameters monitored from the executed business process instances. Currently, some of the available BPM Environments offer basic-level performance prediction capabilities. However, these functionalities are normally too limited to be generally useful for performance related decision support at business process level. In this paper, an approach is presented which allows the non-intrusive integration of sophisticated tooling for what-if simulations, analytic performance prediction tools, process optimizations or a combination of such solutions into already existing BPM environments. The approach abstracts from process modelling techniques which enable automatic decision support spanning processes across numerous BPM Environments. For instance, this enables end-to-end decision support for composite processes modelled with the Business Process Modelling Notation (BPMN) on top of existing Enterprise Resource Planning (ERP) processes modelled with proprietary languages.

1 Introduction

Business processes are the foundation of any enterprise. Their efficiency has an important effect on the profitability and hence on the success of a company regardless of its size or domain. Therefore, the goal of any enterprise is to continuously optimize business process execution and adapt it to changes within the market environment or the company itself. Enterprise application vendors aim to support this by the notion of "closed loop of continuous process optimization" (see Figure 1). In this paper, all tooling related to this loop is bundled under the term BPM Environment. One phase within this loop is the business processs configuration and business process composition (see CONFIGURE and COMPOSE in Figure 1). This phase enables business analysts to use tools like

U. Dayal et al. (Eds.): BPM 2009, LNCS 5701, pp. 97–112, 2009.

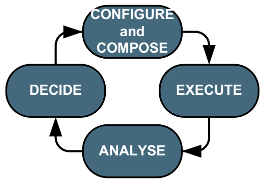

Fig. 1. Decision Support integrated in a closed loop of continuous process optimization

NetWeaver BPM [1], JCOM [2] or EMC Documentum Process Suite [3] to compose business logic, e.g. for Composite Applications, on top of services provided by configured back-end processes, such as the ERP processes offered by SAP Business Suite or Business ByDesign [4]. One example of such an extension is provided in one of our previous works [5]. Additionally, the business execution (see EXECUTE) needs to be supported by BPM Environments as well.

Tooling for the analysis of the business process history is provided by some BPM environments (see ANALYSE) in order to enable business process monitoring and analysis. This Analysis step provides performance data for already executed business processes, and offers functionalities and UI capabilities to the users such as sales unit managers, etc., to monitor and analyse the historic process performance data. This monitoring is then interpreted by users into decisions, such as organisational changes or modifications of the business process itself, meant to improve the future business performance.

However, decisions deduced from monitoring and analysis tooling are not sufficient in case of a high degree of complexity in resource intensive processes (e.g. layered use of resources, complex workflows, etc.) or in the statistical distribution of the monitored performance data or of plan data. The monitoring and analysis based decision making process, therefore, might not always be helpful to completely eliminate performance issues caused by a suboptimal scheduling of resource, under-utilization, bottle-necks, etc.

Thus, business performance related decision support is needed to deal with such cases (see DECIDE). For such support, performance analysis models normally need to be manually built in order to deal with complex resource scheduling problems, for instance, via what-if simulation. This task is time consuming, expensive and requires simulation related skills. A user further needs to have the required performance modelling expertise and the necessary skills to be able to interpret simulation results properly. The same applies for modelling in order to solve an optimization problem. Therefore, it is more appropriate to integrate directly such decision support into the existing business process modelling tools as part of BPM Environments. Integrated decision support is provided by a number of BPM Environments, like the EMC Documentum Process Suite [3], but based on basic level process simulation capabilities. Sophisticated business performance decision support, such as simulation of resource sharing scenarios among different departments or integration of optimization engines is missing in most environments [6]. Finally, non of the existing BPM Engines enables sophisticated end-to-end decision support spanning Composite Application as well as back-end processes [5].

In this paper we define an architecture which enables non-intrusive integration of sophisticated performance related decision support into existing BPM Environments and describe our industrial experiences with applications of this architecture for processes provided by existing ERP software, processes modelled with NetWeaver BPM and processes of the JCOM [2] environment.

The paper is structured as followed: The next section describes different kinds of questions answered by performance related decision support for business processes. Section 3 motivates the need for the integration of performance related decision support into existing BPM environments in a way which abstracts concrete business process modelling. Section 4 describes the proposed architecture which is then evaluated in Section 5. Section 6 provides an overview of the related work. Finally, Section 7 concludes the paper.

2 Background: Performance Related Decision Support for BPM Environments

We experienced that support is needed within complex business processes to investigate questions related to distribution of resources, working times, throughput and utilization. This is especially required in cases where there is a high degree of complexity in resource intensive processes (e.g. layered use of resources, shared use of resources, complex workflows, etc.) or in the statistical distribution of the history data or the plan data. Our business performance related decision support addresses the following type of resource related questions:

1. Can available staff cope with the future business growth?
2. Would a change of business conditions (e.g. change in the lower boundary of sales order approval request) improve the business performance?
3. Would a redistribution of resources between departments help to achieve overall performance targets?
4. How many employees are needed at which point in time?
5. Where is the predicted bottle-neck of the process?

