

Monitoring Business Processes in the Networked Enterprise

Francesco Arigliano⁴, Devis Bianchini⁵, Cinzia Cappiello³, Angelo Corallo²,
Paolo Ceravolo¹, Ernesto Damiani¹, Valeria De Antonellis⁵, Barbara Pernici³,
Pierluigi Plebani³, Davide Storelli², and Claudia Vicari⁴

¹ Dipartimento di Tecnologie dell'Informazione
Università degli Studi di Milano, Italy
{firstname.lastname}@unimi.it

² Centro Cultura Innovativa d'Impresa, Università del Salento, Lecce, Italy
{firstname.lastname}@unisalento.it

³ Dipartimento di Elettronica ed Informazione, Politecnico di Milano, Italy
{firstname.lastname}@polimi.it

⁴ Research & Development Laboratory - Engineering, Ingegneria Informatica S.p.A.,
Italy
{firstname.lastname}@eng.it

⁵ Dipartimento di Elettronica per l'Automazione
Università degli Studi di Brescia, Italy
{firstname.lastname}@ing.unibs.it

Abstract. The Object Management Group (OMG) is promoting the Model Driven Architecture (MDA) approach to support interaction among enterprises based on business process models. Based on this approach, we discuss in this paper how to specify performance indicators among the levels with different degree of abstraction suggested in MDA. These indicators will drive the monitoring activities to check the execution of business processes involving networked enterprises. Connecting the different levels we also decrease the cost of implementing metrics as the measurement of the entities at one level can be based on the lower level.

Keywords: performance indicators, model driven architecture, rules, violations, trends.

1 Introduction

Business Process Management (BPM) [1] is a systematic approach for driving the flow of business activities in accordance with strategic analysis. The idea is that this can be achieved only with a high level of coordination among the activities composing a business process. In particular, design, execution, and monitoring must be interrelated to verify if the running process is in compliance with requirements and to understand which part can be modified for improvement. One of the core functions of any BPM is represented by the Business Activity Monitor (BAM) that checks at run-time if a business process works according to the initial specifications by means of rules. These rules predicate on

data, messages and activities that have to be known in advance by the process designer and to be accessible by the BAM at run-time.

Although there are several BAM implementations, they suffer of two main limitations. Firstly, current BAMs usually focus on business processes that involve a single organization and manage the whole process by relying on a single environment [2,3]. Secondly, rules feeding a BAM predicate on elements that refer to the technological description of the business process. Thus, the BAMs assume that someone is in charge of defining these rules starting from the business needs. In addition, the continuous update of the business process to better fit the business objectives requires tools able to review the strategies adopted, the effectiveness of the information flow implemented, or the services implementing the activities.

The aim of the TEKNE (Towards Evolving Knowledge-based interNetworked Enterprise) project is to create an integrated framework for supporting and guiding Internetworked Enterprises (IEs) focusing on process modeling and rule-based monitoring. IEs, as defined in [4], are borderless organizations that share applications, services and knowledge and whose processes are transformed and integrated with the ones of their partners. The framework supports a methodology that plays a fundamental role in organizations modeling and in the measurement of their performance, in order to allow business managers and IT-specialists to control the organizational complexity in an integrated way. The methodology adopts the *MDA (Model Driven Architecture)* [5] approach in that it allows the representation of organizations and of their business at different levels of abstractions and defines a set of transformations between the models of such levels: *CIM (Computation Independent Model)* level, *PIM (Platform Independent Model)* level, and *PSM (Platform Specific Model)* level. According to this structure, companies are modeled spanning progressively from their strategies and business objectives to the enabling technological services. In this way, monitoring rules can be defined with higher level of abstraction and then transformed to the rules suitable for BAMs.

This implies the management of *directives* and *Key Performance Indicators* (KPI) both at business and technological level [6]. *Directives* are intended to define business rules the IE processes have to be compliant with, whereas *KPIs* are quantifiable measures that are usually used to analyze the trend of a process and its quality. Directives and KPIs are used to define properties that the IE business processes and the related implementation should satisfy. Usually, each organizational level defines the rules and indicators by using its own models and data and there is a lack of connection between indicators defined at different levels.

This paper proposes a framework that enables the specification and measurement of automatable metrics at each level of the modeled IE organizations, allowing as well the definition of relationships among metrics of different levels, thus providing a means to drill down the abstraction levels and bind strategic objectives to organizational and technological performance [7]. Connecting the different levels will also decrease the cost of implementing metrics as the measurement of the entities at one level can be based on the lower level. Metrics at CIM

level are used to monitor process compliance to strategic directives and financial goals, thus mostly highlighting tendencies exhibited by some aggregated values extracted from enacted processes. Metrics at PIM level are directly related to operational aspects of business processes, measuring violations of business rules that affect specific activities. Metrics at PSM level are focused on process performance in terms of timing and QoS. Although each level is focused on different aspects, one of the goals of this paper is to describe how it is possible to relate a metric defined at one level to the other levels. Also the method discussed in Section 5 provides an important instrument to evaluate the feasibility of a monitoring program on a set of distributed services that constitutes the Inter-networked Enterprise.

The paper is structured as follows. In Section 2 we illustrate the TEKNE framework and the approaches for rule modeling at different levels. In Section III a case study is presented. In Section IV a formal definition of Directives and KPIs is proposed and discussed. Section V discusses KPI complexity. In Section VI related work is illustrated.

