A Cloud-based Service-oriented Architecture for Business Process Modeling and Simulation

Paolo Bocciarelli
Andrea D'Ambrogio
Emiliano Paglia
Tommaso Panetti

University of Rome Tor Vergata
Via del Politecnico, 1
00133, Rome, Italy

Andrea Giglio

Guglielmo Marconi University
Viale Plinio, 44
00193, Rome, Italy

ABSTRACT

The adoption of Modeling & Simulation (M&S) approaches is widely recognized as a valuable solution for enacting a timely analysis of business processes (BPs). Despite their relevance, the effective introduction of such approaches in the BP lifecycle is still limited, due to the know-how and skills for building and implementing a simulation model and to the cost and effort for setting up and maintaining the execution platform. In this respect, this paper proposes a cloud-based architecture that exploits the *M&S as a Service (MSaaS)* paradigm and *containerization* technology for the flexible and dynamic composition of M&S services, so to allow business analysts to carry out an effortless and effective M&S-based BP analysis. An example case study dealing with an e-commerce scenario is also presented in order to show the actual application of the proposed approach.

I. INTRODUCTION

Modeling and Simulation (*M&S*) is widely recognized as a valuable solution to analyze the behavior of complex systems and processes at both design time and operation time, in order to assess their performance and evaluate design alternatives from the early lifecycle stages. This work specifically focuses on the use of M&S for the analysis of Business Processes (BPs). The BP management field particularly benefits from the adoption of M&S-based approaches, which allow business analysts, enterprise managers and ICT specialists to get the appropriate performance-oriented understanding of their BPs, according to the desired or required level of abstraction, i.e., from the BP conceptual definition down to the technical BP implementation and operation [1].

Despite the acknowledged relevance of M&S approaches, their actual introduction in the BP lifecycle is still limited, mostly due to the significant know-how, effort and cost that are needed to build, implement, deploy and execute simulation models of BPs.

In order to overcome such limitations and provide a more effective adoption of M&S approaches for BP analysis, this paper proposes a solution based on the *M&S as a Service (MSaaS)* paradigm, which brings the benefits of service-oriented architectures and cloud computing into the M&S field, so to enhance interoperability, composability, reusability and reduce the effort and cost of M&S.

MSaaS platforms enable the flexible and dynamic composition of M&S services that provide on demand access to M&S solutions to be used both at definition time, when the BP is specified by use of a BP modeling language (e.g., BPMN - Business Process Model and Notation), and at operation time, when the BP execution is monitored and proper decisions have to be taken in case of a performance downgrade, so to get a performance improvement.

This paper contribution further enhances standard MSaaS capabilites by exploiting *model-driven development (MDD)* approaches, which allow to significantly reduce the effort of developing, deploying and executing simulations by focusing on the use of formal models and automated model transformations, as illustrated in [2] and [3].

In this respect, the main contribution of this paper is the specification of a platform, denoted as *BP-MSaaS*, which is founded on the use of:

- a *cloud computing* infrastructure, to outsource and to dynamically configure the required hardware and software platform without the necessity of building and maintaining costly internetworked computational nodes;
- a *model-driven development (MDD)* approach, to significantly reduce the effort of building, implementing, deploying and executing BP simulation models;
- a *microservice* paradigm, a granular service oriented architecture which allows to compose the desired M&S capability by integrating loosely coupled self-contained services that cooperate by use of a standard communication protocol [4];
- a *containerization* method, which allows to wrap services in a self-contained runtime environment, in order to ease their cloud-based deployment and execution. Specifically, this paper adopts Docker [5],

an open-source containerization infrastructure that provides native integration with various cloud service providers, such as Microsoft Azure [6] and Amazon AWS [7].

It is worth noting that the proposed architecture is flexible enough to be effectively used not only in the BP management domain, as discussed in Section III.

In order to provide an actual understanding of the proposed contribution, the paper also presents its application to an example e-commerce scenario.

The reminder of this paper is structured as follows: Section II outlines the literature review and summarizes the identified open issues that limit the effective adoption of M&S-based BP analysis. Section III illustrates the proposed BP-MSaaS platform, which is described from both an architectural and a behavioral perspective. Section IV presents an application scenario and provides additional implementation details. Finally, Section V gives concluding remarks.