Questions 1-3 can be answered via discrete event simulations [7]. For such a prediction, the control flow oriented business behaviour model, e.g. a Business Process Modelling Notation (BPMN) conforming model, needs to be combined with business process instance data indicating the resource related behaviour data of business process instances over a period of time. This time period can span historic process instances as well as future ones. Examples of resource related behaviour are the time needed to execute a BPMN Activity by one employee or the working time contingent of this employee.

Thus, *Process Model Data* (control flow related behaviour) needs to be combined with *Process Instance Data* (resource related behaviour). In the literature, numerous transformations can be found [8,9] where process models, such as UML Activity Diagrams are combined with process instance data in order to generate input for a discrete event simulation tool. On the other hand, numerous transformations can be found [10,11] as well where the same data is used to generate models analysed analytically. Analytical performance analysis tools produce results normally significantly faster than simulations but they are using mathematical assumptions which make the results less accurate. Also, they can normally only be used up to a certain size of input model due to the state space explosion problem. For instance, [10] transforms UML Activity Diagrams to Layered Queuing Networks (LQN) which can be analytically solved with the LQN solver (LQNS)[12]. LQN solver, as compared to other analytic approaches, especially considers layered use of resources, for instance, in case one resource needs to wait for another one in a rendezvous like communication scheme [13] in order to process activities. Moreover, if business performance objectives, constraints and requirements are modelled as well, it might be useful to integrate an optimization engine, such as that provided via the tool AnyLogic [14], in order to automatically execute a number of what-if simulations in order to propose an optimal solution. Such an approach can be used to answer question 4. Furthermore, additional computations can be done to answer requests like question 5, which are based on the results from the analytical or simulation based analysis.

3 Motivation

Some BPM Environments already provide basic what-if simulation capabilities [6]. Others turn to specialists to undertake simulation studies, and those specialists often prefer more sophisticated simulation tools [6]. Sophisticated simulation tools, such as the AnyLogic simulation tool [14], enable simulation of resource sharing scenarios among different departments. This functionality is normally not supported by the integrated simulation capabilities of existing BPM tools, such as the EMC Documentum Process Suite [3].

Furthermore, most BPM tools with simulation capabilities do not offer decision support functionality besides simulation functionality. However, a combination of different kinds of performance analysis techniques can help to turn what-if simulation results into information that business domain experts better understand. This is required as business domain experts are typically not performance modelling experts [5]. For instance, the automated combination of what-if simulations with an optimization engine might be useful in order to provide suggestions about, for example, how a perfect resource scheduling should look like. Also a combination of a simulation or a analytic performance prediction tool with a bottle-neck analysis based on the prediction results might be useful for a user to know which process step should be improved to prevent a bottleneck in the future. Moreover, none of the existing BPM Environments enable decision support spanning processes across numerous BPM Environments. This would enable end-to-end decision support for composite processes modelled with BPMN based modelling tools (e.g. NetWeaver BPM) on top of existing back-end processes (provided by existing ERP software) modelled with proprietary modelling tools. Concluding, an approach is required which enables the integration of sophisticated performance related decision support into a number of existing BPM Environments, which don't have this functionality yet or only offer basic level functionality. This integration especially needs to abstract from the business process modelling tool.

4 Proposed Architecture

Our proposed architecture for such an integrated performance related decision support in shown in Figure 2. This architecture refines the so called Model-Driven Performance Engineering architecture (MDPE) [15,16] which was originally designed for rather hardware resource related performance decisions. However, performance is a concept which cross cuts numerous domains. Thus, a performance modelling approach such as MDPE can be used for the BPM domain as well.

In the following paragraphs we define the various actors involved in this architecture which are bundeled within the *MDPE Workbench*:

The *Decision Support Integrator* extends the BPM Environment of choice with performance decision support functionality. A *Performance Modelling Actor* is a part of the Decision Support Integrator. It abstracts sophisticated *Performance Analysis Tools* on the one hand and a BPM Environment of choice on the other hand. Furthermore, the Performance Modelling Actor requires *Performance Parameter* as input. Examples for such parameters are how many sales order requests have occurred/will occur per day in the previous/next 12 months (see *History Data* and *Plan Data* in Figure 2). A more detailed description of the Performance Modelling is given in Subsection 4.1. A *Decision Support Calculator*, another part of the Decision Support Integrator, enables us to interconnect a number of different *Performance Analysis Tools* and to use these to generate a *Decision Support Result* as described in detail in Subsection 4.3.

