All Sciences Proceedings

1st International Conference on Trends in Advanced Research

March 4-7, 2023 : Konya, Turkey

ICTAR

All Sciences Proceedings
http://as-proceeding.com/

Developing a New Conceptual Schema for the Business Process Component

Berraouna Abdelkader1
University of Cherif Messaadia Souk Ahras, Algeria

Corresponding Author Email: a.braouna@univ-soukahras.dz

Abstract – Managers, market analysts, software designer, and others are eager to understand the future progression and dynamics of a market in order to optimize their business process model. This paper suggests a new approach to the conceptual schema of the business process, using MDA and component technology. The proposed method outlines the business process model using a highly abstract component-based specification, allowing for easy refinement and adjustment. The business process model can be updated by the designer through the use of model evolution rules. The design team can then refine the model according to the company manager's desired evolution scenario, guided by rules that facilitate this evolution.

Keywords – Meta-Model, Business Process, E-Commerce Business, MDA, Business Process Component, Flexibility.

I. INTRODUCTION

The fast-paced business environment requires companies to continuously adapt their operations to stay relevant. This includes adjusting to technological advancements and changing customer demands. In order to be successful, companies must be able to quickly and efficiently respond to these changes in their environment. A business process is a visual representation of a company's activities, executed by individuals, groups, services, and organizations. The ability to adapt these processes is critical for managers at both organizational and operational levels[1].

The business environment in which companies operate is constantly changing, forcing them to adapt their processes to keep up with new technological developments and customer requirements. To stay competitive, companies must quickly and effectively respond to these changes. This requires the ability to monitor changes in their environment and adjust their business processes accordingly.

Our goal is to provide a business process model based on encapsulated components, which can be refined using Model Driven Architecture (MDA) concepts. The use of business process components reduces data dependencies and allows for multiple implementations of the same component. Our approach facilitates the adaptation of the business process model by using techniques such as model refactoring and refinement.

The MDA approach involves the organization of development into different layers, from platform-independent models to platform-specific models. Our work aims to apply this concept to the business process model, allowing for refinement without reliance on code. The entire refinement process will be performed through model transformation [4],[12],[13].

However, there are limited methods available for adapting business processes. Existing approaches in the literature lack the necessary flexibility and broad vision required to handle varying circumstances. These solutions are also specific to certain domains and not easily transferable.

Our solution proposes a business process model that is composed of encapsulated activities, known as business process components. Each component is a logical unit that can have multiple implementations and is defined by its provided and required interfaces. Our approach uses a Model-Driven Architecture (MDA) approach to refine the business process model, making it more flexible and adaptable.

A business process is a graph-based representation of a sequence of activities that a company performs. It describes the order in which these activities are executed and any data dependencies between them. These processes can be carried out by individuals, groups, services, and organizations. In this context, adapting business processes is a top priority for managers at both the organizational and operational levels. It refers to the ability to effectively adjust to different conditions[1], [12],[13].

Our goal is to provide a business process model that consists of business process components. These components are logical units seen as black boxes, with their boundaries defined by the interfaces they provide and require. Our vision is that business process modeling based on components will result in a system made up of instances of these components. Each component can be implemented separately, allowing for multiple implementations of the same component.

Our conceptual schema of the business process component allows:

- Dynamic Adaptation of business process model
- Flexibility
- The development of a basic component
- Provides a meta-layer model for business process model
- Enables configuration management
- Provides meta-process support

Our contribution involves describing business processes at a high level of abstraction through the use of a meta-model. The meta-model captures various aspects of a business process, including its functional, structural, informational, and behavioral flows. Additionally, we have created a method for modeling business processes based on their components, which also allows for refining the process model.

We have taken a unique approach to modeling business processes by using a high-level specification in the form of a meta-model. This meta-model encompasses all the key elements of a business process, including its functional, structural, informational, and behavioral aspects. Our method for modeling the business process is based on business process components, providing the flexibility to refine and adjust the model as needed. This approach offers a clear and concise way of capturing and describing business processes, making it easier to understand and analyze their workings.