2 The TEKNE Framework

The aim of the TEKNE project is to create a unique framework for BPM addressing specification, execution, and monitoring of internetworked enterprise processes. The project was also aimed at ensuring compatibility with software engineering standard approaches, such as the MDA. Basically, the company models are represented at different levels of abstractions that progressively span from the definition of business strategies to the technological platform that will implement them in concrete executions. The *Computer Independent Model* (CIM) defines the conceptual elements that are required at a business level, such as for instance actors, resources, and overall strategies and tactics of the business. The *Platform Independent Model* (PIM) describes the activities to be implemented and the flow of information driving activity execution. The *Platform Specific Model* (PSM) defines the technological platform specifying the services and the software components to be implemented. All these models are integrated with each other by appropriate mappings and are related to the methodology of organization change that is proposed in the project¹. With this approach, we can create a complete framework that could grant a real co-design of organization and technology.

In TEKNE, we adopt three different notations for describing the business processes: one for each of the levels. More precisely, SBVR (Semantic of Business Vocabulary and Business Rules) is used at CIM level, BPMN (Business Process Modeling Notation) is used at PIM level, and finally XPDL (XML Process Definition Language) is used at PSM level.

The combination of these models is a key element for addressing the final objective of the work presented in this paper: the definition at design-time of

¹ The discussion about this methodology is out of the scope of this paper that focuses on technological aspects, further details can be found in [8].

automatable metrics for internetworked processes that can be referred directly to business policies and business rules. Indeed, the architecture allows coupling elements of a business domain (i.e., SBVR business rules) with activities of a business process, both at design time and at run time, thus binding generic logical assertions to generic operations and their instances.

Probes are defined for all the three levels in order to measure the performance of the business process execution starting from the data available at each level. In case a business rule is violated during the process execution, the run-time execution environment has the ability of adapting the process execution as a composition of services that are dynamically selected with respect to quality-of-service parameters and to their context of execution. The execution environment is based on the PAWS approach [9] for adaptive service-based processes.

2.1 Rules Modeling with SBVR

SBVR (Semantics of Business Vocabulary and Business Rules) is a formalism proposed by OMG ² that allows for the formalization of business vocabularies and rules through a combination of syntactical and text formatting rules. According to this formalism, at design-time, the business engineer is able to conceptually model (i.e., at CIM level) a business process together by its context and directives. The adoption of SBVR is motivated by two features:

- SBVR models can be expressed by means of SBVR Structured English (SBVR-SE), a controlled English notation that has been designed to enable process/knowledge owners to directly and easily represent their tacit or explicit knowledge;
- the SBVR metamodel is compliant with and mapped to first order logic, thus fully supporting automatic interpretation and reasoning upon its assertions.

As an example, a *rule* indicating that a manager may book hotels in any category can be expressed in SBVR as follows:

Permissibility:

a manager **may** book a Hotel that has-category 5stars, 4stars, 3stars

Thanks to the mapping between elements in the BPMN and SBVR models, it is possible to derive rules from actual instances of process activities. Given a certain rule defined in a vocabulary, a rule can be defined as a real-world instance of another rule. For example “Mario Rossi books the Hotel Morning” is a rule that is an instance of the rule “Role book Hotel”. Assuming that the workflow-engine is aware of roles, one can for instance derive that “Mario Rossi is a Manager”, from the login logs. The resolution of rules is of responsibility of the third major component of the proposed architecture: the **Metrics Framework**. Here, data from the design-time and run-time repositories are retrieved, merged, and transformed into a logical formalism, thus constituting a knowledge base where logical assertions can be evaluated and metrics can be measured. Moreover, a dashboard

² <http://www.omg.org/spec/SBVR/1.0/>

is provided in order to allow business managers to directly monitor the actual values of the metrics they have previously defined.

2.2 Monitoring in Process Modeling Using BPMN

The *Business Process Modeling Notation* (BPMN) [10] has been adopted for specifying the business process at PIM level.

BPMN is the standard notation used for business process modeling. It is a simple graphical specification for defining the control flow of the business process. In particular determining the ordering of activities, and clearly defining pre- and post-conditions for their execution. The primary goal of developing BPMN diagrams is to provide a notation that is readily understandable by all business users, that is the analysts that create the initial drafts of processes, the technical developers who are responsible for implementing the technology automating those processes, and, finally, the business people who will manage and monitor those processes.

With BPMN business process can be seen as the collection of activities that are designed to produce the intended business behavior for a specific customer. In this way, the business engineering can capture both the broad outline and the specific procedures governing a business.

In the process models, activities represented by dotted rectangles are introduced to represent points of monitoring, where Directives or KPIs have to be verified. In addition, in the BPMN business process alternative paths can be introduced at different points of control, to check the defined directives and enforce them, defining where to read data and how to analyze them.

In order to automate the BPM for Directives and KPI definition any measure must be associated with a concrete data model providing the information required to implement that measure. As we are going to discuss in Section 5, this is the preliminary condition to reduce the cost of implementation of any measure. The integrated representation of business process models provided by TEKNE is motivated by the idea of providing a system supporting such a definition of KPIs.