The *Instance Data Manager* described by Subsection 4.2 is needed to provide access to the *Process Instance Data* for all actors of the Decision Support Integrator, and to enable editing and analysis of input and output data for the decision support in a language which a business domain expert understands.

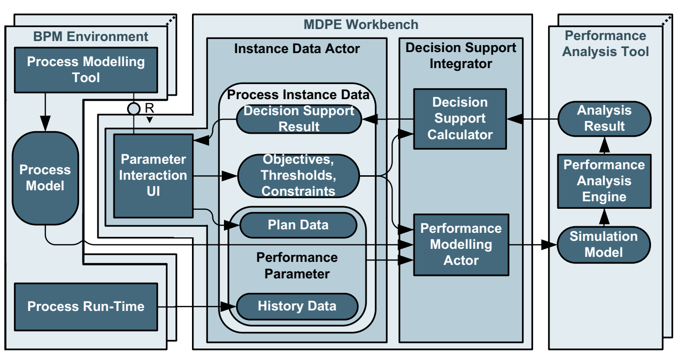

Fig. 2. Proposed Architecture as Block Diagram [17]

4.1 Performance Modelling Actor

The Performance Modelling Actor (see Figure 4) provides an abstraction layer for sophisticated Performance Analysis Tools including the Process Runtime on the one hand and Process Modelling Tools on the other hand.

The MDPE approach uses Tool Independent Performance Model (TIPM) which has been designed based on the Core Scenario Model (CSM) [18] by the TU Dresden, SAP Research and the simulation tool provider XJTech as a generic performance analysis model representation. Each TIPM is transformed to at least one Tool Specific Performance Model (TSPM) as shown in Figure 4. A TSPM is specific for a given Performance Analysis Tool, such as the discrete event simulation engine AnyLogic [14]. Compared to that, the TIPM is an intermediate language between Performance Analysis Tools and Process Modelling Tools. Thus, the TIPM helps to apply performance related decision support for a number of BPM Environments. A description of the TIPM meta-model follows.

TIPM based Abstraction. A TIPM combines the behavioural information from the Process Models with Process Instance Data. The behavioural information is represented in the meta-model of the TIPM (see Figure 3) with the meta-elements *Step* and *PathConnections* which are part of a *Scenario*. An example for such a Scenario in the business process domain is "Sales Order Processing" for a certain sales office in Philadelphia. Resources can be shared among multiple Scenarios, such as the case that the Marketing department with 10 employees is shared between the *Sales Order Processing of a sales office in Philadelphia* and the *Sales Order Processing of a sales office in Chicago*.

Performance Parameters (see Figure 2) need to be collected by an automated parameter importer out of the Process Runtime in the case of History Data (see in Figure 4) as proposed by Rozinat et al. [19] or defined as Plan Data. Performance Parameters are used to populate the following fields in the TIPM (see Figure 3):

– *Resource.multiplicity (called* Capacity *in this paper):* This metric indicates how many units are available in a pool of resources e.g. 10 employees in the Philadelphia sales office.

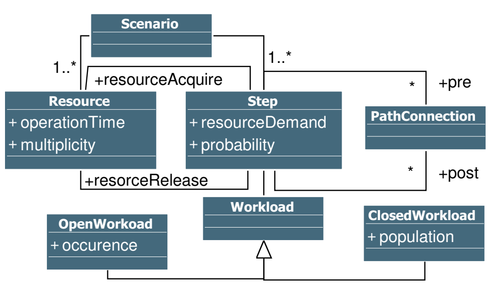

Fig. 3. Simplified Performance Analysis Model called TIPM

- *Resource.operationTime:* This metric indicates how much work can be done by one resource unit in a period of time. For instance, it specifies the resource efficiency of an employee.
- *Step.resourceDemand:* Indicates the net resource consumption of a Step, e.g. how much net working time is needed in order to create a Sales Order.
- *Step.probability:* Indicates the probability that a step is reached from the previous step.
- *Resource Link:* Is the reference between the Step and the Resource (see ResourceRelease and ResourceAcquire in Figure 3). It specifies, for instance, which process steps in Sales Order Processing have to be executed by the Marketing department.
- *Workload:* Specifies the occurrence or the population of arriving requests either in case of an OpenWorkload (occurrence), or a ClosedWorkload (population). An example of a open workload is the number of arriving sales requests per day in a business process for sales order processing; a closed workload example would be number of consultants starting a business trip immediately upon return from the previous one.

In the following subsection a description is provided of how the TIPM interconnects process modelling tools within BPM Environments with the Performance Analysis Tools.

