

Towards Resource-aware Business Process Development in the Cloud

Emna Hachicha, Walid Gaaloul

Computer Science Department,
TELECOM SudParis, CNRS UMR Samovar
Evry, France
{firstname.lastname}@telecom-sudparis.eu

Abstract. In recent years, cloud environments are becoming more and more interesting and useful for the execution and the deployment of business processes. Indeed, it enables organizations to reduce their costs and optimize their processes. Many researches have been realized for providing support and enhancement to the resource perspective in business processes. Nevertheless, they have basically focused on human resources and have neglected other types of resources. This paper fills this gap by proposing an extension to the BPMN metamodel in order to optimally manage resources deployed in the cloud through resource constraints verification. The purpose of our approach is to enable Business Process development to benefit from economies of scale, faster provisioning times, decreased runtime costs, and reduced energy consumption. To do so, we aim at enriching Business Process Models with a semantic knowledge base about the consumed cloud resources that can be used to optimize resource management.

Keywords. BPM, Resource Management, Cloud Computing

1 Introduction

BPM and *Cloud Computing* are two major paradigms nowadays. On the one hand, *Cloud Computing* has recently become a matter of great importance in many sectors. It provides several significant benefits to its users (elasticity, lower costs, reduced energy consumption, etc). On the other hand, *Business Process Management (BPM)* represents an essential approach to continuously improve organizations's processes. Particularly, resource management awareness motivate organizations to better consider the resource perspective of their processes.

A business process might be specified and designed from many perspectives like control-flow perspective or organizational perspective which have been considered in modeling languages. Despite of various efforts on integrating the resource perspective in business processes's field, it remained poorly described and operated. This perspective refers to the link between the different activities constructing the process and the resources which are needed to be consumed. There is a lack of representation of other resources in addition to the human ones. Furthermore, there was a total absence of the cloud resources representation in

such field. The fact of considering resources and its allocation constraints is a crucial issue. Likewise, we can say that dependencies between resources have not been well defined so far. Basically, most of the existing approaches and techniques adopt a non semantic resource modeling. On the contrary in our approach we do need the semantic aspect because of its benefits. Regarding to syntactic models, formal semantic models came to struggle its limitations. They provide common descriptions for heterogeneous contents, decrease ambiguities, and allow more efficient and accurate results. Moreover, seeing that cloud environments are distributed and heterogeneous, the need for a common vocabulary in order to share knowledge is a crucial issue. Thus in order to ensure this interoperability between cloud providers, we aim at building a semantic knowledge base as a first step towards the optimization of resource management.

From the above, it becomes clear that an approach that explicitly integrates resource management knowledge and cloud resources into the business process environment is missing. The goal of this work is to fill this gap by proposing an extension to the BPMN model for the enhancement of the resource perspective. We argue that business processes owners could make use of cloud resources and thereby improve their management by incorporating the semantic notion through a resource management knowledge.

The remainder of this paper is organized as follows: In section 2 we provide an overview of the related work. After that, we illustrate a motivating example in Section 3. Section 4 introduces resource perspective concepts in Business processes and defines used cloud resources. An overview of our approach is presented in section 5. Section 6 discusses the proposed extension to BPMN. Afterwards, in section 7 we present our approach to address previously mentioned shortcomings. Then, we formalize definitions about our ontology aiming to verification in Section 8. The approach validation beneath Signavio platform is then depicted in section 9. Finally, the paper closes with conclusions and future works.

2 Related Work

There exists previous works on the representation of the resource perspective in Business Process Management. In [15], authors propose an approach that enables the implementation of resource perspective in BPMN and BPEL languages. The same authors have addressed the modeling and visualization of resource perspective requirements by extending the BPMN 2.0 metamodel [14] which was validated against workflow resource patterns [12]. These works support the three aspects of resource view: Resource structure, work distribution, and authorization. However they do not enable the representation of cloud resources and do not consider security and privacy properties. On the contrary our approach integrates cloud aspect and seeks for checking resource allocation properties.

A general framework for document-driven workflow systems was proposed in [17]. It enables discovering data dependencies between tasks in a process with the intention of assisting workflow designers and achieving more efficient control

flow design. While in our proposal our purpose is defining more explicitly these interrelations additioning relations depending on resource flow.

Authors in [16] extend the WS-HumanTask architecture with the goal to improve support to the resource perspective requirements in BPEL processes. In [11] authors propose a metamodel for configurable processes which integrates resources: human, data and artifacts. Nevertheless they have overlooked semantics in terms of relationships between resources and activities. Also they have neglected cloud resources. Whereas our proposal deals with these limitations.