II. Open Issues in Business Process Modeling and Simulation

BP modeling and simulation is a widely considered and analyzed research and development field. A complete and rigorous investigation on methodologies, techniques and tools can be found in [8]. An additional analysis on modeling methods such as Gantt Chart, IDEF family methods, Colored Petri-Net, Object Oriented methods, Workflow techniques is provided in [9], while an overview on BPMN, BPEL4WS, UML, Petri-Net can be found in [10], [11]. A quantitative analysis and classification of BPMN, Flow Chart, EPC and UML modeling languages, according to the Moody's quality criterion for a good diagram [12], is provided in [13], in which the authors state that BPMN is the language with the highest score, but still far from the maximum value.

A key point emerging from the literature review is the need to adopt approaches that consider modeling as a continuous activity to be carried out throughout the business process lifecycle, thus evidencing the strict correlation with other activities and, in particular, with simulation. Such a correlation is also discussed in [14], in which the authors present the benefits that a multi-dimensional approach provides, in [15], which presents methods to obtain a model suitable for business process simulation, and in [16], in which simulation techniques are presented as complementary to modeling, since they separately consider dynamic and static aspects of a process, respectively.

Several authors reveal some limitations of current business process modeling and simulation approaches, such as the lack of focus on the actual IT platform that enacts the business process model, and the semantic gap between conceptual and simulation models [17], [18], [19], [20], [21]. In particular, [20] points out that there is no commonly accepted conceptual modeling technique for simulation.

These limitations have been addressed in literature, and strategies to overcome them have been presented, such as multi-dimensional approaches [14], hybrid modeling techniques with performance analysis and process optimization [19], and model-driven approaches that aim at reducing the gap between conceptual and simulation models [20], [21]. Indeed, in [21] traditional modeling methods, such as BPMN, Flow Chart, Gantt Chart, IDEF0, IDEF3, and UML are shown to provide an abstract BP representation, which does not allow to include those details required to enact a direct translation into an executable simulation model. Thus, the authors propose a model-driven framework that contributes to reduce the gap between conceptual and simulation models. Also in [22] the authors state that, due to the aforementioned semantic gap, the conceptual model usually does not contain enough information to directly allow simulation and thus a transformation approach is required. In this respect, model-driven development is considered both as a promising solution and as a challenge [23].

A systematic review of business process simulation tools can be found in [15], in which the authors analyze business process modeling tools that also provide simulation features, business process management tools having simulation capabilities and general purpose simulation tools. Several simulation tools are also compared in [24] in terms of characteristics such as coding aspects, user support, cost, pedigree, modeling assistance, friendliness, simulation capabilities and software and hardware requirements.

Most of the existing simulation tools require M&S know-how and programming skills above those generally available in organizations [25]. For such reason, the adoption of model-driven development, which provides increased levels of automation, is considered a promising approach both for transforming process conceptual models into simulation models and for obtaining the corresponding simulation code [20], [26], [21].

To summarize, the literature review has identified the following open issues as limitations of currently available approaches to M&S-based BP analysis:

- BP modeling is considered as a separate activity rather than being correlated to simulation throughout the BP lifecycle;
- a significant semantic gap exists between the BP model and the simulation model;
- existing simulation tools require M&S know-how and programming skills which business analysts are usually not familiar with.

In order to overcome such limitations, this paper proposes the BP-MSaaS platform, as illustrated in next section.

III. BP-MSaaS Platform for M&S-based BP Analysis

This section describes the BP-MSaaS platform, which enables the M&S-based analysis of BPs according to a MSaaS paradigm. In this respect, Section III-A illustrates the platform architecture, whereas Section III-B gives the behavioral specification.

A. BP-MSaaS Architecture

The proposed architecture, shown in Figure 1, has been designed in order to overcome the most relevant open issues affecting existing M&S-based BP analysis approaches, as discussed in Section II. The rationale of the proposed approach is that a set of cloud-based remote services can be selected, orchestrated and executed in order to carry out the several tasks required to conduct a BP analysis: *modeling services* to specify abstract BP models; *transformation services* to generate the corresponding executable simulation models; *simulation services* to execute simulations and *presentation services* to summarize and show the related results.