Modular Model Transformations. As shown in Figure 4, the TIPM induces the need for a model automated transformation chain in order to first transform Process Models and the Performance Parameters to a TIPM and then to transform the TIPM to one or more TSPMs. The transformations are implemented within so called source- and target adapters as shown in Figure 4. These transformations are modularized into numerous transformation steps as described in [20]. This, for instance, enables separation of the structural concern of the *TIPM2AnyLogic_Sim* transformation (see Figure 4) from the concern of the actual XML representation of an AnyLogic simulation model. This decoupling further enables a high degree of reusability, as we are able to reuse some transformation steps for a number of source- and target adapters [20].

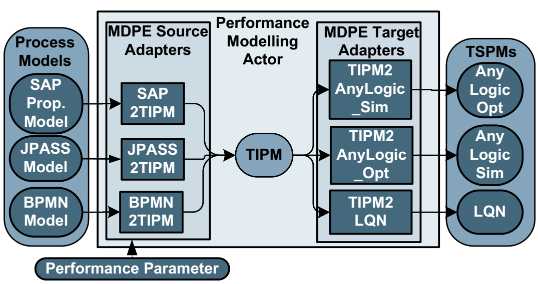

Fig. 4. Performance Modelling Actor as Block Diagram[17]

Figure 4 shows the source- and target adapters that we implemented. It can be seen that we are able to extend three different BPM Environments. Each of these environments is based on different modelling languages. We, therefore, have to support as input for the Performance Modelling Actor: SAP proprietary models, employed for back-end processes delivered by Business Suite or Business ByDesign; JPASS models for the extension of the JCOM environment; and BPMN models to extend the NetWeaver BPM environment employed for composite processes.

We added three different target adapters to our workbench. Thus, three different performance analysis methodologies can be used from the three different BPM environments which shows the high degree of extensibility of our solution enabled by the TIPM. One target adapter contains the transformations between TIPM and the simulation tool AnyLogic that we currently use as discrete event simulation engine (see *TIPM2AnyLogic_Sim* in Figure 4). Another is used for AnyLogic optimization experiments (see *TIPM2AnyLogic_Opt* in Figure 4). Moreover, we are considering analytic performance analysis. Therefore, the current MDPE implementation also supports the transformation of the TIPM to Layered Queuing Networks (see *TIPM2LQN* in Figure 4) in order to be used as input for the LQNS tool [12].

Each transformation in the chain has not only the direct transformation result as output but also, as a by-product, a *Trace Model* which stores the information about which model element(s) a is transformed to which model element(s) b. In [21] we described how this Trace Model is achieved as a by-product without additional effort from the developer of a transformation via the so-called Higher Order Transformations. The use of these trace models is described in the following subsection.

4.2 Instance Data Actor

As shown in Figure 4, the transformations within the source adapters combine Performance Parameters with the behaviour modelled within Process Models in order to generate a TIPM. Most of the Performance Parameters need to be extracted as History Data out of a business process history log provided by a Business Process Runtime (see Figure 2). However, Plan Data can be defined or modified by the user. Additionally, the user needs to specify the Target Values, Objectives and Constraints and to understand the Decision Support Results (see Figure 2). Thus, it is required that the user can set and view this Process Instance Data (see Figure 2) based on the Process Models and by using a vocabulary of his/her business process domain.

The following two subsections, therefore, describe how these Process Instance Data (see Figure 2) are represented and managed through the automated model transformation chain introduced by the abstraction provided by the TIPM; and the high degree of modularity for the implementation of the TIPM related model transformations.

Management of Process Instance Data. Decision Support Results (see Figure 2), such as a simulation based prediction that a threshold will not pass in the future, are set based on a TSPM, but need to be visualized based on the original Process Models. We therefore use the trace models generated as by-products of the transformations within source-and target adapters to navigate backward through the automated transformation chain, from the TSPM model elements to the model elements of the original Process Model. However, we have to deal with a high number of source and target adapters, and therefore a high number of model transformations, trace models and intermediate models of the different model transformation chains. Thus, a systematic solution was required to represent the linkage between source and target models of the different model transformations and the related trace models. This linkage is stored, as proposed by Bèzivin [22], into a so-called *megamodel*, which is a specialized model to represent relationship between modelling artefacts. A specialized version of such a megamodel [23] together with the trace models enables us to modularize the transformation chains between Process Models and TSPMs into as many transformation steps as one wants, as shown in [24].

Representation of Process Instance Data. The representation of the Process Instance Data has to be done in a way that the meta-model of the Process Model is not polluted. This pollution leads to contradicting the separation of concerns principle [25]. Additionally, it is not always possible to have access to the meta-model of the Process Model [24]. Thus, an approach such as UML profiles was not sufficient for our case.

In our approach all Process Instance Data is defined within separate annotation models [24] which are conforming to annotation meta-models. Therefore, for the definition of, for instance, Performance Parameters, Decision Support Results and the Objectives, we had to define a number of separate annotation meta-models [24]. Each of these meta-models is specific for the business domain which enables the user of our architecture to view and edit the different annotation models via a specific Parameter Interaction UI (see Figure 5) [24] in vocabulary he/she understands.