2.3 Flexible Process Execution with XPD L and PAWS

At PSM level we adopt XPD L 2.0 (XML Process Definition Language). This XML-based language is a widely adopted standard maintained by the WfMC (Workflow Management Coalition) and supported by a number of workflow-engines. As an alternative solution, XPD L can be substituted with WS-BPEL (Web Service Business Process Execution Language) to specify the business process in terms of how to orchestrate the Web services able to perform the activities belonging to the process.

These languages allow designers modeling the process as the combination of abstract services. The main assumption is that several services can be used to perform the same activities. Such services can be, in fact, provided by alternative providers and they may present different characteristics, both from the semantic

point of view (e.g. selecting expensive *vs.* cheap hotels) and for the quality of service of the provider (e.g., considering the response time).

Automatic process execution based on XPDL is complemented by the ability of dynamically selecting services to be invoked during process execution. The execution environment is based on the PAWS approach [9] for adaptive service-based processes, in which services are dynamically selected on the basis of process global constraints, service interface similarity evaluation, and QoS is negotiated. Services are retrieved from a service registry in which the service interfaces can be semantically annotated [11].

3 Case Study

A simple but real example is illustrated in Fig. 1 and Fig. 2, about an organization for managing reimbursement requests submitted by workers traveling for business. In particular, Fig. 1 and Fig. 2 represent in BPMN respectively the two processes of hotel reservation and board expenses reimbursement.

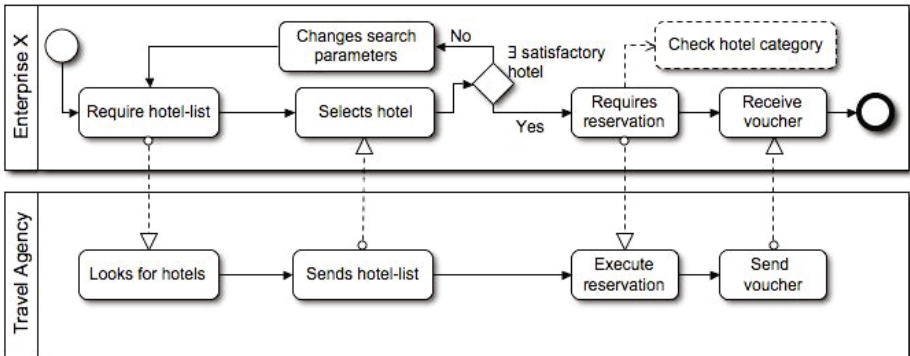


Fig. 1. An example of BP modeling *Hotel Booking*

The example is based on an internal regulation of a real SME regarding missions of its workers. The regulation identifies two kinds of workers: employees and consultants. The regulation is composed of many rules. Here few of such rules are considered for the sake of simplicity. For example, hotels booked for consultants should not be of category higher than 3 star, whereas employees do not have such a restriction. Moreover, both employees and consultants have an upper limit on the reimbursable daily board expenses.

Note that at an operational level it is permitted that the first rule is violated in some exceptional cases (e.g. only four-stars hotels are available at the location of the mission), while the second should never be violated: any exceeding expense should be systematically ignored by the administration of the organization. On the contrary at strategic level monitoring is not applied to the single activity

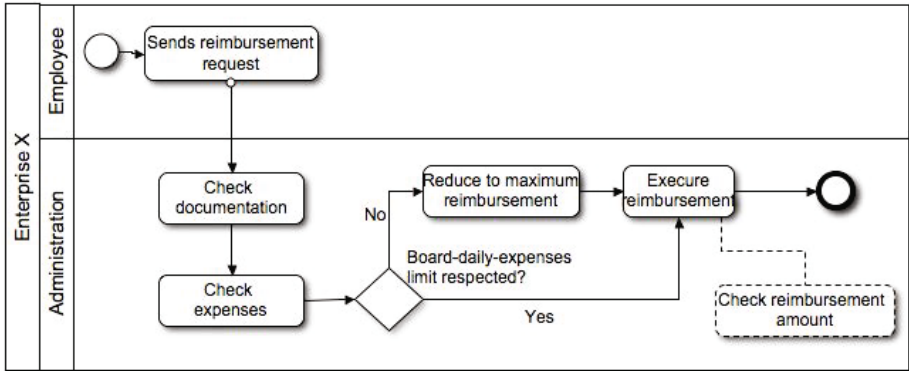


Fig. 2. An example of BP modeling *Request of Reimbursement*

but on trends and aggregation of information. The rules applied insist on the comparison among a specific execution of the process and an objective to be achieved. For example a rule insisting on strategic level could state that the number of missions with daily board expenses greater than 50 euro cannot exceed the 80% of the total number of missions. Note that this rule is based on the assessment of a KPI defined as the ratio between the number of missions that violate the rule defined at the operational level and the total amount of missions.

In Fig. 1 the *Check hotel category* represents a monitoring activity; in Fig. 2 a monitoring activity is *Check reimbursement amount*, while the alternative path to *Reduce maximum reimbursement* is introduced if a business rule about the maximum reimbursement is violated in the process.

We assume that services such as *Look for hotels* in the Hotel Booking process may be dynamically selected based on the given directives (selecting hotel with limited categories), or based on timing constraints, such as for instance *Execute reimbursement* in Process Request of Reimbursement, where a faster service could be selected in the maximum time for executing the process is strict and at risk of being violated.

Given the BPMN process models and the applied rules expressed in SBVR-SE, it is possible to define KPIs and metrics for strategic, operational and execution levels, as discussed in the following of the paper.