The rest of this article is organized as follows: section 2 presents some of basic concepts and terminology of business process component and context of our research work. Section 3 outlines the similar work and highlights the motivation of this work. Section 4 presents our proposed approach, and section 6 will provide conclusion and some perspectives.

A. Component Definition

A software component is a manageable piece of code that is sufficiently small to be developed and maintained, yet large enough to be installed and supported. Furthermore, it is equipped with standard interfaces that allow for seamless integration with other components. [5].

In simpler terms, a software component is a standalone unit of code that is designed to be easily managed, installed, and supported. It has standard interfaces that allow it to work with other components as a part of a larger system.

B. Definition of business process component

A business process component is a specific instance of a business process. It contains the representation of the process and offers a straightforward interface for connecting with the external environment. In terms of the underlying business process, a component can be thought of as an activity or sub-process that helps to achieve a particular objective.

The purpose of a business process component is to create or modify a set of products or services. The end result of the business process is represented by objects such as documents, code, or services. Essentially, the business process component is a building block of a larger business process, with a well-defined function and purpose.

II. RELATED WORKS

In this paper, we have studied several approaches for modeling dynamic business process and business process adaptation.

Currently, there is a wide range of techniques have been proposed for business process adaptation modeling in different application domains, such as:

- Meta-modeling based approaches.
- Web service composition.
- Components based approaches.
- Rule-based approaches.
- Constraint-based approaches.
- Case-based approaches.

Firstly, we discuss the component-based approach. In software engineering, software architecture is used to describe the structure and functioning of a software system. In this context, a system is defined as a collection of software components, their connections, and the way they interact with each other. By designing the software architecture, a deeper understanding of the system can be obtained, which makes the design process easier. Furthermore, it provides a foundation for a robust system design analysis, enabling early detection of design flaws and errors.

In their work, Souvik Barat and Vinay Kulkarni[5] introduce an abstraction for business processes that considers composition, variability, and resolution in a cohesive manner. They explain the abstraction, demonstrate its implementation through a model, and provide an example to illustrate the concept.

To achieve this, they have created a component abstraction that enables decomposing application services into common and variable parts. This allows identifying areas where variations occur, and provides a mechanism for connecting the variable parts to the reserved spaces. Additionally, a resolution mechanism is implemented to ensure that the appropriate variable parts are connected to the designated reserved spaces.

As a result, the component abstraction addresses composition, variability, and resolution in a unified and localized manner. This was achieved through the use of model-based techniques to support a broad range of business applications [5].

The authors have proposed the abstraction of process components to define a behavioral unit that can represent a business process or its parts in a modular, hierarchical, composite, extensible, and configurable manner. This abstraction enables:

1-Composition - the creation of larger process components from existing components.

2-Configuration - the ability to adapt a process component to predefined scenarios.

3-Adaptation - the ability to adjust a process component to unknown situations as they arise.

In this way, the abstraction of process components offers a flexible and scalable approach to modeling business processes.

Gardler and Mehandjiev [6] introduced a new approach to the semi-automatic management and adaptation of e-business support systems through components. The process is automated by creating a link between the business strategy and software structure by mapping the company's process models to software models. The identified software models are used to guide the provision of suitable components through a "test" software harness. This approach serves as a strategic positioning and planning tool and also examines the alignment of these models and systems with an organization's strategic objectives.

The authors provide a methodology for selecting appropriate software models to support an organization's goals, while also allowing for the flexibility of component-based systems to realign implemented systems if the future differs from expectations. Their approach supports the semi-automatic collection and reuse of software models, components, and component groups using a business strategy mapping technique, which maps business process models to software design models and software structure. Business processes adaptation has to some extent benefited from product line architectures related to adaptation to known situations. In its work, they have inspired business processes adaptation with the same idea. An extension of the essential BPMN meta-model that supports business process families and a set of adaptation operators are presented. They describe their realization using model-guided techniques. They present here summaries to describe the model, part, fill character, composition integration structure and structure. They also present operators for the derivation of a larger by selecting a part from the list of available options, and modifying a part so as to meet context-specific requirements.