As described in [24], our annotation meta-models are refining the weaving meta-model provided by the ATLAS group [26]. This meta-model enables the definition of links to other models [27,28]. Thus, due to the fact that our annotation meta-models are based on the weaving meta-model, our approach enables annotation of additional information to Process Model of the BPM Environment without polluting them.

Figure 5 shows the application of our annotation models for two different process modelling tools: The BPMN based NetWeaver BPM editor and an editor for back-end processes based on a SAP proprietary modelling language. The "Start Process" node selected in the back-end process editor (see right side of Figure 5) and the annotated workload for the sales office "Chicago" is visualized as "Process Instance Occurrence" in the Parameter Interaction UI (see left side of Figure 5). Moreover, the bottom right of the figure shows a planned "Occurrence" of a process instance, between 01.10.2009 and 31.12.2009, which is 2 tasks per day.

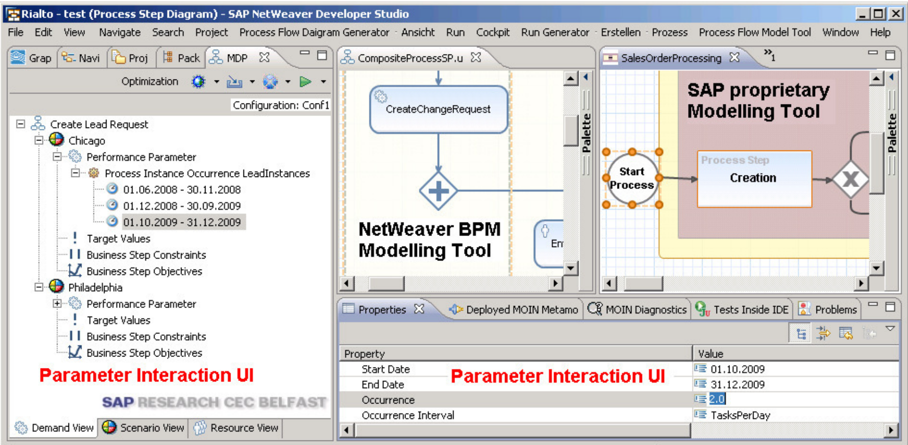

Fig. 5. Integration of the Parameter Interaction UI into two modelling tools: The NetWeaver BPM editor (middle) and an editor for back-end processes (right)

The Parameter Interaction UI therefore encapsulates the functionality to enrich the Process Modelling Tool of the BPM Environment with capabilities to visualize the annotated Process Instance Data based on the Process Model and edit some of this data. The current implementation of this annotation editor is Eclipse framework specific which restricts the application of our implementation to BPM Environments, using Eclipse based Process Modelling Tools. The main concepts can however be applied to any Process Modelling Tool.

For the non-intrusive integration of annotation models into a process modelling tool it is necessary to notify about the currently selected graphical model element to the Parameter Interaction UI. Therefore, it was required to implement a minor extension (less than 100 lines of code) for the SAP proprietary counterpart of the Eclipse Graphical Editor Framework (GEF) [29], to call the Parameter Interaction UI if the selected process flow model element is changed. This extension can be reused for numerous modelling editors. The JPASS tool is however not based on a graphical framework like GEF. We therefore additionally developed a minor extension for the JPASS tool. Hence, the only place where, in few cases, an Eclipse based process model editor needs to be modified in order to extend it with performance related decision support, is to notify about the currently selected graphical model element to the Parameter Interaction UI.

4.3 Decision Support Calculator

In the previous subsection we described how the Process Instance Data is represented as annotation models, and how we interconnect different Process Modelling Tools with multiple Performance Analysis Tools. The current subsection describes how the Decision Support Results (see Figure 2), e.g. if thresholds will be met in the future, are calculated based on the output of one or more

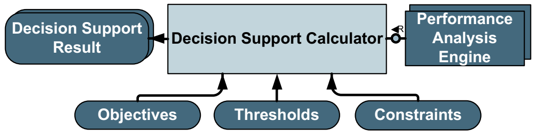

Fig. 6. Decision Support Calculator as Block Diagram [17]

Performance Analysis Tools and user provided Objectives, Constraints and Thresholds. This calculation is done by the *Decision Support Calculator*, which is depicted by Figure 6. This actor combines Performance Analysis Results with its own logic in order to output a Decision Support Result.

Based on the currently available Performance Analysis Tools for our current proposed architecture, a combination of different kinds of decision support is enabled:

- A *Threshold Checker* either executes what-if simulations by calling the AnyLogic discrete event simulation tool based on the history and plan data, or triggers analytical predictions from the LQNS tool based on average calculations from these data. The performance analysis results are compared with the user provided thresholds.
- An *Optimization Engine* executes automatically a number of what-if simulations by calling the AnyLogic Optimization Engine in order to fulfil user provided threshold but also to have the best possible result with regard to user provided Objectives. The possible configurations for the what-if simulations are restricted via the user provided Constraints.