4 Defining Directives and KPIs

As previously said, a primary aim of our framework is to provide a means for monitoring the process at different levels. For this reason, our monitoring system requires the definition of Directives and KPIs at strategic, operational, and executive level.

Directives and KPIs are defined extending the approach described in [12]. First, we distinguish among different components of the specifications. F is a

specification of the flow of activities and information that must result from the execution of the process to be developed; W represents a set of world properties, i.e. the context where the process is executed; L is the log tracking the execution of the activities described in F ; and R represents a set of requirements to be met by the process, our Directives and KPIs. In TEKNE the requirements are expressed using the SBVR rules. All these components are related by the following implication:

$$F, W, L \rightarrow R \quad (1)$$

To prove that the specifications will satisfy the requirements (our rules), it is necessary to show that this implication holds. This can be done under the condition of providing an algorithm which is able to conjunctively evaluate the components, maintaining their independence. This algorithm has to work as a black box taking in input F , W , L , and R , and computing the consistency of the knowledge base. In case of inconsistency a violation has to be pointed out. Such an algorithm is compatible only with a representation of F , W , L , and R in declarative form, and assuming a uniform naming space.

Taking this approach, the definition of a Directive or a KPI reflects the requirement the system has to comply with. Of course, the hard part is achieving the uniformity in representing the specifications, maintaining the independence of the components. Another key point is to provide definitions detailing the points of control to be activated in order to equip our representation with the data values required to the resolution of the implication 1.

In addition, a definition is enriched by other information required to set up the rule resolution. In particular:

- the components of the specification to be involved, in correspondence to the related MDA level;
- the points of control to be activated to extract data values.

Based on this approach, considering the example represented in Fig. 1 and Fig. 2, we now provide the definitions of the rules previously described in Section 3 using SBVR.

Strategic Level. KPIs and directives aimed at monitoring strategic goals insist on trends exhibited by the process, as shown for instance by Rule 1a and Rule 1b.

Obligation:

Rule 1a: It is obligatory that the number of reimbursement that *have* board-daily-expense greater than 50 euro is less or equal than 80%

| | |
|--|---------------------|
| Specification: R, F, L | KPI type: Strategic |
| Point of control: Check reimbursement amount | |

Obligation:

Rule 1b: It is obligatory that the number of reimbursement processes that *have duration greater than 7 days is less or equal than 95%*

| | |
|--------------------------------|----------------------------|
| Specification: R, F, L | KPI type: Strategic |
| Point of control: Tasks | Execution Time |

Operational Level. At Operational level the aim is to implement a control on the flow of activities that has to comply to specific constrains. Rule 2 is an example of acting only on the contextual knowledge (W), without involving the process flow or its execution. Rules 3 and 4 are instead examples of rules requiring to know data values created at execution time.

Rule 2: Necessity:

a worker **has to be** an employee or a consultant

| | |
|-------------------------------|---------------------------------------|
| Specification: R, W | Directive type: Operational |
| Point of control: none | |

Rule 3: Obligation:

board-daily-expense **is at most** 50 euro

| | |
|--------------------------------|---------------------------------------|
| Specification: R, F, L | Directive type: Operational |
| Point of control: Check | reimbursement amount |

Rule 4: Obligation:

a consultant **books** a Hotel **that** *has-category* 1stars, 2stars, 3stars

These rules described above are strictly related to the domain. Rules that can be generally applied to the execution of any process are related to the duration of process or activities, as described in Rule 5.

| | |
|--------------------------------|---------------------------------------|
| Specification: R, F, L | Directive type: Operational |
| Point of control: Check | hotel category |

| | | |
|---|------------------|-------|
| Specification: R, F, L | Directive | type: |
| | Executive | |
| Point of control: Process execution time | | |

Rule 5: Obligation:

a reimbursement-process *has-duration* duration **that is less than** 7 days

Rule 5 is a directive linked to Rule 1b, where a KPI is defined based on execution time for the process. While the operational level rule can be used to check processes during execution and might be violated in exceptional cases, Rule 1b defines a criterion to be checked over multiple executions of the process, defining a KPI linked to the CIM level of IE process definitions.

In general, correspondences may be defined between rules at strategic level and operational level, with data derived from monitoring points defined in processes at the PIM level, using the extended BPM notation.

4.1 Rule Validation

To better understand how these rules can be processed in the **TEKNE Metrics Framework** let us detail how the predicates forming our knowledge base are structured. In **F** we have the predicates describing the process flow. These predicates are a specialization of the predicates in (2).

$$\begin{aligned}
&\rightarrow Process(p) \wedge executeActivity(p, a) \\
&\quad \wedge hasStartEvent(p, s) \\
&\quad \wedge hasEndEvent(p, e) \\
&\quad \wedge hasDecisionPoint(p, d) \\
&\wedge involveRole(p, r) \wedge Activity(a) \wedge Role(r)
\end{aligned} \tag{2}$$

For example the process in Fig. 1 is represented using predicates as in (3).

$$\begin{aligned}
&\rightarrow Process(p) \wedge executeActivity(p, a) \\
&\quad \wedge hasStartEvent(p, s) \\
&\quad \wedge executeDecisionPoint(p, d) \\
&\quad \wedge involveRole(p, r) \\
&\quad \wedge hasName(r, TravelAgency) \\
&\quad \wedge hasName(a, Looksforhotels) \\
&\quad \wedge hasName(d, BoardDailyExpense) \\
&\quad \wedge hasName(s, Start1)
\end{aligned} \tag{3}$$

