A Model Checking Based Business Process for Monitoring Company's Strategies

Hanane Ouaar and Mahmoud Boufaida

August 28, 2022

A Model Checking Based Business Process for Monitoring Company's Strategies

Hanane Ouaar
Department of computer science
Mohamed Khider University
Biskra, Algeria
hanane.ouaar@univ-biskra.dz

Mahmoud Boufaida
LIRE laboratory
Constantine 2 - Abdelhamid Mehri University-
Constantine, Algeria
mahmoud.boufaida@univ-constantine2.dz

***Abstract*—This study presents an agile system architecture for monitoring the company's strategy based on both paradigms: the agent and the model checker. This system covers the strategic and the supervisory levels for building an agent system that performers a strategy analysis with real measures taken by mobile agents, to increase the business efficiency of the internal or external operations of business processes using monitoring protocol primitives. Therefore, the present approach leads to improve the decision making with providing visibility on the progress of business processes using an extended Balanced ScoreCard. The traces of business process as a cooperative behavior can be recorded at a run time without conflicts when its interaction scenario is modeled in a formal method and proved with the temporal logic. The specificity of the developed system is related in the agility property, when it is easy to refine or change of a concept without altering the others.**

***Index Terms*— Business Process Strategy System, Formal Methods, Verification, Agent Monitoring Protocol, Agility.**

I. INTRODUCTION

Business Process Management (BPM) [2] is a discipline for managing life cycle of Business Processes (BP) from the modeling phase to process enactment and improvement, taking into account all different involved stakeholders. BPM has received considerable attention recently by both business administration and computer science communities. According to [13] BPM is a mature discipline that drives corporate success through effective and efficient business processes. BPM is commonly structured via capability frameworks, which describe and bundle capability areas relevant for implementing orientation process in organizations. Otherwise, BPM aims to help companies to improve their effectiveness through a better coordination between human resources and systems. It optimizes processes and provides a greater visibility in the area of business operations [7].

Therefore, companies that adopt the methodologies and BPM technology get a rapid return on investment and make better use of their existing systems. Moreover, BP provides a means of coordinating interactions between workers and organisations in a structured way. However, the dynamic nature of the modern business environment means that some BP should be externalised i.e. accept new BP from outside, or let local BP to displace off boundary. So, the challenge is to provide a flexibility and to offer an external process support at the same time. However, current BPM suffers from some limitation in optimisation due to the lack of good monitoring methods, because the involved control of internal and external BP achieving both company business strategy and its global objectives.

In the same field, a new work has been proposed a simple method [10] for developing a business supervision system, which covers the three phases: analysis, design and implementation. But this method suffered from lack of a coordination model and mobility quality. Otherwise, the aim of the present study is to provide an approach to build a system, named Business Monitoring System (BMS), respecting the agility property, which is certainly one of the most used quality factor today in the computer science field. Companies are particularly sensitive to this property. The latter evokes the idea of speed, talent, flexibility, suppleness, ability to get out of an unstable or danger to reach a safer position or a sustainable one.

The contribution of this research consists of proposing an architecture based on two levels: strategy and supervision.

The first level permits through a transformer to identify the company business strategy, to build a new structure of Balanced Scorecard (BSC) [1]. BSC is a performance measurement method that includes not only traditional financial measures but also qualitative ones as employee satisfaction, corporate mission and customer loyalty. Its structure respects the balance with the following four perspectives: financial, customer, internal BP and learning and growth. This new structure, with adding a 'Public Process' as a new perspective, is called a monitoring dashboard. After that, this dashboard is translated into an XML file.

In this level, a formal specification of the interaction behavior of the internal and external supervised BP is modeled with the timed automata and proved in TCTL (Timed Computation Tree Logic) [4] by applying the model checking [5]. TCTL is an extension of CTL (Computation Tree Logic) [3] temporal which makes it possible to express properties involving temporal quantifications. Thus, the formal methods provide a rigorous development process based on formal notations with defined semantics. i.e., they are characterized by their ability to express a precise meaning, thus allowing the verification of the coherence and the completeness of a system. This constitutes their main advantage. Formal methods can also help a user to understand a system, find errors and reduce their impact on a project delivery, through a complete specification that can be applied to any type of systems, software or material [6].