The outermost layer of the BP-MSaaS architecture is the *Web Layer*, which implements the portal through which the user interacts with the platform. The portal is intended to be a one stop shop that provides an easy-to-use interface to the platform and its offered services. Underneath the Web Layer, the *Middleware Layer* provides the access point to the M&S services that are made available through the following components of the underlying cloud infrastructure:

- **User Identification and Authorization**: component providing authentication services.
- **Model Repository Manager**: component providing services for storing/retrieving models to/from the cloud-based repository.
- **Modeling Service Manager**: component providing modeling services for creating, editing and storing models. In order to make the architecture flexible enough to fit the needs of several application domains and modeling languages, modeling services can be automatically generated from a domain metamodel that defines a given modeling language, by use of *transformation services*.
- **Transformation Service Manager**: component providing services for storing, retrieving and executing model transformations on the cloud infrastructure. Model transformations are used to generate models (e.g., simulation models from BP models), by applying *model-to-model* transformations, as well as to generate the code that implements executable services (e.g., modeling services or simulation services), by applying *model-to-text* transformations.
- **Simulation Service Manager**: component in charge of managing the required tasks needed to first select a simulation service image available in the cloud repository and then deploy and execute it on the cloud infrastructure.
- **Presentation Service Manager**: component providing services for the visual rendering of simulation results.
- **Docker Infrastructure**: component providing services for interacting with the underlying cloud infrastructure by use of the Docker containerization approach.

The portal of the BP-MSaaS platform relies on the underlying cloud infrastructure layer, which provides the following services:

- **Storage Service**: the cloud-based repository stores the several data and entities used by M&S services: *models*, which can be generated by a model transformation or edited by use of a modeling service, *configurations data* and *service images*. The executable code that implements a M&S service (i.e., a simulation, transformation or modeling service) is stored in the repository, along with the required library and runtime, as an executable image, which has to be deployed to a given container for execution.
- **Computation Service**: to ease the execution of services on the computational nodes provided by the cloud infrastructure, the architecture exploits virtual containers, e.g., Docker [5], within which service images are deployed and executed.

Docker plays a relevant role in the proposed architecture as it provides the containerization feature enabling easy deployment of service implementations. The Docker architecture is founded on three main concepts: *images*, *containers* and *registry*. An image is a stand-alone and executable package that includes everything needed to run a piece of software (e.g., code, runtime, libraries, etc.). A container is a runtime instance of an image when actually executed. Finally, the registry, referred to as *Docker Hub*, is a repository of images. Docker also includes *volumes*, or directories stored outside of the containers filesystem, which are used to hold data that can persist even after container termination, thus providing data persistence to stateless Docker images.

Regarding the availabity of modeling, transformation and simulation services, it is worth noting that the following services have been already provided as standalone applications that exploit QVT [27], the standard provided by the OMG's Model Driven Architecture (MDA) [28] for the specification of model-to-model transformations, and Acceleo [29], the model-to-text transformation language compliant to the MOFM2T MDA standard [30]:

- **Simulation Services**: *jEQN* [31] and *eBPMN* [32] domain specific languages for enabling the simulation of *EQN* (Extended Queueing Networks) and *BPMN* models, respectively;
- **Model-to-Text Transformation Services**: Acceleo-based transformations to generate *jEQN* code from *EQN* models [2] and *eBPMN* code from *PyBPMN* models, being PyBPMN the performance-oriented BPMN extension proposed in [33] and [34]. Moreover, an *Ecore to Javascript and HTML* Acceleo-based transformation to generate modeling services from domain metamodels specified in terms of Ecore models [35];
- **Model-to-Model Transformation Services**: a QVT-based transformation to generate EQN models from UML activity diagrams [2] and a QVT-based transformation to generate PyBPMN models from BPMN models with text annotations specifying the related performance-oriented characterization [3].

Development activities are currently in progress to finalize the porting of such applications as microservices into the BP-MSaaS platform. In this respect, the flexible and modular approach at the basis of the BP-MSaaS platform design and the availability of several modeling, transformation and simulation services make it possible to effectively use the BP-MSaaS