Some other studies have capitalized on social networking and argue its utility for web services and Business Process Management by means of taking into account and defining dependencies between its components [9], [10], [8] and [7]. Approaches in [2] and [13] have considered the human resource as cloud compute units via respectively proposing a framework considering group of people as a social compute unit (SCU) and applying this concept in the context of resolution of incidents in an IT service organization. Notably, the authors in [4] have been proposed a resource-efficient scheduling algorithm for BP and cloud-based computational resources. Different from our proposed approach, cloud resources and resource dependencies are not well defined in business processes.

Table 1 depicts a comparative table for the evaluation of some of the above mentioned approaches according to various assessment criteria mentioned in the first row of this table (resource perspective, resource dependencies, etc). The choice of these evaluation criteria is argued by the fact that we are interested in integrating the representation of resources as well as their dependencies whether classical, human or even cloud resources. Seeing that BPMN is the most widely used language, it is important to consider whether this approach adopt it as a modeling language. Also we are interested to verify if there is any support of UML modeling and semantic representation. Besides, Cloud Computing is receiving a high level of attention, which is an evidence given the importance of its advantages such economies of scale, reducing runtime costs, improving flexibility, etc. To allow benefiting from these advantages in the field of Business Process Management, we propose as well to integrate cloud resources.

Table 1: Evaluation of previous approaches

Approaches	Assesment Criteria						
	Resource Perspective	Resource Dependencies	Considering HumanTask	Cloud resources	BPMN	UML Modeling	Semantic Stack
[16]	+	-	++	-	-	+	-
[14]	+	-	+	-	+	+	-
[8]	+	+	+	-	-	-	-
[4]	+	-	+	+	-	-	-
Our Approach	++	++	+	++	+	-	++

All approaches have considered resource perspective and particularly focused on human task however almost all of them neglected cloud resources except [4]. Likewise, we can deduce that there is a lack of semantic modeling models. Moreover, resource dependencies have been pointed out only in [8]. To overcome these gaps, our proposal aims at better handling these assessment criteria.

3 Motivating Example

Let us suppose the following scenario as an illustrating example represented in Fig. 1 that depicts a sequence of a set of activities that belongs to a BPMN process model. This example describes the process of ordering goods in a company. The process here starts with a customer sending an order to this company. Once the *Receive Order* is terminated, the *Check Credit* task is activated for verifying whether it is sufficient or not to satisfy the order. After this latter activity is checked, two possibilities can be presented: if the credit is not sufficient, a *Credit Order* task is launched. In case of enough credit, a set of activities are triggered (*Take From Warehouse*, *Invoice*, and *Ship*). Finally, in all cases a message indicating the response is sent to the concerned customer.

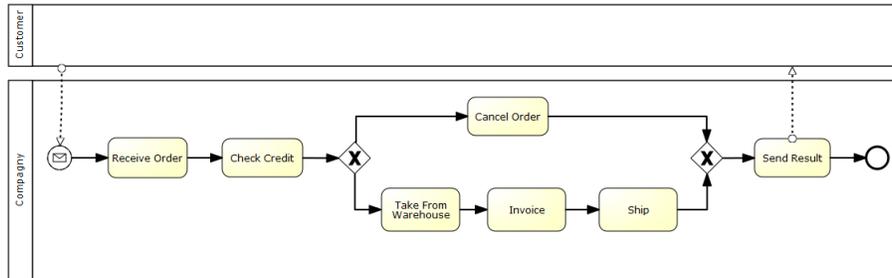


Fig. 1: BPMN Process Example

We require to attach resources to process activities whether from service providers deployed in the cloud or other organisations. In this case, this process necessitates to consume a set of various resources which are semantically interdependent. For instance the *Check Credit* activity requires a compute resource for calculating if the product price is lower than his bank credit and also if the bank allows this flow. In addition, we have to verify resource allocation constraints. In fact these restrictions represent a set of properties of security, privacy and optimization. For example a resource could be private for one particular task so it cannot be shared by any other activity. In order to guarantee security for instance when a resource disappears another or some resources could replace it depending on their characteristics.

In fact, current business processes descriptions are not well defining resource interrelationships and cloud resources. Relationships between activities and resources have to be taken into account as well as resources restrictions. For this purpose, we developed a semantic model which we use to annotate business processes and better manage consumed resources than previous researches.

