Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 100 (2016) 229 – 236

Procedia Computer Science

Conference on ENTERprise Information Systems / International Conference on Project MANagement / Conference on Health and Social Care Information Systems and Technologies, CENTERIS / ProjMAN / HCist 2016, October 5-7, 2016

A Study of Petri Nets, Markov Chains and Queueing Theory as Mathematical Modelling Languages Aiming at the Simulation of Enterprise Application Integration Solutions: a first step

Cássio L. M. Belusso[a,b]*, Sandro Sawicki[b], Fabricia Roos-Frantz[b], Rafael Z. Frantz[b]

[a]*Federal University of Fronteira Sul, Rua Jacob Reinaldo Haunpenthal, 1580, Cerro Largo, 97900-000, RS, Brazil*
[b]*Department of Exact Sciences and Engineering, Unijuí University, Rua do Comércio, 3000, Ijuí, 98700-000, RS, Brazil*

Abstract

Enterprise Application Integration (EAI) is a research field that seeks to develop methodologies, techniques and tools to design and development integration solutions. The software ecosystem of companies is comprised of several applications, usually obtained from third parties or developed internally and custom-made for their business processes. Interest in EAI has arisen due to the need to integrate different applications composing the software ecosystem to allow business processes to evolve in response to the constant demands of the business market. The main challenge facing companies in this context is that most of their applications are not designed considering integration with other applications. The development of integration solutions is not a simple task. Guaraná technology provides a domain-specific language that allows for the design of conceptual models to represent integration solutions. This paper reports on a study of Petri Nets, Markov Chains and Queueing Theory, aiming to construct simulation models from conceptual models of integration solutions modeled with Guaraná. We map the building blocks Slot and Task of Guaraná to their corresponding elements in the mathematical modelling languages studied.

Peer-review under responsibility of the organizing committee of CENTERIS 2016

Keywords: Enterprise Application Integration; Domain-Specific Language; Conceptual Model; Simulation.

* Corresponding author. Tel.: +55-55-3359-3950; fax: +55-55-3359-3951.
E-mail address: cassio.belusso@uffs.edu.br

Peer-review under responsibility of the organizing committee of CENTERIS 2016
doi:10.1016/j.procs.2016.09.147

1. Introduction

Business processes often need a set of applications operating together in order to carry out their services. The inclusion of new services into the company's software ecosystem is a difficult task. It has to find ways to incorporate new and different functionalities capable of interacting in a synchronized fashion with existent applications, due to the fact that most applications are not designed to be integrated with others. Enterprise Application Integration (EAI) was developed to solve this problem, by enabling numerous applications not designed to work together to share information and functionalities, providing an immediate solution for this need. Hohpe and Woolf (2003) reported a large set of patterns that could be used to develop integration solutions, depending on the most adequate type of solution for a particular integration problem[1].

Integration solutions are softwares whose main function is to synchronize data among different applications or reuse functionalities. Integration solutions are implanted into the software ecosystem as a new application that offers its users a high-level vision of the integrated applications with which they can interact[2]. Integration solution development consists of design, implementation, execution and monitoring stages. In order to facilitate the understanding of an integration solution's structure and operation, conceptual models are built to represent it in the design phase. There are a number of technologies to build that models, based on a messaging system style, which are appropriate to heterogeneous ecosystems and are ideal for environments that require flexibility[1]. These technologies also allow for the construction of conceptual models with a high level of abstraction, using an easy to understand graphic interface. Of the open source technologies, we can highlight: Spring Integration[3], Apache Camel[4], Mule ESB[5] and Guaraná[6]. Each of these offers a domain-specific language (DSL) to design conceptual models following the Pipes and Filters architectural pattern, in which larger processes are divided into smaller independent services (Filters), which are usually desynchronized by channels (Pipes)[1]. This study utilizes the Guaraná DSL.

Integration solution implementation can be an expensive and time-intensive process, which has encouraged studies that use integration solution simulation approaches. When developing an integration solution, it is important to meet all the demands of the ecosystem in which it will be included or the solution will run the risk of becoming vulnerable to performance issues[7,8,9].

In this paper, we aim to show that it is possible to translate a conceptual model based on Pipes and Filters developed in Guaraná to mathematic models capable of simulation in specific simulation tools. It is thus necessary to create the simulation model based on the conceptual model, which is capable of being processed by a simulator by way of equivalence between elements of the Guaraná model and the translated model. These elements are translated to three different models based on: Petri Nets[10], Markov Chains[11] and Queueing Theory[12]. The simulation of conceptual models is an important contribution in that it can enable the identification of conditions under which an integration solution could fail, while still in the design phase, thus reducing risks and costs involved in its construction.

The notion of simulation refers to the precise representation of all characteristics present in a real system. Simulation is an important ally when facing problems that are highly complex, often impossible to solve analytically or even by actual experimentation, which may be considered expensive. In addition, it allows an improved understanding of the real system and its evolution over time. The main objective of this study is to step forward in the simulation of enterprise application integration solutions.

This paper is organized as follows: Section 2 briefly reports some previous related works from the literature; Section 3 outlines the basic concepts used in the elaboration of conceptual models of integration solutions using Guaraná technologies, presenting the three approaches, Petri Nets, Markov Chains and Queueing Theory, as the objects of the study; Section 4 presents the translation of the elements from a simple model developed in Guaraná to simulation models using the three chosen approaches, as well as the mathematical formalism that serves as a foundation in each approach; finally, Section 5 presents final considerations and perspectives for future work.