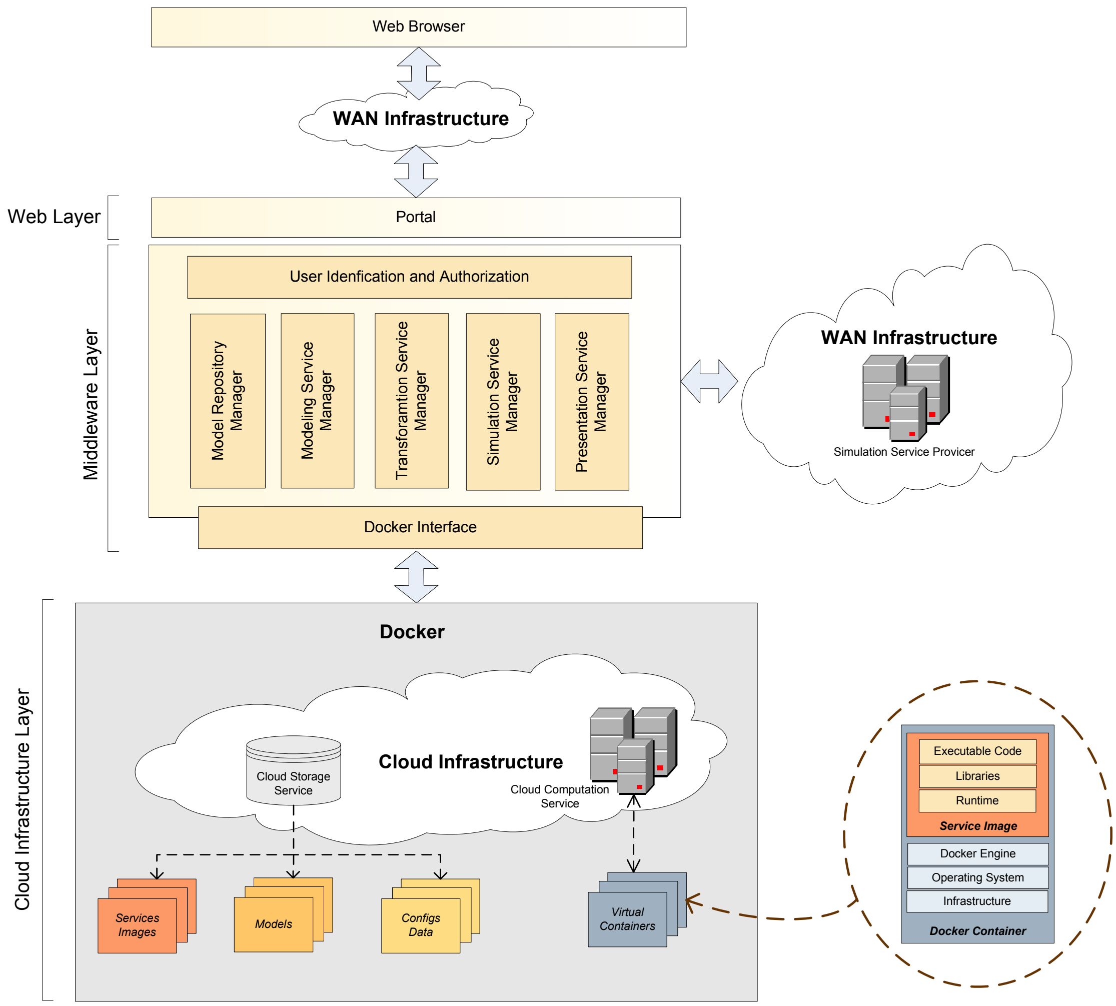

Fig. 1. BP-MSaaS plaform architecture.

platform in various application domains, by selecting the relevant set of services.

Finally, it is worth noting that the proposed architecture still provides the opportunity to make use of remote SOAP/REST Web services offered through a conventional WAN.

The next Section provides an operational view of the proposed approach by giving the behavioral specification of the BP-MSaaS platform.

B. Behavioral Model

Figure 2 shows the UML use case diagram that identifies relevant actors with related responsibilities. The specification of most important use cases is given later on in this Section.

Service Provider is the actor in charge of developing the several services available in the cloud, e.g., the above-mentioned *eBPMN simulation service* or the *UML-to-EQN transformation service*. The *Service Provider* also manages the service catalog by use of the *Modeling Service Manager* component provided by the BP-MSaaS platform, to deploy/delete services onto/from the cloud repository.

End User is the actor carrying out the simulation-based analysis of BPs. In this respect, it is in charge of configuring and executing a simulation experiment by selecting the simulation model and the required services from the catalog. The *Simulation Expert* supports the *End User* and is specifically involved in the definition of simulation models, which can be built from scratch or obtained by executing an automated model transformation (in such a case, models obtained by use of automated model transformations may need to be manually refined and configured).

Domain Expert is the actor that builds the domain models taken as input by model transformations used to generate other models (e.g., model-to-model transformations to generate simulation models) or the code that implements a service (e.g., a model-to-text transformation to generate a modeling service providing a Javascript-based visual editor). Finally, the *System Administrator* is in charge of managing the BP-MSaaS platform and the underlying cloud-based infrastructure.