4 Preliminaries

This section provides preliminaries on resource perspective aspects and concepts of cloud computing resources in order to support our proposal. This section is structured as follows. Section 4.1 distinguishes three aspects of the resource perspective. Section 4.2 identifies cloud resources.

4.1 Resource perspective in Business Processes

As mentioned above, the resource perspective in business processes is not well defined. But with regard to recent efforts realized to bridge this gap, three important aspects are defined in a process model with the aim to describe the resource perspective: resource structure, work distribution, and authorization [15].

The resource structure component involves two aspects: the characterization and classification of resources. The characterization is the definition of informations about resources. The classification is joining resources with a concept. It enables assigning resources with sets of common properties. The work distribution aspect is concerned with the work distribution and its binding to specific resources for execution. In other words, it defines the manner in which the work of a process is distributed and allocated to resources. The authorization aspect deals with the definition of privileges that resources own with regard to the execution of operations in order to organize the work advertised to them.

As mentioned in section 2, these concepts have been considered through efforts on extending the resource perspective description based on the BPMN 2. In our work, we have considered these concepts to construct a semantic model that aims at enhancing the description of resource perspective in business processes.

4.2 Cloud Computing resources

The Cloud permits to deliver three important types of resources on demand which are: computing, storage, and networks.

The computing resources provide mechanisms in order to deploy and run softwares. They are capable of delivering Human-based services (HBS) [2]. This type of resource depicts an information processing resource (e.g. virtual machine).

The Cloud Storage simply represents an information recording resource in data storage devices. It offers great benefits e.g: reliability, faster deployment, reduced costs, ensuring protection in case of loss. Actually there is different ways to use this type of resources depending on users requirements: private data, shared data. Besides enterprises can request a partial or a total management

of its data by the cloud storage provider. Whereas network resources allow to have mechanisms that are used for communication, and might also offer added-value services such as load balancers [5]. The network type plainly denotes an interconnection resource (e.g a virtual switch).

Obviously these resources types are caught from the cloud resource description in specific cloud computing APIs such as OCCI [1] and/or TOSCA [3].

5 Approach Overview

In this section, we present an overall overview of our approach, as shown in Fig. 2, which is based on three key inputs: (i) Business processes described in BPMN language, (ii) Cloud resources deployed using one of the two most successful standards which are OCCI and TOSCA, and (iii) Cloud resource restrictions that comprise rules and properties to be verified and checked. This latter entry is of a high importance because, by properties verification and constraints compliance, it aims at ensuring the proper management of resources utilization. The constraints and properties can formally specify such requirements: for instance, an activity consumes a resource that cannot be shared by other activities, or a network resource must be only in a private cloud. As another example, a task can only consume a specific human resource. Or, if a storage resource with a specific capacity disappears at run-time, a rule can indicate that two storage resources can replace it with equivalent capacities.

We have established a CloudPrO ontology (see Fig. 5) which takes into account these three entries and formally describes dependencies between resources of business processes in a semantic way. Indeed, business processes and cloud resources components are modeled using RDF/RDFS language. Whereas, the cloud resources constraints is modeled using rules SWRL ¹.

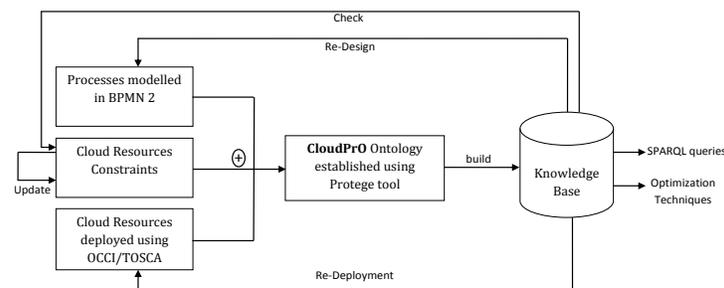


Fig. 2: Approach Overview

¹ SWRL: A Semantic Web Rule Language Combining OWL and RuleML, <http://www.w3.org/Submission/SWRL/>

We advocate that our approach allows to benefit at both levels: on the one hand, at design-time we hold some rules to check targeting resource correct distribution and management. On the other hand, at the run-time level we also have rules aiming at a reliable resource consumption.

In the following, we present in section 6 our proposed BPMN extension. Whereas in section 7 we discuss our model to semantically specify cloud resources in business processes. Resource allocation properties are defined in section 8. Section 9 elaborates the validation of our approach.