In **W** we can have any predicate extending the representation of the process flow with facts and constraints pertinent to the business domain. For example Rule

2 prescribes that we can refer to the union of employees and consultants using the term worker, as in (4).

$$\begin{aligned} & \textit{Employee}(x) \rightarrow \textit{Worker}(x) \\ & \textit{Consultant}(x) \rightarrow \textit{Worker}(x) \end{aligned} \quad (4)$$

In L we have the predicates describing the concrete execution of a process, i. e. the specific user executing the activities, time of execution, input and output produced. An example of the predicates we can use is in (5).

$$\begin{aligned} & \rightarrow \textit{Process}(p) \wedge \textit{executeActivity}(p, a) \\ & \quad \wedge \textit{hasStartEvent}(p, s) \\ & \quad \wedge \textit{hasEndEvent}(p, e) \\ & \quad \wedge \textit{involveRole}(p, r) \\ & \quad \wedge \textit{hasUserName}(r, \textit{John}) \\ & \quad \wedge \textit{hasStartTime}(s, 12 : 02 : 34) \\ & \quad \wedge \textit{hasEndTime}(e, 12 : 53 : 38) \end{aligned} \quad (5)$$

To represent the rules set R expressing Directives and KPIs we have to distinguish predicates insisting on the antecedent or on the consequent. Our task is to check if one interpretation of the knowledge base, formed by the union of W, F and L, is implied by R. Any rule is organized according to an antecedent defining a specific state of the interpretation of the predicates in the knowledge base plus a consequent that must be true if the antecedent is true. Because the theorem prover we are using implements the material implication, rules are valid if both antecedent and consequent are true or if the antecedent is false. Without distinguishing the two part of the rule we could turn up to have a valid rule only because the antecedent is false. In 6, 7 we provide an example of the representation of Rule 3 and Rule 5 according to this distinction.

$$\begin{aligned} & \textit{Process}(p) \wedge \textit{executeDecisionPoint}(p, d) \\ & \quad \wedge \textit{hasName}(d, \textit{BoardDailyExpense}) \\ & \quad \rightarrow \textit{ValidCondition}(p) \\ & \textit{Process}(p) \wedge \textit{executeDecisionPoint}(p, d) \\ & \quad \wedge \textit{hasName}(d, \textit{BoardDailyExpense}) \\ & \quad \wedge \textit{hasOutput}(d, 65) \\ & \quad \rightarrow \textit{ValidConsequence}(p) \end{aligned} \quad (6)$$

$$\begin{aligned}
& Process(p) \wedge executeActivity(p, a) \\
& \wedge hasName(a, ReimbursementProcess) \\
& \quad \rightarrow ValidCondition(p) \\
& Process(p) \wedge hasDuration(p, d) \\
& \quad \wedge lessThen(d, 7) \\
& \quad \rightarrow ValidConsequence(p)
\end{aligned} \tag{7}$$

4.2 Rule Monitoring

The PSM level receives the BPMN process representation and translates it in term of the technology that has to be used to implement the IT platform. As described in Section II, our approach is based on service-based processes. Thus, applications and related business processes can be modeled as the composition of different services provided by the organizations involved in the IE network. Formally, the process \mathcal{P} specified and modeled at the PIM level that contains a set of tasks $\mathcal{T} = \{t_i\}$ can be associated with a set of services $\mathcal{S} = \{s_j\}$ to be performed. Each service s_j can execute one or more tasks t_i specified in the process model. Note that in a real business process not all the tasks can be executed by using a Web service (e.g., manufacturing activities) since they manipulate physical objects and produce tangible outputs. Thus, considering the set of services \mathcal{S} and the set of tasks \mathcal{T} , it is possible to assume that for every process $\mathcal{S} \subseteq \mathcal{T}$.

Considering the example Hotel Booking we can identify the services as represented in Figure 3.

