Abstract—Market requirements change all the time, that’s why a business organization needs to increase the level of agility in order to respond quickly to this change. Business process modeling is a key phase in the business process engineering chain because it’s the first phase and all the other phases depend on it. Providing reusability in the process modeling can be very useful in order to reduce high complexity, time consuming and error in designing a new or redesigning an existing process. In this paper we present a business process model querying system based on Object Oriented Database Management system OODBMS. Such a system facilitates reuse of process models and can be used also to ensure compliance of business processes with legal regulations or with standards.

BPMN, Querying Business Process, Process Models, OODBMS

I. INTRODUCTION (HEADING 1)

Business environment is very complex and change constantly, that is why an enterprise needs to be agile in order to maintain its competitiveness in the market. Business agility requires access to information at business speed and responding quickly to accommodate to constant changes. Conscious about this, many enterprises are investing in business process management systems which facilitate the definition, deployment, execution, and monitoring of the business processes (BP for short). In such system, the first step is modeling. Business analysts capture business requirements and create a new process which is a complex, time consuming and error prone task. To facilitate this, what will be useful and practical is to provide business process designers with a tool that allows them to reuse previously designed business process models minimizing designs made from scratch and also validate the new designed models using techniques of checking for compliance. These, will not only simplify the modeling phase but also increase the quality and the maturity of the developed process models and reduce the cost of this phase. Process models are stored and maintained in a central place in an organization called process repository. And in order to reuse the pervious suitable models, we need to filter and search in this process repository.

In this paper, we present a framework for querying and reusing business process models; the framework is based on oriented object database management system which constitutes our repository. The remainder of the paper is organized as follow: In Section 2 we present a state of the art for business process querying. Section 3 reports on our approach for storing and querying business process models. We conclude and give an outlook on future research in section 4.

II. STATE OF ART

Ahmed Awad classifies querying into business process in [2] into three stages:

A. Querying process models

Consists on determining similarity between business process models by searching a process repository with a structural query allowing finding processes where activities and other constructs are combined in a specific constellation.

B. Querying running process instances

Mechanism for monitoring execution state enacts process instances. Querying here is a tool of Business Activity Monitoring BAM.

C. Querying execution logs

Also known as business process mining [16]. Consists on searching over already terminated business process instances and is concerned with extracting knowledge about business processes from their execution traces in order to discover the process definition from logs by a reverse engineering approach.
Each area receives so much attention by the research community and many publications are currently related to these fields.

As stated in the introduction, focus in our work is on querying process models. A group around Ahmed Awad and Sherif Sakr conducted research on the BPMN-Q language [1], [2]. It is a graphical language for querying control-flow-related aspects. It has been applied to detection of anomalies, e.g., deadlocks, and compliance checking [17]. An existing prototype implementation of BPMN-Q framework based on RDBMS repository [7] proves that the concept is feasible, but the language still doesn’t cover all process aspects. Steen Ryll in [11] extends the language to consider data perspective but resources and messaging between interacting processes are not yet covered.

Markovic proposes an approach for querying and reasoning on business process models using semantic web technologies [19]. He presented a formal model [18] for describing business processes integrating all perspectives describing a business process (functions, resources, data, and roles). He divided queries on static ones executed on the models semantic knowledge and dynamic ones conceptually close to process similarity calculation and querying-by-example approaches.

Milo and Beeri conducted research on the BP-QL language [20], [21], [22]. It is a query language with graphical representation, primarily designed for querying BPEL-specified process models. However, their work addresses monitoring of running processes as well, i.e. it belongs to two classes of search approaches.

III. OUR APPROACH

Our framework intends designers who search business process models fragments in order to model a new BP and reuse existing ones. In our approach we have chosen to use object model to store models into the repository instead of the relational one [1] due to the advantages of this kind of model comparing to the relational. This will facilitate models storing and also the translating phase from the modeling language to the storage format. Hence our framework is organized in three layers, the front-end, the middle and the back-end. The front-end of our framework is the user querying interface. In this layer we reuse the BPMN-Q [2] querying language. The back end is the business process models repository stored in the OODBMS [4]. In this repository business process models are defined using BPMN notation [3]. The repository can also be filled by another module using process mining and process discovery using ProM [5] as shown in Figure 1. The core is the object query process that evaluates the user query over the object schema of the business process repository.