6 BPMN Extension

The *Business Process Model and Notation (BPMN)*² is widely applied as business process modeling language. The primary goal of BPMN is to provide a notation that is easy to understand by all business users whether business analysts, technical developers, or also business people. It offers a set of diagrams to design business processes. This latter version of BPMN incorporates resource perspective concepts which are resource assignment and human interactions. Nevertheless, it does not provide explicit definition of resource type, also it does not include a semantic level.

In the following, we propose an extension, which called *ResourceExtension*, to the BPMN element named *Resource* (see Fig. 3).

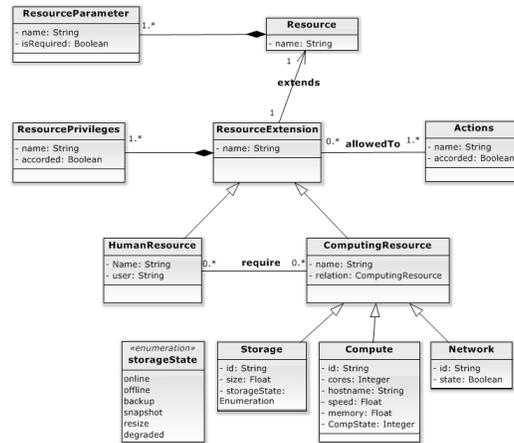


Fig. 3: Extending the resource element in BPMN

We have described the *ResourceExtension* as an *ExtensionDefinition* which is linked with other proposed extension elements: *ResourcePrivileges*, *Actions*,

² OMG: Business Process Model and Notation (BPMN), v.2 (2011), <http://www.omg.org/spec/BPMN/2.0/PDF/>

HumanResource, *ComputingResource*, *Storage*, *Compute*, and *Network*. We note that we have adopted and relied on the OCCI standard in order to define cloud resources structure, however the occi integration is out of scope and we will not handle this issue in this paper.

Concretely, we specified firstly these proposed extension definitions in an xsd document including the dependencies described between different entities. Thus by importing our xsd extension in the BPMN document, we extend the BPMN *Resource* by putting into the core of its tag the corresponding extension elements. After that, we fill in the attributes with the suitable values in accordance with the definition that we did in the xsd document.

7 Semantic Model for resource management in business processes

As discussed above, our proposed resource-aware approach differs from previous approaches in that it relies upon making use of the cloud resources: storage, compute, and network into BPMN processes to lower runtime costs, economies of scale, decreased energy consumption. Additionally, it focuses on the manner of managing these resources by means of a semantic knowledge base. We have also represented a semantic description defining more explicitly dependencies between entities. This description is evidently linked with the existing BPMN description. To do so, we have modeled a *CloudPrO* ontology written in *RDF/RDFS* format (See Fig.5). Our modeling is an extended view from the basic model ontology (See Fig. 4), which is defined as a tuple $\langle C_{bp}; D_{bp}; P_{bp}; R_{bp} \rangle$ where C_{bp} represents concepts, D_{bp} data or attributes, P_{bp} are properties, and R_{bp} are rules set, adding semantic enhancement on resource concept. The conceptual model depicted in Fig. 4 is based on *BPMO* [6] ontologies developed in the European project *SUPER*³ that aims at providing a semantic-based and context-aware framework. Formally, the *CloudPrO* ontology is a 4 tuple $\langle C; D; VP; R \rangle$ where C represents concepts set, D data about concepts, VP the verification properties, and R are the set of rules to be respected.

Activities composing a process require resources to meet the users needs. Dependencies between the activity and the resource elements can be defined and analyzed according to three several ways: *Flow*, *Fit* and *Sharing* [17]. A *flow* dependency means that one activity generates a resource that is consumed by another activity. Concerning the *Fit* relation, it deals with the case when various tasks produce one resource. Whereas the *Shared* dependency appears when a single resource is consumed by multiple activities.