The final Decision Support Results are also represented as annotation models which are used to enrich the original Process as described in the previous subsection. Hence, a user is, for instance, able to see which activities in a BPMN process will not fulfil certain thresholds in case of future business growth.

5 Experiences Gained

From the architectural point of view, the high degree of modularity within the proposed architecture enabled us to gain advantages in terms of extensibility and reusability. Extensibility is demonstrated when we recently extended our solution with the LQNS tool. This additional performance analysis methodology is usable for the users of all the BPM Environments which we have already extended. Reusability has been demonstrated also: The JPASS modelling tool (within the JCOM BPM Environment), has been recently extended with our architecture. The effort of writing the required transformations took less than one week of development effort. However, the effort of integrating the first Process Modelling Tool in our architecture took us around six weeks of development effort. This is due to the fact that all TIPM to TSPM transformations could be reused for the JPASS integration including some transformation steps provided by the already existing Process Model to TIPM transformation.

The cost for this high degree of reusability and extendibility, introduced with our architecture, is additional memory consumption and performance footprint-which is however, for current applications of our architecture, not yet critical.

In order to gain experiences with our tooling from the functionality point of view, we applied it for the SAP demo company called Akron Heating. Akron Heating does not exist in reality but the business processes and data of this company are maintained within SAP just as the data of a real one, for experimental purposes. Below we discuss an example combination of different what-if questions and an process optimization based on Akron Heating in order to demonstrate that the current combination of Performance Analysis Tools provided by our architecture is applicable for industrial usage.

Akron has three sales offices in the US (Denver, New York and Philadelphia). To leverage business in the USA Midwest area the head of the company decided to set up a new sales office in Chicago in September 2008. Based on the monitoring and analysis tooling, the head of sales discovers that one of the processes has not been executed with the expected efficiency across all sales units. Thus, based on a process model he/she is able to investigate the source of this poor business process performance based on predefined thresholds of Key Performance Indicators (KPI) which are e.g. indicating the historic end-to-end processing times. Based on a drilldown of these processing times, he/she finds out that especially the historic performance of one process step executed in the Chicago office is not sufficient.

Since we extended a SAP proprietary modelling editor for back-end processes with our tooling, the user is now able to investigate the impact of a number of potential changes in the process execution with a combination of different automatically generated what-if simulations (discrete-event simulations) and process optimizations. For the what-if simulation, two Scenarios are annotated (see left side of Figure 5) to the process model and transformed to the TIPM (see 4.1): one for the process executed in Philadelphia and one for the Chicago business process. It follows a description of a four step performance analysis:

- In a first step, the user does a what-if simulation in order to predict the outcome of training by reducing the annotated planned working time consumption (called Resource Demand in the TIPM) of one process step. A discrete event simulation based prediction which utilizes the AnyLogic tool shows, based on the process model, that if the training of the department in Chicago made the employees as efficient as the employees in other departments, all processing targets would be met.
- In a second what-if question the user wants to investigate, again via a discrete event simulation, if the staff in the Chicago office can handle the future business growth by increasing the planned Process Instance Occurrence (called OpenWorkload in the TIPM). The result of the simulation demonstrates that a business growth would lead to a resource problem within the department in Chicago.
- A third what-if simulation shows how our decision support tooling can help to identify if staff of other departments can compensate this resource problem in the case that some resources are shared among the departments. This is done

by modifying annotated responsibilities of the Philadelphia staff for marketing related process steps of the Chicago office (responsibilities are represented as Resource Links in the TIPM). In the TIPM the Philadelphia resource can be linked between the marketing related Steps of the Chicago process. This is possible as the staff of the two departments is represented in the TIPM independent of the formerly mentioned Scenarios.
– In the fourth step, the application of the AnyLogic optimization engine shows, via Optimization Assessment, what is the optimal sharing of resources among the departments; e.g. how many working hours have to be provided for the Chicago process related tasks by Philadelphia staff.

Additionally, we could have predicted the impact of changing a business condition, such as the lower boundary of an approval request. Also, in case the analysed process gets extended with a NetWeaver BPM process, our tool is still able to support the head of sales at Akron heating.

Furthermore, we are able to combine simulations with a bottle-neck analysis in order to indicate future bottle-necks. We anticipate to also gain industrial experiences with this additional Performance Analysis Engine.