The literature provides various meta-models for business process modeling that are used to define the elements of the business process language and their relationships. These meta-models are utilized

to identify the structures and types of business process models from their constituent components. A set of primitives is typically defined, which can be used to perform change operations on the models.

In [7] authors addressed the security issue in adaptive business process frameworks and proposed a role-based access model for auto-adaptive business processes.

In [8] Liu proposed a method for continuous improvement of business process execution through effective resource allocation. To achieve this, they use business process execution logs and apply a data mining algorithm to extract new knowledge about task-resource bindings.

Muller et al proposed an agent-based framework for automatic business process adaptation in response to adverse events. This framework assesses the execution time of the business process to determine when adaptations are necessary[9].

The authors Hamri et al [10] have proposed a new approach to building a business process meta-model using MDA engineering and ontologies. This approach uses the Meta-Object Facility (MOF) to define the concepts that describe the information of the business process domain and incorporates an ontology definition meta-model (ODM). The basic meta-model and ODM are considered as two platform independent models (PIMs) and the OWL meta-representation is the platform specific model (PSM).

Popp et al [75] proposed a different approach to business process modeling by demonstrating high-level reference processes with less detail. They suggest that the missing details can be captured in additional business decisions that specify how an organization performs specific tasks. Their application to a high-level reference process leads to its refinement. They propose a division of process models, represented using Business Process Model and Notation (BPMN), and details of business rules. Model changes to process models, made at configuration time, result in either refined or balanced models of business processes. This allows for the required changes of business processes to be managed even without an expansion of the BPMN 2.0 standard.

The previous approaches described in the literature aim to address various issues related to business process modeling, adaptation, and management.

- Component Abstraction: A unified approach to addressing composition, variability, and resolution in business processes by using model-based techniques.

- Semi-Automatic Management of E-business Systems: An approach to automating the adaptation of e-business related support systems by establishing a link between business strategy and software structure.

- Meta-model for Business Process Modeling: A meta-model approach used to identify the structure and types of business process models from constituent components.

- Security Issues in Adaptive Business Process Frameworks: A proposal to handle security issues in adaptive business processes using a role-based access model.

- Continuous Improvement of Business Process Execution: A procedure for continuously improving business process execution through productive resource designation.

- Agent-based Framework for Automatic Business Process Adaptation: An agent-based framework to automatically adapt business processes in response to undesirable events.

- MDA Engineering and Ontology Definition: A proposal to build a business process meta-model using MDA engineering and ontologies for defining business process concepts.

- High-level Reference Processes: A proposal to demonstrate high-level reference processes with less detail and to refine them with additional business decisions.

Overall, these approaches aim to improve the efficiency and flexibility of business process modeling, adaptation, and management.

These approaches presented different solutions that are very specifics to their respective domains and lack the ability to be easily reused in other areas or applications. They lack sufficient flexibility and a comprehensive vision of business process adaptation. Additionally, there is limited research on the automated refinement of business process models, and limited exploration of using model

transformation rules based on MDA with conformance to a common metamodel for the source and target models.

III. OUR PROPOSED APPROACH

Our proposed approach seeks to address the limitations of existing solutions by introducing the creation of a comprehensive business process meta-model that is based on the component-based approach. This meta-model aims to support companies in their efforts to establish and develop their business processes. The model encompasses specification of component interface, control specification, and content specification, and possesses the following unique features:

- The business process metamodel has been designed to allow for the adaptation of business process models, with the adaptation problem studied and addressed in the model.
- The metamodel explicitly supports the adaptation of business processes, allowing for the creation of new versions or modifications of the model using the proposed operations.