The *Resource* element has two subclasses: *HumanResource* and *ComputingResource*. *HumanResource* resources defines persons that need activities. Dependencies between humans are defined by properties linked between each others. Here we defined three types of relationships: *Substitution*, *Delegation*, *Peering* (see Table 2). For instance, the *Substitution* relation allows to a person pj to

³ <http://www.ip-super.org/>

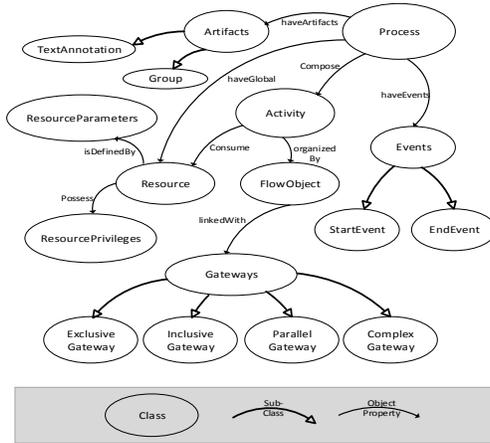


Fig. 4: Model Ontology for Business Process

replace pi if this latter is planned to be not available. *Delegation* dependency has the same meaning but pi is not available in unexpected way. Whereas two persons are in a *Peering* relation if the activity needs their both capacities.

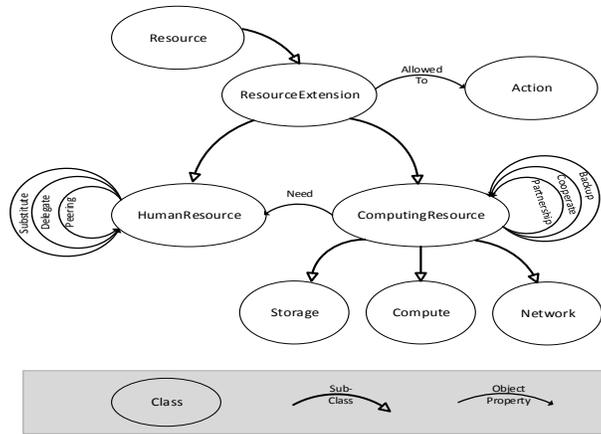


Fig. 5: The CloudPrO Ontology

ComputingResource enables the definition of another type of resources including specifically the cloud resources (storing, computing, networking). There are relations linking this resource type: *Cooperation*, *Partnership*, *Backup*. Two resources are in a *Cooperation* relation if the activity requires both capacities and

those capacities are similar. It stills the same thing for the *Partnership* relation however the two resources have complementary capacities. Finally, a resource might replace an another if this latter fails and have obviously similar capacities. Indeed, we define a *need* dependency between the two major resource types exemplifying cases when a person calls for a computing resource or the inverse. From a work items point of view, both resource subclasses are depending on available list of actions which may be executed and dedicated to resources, i.e: creating, editing, duplicating resource, etc. These operations are established by the class *Actions*. In addition, the *ResourcePrivileges* refers to the authorization aspect and organizes the distribution of work operations at the resources level.

Table 2: Resource Dependencies Descriptions

Dependency Name	Description
<i>Substitution(pi,pj)</i>	pj replaces pi if expected unavailability of pi
<i>Delegation(pi,pj)</i>	pj replaces pi if unexpected unavailability of pi
<i>Peering(pi,pj)</i>	Execution of pi and pj if concerned activity needs both capacities
<i>Cooperation(ci,cj)</i>	Execution of both resources ci and cj if concerned activity needs both similar capacities
<i>Partnership(ci,cj)</i>	Execution of both resources ci and cj if concerned activity needs both complementary capacities
<i>Backup(ci,cj)</i>	cj replaces ci if failure of ci and capacities of ci and cj are similar

8 Privacy, security and optimization properties

After acheiving the infrastructure design, we define in this section a set of properties and rules which serve as a template that have to be followed aiming at assisting business process designers for defining their proper properties and rules.

8.1 Verification Properties

Resource Description Framework (RDF/RDFS) as a family of *W3C* specifications formalize informations on web resources. It describes relationships among resources. Indeed, the *RDF* metamodel ⁴ is composed of: *RDFSResource*, *RDF-Statement* which plays a triple role which are a subject, an object or even as a predicate, and *RDFProperty* which links subject resources and object resources.

Concretely, a verification property is a sub-class of the *RDFProperty* which has four other sub concepts. We formalize its definition as follow.

As depicted in Definition 1, the first set of properties describe relationships between concepts composing business process included in Fig. 4. For instance,

⁴ OMG: Ontology Definition Metamodel, v1.1 (2014), <http://www.omg.org/spec/ODM/1.1/PDF/>

activities can be joined to gateways via the *linkedWith* property passing by the *FlowObject*. Also, a process can own global resources via the *haveGlobal* property.