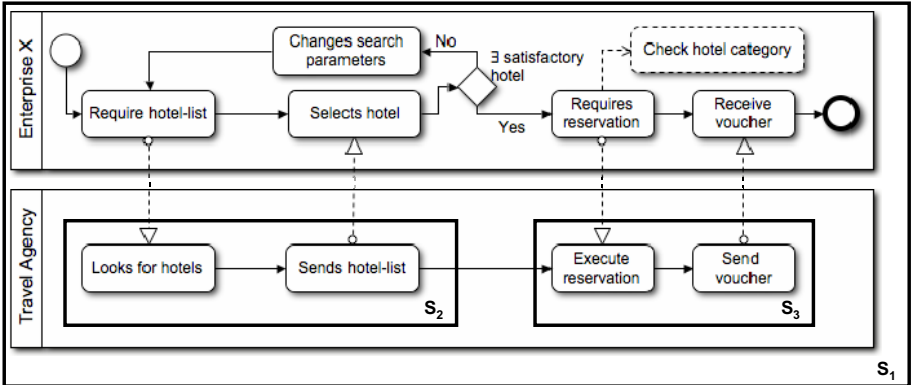


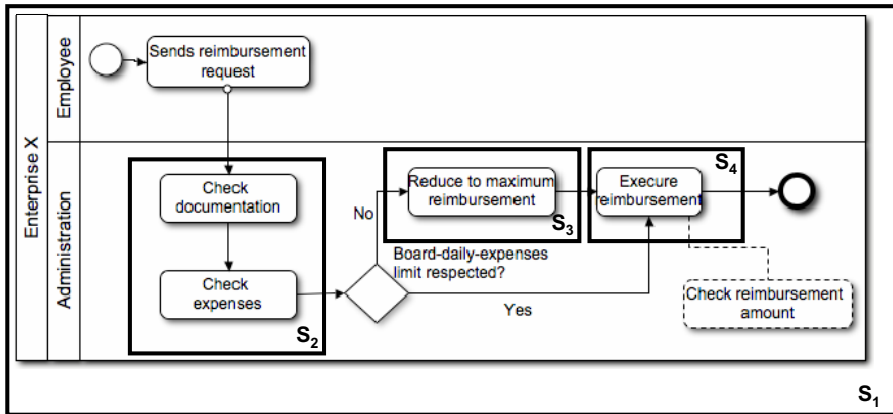
Fig. 3. *Hotel Booking* services

Each service is associated with a set of tasks as described in Table 1 (we assume that services are numbered starting from 1 inside each process description).

Table 1. Hotel Booking Service description

| Service No. | Tasks Name |
|----------------|---|
| s ₁ | All tasks in the process |
| s ₂ | Looks for hotels and <u>Send hotel-list</u> |
| s ₃ | Execute Reservation and <u>Send Voucher</u> |

Considering the example Request of Reimbursement we can identify the services as represented in Figure 4.

**Fig. 4.** *Request of Reimbursement* services

Each service is associated with a set of tasks as described in Table 2.

Table 2. Hotel Booking Service description

| Service No. | Tasks Name |
|----------------|---|
| s ₁ | All tasks in the process |
| s ₂ | <u>Check documentation and Check expenses</u> |
| s ₃ | <u>Reduce to maximum reimbursement</u> |
| s ₄ | <u>Execute reimbursement</u> |

Each service s_j is characterized by functional and non-functional aspects. To represent functional aspects, a service can be modeled as

$$s_i = \langle name, in_j, out_k, f, r, o, QD \rangle$$

where the service is described by the *name*, the set of input data in_j and the set of output data out_k , the related transformation function $f : in \rightarrow out$, and the role r inside the organization o responsible for the task. Non-functional aspects are related to quality attributes specified in the set QD . The quality of a Web

service is defined by a set of quality dimensions QD in which each dimension qd_i associated with a given quality aspect and represented as:

$$qd_i = \langle qdname, V \rangle$$

The *name* uniquely identifies the quality dimension. The element V corresponds to the quality dimension value that could be either categorical, numerical or an interval of admissible values. Both functional and non functional aspects can be used in the verification of the expressed requirements (rules validation and KPI assessment). In fact, for example the Rule 4 for the verification of the Hotel Category can be validated by considering input and output data associated with the s_2 in the Hotel Booking process. Rule 5 instead requires the analysis of non-functional properties. In particular, the assessment of the execution time of the service Request of Reimbursement has to be performed. In particular, the evaluation has to consider the *response time* dimension values associated with the services that compose the process analyzed. In formula:

$$s1.ExecutionTime \approx s2.ResponseTime + s3.ResponseTime.$$

For the assessment of Rule 1b we can consider the result of this formula and compare the number of times in which the results satisfy the Rule 5 with respect to the number of times in which the process is executed.

As regards Rule 1a we can not use any functional or not functional service properties. In this case, the number of reimbursements with limited expenses can be calculated by looking at the service execution. In fact, in the Request of Reimbursement process, this parameter can be calculated as the differences between the number of invocations of s_1 and the number of invocations of s_3 .

During execution, monitoring at PSM level can support process adaptivity with respect to the defined business rules and KPI. For instance, during hotel selection, the proposed hotels may depend on the category of the employee for which the reservation is being made. Considering quality of service, different services can be employed (or selected) depending on the probability of violating a timing constraint (e.g., if the reimbursement process has been running for more than a given threshold time, a faster execute reimbursement service may be selected for s_4).

Monitoring at the PSM level is therefore needed to perform adaptivity and constraint enforcement functionality. In addition, monitoring rules can be defined to select the messages to be logged in order to be able to perform monitoring at the above levels. For instance, if the daily limit for board expenses is violated, the Request for reimbursement process is executed taking the "no" path as shown in Figure 4. However, to analyze the requirements satisfaction at PIM level, as noted above, information about the number of reimbursements above the daily limit and the total number of reimbursements is to be collected by the monitoring system. Monitoring rules can be defined using a monitoring system able to analyze the exchanged messages, such as defined in [13].

5 KPI Complexity

The literature on Business Process Monitoring has achieved today huge dimensions. Around this problem we have studies involving BPM infrastructures,

Performance Indicators categorizations, BAM methods. In spite of this strong development very few lines were wrote on criteria for comparing the performances of two monitoring systems. The main reasons behind this lack are related to the difficulty to identify a uniform term of comparison. For instance the performance of a system are dependent on the dimension of the data in analysis or on the computational power in use.