A. Developing of business process meta-model based on the business process component

In our approach, we have utilized the CMP [2] conceptual model to specify the components required to construct an Information System from the abstraction of business processes. This approach is inspired by the widely adopted engineering of information systems through reuse. In line with this, we have proposed the Business Component approach where we use the CMP model to model the activities in the business process and design it in a modular fashion by breaking it down into quasi-autonomous components.

In the CMP approach, there are two main categories of business components: Entity-Type Components (CMEs) and Process-Type Components (CMPs). CMEs enable the abstraction and manipulation of entities, while CMPs are specifically designed to support specific business processes. In our approach, we are focused on reusing process-type CMs (CMPs) rather than entity-type CMs (CMEs). The reason behind this is that processes are more suitable for reuse as compared to entities, as the reuse of entities can lead to reduced reuse of information if they are not integrated into a process or a process fragment. We have taken inspiration from the CMP metamodel and have introduced the concept of business process component into our metamodel, as business processes describe how actors manipulate application data [2].

The "common conceptual schema" approach, as proposed by Armitage and Kellner [3], involves the creation of a concept map to represent all the entities involved in a process and their relationships. The aim of this approach is to provide a single, integrated view of the information needed to model and define a process.

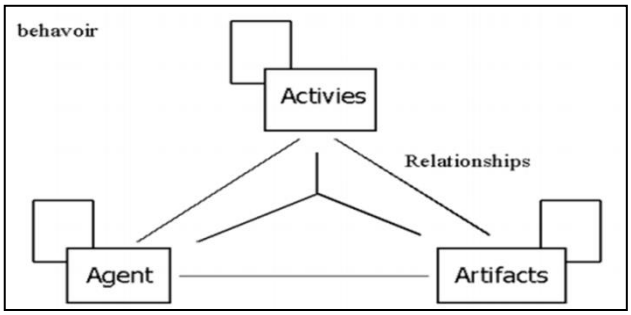

Fig.1 common conceptual scheme[3],

We can interpret the schema as follows: An agent performs an activity that produces an artifact. In addition, an artifact is derived from another artifact; an agent is managed by another agent; and an activity verifying the work of another activity. A process is dynamic in nature.

In our approach, we have combined two models, the Business Process Schema and the CMP, to model the business process component. The Business Process Schema defines the relationships between agents, activities, artifacts, and processes. It states that an agent performs an activity that produces an artifact, and an artifact can be derived from another artifact, an agent can be managed by another agent, and an activity can verify the work of another activity. The CMP provides important information on all entities involved in the process and their relationships. This combination of models results in a dynamic representation of the business process component.

B. Component approach to process modeling

We were particularly drawn to component-based process modeling approaches, as they allow for the implementation of data control and integration through a component-based model. The process is viewed as a collection of interconnected components, and data dependencies are minimized

by encapsulating components within clearly defined interfaces.

C. The conceptual schema of the business process component

We use the Business Process Management Coalition's reference model [11] to model processes from a set of basic abstractions. Figure 2 serves as a guide for identifying and grouping process elements that can be reused and defined to interact with other components.

Conceptually, we view the entities in Figure 2 as abstract classes, each possessing properties and methods, along with a predefined default behavior. Abstract classes are used to handle a specific approach to the process, and are further refined through specialization to match a particular approach to process modeling. An organization can develop its own process models by specializing the classes to fit its needs.

To create a new business process model, we can conceptually specialize the process entity in Figure 1 to form a business process entity. This allows us to identify the necessary process components to structure the model for reuse and implementation."