Figure 1 shows the framework architecture with the following main components:

A. Repository

A central object database OODBMS, it stores an abstract uniform representation of process models. Figure 2 shows the object schema of our database, it consists of the BPMN metamodel. Each BP model is represented by a set of objects that compose a process. An object can be an activity, an event or a gateway. Event have a type (start, intermediate or end) and can be a message, a timer, a rule, a link, an error, a compensation, a cancel or a multiple event. There are also different types of gateways: xor, or, complex and parallel (fork or join). Activities are linked with a flow object: sequence flow or message flow. With this schema we can also store data object that transit into our models. Data objects are associated to activity or gateway as shown in figure2. OODBMS were created to handle big and complex data that relational databases could not. OODBMS have many advantages and benefits that justify our choice. First, object-oriented is a more natural way of thinking. Second, the defined operations of these types of systems are not dependent on the particular database application running at a given moment. Third, the data types of object-oriented databases can be extended to support complex data such as images, digital and audio/video, along with other multi-media operations. Other benefits of OODBMS are reusability, stability, and reliability. Another benefit of OODBMS is that relationships are represented explicitly, often supporting both navigational and associative access to information. This translates to improvement in data access performance versus the relational model. Another important benefit is that users are allowed to define their own methods to access to data and how it will be represented or manipulated. The most significant benefit of the OODBMS is that these databases have extended into areas not known by the RDBMS. Medicine, multimedia, and high-energy physics are just a few of the new industries relying on object-oriented databases. As with the relational database method, object-oriented databases also have disadvantages or limitations. One disadvantage of OODBMS is that it lacks a common data model. There is also no current standard, since it is still considered to be in the development stages [6].
B. Generic Querying Interface

The interface is a visual web where users can compose their queries using the BPMN-Q notation [2]. This notation was invented in 2007 by Ahmed Awad from the Hasso-Plattner-Institute, University of Potsdam, Germany. The language currently addresses a subset of modeling notations available in BPMN (Activities, Events, Simple gateways). The language defines seven elements as shown in figure 3.

The BPMN-Q version 1.0.3 is open source and can be downloaded from [8]. We have two options: integrate the BPMN-Q interface in our platform or use Oryx [9], [10] as the basis of our implementation effort. Oryx is an open source business process management platform and it is one of the first implementations of a process model editor, which runs completely in the browser. We note that in [11], BPMN-Q was integrated into Oryx project.

C. Querying Object Processor

It is the core engine of our architecture. It receives a BPMN-Q query, translates it into OQL [12] (object Querying Language) scripts which are executed into the OODBMS. The query result is a set of objects, our object processor performs in this phase the opposite operation and passes the relevant models to the Model Editor for displaying purposes. What we need here is to convert the OQL result to a specific format (BPMN or XPDL [13]) in order to display the model into the editor.

D. Model Editor

Designers use this interface to create a new BP and to store it in the repository. Model editor is used also to display model process results returned by the query.

E. Business process mining module

This module supplies our repository with business process models using process mining techniques. Taking a set of real process executions called “event logs” as the starting point; these techniques can be used for process discovery. Process discovery [14] allows to automatically constructing a process model reflecting the behavior that has been observed and recorded in the event log. Process models constructed are then stored in our OODBMS repository. ProM [15] is an extensible framework that supports a wide variety of process mining techniques in the form of plug-ins and we will use it to supply our repository. ProM takes as input an event logs and generates a Petri net process. The petri net process has then to be translated to BPMN format in order to save it in our repository.

IV. CONCLUSION AND FUTURE WORK

In this paper we introduce the architecture of our framework for searching over an OODBMS repository of business Process models. We have shown throughout Section 2 the state of art of business querying systems and in section 3 the different components that compose our framework. As Future work: implement the solution and prove its feasibility and integrate the semantic aspect in the framework.

REFERENCES


[8] Distribution :bpt.hpi.unipotsdam.de/pub/Public/AhmedAwad/BPMNQ_distribution_1.0.3.zip