Definition 1 (*Process elements related Properties*)

A *VP* is sub-type of *RDFProperty*

where *VP* is an *RDFStatement* which have as a subject C_i and an object C_j and $C_i, C_j \subset C_{bp}$

The second set, as specified in Definition 2, establish the connection between activities as a subject and resources as an object. The opposite is possible with a resource subject and an activity object. This is explained by the fact that an *Activity* needs one or more *Resource* through the *consume* dependency.

Definition 2 (*Activity/Resource related Properties*)

A *VP* is sub-type of *RDFProperty*

where *VP* is an *RDFStatement* which links a subject C_i and an object C_j and C_i is an instance from the *Activity* concept and C_j instance extending from the *Resource* concept.

The third set of properties, defined in the Definition 3, bears on work distribution concept for resource perspective. As discussed earlier in Section 7, each resource is allowed to execute specific operations including in the *Action* concept.

Definition 3 (*Action/Resource related Properties*)

A *VP* is sub-type of *RDFProperty*

where *VP* is an *RDFStatement* which links a subject C_i and an object C_j and C_i is an instance from the *Resource* concept and C_j instance extending from the *Action* concept.

Finally, the latter set concern relationships between different specializations of *ResourceExtension*. It can be defined with various dependencies. It exists three types of properties binding human resources (*substitution*, *delegation*, *peering*). About relationships between *ComputingResource*, it exists the *backup*, *cooperation*, and *partnership* as verification properties. Moreover, we designate a *need* property as a predicate between the *HumanResource* and *ComputingResource*.

Definition 4 (*Process Resource related Properties*)

A *VP* is sub-type of *RDFProperty*

where *VP* is an *RDFStatement* which links a subject C_i and an object C_j and C_i, C_j are instances extending from the *Resource* concept.

8.2 Rules formalization

We define herein concretely a set of rules models. These rules, which provide coherency description between concepts, are expressed using the *Semantic Web Rule Language (SWRL)* which is based on a combination of the *Web Ontology Language (OWL)* and the *Rule Markup Language (RuleML)*. As we define after, a rule axiom consists of an antecedent and a consequent. It may be read as meaning that if the antecedent is “true”, then the consequent must be “true”.

Rule: (“Antecedent \Rightarrow Consequent”)

i.e: $A_1 \wedge A_2 \dots \wedge A_n \Rightarrow C$

In our context, we specify three models of rules: simple rules, complex rules, and dependency-based rules.

Firstly, the simple rules as defined in Definition 5 concerns only one resource type. For instance, a rule may signify that a network resource can be only in a private cloud not in a public one.

Definition 5 (*Simple Rules*)

A simple rule R is defined

$$iff \wedge_i Pr_{P_i} \wedge_j Pr_{C_j} \Rightarrow Pr_{CloudPrOn}$$

where Pr_{P_i} and Pr_{C_j} are respectively predicates defining relationships and concepts concerning only one resource type.

Then according to the Definition 6, the second model of rules deals with more than one resource type. For example, a complex rule can imply that a certain number of storage resources with specific attributes must substitute a storage resource which was vanished.

Definition 6 (*Complex Rules*)

A complex rule R is defined

$$iff \wedge_i Pr_{P_i} \wedge_j Pr_{C_j} \Rightarrow Pr_{CloudPrOn}$$

where Pr_{P_i} and Pr_{C_j} are respectively predicates defining relationships and concepts concerning more than one resource type.

The latter model of rules $DbRules_{CloudPrO}$ relies on dependencies addressed in Table 2. As example, a computing resource c_i can appeal to another computing resource c_j to execute together in order to achieve a specific processing (*Cooperation*(c_i, c_j) or *Partnership*(c_i, c_j)).

Definition 7 (*Dependency-based Rules*)

A dependency-based rule R is defined

$$iff \wedge_i Pr_{P_i} \wedge_j Pr_{C_j} \Rightarrow Pr_{CloudPrOn}$$

where Pr_{P_i} and Pr_{C_j} are respectively predicates defining relationships and concepts concerning dependencies already defined.

9 Validation

The *Signavio Core Components*⁵ tool is an open source web-based application and a powerful tool for mastering process management supporting the BPMN 2.0 standard. In order to validate our proposed approach⁶, we have extended this modeling tool. We developed a plugin that takes into account our CloudPrO ontology and highlights resource management in business processes modeling.

To do so, we refer herein to the motivating example discussed in Section 3 to show how to capitalize on the CloudPrO ontology for business processes.