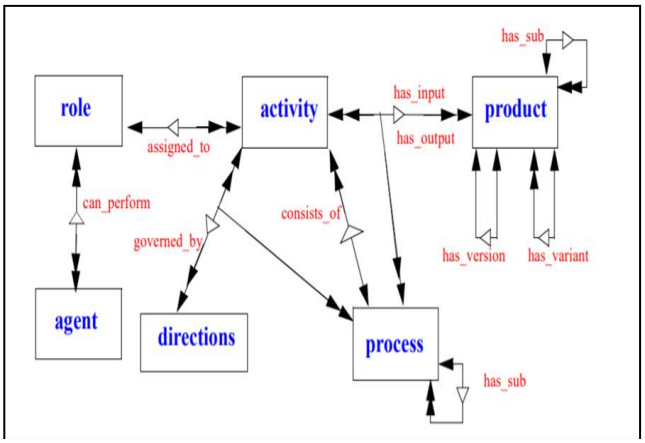

Fig. 2 Process Components Meta-Model

The components of the business process have properties have attributes that define their responsibilities, objectives, and outputs. These components are designed in a way that any modifications made to these attributes can be communicated to other process components. This conceptual diagram of the business process is used in modeling the business process component. The figure shows our approach to business process modeling based on the business process component metamodel.