Whether more, in the literature there are several classifications of formal models [12]. In this work we will focus on the two following methods: first, the data-oriented method for describing system states derived from formal models, are easy to be understood, and therefore more accessible to non-specialists. Graphical notations are widely used for specifying systems. This method combines graphical languages with formal semantics, which makes it possible to draw advantages from these two axes in the development of systems. In our context, we will use timed finite state automata. Second, the operation-oriented method for describing the operations of the system and its behavior is derived from formal languages. The properties of the system including temporal behaviors could be described with using a formal language such as logic. In some cases a subset of the logic may be executed. The executable specification is then used for simulation and rapid prototyping. In our work we will use a specification of temporal logic TCTL executed by a model checker [9].

The second level is based on a Multi-Agent System (MAS) [8] respecting the monitoring protocol primitives. We use a set of mobile agents, which migrate for taking the requested measures according to remote destinations, and transfer the obtained results to their owner. Therefore, the agent paradigm is probably the most suitable technology to deal with decision making strategies in such environments, because they permit an easy combination of various artificial intelligence techniques, distribution, openness, and highly dynamic interaction. Moreover, agents are exploited for the design and/or the simulation of complex systems, as autonomous entities that are able to perform their functions without the need of continuous interaction from the user. To evaluate this approach, a case study has been developed in the Algerian Gulf Bank (AGB) by providing a prototype system.

II. OVER VIEW OF THE BUSINESS MONITORING SYSTEM (BMS)

A research contribution has proposed a simple architecture [11] for building a monitoring BP system in the e-banking domain, which involves two levels: a conceptual and an operational. However, this architecture suffers from important issues. First, it leads to a weak performance for coordinating the system's components in order to accomplish the common goals. Second, it reveals a deficiency of the mobility for taking measures trough outlying designations in a dynamic environment. Finally, it lacks a BPM flexible solution for their components reuse. In order to solve the previous issues, this architecture has been enriched with a new version, when it is managed by a set of primitives of a monitoring protocol using intelligent agents and the mobile agents in order to take measures from one node to another.

The main contribution of this research work resides in the proposition of an approach composed of two important levels: the strategy level and the supervision one.

The first level ensures the BP interaction cooperation scenario, through the formal specification of internal and external BP to supervise it with the timed automat and the timed logic language TCTL [15], [16] in order to provide the most important properties, saved in the end as XML file. Moreover, a formal language is indeed a language endowed with an adequate mathematical semantics based on rules of interpretation which guarantee the absence of ambiguity in the produced descriptions and rules of deduction which make it possible to reason on the specifications in order to discover potential incompleteness, inconsistency or to prove properties [14].

This level proposes also a transformer that accepts a company business strategy as an entry. After that, it extends the BSC structure that is called a dashboard, with adding "Public Process" as a fifth perspective. Then, this transformer engine find, since for each of the five perspectives, their objectives, measures, targets and initiatives. Finally, the transformer provides as an output the dashboard XML file.

The second level provides a monitoring protocol as a set of communication primitives that uses the agent technology, with exploiting the results of the upstream one.

Multi-Agents System (MAS) [8], has been exploited in this methodology for designing the agent roles and for simulating their coordination, as autonomous entities that are able to perform their functions without the need for continuous interaction from the user. In this paper, the notion of mobile agent is used for the measures taking task in a distributed environment. This protocol coordinates four independent roles associated with an agent: Settings Agent, Coordinator Agent, Collector Agent and Mobile Agent. The Settings Agent provides the user profile and all the configuration settings. The Coordinator Agent communicate and coordinate with the other agents, and it responsible for requesting the tacking of measures to applying report edition, analysis and alert activities, and saving the traces in a History Database. The Collector Agent activates the Mobile Agents to create a set of clones, receives and collects from it the taken measures.