The UML sequence diagram shown in Figure 3 provides the specification of the *Build simulation experiment* use case.

As specified in Figure 2, the *Build simulation experiment* use case is extended by the *Select and execute transformations*

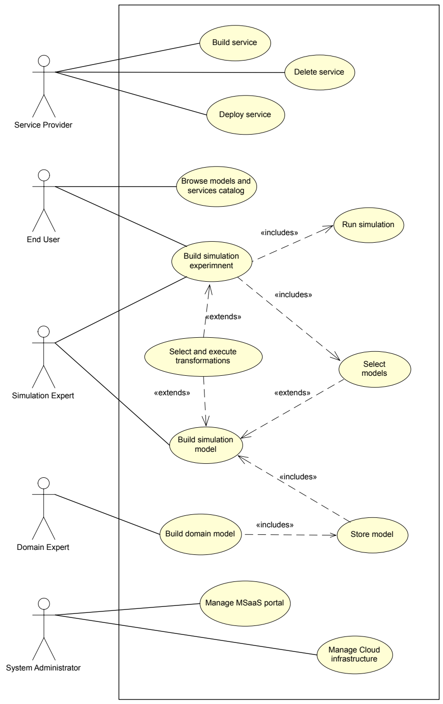

Fig. 2. Use case diagram of the BP-MSaaS platform.

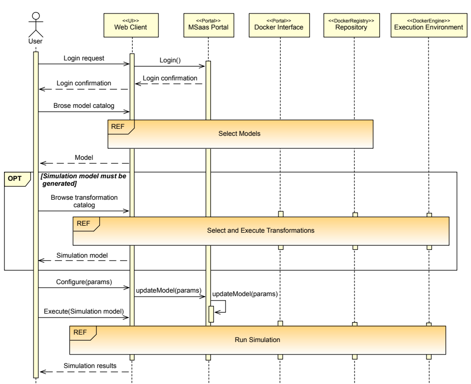

Fig. 3. Specification of the *Build simulation experiment* use case.

use case and includes *Select model* and *Run simulation* use cases.

The *Select model* use case is first activated to browse the catalog and select a model, which may be at various levels of abstraction. In other words, if the selected model is a domain model or an abstract BP model, then the appropriate model transformations have to be selected and executed in order to generate an executable simulation model, as specified by the *optional* fragment in the diagram that refers to the *Select and execute transformations* sequence diagram. Differently, if an executable simulation model is the one selected by the user, then it is directly given as input to the appropriate simulation engine.

Figure 4 shows the sequence diagram specifying the *Select and execute transformations* use case, which is optionally executed when the selected model is not in executable format and thus one or more model transformations are to be carried

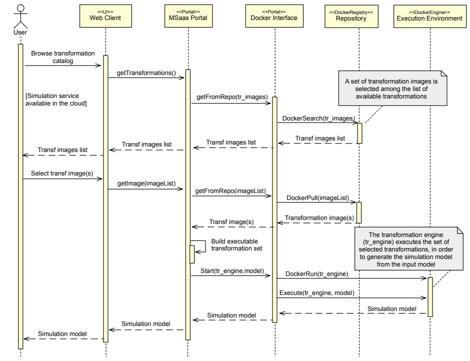

Fig. 4. Specification of the *Select and execute transformations* use case.

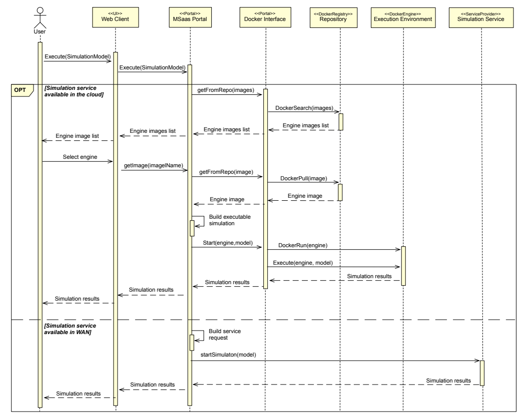

Fig. 5. Specification of the *Run simulation* use case.

out. In the considered scenario, the logged user browses the catalog of available model transformations and selects the set of required ones. It should be noted that the type and the number of the required transformation is strictly related to the type of the input model and the simulation engine which executes the simulation. As an example, an input UML activity diagram to be analyzed by use of an EQN model may require two different model transformations: a *model-to-model* transformation, to generate an abstract EQN model from the input UML activity diagram, and a *model-to-text* transformation to map the EQN model to an executable code ready to be taken as input by the EQN simulation engine. Differently, an input BPMN model may only require a single *model-to-text* transformation to directly generate the simulation implementation code for the available BP simulation engine. The required engine(s) are eventually deployed to the Docker container that executes the transformation(s) yielding the executable simulation model as output.