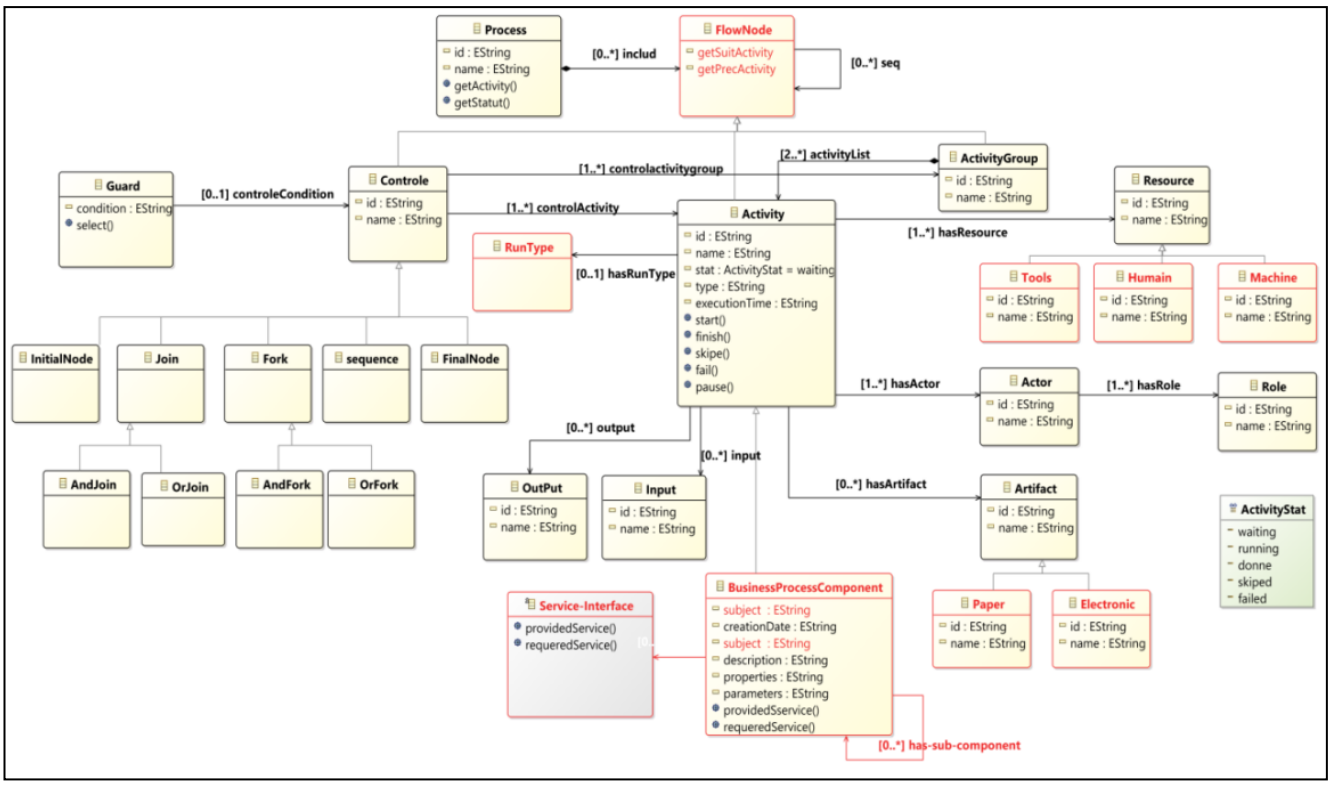

Fig.3 meta-model captures various aspects of a business process

In our approach to business process modeling, activities are described with a high level of abstraction. They represent tasks that are carried out by actors and may involve the use of external services, but their details are not specified.

We have adopted the concept of business process components, which allows us to build a model by assembling and connecting components. This modeling approach is particularly useful for describing the specific behaviors of business actions and service functionalities.

The business process component metamodel provides a way to capture the abstract concepts that serve as a foundation for other parts of the metamodel. This makes it easy to extend the metamodel to incorporate new business concepts.

Activities, data flows, and control flows, which are essential for describing the specifics of a business process. The meta-model is designed to capture all the necessary characteristics and functionalities of business processes in the commercial domain, and the relationships between these elements. The designer can extend the meta-model by incorporating new business concepts and specify the relevant aspects of the business process, such as the decomposition of activities into sub-activities, the control and data flows, and the Assignment of activities to the processing entities and actors.

The abstract description of a process definition made possible by the Process concept in the meta-model provides the designer with the ability to incorporate additional concepts into the model, such as the sub-process concept. This approach of using abstract concepts enhances the ease of transforming business process models during processing.

The Activity class is the central part. A process contains several FlowNodes that can be an activity group, a simple activity or a control node. The FlowNode class has a reflexive association called seq to characterize the sequence relationship. An activity has other relationships with the Resource class and the Actor class.

A Business Process Component inherits the Activity class and has all these characteristics, it has Inputs and Outputs and produces Artifact also it has an Actor with a well defined Role.

In our approach, a Business Process Component is an extension of the Activity class and possesses all its properties. This component is defined by inputs, outputs, an artifact produced, and an actor with a clear role.

A business process component also has the following attributes:

- Subject: which specifies the subject of the component.
- Creation date: the date when the component was created.
-Description: a detailed explanation of the component's purpose and functionality.
-Property: the attributes and characteristics that define the component.

Examples of properties

- Persistence
  - granularity
  - volume
  - duration
  - access mechanism
  - access frequency (creation / deletion, update, read)
-Reliability
  - Inter-process communication mechanism
  - latency
  - Synchronity
  - message size
  - Protocol
-Interfaces with other systems
  - latency
  - duration
  - access mechanism
  - access frequency

Safety and security
- data granularity
- user granularity
- security regulations

Also process business component has two methods:

-provided-service() : allows to define the services offered by the component.

-requered-service(): allows to define the services required by the component.

These two methods are declared in the Service-Interface

A business process component can be decomposed into process business sub-components using the has-sub-component relationship.

In addition, business process components have the ability to be decomposed, providing a different level of detail and granularity in the components. The interface parameters can be used to control the components and can be adjusted using predefined parameter values. This makes it possible to modify the configuration of the component in order to adapt to any future changes or transformations.

The Artefact class has two subclasses Paper and Electronics, as well as the Resource class has a tree subclass: Tools, Humain and Machine.

The Activity class has an attribute state assigned to the enumeration type State which currently indicates the state of execution of the activity, this enumeration has five states (waiting, execution, completed, skipped and failed). To guarantee the control of various activities, we have the Control class which contains three types of control Join, Fork and Sequence as well as the InitilaNode and FinalNode classes. The condition for choosing the right path is modeled by the Guard class which is associated with the following activities. At runtime, the user can choose the Guard with the operation select().

the condition relationship allows to describe connections (between control node and Guarde) whose guards are introduced in the guard parameter as an OCL expression,

IV. CONCLUSION

The adaptation of business processes has long been a subject of study, as researchers aim to understand how it evolves in order to reduce the reuse cost of business process components. Although there are business process model adaptation principles that apply to systems in any domain, we are particularly interested in the adaptation of business process components.