The use of the agent paradigm has been intended because it brings some useful characteristics: system consistency, such as intelligence, autonomy, cooperation and mobility. The mobility aspect permits an agent to carry the company's measures and it sends the relevant results toward its owner, due to its ability to move between different distant data sources. Therefore, behind the mobility are found the minimization of the communication costs and reduce significantly the execution time of tasks.

Consequently, an agent-based architecture is proposed in the BPM field, which aims at realizing a BP monitoring system. The strength of our contribution resides in their protocol primitives among different agents, their roles and business strategy. Another important improvement of the proposed system is related to the respect of the agility property, which enables this system to support a BPM redesign and a reuse. This property is ensured in the coordination, the collaboration and in the interoperability of the different agents involved in the system, where each agent is dynamically adaptive, robust and flexible.

III. DESCRIPTION OF THE BMS COMPONENTS

This section presents the role of different components in the proposed architecture system, respecting the strategy level and the supervision one, as shown in Figure1, they are described below:

A. Strategy level

1) *XML dashboard*: this part permits the construction of the monitoring dashboard and the generation of its corresponding XML files, using the following components and resources:

Transformer: It is an engine that executes the following rules:

This level provides two resources files:

Rule 1: Extending the original structure of the BSC that called the "monitoring dashboard" into five dimensions: financial, customer, private process, public process that has been created as new perspective, and learning and ground. Then, this transformer builds this dashboard according to the organisation business strategy as intput resource.

Rule 2: Identifying the objectives for each of the five perspectives of the selected enterprise.

Rule 3: Identifying the measures, targets and initiatives for each identified objective.

Rule 4: Elaborating the structure DTD (Document Type Definition) of this monitoring dashboard.

Rule 5: Generating the XML file of the monitoring dashboard by the transformer as output resource.

2) XML file of the formal behavior: This section is composed of the four successive steps:

a) Identification of the intern and the extern BP List: specify exactly the set of the synchronous public and private BP in the company system that could be monitored respecting the strategy of the company.

b) BP cooperative behavior formal modelling: that represents the behavior of a cooperative business processes by developing their synchronous timed automata as a graphical notations.

c) Temporal property verification: writing the formal specifications of BP already formally modelling in the previous stage by the CTL temporal logic, this applies Model Checking as a technique for automatic formal verification.

d) XML file of the formal behavior construction: it is a result component (the model and his verification) that will exploited in the supervised level as a setting in the BPM, independent to the other framework components.

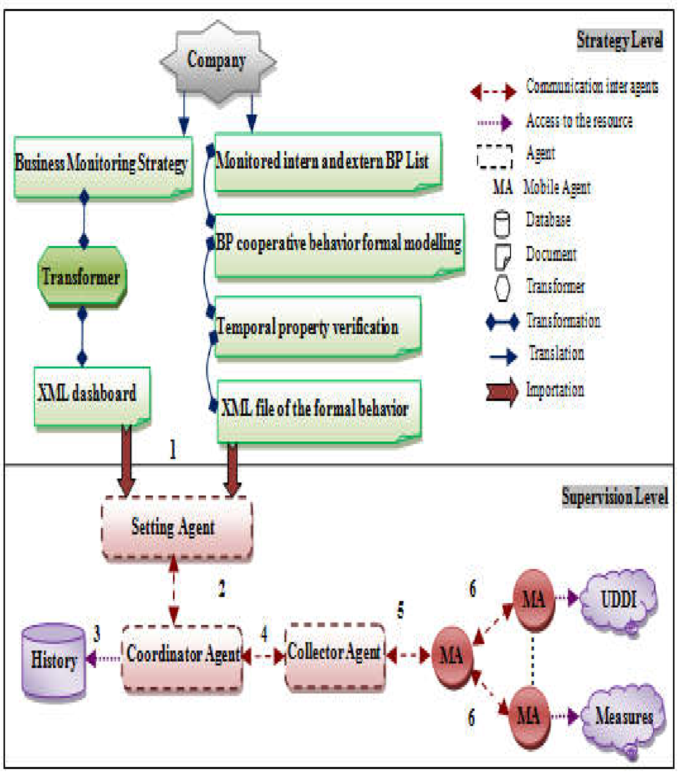

Figure 1. Global overview of the system architecture

B. *Supervision level*

In this level, the defined agent society is composed of: the Setting agent (SA), the Coordinator Agent (CA), the Collector Agent (LA) and the Mobile Agent (MA). Furthermore, this system exploits the History Database as resource. It provides a source of all the monitored measures traces and allows updating him. Its implementation model respects the extended BSC, and it is described with the following relationship:

1) **Perspectives**: < perspectives_Id: Integer, perspectives_name: String>.
perspectives_Id: perspective code of the dashboard.
perspectives_name: such as: Financial, Customer, Private_Processes, Public_Processes, learning_growth.

2) Objectives: <objectives_Id: Integer, objectives_name: String>.
objectives_Id: objective code for a perspective.
objectives_name: objective name.
Across the model connectivity, this table inherits the key of Perspectives table to use it.

3) Measures: < measures_Id: Integer, measures_name: String, measures_val: Integer >.
measures_Id: measure code, each objective can provide various measures.
measures_name: It is the name of a measure.
measures_val: a measure value, which will contain the information.
As the previous case, this table will use the inherited key of Objectives table.

4) Traces: <agent_mobil_Id: Integer, address_distination: String, measures_date: Date, measures_time: Time >.
gent_mobil_Id: identifier of the mobile agent that takes this measure.
address_distination: address destination of the source of measure taken by this mobile agent.
measures_date: measure date.
measures_time: measure time.

IV. ROLE AND BEHAVIOUR OF THE BMS AGENTS

This section is devoted to the study of both the role and their behaviour of the different agents mentioned above.

A. Setting Agent (SA)

It manages the communication between the supervision system and its users (administrator and the other users) via a graphical interface. The structure of this agent is specified as:

- Communication Module;
-Dialogue Manager Interface;
- Library of monitoring interactions;
- Dashboard table;
- Supervised results Table.

B. Coordinator Agent (CA)

It is the basic agent that coordinates with the SA and the LA in order to accomplish softly the operations sequence. The structure of this agent is specified as:

- Communication Module;
- Monitoring and Coordination Module: From the collected measures, this module provides three essential activities based on the monitoring tasks:
  - Report edition activity.
  - Analysis activity.
  - Alerts activity.
- Measuring Module;
- Knowledge;
- Report File.

C. Collector Agent (LA)

According to the well-defined cycle times, the LA triggers the operation of measures taking, by the activation of the MA specialized in the measures taking toward specific destination. This LA receives and collects the transferred measures by this MA. The structure of this agent is detailed as following:

- Communication Module;
- Launching and Collecting Module;
- Measures settings Table;
- Collected Measures File.

D. Mobile Agent (MA)

According to specific destinations, this agent has the ability to clone itself in order to take measures and to migrate from each clone to an outsourcing destination. After that, the taken measures are transferred toward the LA. The structure of this agent is specified as:

- Communication Module;
- Processing Module;
- Mobility management module;
- Measures file.

V. SPECIFICATION OF THE MONITORING PROTOCOL PRIMITIVES

The sequence of the messages exchanged between agents of the BMS is specified in Figure 2 with a monitoring protocol as a set of transitions between the different agents of the system and the History database.