Concluding, the possibility to combine different sophisticated decision methodologies with a number of process modelling tools provided by different BPM Environments has been identified as very beneficial. However, we identified the need for an automated History Data import for our solution which we have not implemented yet. Thus, History Data is currently annotated manually which is too time consuming for industrial application. An automated History Data import would calculate the historic Probabilities and Occurrences for a specific process, e.g. by counting the number of executed process instances. Furthermore, the working time consumptions can be calculated based on the process step durations. Additionally, resource Capacities and Resource Links can be calculated by interpreting those resources as part of the Capacity which has been used in the past. This importer also needs to provide a way to systematically deal with uncertainties in the History Data, for instance, due to a high variance or too few executed process instances. This should especially enable users to provide assumptions for cases where the confidence in the historic data is too low. Moreover, an integration of additional data sources, such as Human Resource (HR) data from the organizational management is required for future versions of our tooling. Especially, the allocation of persons to projects or organisational units needs to be accessed from HR data in order to calculate capacities. We, therefore, also require a mechanism to enable import of such additional data sources.

6 Related Work

From the application point of view, the closest related work to our knowledge is that concerned with BPM Environment such as EMC Documentum Process Suite [3] which provides simulation capabilities which are normally simplistic [6]. Our approach enables one to benefit from the know-how and functionality contained in a sophisticated performance decision support system, which enables, for example, sophisticated model simulations, optimizations and static analysis including a combination of them. The closest work to our knowledge from the architectural point of view is the PUMA architecture [30] which is based on a Core Scenario Model (CSM) [18], similar to the TIPM. However, the PUMA approach cannot be applied to BPM Environments as it is modelling Performance Parameters as UML Profiles, which are applicable only when UML models are employed as Process Models. Our approach is based on annotation models and provides a significantly higher degree of flexibility. Our approach can therefore be used to annotate any kind of Process Model, for instance, BPMN models used for NetWeaver BPM, or numerous SAP proprietary models used in existing ERP solution. Furthermore, we are able to support visualization of the Process Instance Data, such as Plan Data and the Decision Support Results, in the language a domain expert can understand and based on the original Process Models. Finally, our approach considers multiple views, namely: Objectives, Constraints and Requirements, as proposed in [16]. This enables better decision support than that provided by the PUMA approach, as we can, for instance, automatically propose optimal solutions.

7 Conclusions and Future Work

In this paper we proposed a generic architecture which enables extension of existing BPM Environments, having basic-level or no decision support, with capabilities for sophisticated performance related decision support. In case the proposed solution is applied, this decision support is executed via a mouse click. It is especially useful for resource scheduling questions, which arise particularly in the case of highly complex resource intensive processes (e.g. layered use of resources, complex workflows, etc.) or where the statistical distributions of the history data are complex. Thus, our approach helps to improve understanding of resource usages within complex business processes.

Due to the integration of sophisticated decision support tooling, existing BPM Environments can benefit from the know-how and functionality contained in such tools, which enables, for example, sophisticated model simulations, optimizations and static analysis. Furthermore, our architecture enables the integration and combination of multiple sophisticated decision support tools in an efficient way and without polluting original models with additional information for performance analysis, which is sometimes not possible [24]. Our architecture further enables to integrate sophisticated decision support tooling in such a way that it is straightforward to be used by business domain experts using the BPM Environments at runtime and design time of a business process.

Additionally, we abstract the BPM Environment itself which enables us to apply our decision support for end-to-end processes which are possibly managed with a number of BPM Environments.

We anticipate to extend our approach with a graphical indication of uncertainties in the historic performance data, which is used as input for the automatically generated performance analysis models. Such uncertainties, for instance, historic resource demands with a high variance, etc., will be presented to the user to allow input of available assumptions.

Disclaimer

References

1. Snabe, J.H., Rosenber, A., Molle, C., Scavillo, M.: Business Process Management: The SAP Roadmap (2008)
2. JCOM (2008), `http://www.jcom1.com/`
3. Associates, B.S.: The BPMS Report: EMC Documentum Process Suite 6.0 (2007)
4. SAP AG (2009), `http://www.sap.com/solutions/sme/businessbydesign/index.epx`
5. Fritzsche, M., Gilani, W., Fritzsche, C., Spence, I.T.A., Kilpatrick, P., Brown, J.: Towards utilizing model-driven engineering of composite applications for business performance analysis. In: Schieferdecker, I., Hartman, A. (eds.) ECMDA-FA 2008. LNCS, vol. 5095, pp. 369–380. Springer, Heidelberg (2008)
6. Harmon, P., Wolf, C.: The state of business process management (2008)
7. Banks, J., Carson, J.S., Nelson, B., Nicol, D.: Discrete-Event Simulation. Prentice-Hall, Englewood Cliffs (2005)
8. Rozinat, A., Wynn, M.T., van der Aalst, W.M.P., ter Hofstede, A.H.M., Fidge, C.J.: Workflow Simulation for Operational Decision Support Using Design, Historic and State Information. In: Dumas, M., Reichert, M., Shan, M.-C. (eds.) BPM 2008. LNCS, vol. 5240, pp. 196–211. Springer, Heidelberg (2008)
9. Balsamo, S., Marzolla, M.: A simulation-based approach to software performance modeling. In: ESEC/FSE-11th. ACM, New York (2003)
10. D'Ambrogio, A.: A model transformation framework for the automated building of performance models from uml models. In: WOSP 2005. ACM Press, New York (2005)
11. Bertolino, A., Marchetti, E., Mirandola, R.: Real-time UML-based performance engineering to aid manager's decisions in multi-project planning. In: WOSP 2002. ACM Press, New York (2002)
12. Franks, R.G.: DISSERTATION: Performance Analysis of Distributed Server Systems (1999)
13. Woodside, C.M., Neilson, J.E., Petriu, D.C., Majumdar, S.: The Stochastic Rendezvous Network Model for Performance of Synchronous Client-Server-like Distributed Software. IEEE, Los Alamitos (1995)
14. XJ Technologies: AnyLogic — multi-paradigm simulation software, `http://www.xjtek.com/anylogic/`