To avoid being affected by all these variable factors, we propose here to evaluate the complexity of implementation of a measure, reducing its representation to the set of inputs required by the measure itself. The complexity of implementation is the effort required to transform the primitive data in our system into the inputs involved in the given measure. More concretely we can represent the complexity by the following formula:

$$C_m = \sum_{i=1}^n c_{im} * K. \quad (8)$$

Where c_i are the single inputs used in the measure m and K is a generic complexity of integration of distinct inputs. The complexity represented by K can vary according to the degree of homogeneity the data we are integrating have. Any c_i must be mapped to a specific set of data available in output from the system. To understand how much this mapping is complex we need to define:

- d_i for the single input data required by the measure under evaluation;
- a_l for output data that can be mapped without any manipulation;
- $b_j = f(a_j, \dots a_n)$ for output data that must be manipulated by a single function;
- $e_k = g(a_l, \dots a_n, b_j, \dots b_n)$ for output data that must be manipulated by combination of functions.

Now we need to assign a value to this data according to a unit of measurement of information storage, and to assign a multiplicative value to these functions, for instance staying in the range [1 – 5]. This way the complexity of mapping one input of our measure with the output data from the system can be see as the ratio among the dimension of output data and the dimension of the output data multiplied by the required manipulations. In formula:

$$c = 1 - \frac{\sum d_i}{\sum a_l + \sum b_j + \sum e_k}. \quad (9)$$

The last passage is to show how this formula works. Let take as an example the Rule 3 and Rule 5. Using the **TEKNE** infrastructure Rule 3 insists on data that are directly produced by the system, when we estimate the *Board_day_expense* we directly have the information to verify that rule. Its cost is equal to 0 because in the ratio we have two factors with equal value:

$$c = 1 - \frac{d_1}{a_1}. \quad (10)$$

This is because $d_1 = a_1 = \dim(\text{Board_day_expense})$. With Rule 5 the result will be different because the predicate *ReimbursementProcess* is not mappable as a

simple output of our system but we have to evaluate the time difference among the *start event* of a process and the *end event*. In this case the cost will be positive. The function will be a ratio among one simple input and one composite function of outputs.

$$c = 1 - \frac{d_1}{b_1}. \quad (11)$$

For example if the $b_1 = f(a_1, a_2)$ and $f = 3$, $b_1 = 3(a_1 + a_2)$, and the complexity of implementation of Rule 5 is equal to $c = 1 - \frac{1}{6} = 0.83$.

6 Related Work

In [14], SBVR rules are used for defining policies to be conformed, but this work is limited to access control. In paper [15] authors introduce the assertion language XSAL whose main purpose is to express business rules in the form of assertions over business processes in order to verify this rules when executing a business process. Another work [16], leveraging semantic control of BP, is focused on access control on transaction execution and on supporting financial control objectives. The work in [17] aims at modeling compliance measures based on policies and presents a framework for managing and enforcing compliance policies on enterprise models and BPs. Such work suggests to utilize SBVR to state the policies through business rules. In [18] authors define an approach to check business processes against rules emerging from business contracts. In [19] the logical language PENELOPE is presented. Such language enables verification of temporal constraints arising from compliance requirements on effected business processes. Compared with the existing research, our work explores a peculiar approach based on the idea of directly use business rules to define indicators in the monitoring activity. Such indicators can be referred to different levels (strategic, operative and execution) and are described using languages and notations familiar to business people.

Monitoring of service-based compositions is being studied in software and services areas. In [20], rules are inserted in a process to monitor its functionalities and performance, focusing on the execution of a single service. In [13], the monitoring infrastructure allows analyzing logs of messages according to monitoring rules that are specified by the designer of the monitoring functionality. In this case the focus is on analyzing series of messages arising from multiple executions of the same process model. In the approach propose in TEKNE, the goal is to be able to monitor the process at different architectural levels, and therefore analyzing at the same time both short term and long term behavioural patterns.

Furthermore, the relationships between business requirements, expressed by the definition of KPIs, and QoS have been scarcely addressed in past contributions. KPI and services have been initially compared in [21]. In this work, author indicates that business requirement definition at the PIM level should drive QoS, which in turn should drive the infrastructure requirements. In [22], the need for mapping KPIs to SLA parameters of services and infrastructure is expressed, while in [23] a service requirement can be associated with many KPIs so as to be able to move

from the service space to the KPI space and introduce constraints on the KPIs. The approach presented in this paper considers the linkage between service characteristics and business requirements already analyzed in these contributions but also proposes the Web service monitoring as a method to enable the KPI assessment.

In conclusion we remark that today market provides professional Business Process Management (BPM) tools to monitor, administrate, measure, and analyze the performance of individual and end-to-end processes. Anyway these solutions are often limited to centralized organizations. In the TEKNE project we explored the potentialities of SW technologies for BPM, especially in distributed environments. Our framework supports design and monitoring at strategical, operational, and executive level, allowing the definition of ad-hoc Directives and KPIs.

7 Concluding Remarks

Current technology provides professional Business Process Management (BPM) tools to monitor, administrate, measure, and analyze the performance of individual and end-to-end processes. Anyway these solutions are often limited to centralized organizations. Making BPM available to the Networked Enterprises is still an open issue. Networked Enterprises are decentralized by definition, with disparate business units managing different business platforms. In this context the definition of standards supporting data portability, but maintaining the independence of the data sources is a basic requirement. In the TEKNE project we are exploring the potentialities of SW technologies for Internetworked Enterprises based on BPM, focusing on metrics for monitoring processes at different levels defined according to the MDA approach. Our framework supports design and monitoring at strategic, operational, and executive level, allowing the definition of ad-hoc Directives and KPIs for internetworked processes. We assume that monitoring rules can also result on adaptation during process execution, through dynamic service selection to support the enforcement of business rules. Also we provided a method for evaluating the complexity of implementation of a measure. This method is based on the comparison among the inputs required by the measures and the data provided in output by the informative system. This method provides an important instrument to evaluate the feasibility of a monitoring program on a set of distributed services that constitutes the Internetworked Enterprise.