Every activity in this process can be semantically linked to resources to consume. Both of *Receive Order* and *Send Response* activities need the consumption of two instances of the Storage resource which is annotated as a concept in the

⁵ The Source Code at: <https://code.google.com/p/signavio-core-components/>, The Academic version at: <http://academic.signavio.com/>

⁶ <http://www-inf.it-sudparis.eu/SIMBAD/tools/CloudPrO/>

CloudPrO ontology inheriting from the ComputingResource concept. These instances have respectively identifiers “stor1” and “stor2” and have both 1 Go as a size. To specify these relations, the user selects the concerned activity. Then, a panel encompassing the set of resource instances, including the cloud ones, appears. Once the user chooses the appropriate instance to consume, our tool engenders a text annotation binding the selected activity with this resource instance, which comprises the resource properties and if exist its dependencies with other instances (See Fig. 6). Likewise, the *Invoice task* consumes a Compute resource which its identifier is “compute2” and in a partnership relation with a storage resource “stor2”. The *Check Credit* and *Invoice* tasks consume Compute resources which respectively have identifiers “comput1” and “comput2” to do the appropriate processing. In addition, a human resource that have identifier “human1” is consumed by the *Cancel Order*.

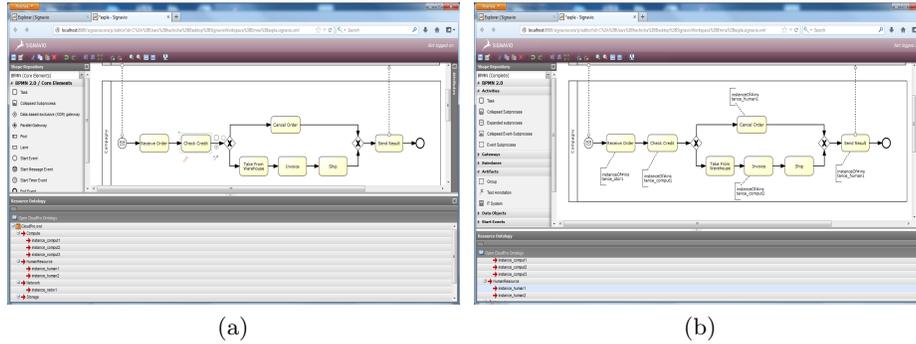


Fig. 6: Two screenshots of the application, (a) the user selects the appropriate resource from the ontology instances displayed in the bottom window, (b) resources are attached with activities as a text annotation

As explained in our approach, we defined also two types of constraints: at run-time and at design-time. Concretely, two run-time restrictions are specified as follow: if one of both storage resources disappears than two other storing resources can replace it with capacities of 500 Mo for each. The second restriction indicates that the compute resource “comput1” is a not shared resource. In other words, once it is instantiated it cannot be consumed by another activity at the same time. So if another activity selects this resource to consume it, this implies a constraint violation. Second, a constraint, which is applied at design-time, reflects the fact that the *Cancel Order* activity have only the right to use this particular instance of HumanResource with its specific attributes values. We use Protege tool⁷ including Pellet reasoners, which is an open-source ontology editor and framework, to detect such anomalies.

⁷ The Source Code at: <http://protege.stanford.edu/>

As discussed above, these rules enrich our approach and may serve for checking and verification at design-time, and management at run-time. We present below the SWRL descriptions referring to our rules.

Table 3: SWRL Rules for resource constraints

Rule Name	SWRL
$Rule_{CloudPrO1}$	$Activity(A) \wedge consume(A,X) \wedge not Storage(X)$ $\Rightarrow Storage(Y) \wedge Storage(Z) \wedge hasSize(Y,size1)$ $\wedge hasSize(Z,size2) \wedge consume(A,Y) \wedge consume(A,Z)$
$Rule_{CloudPrO2}$	$Activity(A) \wedge consume(A,X) \wedge Compute(X)$ $\Rightarrow Activity(B) \wedge not consume(B,X)$
$Rule_{CloudPrO3}$	$Activity(A) \wedge HumanResource(X) \wedge hasName(X,name1)$ $\Rightarrow HumanResource(Y) \wedge hasName(Y,name2) \wedge not consume(A,Y)$

As figured in Table 3 the $Rule_{CloudPrO1}$ means that if an activity A consumes a storing resource X with a specific size but X has vanished, A consumes two storing resources Y and Z with particular sizes. For the $Rule_{CloudPrO2}$, it refers to the not shared compute resource X. It means that no other activity B can consume it if an activity A is already utilizing it. Whereas $Rule_{CloudPrO3}$ implies that the activity A can only use the human resource X with a specific name.