Finally, Figure 5 shows the sequence diagram specifying the *Run simulation* use case. The identified scenario illustrates how the BP-MSaaS platform allows users to execute simulation services deployed in the cloud or available as conventional SOAP/REST Web services. In the former case, the scenario overlaps with the *Select and execute transformations* use case, and thus the selected engine is deployed to a Docker container to enable the execution of the simulation model. In the latter case, the simulation model is used as input parameter of the invoked remote simulation service.

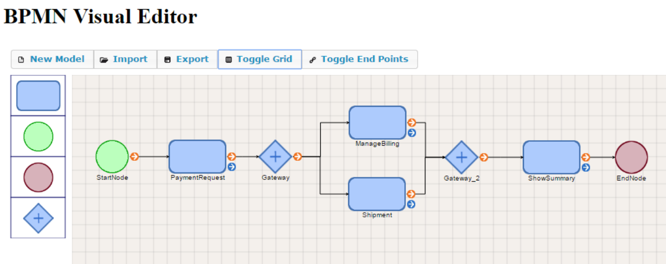

Fig. 6. BPMN model of the *Checkout BP*.

IV. Application Scenario

The proposed BP-MSaaS platform is used in this section to carry out the simulation-based analysis of a BP dealing with the check out of a purchase order submitted to an e-commerce website.

E-commerce application scenarios, being closely related to Business to Business Integration (*B2Bi*), can be effectively improved by use of the BP-MSaaS platform. *B2Bi* tasks are challenging because of the heterogeneous and distributed nature of systems in the enterprise network environment. ERP, CRM and logistic department are always triggered by e-commerce processes. The BP-MSaaS platform contributes to facilitate communication among the disparate systems and to recognize their dependencies with each other, which ultimately enables business process automation.

The model of the BP under study can be either retrieved from a catalog of available models or directly edited by use of a modeling service. Let us assume the BP is specified as a BPMN model built by use of a visual editor provided by the modeling service illustrated in [35]. Figure 3 shows the BPMN visual editor along with the BPMN model of the considered example.

The BPMN model has to be transformed into a simulation-oriented implementation and thus a model-to-text transformation is executed to take as input the BPMN model and yield as output the corresponding simulation code. The image of a *transformation service* is first searched and pulled from Docker Hub. Then, it is deployed and executed in a Docker container in order to get the desired simulation code. Such activities are transparently accomplished and supported by the BP-MSaaS platform.

Specifically, from the set of available services (as summarized in Section III-A) the `BPMN-to-eBPMN` *Model-to-text Transformation Service* is selected and executed in the example application scenario. The transformation takes as input the BPMN model and yields as output the simulation code specified in eBPMN [32], a BP simuation language built on top of the SimArch discrete-event simulation architecture [31].

Finally, the image of a simulation service is selected and executed, as specified by the sequence diagram in Figure 5. In this respect, an eBPMN-compliant simulation service is deployed and executed.

The Docker image which implements the eBPMN simulation engine is based on the official `tomcat` image availabe at https://hub.docker.com/_/tomcat/, a minimal run-time environment that provides Linux OS, Oracle JRE 8 and the Apache Tomcat 8.0 application service, which deploys the `ebpmnService` *war* file into the `webapps` folder.

Similarly to the generation of the eBPMN code, the BP-MSaaS platform transparently provides what needed to build and execute the Docker container, as well as to package the *war* file that contains the executable simulation code.

Finally, an `eBPMN`-based *presentation service* is executed to retrieve the simulation results and provide them in visual form to the user.

As stated in Section I, the BP-MSaaS platform can be extended to cover additional application domains, which can be easily served by selecting a different set of M&S services. As an example, the performance analysis of a software system specified in UML can be enacted by selecting a UML-compliant modeling service, a different simulation service (e.g., the available `jEQN` *simulation service*) and an appropriate set of transformation services (e.g., the available `UML-to-EQN` and `EQN-to-jEQN` *transformation services*).