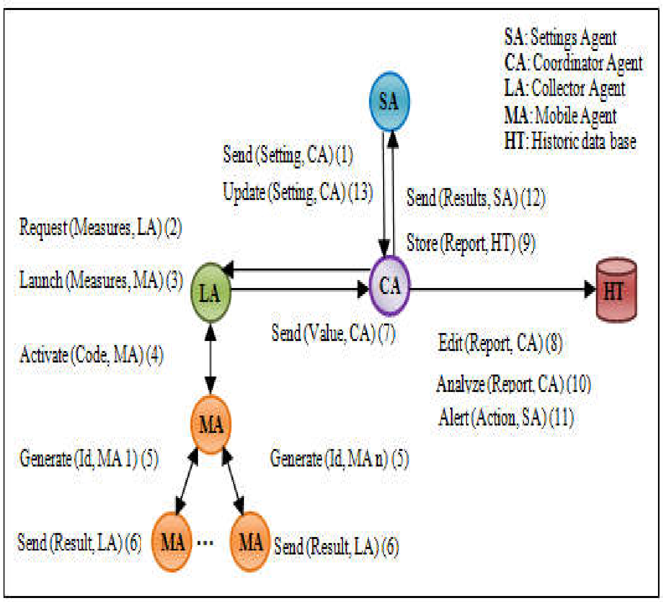

Figure 2. Monitoring protocol primitives

The rules defining this protocol are (Figure 2): Firstly, the administrator of this system should import, through the SA interface, the business strategy as a monitoring dashboard XML file, XML file of the formal behaviour that links all the supervised BP and organise all other configurations setting. The SA agent sends all the required settings to CA and updates it (primitive 1). The CA sends all the settings toward the LA. As well it requests the operation of taking measures, (primitive 2). In a well determined time cycle, LA triggers the operation of measures taking, (primitive 3). After having received the request from the coordinator agent, the LA sends a message to activate the mobile agent, (primitive 4). After having activated the MA, this latter extracts the different parameters of the message that have been received to generate clones of it, (primitive 5). Such clones migrate toward their intended destination to take the requested measure. In each node destination, the MA transfers measures toward the LA node address, (primitive 6).The LA collects all the transferred measures from each MA, stores them in a file, and sends them toward the CA (primitive 7).The CA launches first (primitive 8), the report edition activity that computes metrics and fill the report according to dashboard structure. This activity calls the analysis activity (primitive 10) and the alert one to compare the taken measures with the target values already contained in this dashboard (primitive 11). In this level, The CA also stores this trace in the history database, (primitive 9). The CA informs the SA for all this monitoring results reports, alerts and notifications (primitive 12). Finally, SA shows the results with the user view. It can also update its settings if necessary and sends it to the CA, (primitive 13).

VI. USE CASE: AGB BANK

In order to establish the exploitation of the proposed architecture system, it has been chosen to validate it with a case study related to a modern banking company, called AGB (Algerian Gulf Bank) [17]. The choice of such a company has been motivated for several reasons. First, the banking domain provides the most convenient environment to prove all the aspects of this architecture. Second, this bank pursues its own strategy. Third, it provides many intern services and it also publishes some extern services via the net like the e-banking. The authors' objective was to implement a monitoring system of the various transactions of internal and external BP in a bank company. Their evaluation is demonstrated in the following application:

A. At the strategy level

In this side, according to the selected case study and in a flexible way, the strategy of AGB has been defined and based on several axes. A monitoring dashboard has been generated and figure 3 shows a part of its generated XML file.


```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE BSC (View Source for full doctype...)>
< monitoring_dashboard ="AGB">
< Perspective >
< Financial>
< Strategy >
  < Objectives > Increase in revenue
        < Measure > Own capital </ Measure >
        < Measure > Total revenue </ Measure >
        < Measure > Charges </ Measure >
        < Measure > Tax </ Measure >
        < Target > Social capital> 20% en 2022</ Target >
        < Target > ROA +3 %< / Target >
        < Target > ROE +2 %< / Target >
        < Initiative > Increase sponsoring </ Initiative >
        < Initiative > Analyze reports </ Initiative >
  </ Objectives >
< /Strategy >
</ Financial >
< Customer >
< Strategy >
  < Objectives > Evolution of the customer wallet
        < Measure > Number of customers </ Measure >
        < Measure > Deposits from customers </ Measure >
        < Target > Total account +5 % </ Target >
        < Target >Total Credit ±3% </ Target >
        < Target > Total assets+0, 1 % </ Target >
        < Target > Own Fond+1 % </ Target >
        < Initiative > Align claims with the trade </ Initiative >
        < Initiative > provide easily available and accessible services </ Initiative >
  </ Objectives >
</ Strategy >
</ Customer >
  ......
< /Perspective >
```


Figurer 3. Monitoring dashboard XML file

In the other side, the following resources have been used:

1) UPPAAL [20] is a software tool for modeling, validation and verification of real-time systems. It is appropriate

for systems that can be modeled as a collection of non-deterministic processes with finite control structure and real-valued clocks (i.e. timed automata), communicating through channels and (or) shared data structures. Typical application areas include real-time controllers, communication protocols, and other systems in which timing aspects are critical.