The resource consumption brings along various dependencies between resources. Through CloudPrO ontology, we better manage the use of resources in a business process and handle difficult situations like resource execution failures.

10 Conclusions & Future Works

This paper is a starting point into a broader work to enhance resource operating in business process development as well as integrating cloud resources. Based on the proposed extension, we improved the later version of BPMN through proposing an ontology that ameliorates the resource perspective description in a semantic way while validating resource constraints. We implemented our approach through an evaluation demonstrating the efficiency of our solution.

Actually, we are working on building a semantic knowledge base using our CloudPrO ontology and thereafter executing SPARQL queries aiming to an optimized management of consumed resources. Additionally, as future work, we intend to specify monitoring technique to manage the dynamic change of resources to match new requirements.

Acknowledgments

Work funded by the European Commission under the Erasmus Mundus GreenIT project (GreenIT for the benefit of civil society. 3772227-1-2012-ES-ERA MUNDUS-EMA21; Grant Agreement n 2012-2625/001-001-EMA2)

References

1. Binz, T., et al.: Opentosca - a runtime for toasca-based cloud applications. In: Service-Oriented Computing - 11th International Conference, ICSOC, Berlin, Germany, December 2013, Proceedings. pp. 692–695
2. Candra, M.Z.C., et al.: Provisioning quality-aware social compute units in the cloud. In: Service-Oriented Computing - 11th International Conference, ICSOC, Berlin, Germany, December 2013, Proceedings. pp. 313–327
3. Edmonds, A., et al.: Toward an open cloud standard. *IEEE Internet Computing* 16(4), 15–25 (2012)
4. Hoenisch, P., et al.: Workflow scheduling and resource allocation for cloud-based execution of elastic processes. In: IEEE 6th International Conference on Service-Oriented Computing and Applications, Koloa, USA, December 2013. pp. 1–8
5. Kächele, S., et al.: Beyond iaas and paas: An extended cloud taxonomy for computation, storage and networking. In: Proceedings of the 6th IEEE/ACM International Conference Utility and Cloud Computing UCC. IEEE, USA (12 2013)
6. L. Cabral, A. Filipowska, P.G.J.N.B.N.C.P.G.Z., Zoeller., S.: Process ontology stack. project ist 026850 super deliverable 1.5. In: March 2009
7. Maamar, Z., et al.: Network-based conflict resolution in business processes. In: IEEE 10th International Conference on e-Business Engineering, ICEBE 2013, Coventry, United Kingdom, September 2013. pp. 132–137
8. Maamar, Z., et al.: Towards a user-centric social approach to web services composition, execution, and monitoring. In: Web Information Systems Engineering, WISE, 13th International Conference, Paphos, Cyprus, November 2012. Proceedings. pp. 72–86
9. Maamar, Z., et al.: Using social networks for web services discovery. *IEEE Internet Computing* 15(4), 48–54 (2011)
10. Maamar, Z., et al.: Why web services need social networks. *IEEE Internet Computing* 15(2), 90–94 (2011)
11. Rosa, M.L., et al.: Beyond control-flow: Extending business process configuration to roles and objects. In: 27th International Conference on Conceptual Modeling, Barcelona, Spain, October 2008. Proceedings. pp. 199–215
12. Russell, N., et al.: Workflow resource patterns: Identification, representation and tool support. In: Advanced Information Systems Engineering, 17th International Conference, CAiSE, Porto, Portugal, June 2005. pp. 216–232
13. Sengupta, B., et al.: Who do you call? problem resolution through social compute units. In: Service-Oriented Computing - 10th International Conference, ICSOC 2012, Shanghai, China, November 2012. Proceedings. pp. 48–62
14. Stroppi, L.J.R., et al.: A bpmn 2.0 extension to define the resource perspective of business process models. In: XIV Congreso Iberoamericano en Software Engineering (CIbSE 2011), Rio de Janeiro, Brasil
15. Stroppi, L.J.R., et al.: Extended resource perspective support for bpmn and bpel. In: Proceedings of the XV Iberoamerican Conference on Software Engineering, Buenos Aires, Argentina, April 2012. pp. 56–69
16. Stroppi, L.J.R., et al.: Extending the ws-humantask architecture to support the resource perspective of bpel processes. *CLEI Electron. J.* (2013)
17. Wang, J., Kumar, A.: A framework for document-driven workflow systems. In: Business Process Management, 3rd International Conference, BPM 2005, Nancy, France, September 2005, Proceedings. pp. 285–301