V. Conclusions

The paper has illustrated the proposed BP-MSaaS platform, which enables the effective adoption of M&S-based approaches throughout the BP lifecycle. Specifically, the BP-MSaaS architecture exploits a cloud infrastructure and the Docker containerization architecture to build a flexible service-oriented platform which provides the facilitated selection, deployment and execution of M&S services that support the business analyst in the specification and simulation-based analysis of BPs. The adoption of model-driven development approaches allows to ease the automated development of services and configuration files, in order to hide the complexity of simulation modeling and implementation activities, as well as platform management activities.

An application scenario dealing with an e-commerce BP has been illustrated to show a concrete example of how the BP-MSaaS platform can be effectively used to support and automate the selection, deployment and execution of modeling, transformation and simulation services.

The prototype Docker-based implementation of the BP-MSaaS platform has been used to assess the feasibility and the effectiveness of the proposed architecture. Ongoing work includes the development of a full-featured implementation of the BP-MSaaS platform, by porting already existing applications as microservices into the platform, as well as by adding simulation engines that cover application domains not limited to the analysis of BPs.

References

[1] D. Gianni, A. D'Ambrogio, and A. Tolk, Eds., *Modeling and Simulation-Based Systems Engineering Handbook*. CRC Press, 2014. [Online]. Available: http://www.crcnetbase.com/isbn/9781466571464

[2] P. Bocciarelli and A. D'Ambrogio, "Automated Performance Analysis of Business Processes," in *Proceedings of the 2012 Symposium on Theory of Modeling and Simulation - DEVS Integrative M&S Symposium*. Orlando, Florida: Society for Computer Simulation International, 2012, pp. 10:1–10:9.

[3] P. Bocciarelli, A. D'Ambrogio, A. Giglio, E. Paglia, and D. Gianni, "Empowering Business Process Simulation Through Automated Model Transformations," in *Proceedings of the Symposium on Theory of Modeling & Simulation - DEVS Integrative*, ser. DEVS '14, Tampa, Florida, 2014, pp. 39:1–39:9.

[4] A. Karmel, R. Chandramouli, and M. Iorga, *NIST Special Publication 800-180: NIST Definition of Microservices, Application Containers and Virtual Machines*, 2016. [Online]. Available: http://csrc.nist.gov/publications/drafts/800-180/sp800-180_draft.pdf

[5] Docker, "Docker - Build, Ship, and Run Any App, Anywhere," Website: https://www.docker.com, 2016. [Online]. Available: https://www.docker.com/

[6] Microsoft, *Microsoft Azure*, 2017. [Online]. Available: https://azure.microsoft.com

[7] Amazon, *Amazon Web Services (AWS)*, 2017. [Online]. Available: https://aws.amazon.com/

[8] W. J. Kettinger, J. T. C. Teng, and S. Guha, "Business Process Change: A Study of Methodologies, Techniques, and Tools," *MIS Quarterly*, vol. 21, no. 1, p. 55, 1997.

[9] R. S. Aguilar-Savén, "Business process modelling: Review and framework," *International Journal of Production Economics*, vol. 90, no. 2, pp. 129–149, 2004.

[10] C. Ren, W. Wang, J. Dong, H. Ding, B. Shao, and Q. Wang, "Towards a flexible business process modeling and simulation environment," in *Proceedings of the 2008 Winter Simulation Conference*, 2008, pp. 1694–1701.

[11] L. Aldin and S. De Cesare, "A comparative analysis of business process modelling techniques," in *U.K Academy for Information Systems (UKAIS 2009), 14th Annual Conference*, Oxford, UK, 2009, pp. 1–17.

[12] D. Moody, "What Makes a Good Diagram? Improving the Cognitive Effectiveness of Diagrams in IS Development," in *Advances in Information Systems Development*. Springer, 2007, pp. 481–492.

[13] L. O. Johansson, M. Wärja, and S. Carlsson, "An evaluation of business process model techniques, using Moody's quality criterion for a good diagram," *CEUR Workshop Proceedings*, vol. 963, pp. 54–64, 2012.

[14] B. Gladwin and K. Tumay, "Modeling Business Processes with Simulation Tools," in *Proceedings of the 26th Conference on Winter Simulation*, 1994, pp. 114–121.

[15] M. H. Jansen-Vullers and M. Netjes, "Business Process Simulation - A Tool Survey," *Workshop and Tutorial on Practical Use of Coloured Petri Nets and the CPN*, vol. 834, pp. 77–96, 2006.

[16] V. Bosilj-Vuksic, V. Ceric, and V. Hlupic, "Criteria for the Evaluation of Business Process Simulation Tools," *Interdisciplinary Journal of Information, Knowledge, and Management*, vol. 2, pp. 73–88, 2007.