2) List of monitored intern BP: the statement of account, the currency exchange, a new check book request and deliver Card Inter Bank (CIB), calculate statistics, the main AGB system.

3) List of monitored extern BP: e-banking (Website), Telephone banking (Fax), SMS banking (SMS), Automatic Teller Machines (ATM) and e-Payment (Electronic Payment Terminal). These external BP are published as Web Services when their user contract is mentioned below:

4) BP cooperative behaviour formal modelling: According to the AGB environment, among the several BP, Figure 4 and figure 5 present respectively a summarized example of two synchronised BP: first, the BMS as the global and the local BP, second, the ATM as the extern published BP.

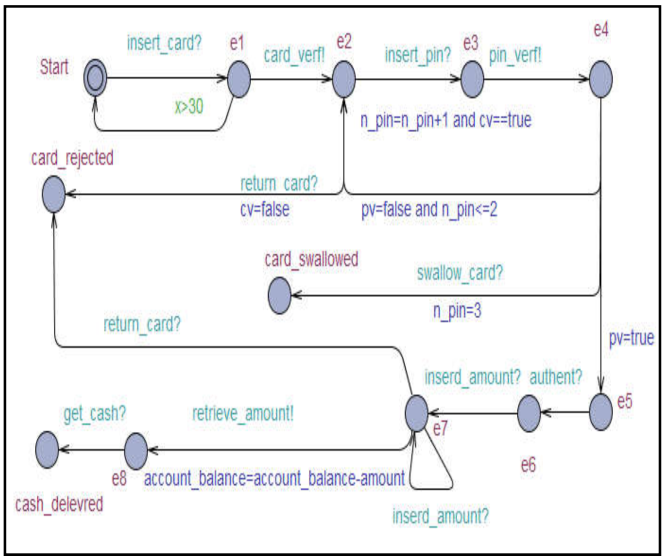

Figure 4. BMS banking timed automata

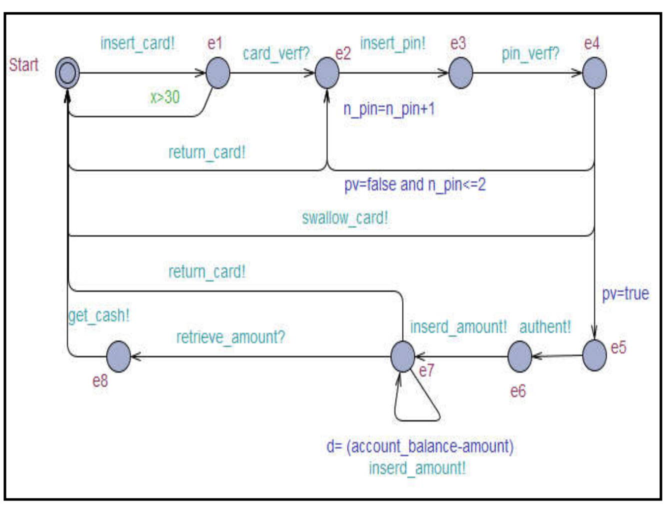

Figure 5. ATM timed automata

5) Temporal property verification: The expression described in Figure 6 summarizes some formal of both BP BMS banking and AGB properties with TCTL(Timed Computation Tree Logic) that are proved in UPPAAL software tool, by applying the modal checker.

6) XML file of the formal behaviour: The final project as models and TCTL properties is in a XML file, to be exploited in the supervision level.