15. Fritzsche, M., Johannes, J.: Putting Performance Engineering into Model-Driven Engineering: Model-Driven Performance Engineering. In: Giese, H. (ed.) MODELS 2008. LNCS, vol. 5002, pp. 164–175. Springer, Heidelberg (2008)
16. Fritzsche, M., Gilani, W., Spence, I., Brown, T.J., Kilpatrick, P., Bashroush, R.: Towards performance related decision support for model driven engineering of enterprise soa applications. In: 15th ECBS 2008, vol. 0. IEEE, Los Alamitos (2008)
17. Knöpfel, A., Gröne, B., Tabeling, P.: Fundamental Modeling Concepts: Effective Communication of IT Systems. John Wiley & Sons, Chichester (2006)
18. Petriu, D.B., Woodside, M.: An intermediate metamodel with scenarios and resources for generating performance models from UML designs. Software and Systems Modeling 6(2) (2007)
19. Rozinat, A., Wynn, M., Aalst, W., Hofstede, A., Fidge, C.: Workflow Simulation for Operational Decision Support Using Design, Historic and State Information. In: Dumas, M., Reichert, M., Shan, M.-C. (eds.) BPM 2008. LNCS, vol. 5240, pp. 196–211. Springer, Heidelberg (2008)
20. Fritzsche, M., Jouault, F., Lämmel, R., Gilani, W.: Model Transformation Chains to integrated Performance related Decision Support into BPM Tool Chains. In: Invited submission fro the post- proceedings of the GTTSE 2009. LNCS. Springer, Heidelberg (2009)
21. Fritzsche, M., Johannes, J., Zschaler, S., Zherebtsov, A., Terekhov, A.: Application of tracing techniques in model-driven performance engineering. In: 4th ECMDA Traceability Workshop, ECMDA-TW (2008)
22. Bézivin, J., Jouault, F., Rosenthal, P., Valduriez, P.: Modeling in the Large and Modeling in the Small. In: Aßmann, U., Aksit, M., Rensink, A. (eds.) MDAFA 2003. LNCS, vol. 3599, pp. 33–46. Springer, Heidelberg (2005)
23. Barbero, F., Jouault, J.: Model Driven Management of Complex Systems: Implementing the Macroscope's Vision. In: 15th ECBS 2008. IEEE, Los Alamitos (2008)
24. Fritzsche, M., Johannes, J., et al.: Systematic usage of embedded modelling languages in model transformation chains. In: SLE 2008. LNCS, vol. 5701. Springer, Heidelberg (2009)
25. Mehr, F., Schreier, U.: Modelling of Message Security Concerns with UML. In: 9th ICEIS (2007)
26. Fabro, M.D.D., Bézivin, J., Valduriez, P.: Weaving Models with the Eclipse AMW plugin. In: Eclipse Modeling Symposium, Eclipse Summit Europe (2006)
27. Vara1, J.M., Castro1, M.V.D., Fabro, M.D.D., Marcos, E.: Using Weaving Models to automate Model-Driven Web Engineering proposals. In: ZOCO 2008/ JISBD (2008)
28. Voelter, M., Groher, I., Kolb, B.: Mechanisms for Expressing Variability in Models and MDD Tool Chains. In: MDSD in Embedded Systems (2007)
29. Eclipse, C.: Eclipse graphical editing framework (gef) – version 3.4.2 (2009), `http://www.eclipse.org/gef`
30. Woodside, M., Petriu, D.C., Petriu, D.B., Shen, H., Israr, T., Merseguer, J.: Performance by unified model analysis (PUMA). In: WOSP 2005. ACM Press, New York (2005)