[17] G. Abeysinghe and K. Phalp, "Combining process modelling methods," *Information and Software Technology*, vol. 31, no. 2, pp. 514–124, 1997.

[18] F.-R. Lin, M.-C. Yang, and P. Yu-Hua, "A generic structure for business process modeling," *Business Process Management Journal*, vol. 8, no. 1, pp. 19–41, 2002.

[19] K. Vergidis, A. Tiwari, and B. Majeed, "Business Process Analysis and Optimization: Beyond Reengineering," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 38, no. 1, pp. 69–82, 2008.

[20] D. Cetinkaya, A. Verbraeck, and M. Seck, "Applying a Model Driven Approach To Component Based Modeling and Simulation," in *Proceedings of the 2010 Winter Simulation Conference*, 2010, pp. 546–553.

[21] I. Rust, D. Cetinkaya, M. Seck, and I. Wenzler, "Business Process Simulation for Management Consultants: a Devs-Based Simplified Business Process Modelling Library," in *Proceedings of the 23rd European Modelling and Simulation Symposium (EMSS 2011)*, 2011.

[22] O. Kloos, V. Nissen, and M. Petsch, "From Process to Simulation - A Transformation Model Approach," in *Proceedings of the 3rd International Workshop on Enterprise Modelling and Information Systems Architectures*, 2009, pp. 83–96.

[23] M. Indulska, J. C. Recker, M. Rosemann, and P. Green, "Business Process Modeling: Current Issues and Future Challenges," in *Proceedings of the 21st International Conference on Advanced Information Systems Engineering*. Springer-Verlag, 2009, pp. 501–514.

[24] I. Imam, N. Nounou, A. Hamouda, and H. A. A. Khalek, "Survey of business process simulation tools: A comparative approach," *Proceedings of SPIE - The International Society for Optical Engineering*, vol. 8285, no. July, pp. 0–7, 2011.

[25] K. D. Barber, F. W. Dewhurst, R. L. D. H. Burns, and J. B. B. Rogers, "Business-process modelling and simulation for manufacturing management: A practical way forward," *Business Process Management Journal*, vol. 9, no. 4, pp. 527–542, 2003.

[26] D. Cetinkaya, A. Verbraeck, and M. Seck, "MDD4MS: A Model Driven Development Framework for Modeling and Simulation," in *Proceedings of the 2011 Summer Computer Simulation Conference*, 2011, pp. 113–121.

[27] OMG, "Meta Object Facility (MOF) 2.0 Query/View/Transformation," 2015. [Online]. Available: http://www.omg.org/spec/QVT/

[28] OMG, "MDA Guide, version 1.0.1," 2003. [Online]. Available: http://www.omg.org/MDA/

[29] Eclipse Foundation, "Acceleo," sep 2016. [Online]. Available: http://www.eclipse.org/acceleo/

[30] OMG, *MOF Model to Text Transformation Language*, 2008. [Online]. Available: http://www.omg.org/spec/MOFM2T/

[31] D. Gianni, A. D'Ambrogio, and G. Iazeolla, "A software architecture to ease the development of distributed simulation systems," *Simulation*, vol. 87, no. 9, pp. 819–836, 2011.

[32] P. Bocciarelli, A. D'Ambrogio, and E. Paglia, "A Language for Enabling Model-Driven Analysis of Business Processes," in *Proceedings of the 2nd International Conference on Model-Driven Engineering and Software Development*, ser. MODELSWARD'14. Lisbon, Portugal: SciTePress, 2014.

[33] P. Bocciarelli and A. D'Ambrogio, "A model-driven method for describing and predicting the reliability of composite services," *Software & Systems Modeling*, vol. 10, no. 2, pp. 265–280, May 2011.

[34] P. Bocciarelli and A. D'Ambrogio, "A model-driven method for enacting the design-time QoS analysis of business processes," *Software & Systems Modeling*, vol. 13, no. 2, pp. 573–598, May 2014.

[35] P. Bocciarelli, A. D'Ambrogio, A. Giglio, and A. Mastromattei, "Automated Development of Web-based Modeling Services for MSaaS Platforms," in *International Workshop on Model-Driven Approaches for Simulation Engineering (Mod4Sim 2017), part of the 2017 Spring Simulation Multiconference (SpringSim 2017)*, Virginia Beach, VA, USA, 2017.