……………………some BMS BP satisfied properties ………………

E<> (AGB.card_swallowed and x==3) *The property is satisfied*

A<> AGB.Start and x<30 *The property is satisfied*

E<>AGB.card_rejected and cv==false *The property is satisfied*

E<>AGB.cash_delevred and account_balance>=amount *The property is satisfied*

A<> not (AGB.cash_delevred) *The property is satisfied*

……………………some AGB BP satisfied properties………………

A<> A1.Start and x<30 *The property is satisfied*

E<>A1.e8 and pv==true *The property is satisfied*

E<>A1.e4 and pv==false *The property is satisfied*

E [] not (A1.e4 and cv==true) *The property is satisfied*

A [] not (A1.e6 and cv==true) *The property is satisfied*

Figure 6. Model checker applied to the TCTL properties

B. At the supervision level: Some implementation aspects

In this section, the behaviour of the different BMS agents has been simulated using a multi-agent platform. JADE (Java Agent DEvelopment Framework) [19].

The different agents (SA, CA, LA, MA) are created in the main container, when the mobile agent and its cloned agents are created in runtime within container that represent the host destination. For the monitoring protocol, a supervision system package has been defined that includes four classes corresponding to the four agents. As the implementation is based on the communication between the different agents, cognitive agents have been used and are able to plan their actions and remember their state in order to evaluate the offers that are available to them. A prototype has been implemented using standards. In fact, the XML technology is used to represent the information exchanged between agents via the standard communication language FIPA-ACL [18].

VII. CONCLUSION

A new approach has been presented for providing a monitoring system in the form of architecture based on two layers: the strategy layer and the supervision one. It combines concepts such as company business strategy, formal model, TCTL logic language specification, BSC, intelligent agent, and the mobile agent paradigm. The defined agents interact with each other to access, transfer, and to evaluate dynamically information of BMS, with respecting monitoring protocol primitives. The strategy enrichment in the supervision system permits to raise the analysis and enables multiple future issues for improving the decision making. To assess this architecture, a real case study has been analyzed in the e-banking domain using a prototype system. The proposed solution provides a profit of agility, with an added value to this approach.

As a matter of fact, only agile businesses in a monitoring system can flexibly redesign or reconfigure their operations and processes in a declared analysis, and then they can survive in the currently rapidly evolving social and economic ecosystem by testing currently the taken measures. In our context, the agility resides in the flexibility, which manipulates independently the BSC properties: objectives, measures and targets for each dimension. This property exists again in the formal models being easy to understand and therefore more accessible to non-specialists, graphical notations are widely used for specifying systems allowing the verification of the coherence and the completeness of a system, also to ensure system setting to find errors and reduce their impact before project delivery. This property located also in the mechanisms of coordination, collaboration and interoperability between the independent agents with the respect of transition primitives of the monitoring protocol. Consequently, a new agile solution has been presented when it is easy to refine a concept without involving the others.

Therefore, this approach can be applied to various business processes systems, but it has some limitations. First, a difficulty to find the best indicators (objectives and measures) of BSC, the suitable time to update them, and to associate a real value (target) with them in the market by applying the analysis. Second, a strict identification of BP that has intern and extern activities in the same time.

In a future work, we think to the definition of other monitoring settings like the execution time aspects, by the consideration of other Key Performance Indicators (KPI) as part of the new BSC measures, which will reinforce the analysis. Moreover, the build of an automatic transformation engine from a business strategy to an XML dashboard.

References

Books:

[1] Norton, D. P. and Kaplan, R. S., 'Balanced Scorecard ', M. Augier, D. J. Teece (eds.), The Palgrave Encyclopedia of Strategic Management. 2016.

[2] Weske, M., 'Business Process Management: Concepts, Languages, Architectures', third Edition book, Springer-Verlag GmbH Germany, pp. 3, 2019.

Technical Reports:

[3] JanT sil , Xiaohong Chen , "LTL Model Checking in Matching Logic", https://hdl.handle.net/2142/114117, IDEALS, 2022.

Unpublished Papers:

[4] Siti Nuraishah Agos Jawaddi, Azlan Ismail, Valeria Cardellini , "Modeling and Verifying Microservice Autoscaling Using Probabilistic Model Checking", DOI: https://doi.org/10.21203/rs.3.rs-1682990/v1, Institute for Big Data Analytics and Artificial Intelligence (IBDAAI), Kompleks Al-Khawarizmi, Universiti Teknologi MARA (UiTM), 40450, Shah Alam, Selangor, Malaysia, 2022.