Acknowledgments. This work has been partly funded by the Italian Ministry of Research Fund for Basic Research (FIRB) under project RBNE05FKZ2_004 TEKNE.

References

1. van der Aalst, W.M.P., ter Hofstede, A.H.M., Weske, M.: Business Process Management: A Survey. In: van der Aalst, W.M.P., ter Hofstede, A.H.M., Weske, M. (eds.) BPM 2003. LNCS, vol. 2678, pp. 1–12. Springer, Heidelberg (2003)
2. Oracle: Business Activity Monitoring
3. Workflow Management Consortium: The workflow reference model
4. O'Brien, J.A.: Introduction to Information Systems: Essentials for the Internetworked Enterprise. McGraw-Hill Education (2000)

5. Aßmann, U., Aksit, M., Rensink, A. (eds.): MDAFA 2003/2004. LNCS, vol. 3599. Springer, Heidelberg (2005)
6. Ceravolo, P., Damiani, E., Fugazza, C.: Representing and validating digital business processes. In: Proc. of the 3rd International Conference on Web Information Systems and Technologies, WEBIST (2007)
7. Arigliano, F., Ceravolo, P., Fugazza, C., Storelli, D.: Business Metrics Discovery by Business Rules. In: Lytras, M.D., Damiani, E., Tennyson, R.D. (eds.) WSKS 2008. LNCS (LNAI), vol. 5288, pp. 395–402. Springer, Heidelberg (2008)
8. Corallo, A., Taifi, N., Passiante, G.: Strategic and managerial ties for the new product development. In: The Open Knowledge Society. A Computer Science and Information Systems Manifesto, pp. 398–405 (2008)
9. Ardagna, D., Comuzzi, M., Mussi, E., Pernici, B., Plebani, P.: PAWS: A framework for executing adaptive web-service processes. *IEEE Software* 24(6), 39–46 (2007)
10. BPMN: Business process modeling notation (bpmn). Misc, OMG (2006), <http://www.bpmn.org/Documents/BPMN%20V1-0%20May%203%202004.pdf>
11. Plebani, P., Pernici, B.: URBE: Web service retrieval based on similarity evaluation. *IEEE TKDE* (November 2009)
12. Jackson, M.: Problem Frames: Analyzing and Structuring Software Development Problem. Addison-Wesley (2001)
13. Buccafurri, F., Meo, P.D., Fugini, M.G., Furnari, R., Goy, A., Lax, G., Lops, P., Modafferi, S., Pernici, B., Redavid, D., Semeraro, G., Ursino, D.: Analysis of QoS in cooperative services for real time applications. *Data Knowl. Eng.* 67(3), 463–484 (2008)
14. Goedertier, S., Mues, C., Vanthienen, J.: Specifying Process-Aware Access Control Rules in SBVR. In: Paschke, A., Biletskiy, Y. (eds.) RuleML 2007. LNCS, vol. 4824, pp. 39–52. Springer, Heidelberg (2007)
15. Lazovik, A., Aiello, M., Papazoglou, M.: Associating assertions with business processes and monitoring their execution. In: Proceedings of the 2nd International Conference on Service Oriented Computing (2004)
16. Namiri, K., Stojanovic, N.: A formal approach for internal controls compliance in business processes. In: 8th Workshop on Business Process Modeling, Development, and Support (2007)
17. Kharbili, M.E., Stein, S., Pulvermiller, E.: Policy-based semantic compliance checking for business process management. In: CEUR Workshop Proceedings, Gemischer Workshop zu Referenzmodellierung und Semantische Geschäftsprozessmodellierung, Saarbrücken, Germany, vol. 420, pp. 178–192 (November 2008)
18. Governatori, G., Milosevic, Z., Sadiq, S.: Compliance checking between business processes and business contracts. In: Proc. EDOC 2006, pp. 221–232 (2006)
19. Ghose, A., Koliadis, G.: Auditing Business Process Compliance. In: Krämer, B.J., Lin, K.-J., Narasimhan, P. (eds.) ICSOC 2007. LNCS, vol. 4749, pp. 169–180. Springer, Heidelberg (2007)
20. Baresi, L., Guinea, S.: Towards Dynamic Monitoring of WS-BPEL Processes. In: Benatallah, B., Casati, F., Traverso, P. (eds.) ICSOC 2005. LNCS, vol. 3826, pp. 269–282. Springer, Heidelberg (2005)
21. Wustenhoff, E.: Service level management in the data center. Technical report (2002)
22. Wetzstein, B., Karastoyanova, D., Leymann, F.: Towards management of SLA-aware business processes based on key performance indicators. In: 9th Workshop on Business Process Modeling, Development, and Support, BPMDS 2008 (2008)
23. Motta, G., Pignatelli, G.: Performing business processes knowledge base. In: First International Workshop and Summer School on Service Science (2007)