In this paper, we have proposed a new approach for business process adaptation. Our approach enables the adaptation of the business process model based on business process components. To implement this approach, we have introduced a business process meta-model that captures various aspects of business process components, including functional, structural, informational, and behavioral elements. The business process model is described at a high level of abstraction using the meta-model concept.

We have also demonstrated how business process models can be automatically adapted through model transformations. Based on MDA (Model-Driven Architecture), our approach provides an efficient solution to business process model transformations

and proposes a set of element change operations and generic transformations that are not specific to any particular field.

In future work, we plan to evaluate our proposed conceptual schema of the business process component to assess its performance and test it in other domains. We also aim to extend our meta-model to cover other areas of the business process.

REFERENCES

[1] Dumas, M., Fournier, F., Limonad, L., Marrella, A., Montali, M., Rehse, J. R., ... & Weber, I. (2023). AI-augmented business process management systems: a research manifesto. ACM Transactions on Management Information Systems, 14(1), 1-19.

[2] Saidi, R., Front, A., Rieu, D., Fredj, M., & Mouline, S. (2009, April). Component-based development: Extension with business component reuse. In 2009 Third International Conference on Research Challenges in Information Science (pp. 165-176). IEEE.

[3] Armitage, J. and Kellner, M.(1994), A Conceptuel Schema for Process Definitions and 163 Models. In Procedings of the third International Conference on the Software Process (ICSP3), pp. 153-165, Reston, VA.

[4] Khelifi, N. Y., Dekhici, L., & Zennaki, M. A model-driven approach to automatically generating CRUD functionalities from use case stories: A Pet clinic example study.

[5] Barat, S., Kulkarni, V., 2011. A component abstraction for business processes. Business Process Management Workshops 2011, pp: 301-313.

[6] Gardler, R., & Mehandjiev, N. (2002, October). Supporting component-based software evolution. In Net. ObjectDays: International Conference on Object-Oriented and Internet-Based Technologies, Concepts, and Applications for a Networked World(pp. 103-120). Springer, Berlin, Heidelberg.

[7] M. Leitner, S. Rinderle-Ma, J. Mangler(2011), "AW-RBAC: Access Control in Adaptive Workflow Systems", In Proceeding of the sixth International Conference on Availability, Reliability and Security (ARES), pp.27-34.

[8] T. Liu, Y. Cheng, Z. Ni,(2012) "Mining event logs to support workflow resource allocation",Knowledge-Based Systems, ELSEVIER, vol. 35, pp. 320-331.

[9] R. Muller,U. Greiner, E. Rahm,(2004), "Agent Work: a workflow system supporting rulebased workflow adaptation", Data & Knowledge Engineering, ELSEVIER, vol. 51, no. 2, pp.223-256.

[10] Hamri, S., Boudjlida, N., & Boufaida, M. (2007). An approach for building an OWL Ontology for Workflow Interoperability. In Enterprise Interoperability II (pp. 357-360). Springer, London.

[11] Popp, R., Kaindl, H., Nurcan, S., Pimenidis, E., Pastor, O., & Vassiliou, Y. (2014). Automated Adaptation of Business Process Models Through Model Transformations Specifying Business Rules. In CAiSE Forum/Doctoral Consortium (pp. 65-72).

[12] Abdelkader, B., Abdelkrim, A., & Meslati, D. (2016). A Meta-model for Dynamic Workflow Evolution. In COMPLEXIS (pp. 59-65).

[13] Abdelkader, B., Abdelkrim, A., & Djamel, M. (2019). A new approach for workflow evolution using MDA technology. International Journal of Intelligent Information and Database Systems, 12(4),304-334.