Papers Published in Translation Journals:

[5] Bader M. AlFawwaz, Faisal Al-Saqqar, Atallah AL-Shatnawi, "Reduction Model Checking for Multi-Agent Systems of Group Social Commitments", Computation 2022, 10, 84. https://doi.org/10.3390/computation10060084, pp.1-19, 2022l

[6] Benjamin Herd, Simon Miles, Peter McBurney, Michael Luck, "MC2MABS: A Monte Carlo Model Checker for Multiagent-Based Simulations", Conference: 16th International Workshop on Multiagent-based Simulation (MABS), At: Istanbul, Turkey, May 2015.

[7] Hammer, M. Introduction. In Jan vom Brocke & Michael Rosemann (2th ed.), "What is business process management", In Springer-Verlag Berlin Heidelberg(Ed.). Hand book on Business Process Management, Methods, and Information Systems, Cambridge, USA, pp. 6, 3, 2015.

[8] Luck, M., Burney, P. M ,Shehory, O. and Willmott, S., 'The agent-link community: agent technology: computing as interaction a roadmap for agent-based computing ', Proceedings of In Agent Link III, pp. 4, 2005.

[9] Marta Kwiatkowska, Gethin Norman, David Parker, "Probabilistic Model Checking and Autonomy", Annual Review of Control, Robotics, and Autonomous Systems, Vol. 5, https://doi.org/10.1146/annurev-control-042820-010947, Kwiatkowska, pp 385-410, 2022, pp.

[10] Ouaar, H. and Boufaida, M., 'A BSC-based method for the supervision of business processes', proceedings of the Sixth International Conference on Business Intelligence and Technology (BUSTECH'2016), Rome, Italy, pp. 1-6, 2016.

[11] Ouaar, H. and Boufaida, M., 'An architecture for managing e-banking business processes', Paper Presented at the first International Conference on Business Intelligence and Applications (ICBIA'2016), Bilda, Algeria, 2016, pp. 14-19, 2016.

[12] Peter Gjøl Jensen, Jiří Srba, Nikolaj Jensen Ulrik, Simon Mejlby Virenfeldt, "Automata-Driven Partial Order Reduction and Guided Search for LTL Model Checking", International Conference on Verification, Model Checking, and Abstract Interpretation, VMCAI, Verification, Model Checking, and Abstract Interpretation pp. 151–173, 2022.

[13] Rosemann, M., 'An Exploration into Future Business Process Management Capabilities in View of Digitalization', Georgi Dimov Kerpedzhiev, Ulrich Matthias Ko¨nig, Maximilian Ro¨glinger, Michael Rosemann, Bus Inf Syst Eng 63(2), pp. 83–96, 2021.

[14] Takeshi Tsukada, Hiroshi Unno, "Software Model-Checking as Cyclic-Proof Search", rXiv:2111.05617v1, https://doi.org/10.48550/arXiv. 2111. 05617, Takeshi Tsukada, 2021.

Dissertations

[15] Ouaar, H. and Melhagueg, A, 'Analysis of the formal specification of a business process', master's thesis, University of Biskra ,computer Science department, Biskra, Algeria, pp. 28-41, 2021.

[16] Ouaar, H. and Sakhraoui, M, 'Analysis of the formal and semi-formal modelling of an ATM', master's thesis, University of Biskra, computer Science department, Biskra, Algeria, pp. 35-54, 2016.

Standards:

[17] Algeria Gulf Bank. [online] https: //www.agb.dz/article-view-7.html, (accessed April 2022).

[18] Foundation for Intelligent Physical Agents.onlinehttp://www.fipa.org/repository/aclspecs.html ,(accessed April 2022).

[19] JAVA Agent DEvelopment Frame-work JADE. [online] http://jade.tilab.com, (accessed June 2022).

[20] UPPAAL Help. [online] https://docs.uppaal.org/, (accessed july 2022).