2. Related work

Some works in the literature have shown different applications of Petri Nets, Markov Chains and Queueing Theory, but few of these have been in the context of EAI. However, it has been established that integration solutions can be characterized as discrete-event systems, which possess well-established simulation techniques[13]. Although we were unable to find any work which used one of these three techniques to simulate enterprise integration solutions, there are a number of works in diverse areas that have used them to analyze models from simulation.

Petri Nets, for example, were used by Varela et al. (2015) to propose a model that could aid in operational and structural decision-making at a tire industry, seeking to minimize raw material waste from manufacturing processes[14]. Yamada et al. (2002) suggested a simulation of the operational stages in a sugarcane processing industry utilizing Petri Nets[15]. Miyagi et al. (2002) modeled healthcare systems using Petri Nets applied to outpatient services at the Hospital das Clínicas (HC) in São Paulo to study the patient flow and analyze the system evolution as well as the behavior of its sectors[16].

Markov Chains have been applied by Gomes and Wanke (2008) to the management of replacement parts, emphasizing that in a Markov Chain applied to stock management, the possible states denote different stock positions that could occur over time[17]. Paulevé et al. (2011) propose a technique to adjust temporal characteristics in stochastic calculations, introducing a stochastic absorption factor to the classical stochastic calculation, with exponential rates[18]. Basagiannis et al. (2008) introduced a verification of a probabilistic model as an assisted tool approach, feasible to systematically quantify Denial of Service safety threats[19].

Based on Queueing Theory, Doy et al. (2006) proposed an investigation of the e-mail service of the computer network of the Mathematics and Statistics Institute of the University of São Paulo (*Instituto de Matemática e Estatística da Universidade de São Paulo*) to identify statistical characteristics and evaluate performance[20]. Camelo et al. (2010) used Queueing Theory and simulation to analyze characteristics of the service for departing mineral ships at the Ponta da Madeira Maritime Terminal, in order to evaluate the mean number of ships in queue and in the system and the mean waiting time in the queue[21]. Abensur et al. (2003) reported a method for formulating compatible strategies for the Brazilian automatic teller system by evaluating performance variables obtained using Queueing Theory[22].

3. Preliminaries

In this section, we briefly describe Guaraná technology and present the theories upon which the conceptual models of this study are translated: Petri Nets, Markov Chains and Queueing Theory, each of which is suitable for particular types of problems, based on the given approach and the variables to be analyzed.

3.1. Guaraná technology

Guaraná is a technology that provides support for developing integration solutions, enabling solutions with a high level of abstraction through an easy to understand graphical interface that provides a general vision of all of the elements in the integration solution of a software ecosystem. In Guaraná, conceptual models are built using the domain-specific language, called Guaraná DSL. This language follows integration patterns present in the work of Hohpe and Woolf (2003), which make up the base of the Pipes and Filters system, which in Guaraná, correspond to slots and tasks, respectively. Tasks are operations that can be executed in messages. They determine how a set of incoming messages should be processed to produce a set of outgoing messages. The messages are an abstraction of information that is exchanged and transformed within the integration solution. They are composed of a title, a body and attachments. Slots connect the entry of a task to the exit of another task, allowing the messages to be processed in an asynchronous form. All integration solution models developed in Guaraná are independent of technology, that is, it is possible to implement the solutions in different technologies and not only in the technology in which it was developed.

3.2. Petri Nets

Petri Nets are used to model and simulate discrete-event systems, allowing a joint analysis of the behavior and structure of these systems, thus improving understanding of them. Petri Nets are graphs formed by two types of nodes: places, connected by arcs, which represent variables of system states and are represented by circles; and transitions, which represent actions made by the system and are represented by rectangles. Petri Net formalism enables a description only of the logical structure of systems, as it does not account for time. However, time interferes in the functioning of most systems, necessitating its consideration in the system representation[23].

In the circles representing a place in a Petri Net, there may be flow indicators called tokens, whose quantity may vary depending on the functioning of the net. When a transition is activated, it uses a certain number of tokens contained in the input places, generating output places. The execution of a Petri Net only occurs by the firing of transitions, which only occurs if the number of tokens consumed by a transition is equal to the weight value given by the input arc. Occasionally, more than one transition may be ready to fire. When this happens, the Petri Net does not define which will fire first, in other words, there is not a firing priority rule.

Although Petri Nets are widely used in the context of modeling and simulation of discrete-event systems, there is still insufficient information related to their utilization in application integration. However, it is known that conceptual models developed in Guaraná can be understood as discrete-event systems, which can be translated to stochastic Petri Nets, thus making simulation feasible. In this work, a stochastic process is considered a process with a certain level of uncertainty, representing the evolution of a system of random variables as a function of time.

3.3. Markov Chains

Markov Chains are a system modeling formalism that describe systems as a stochastic process with discrete states, characterizing them by their states and the way they alternate. Generally, time is the parameter, discrete or continuous, where the future state in a Markov Chain is not affected by past states, but only by the present state. They are used because, in addition to being simple, they allow performance analyses of varied realities through modeling techniques.

Markov Chains in discrete time represent stochastic processes, describing activities that end in events, which generate state transitions. Probabilities of transitions from one state to another form a matrix, called the transition matrix. A Markov Chain is represented by a state-machine, a sequence of random variables that represents the state at a given time. The executions of discrete time Markov Chains are trajectories, whose probability calculation allows an analysis of behavior, of probabilistic accessibility and of quantitative and qualitative properties. The reward structure representing benefit or cost is modeled by reward functions and can measure the intervals of time spent in each state after a certain number of intervals of time[13].

The use of Markov Chains is suitable for this study, as it deals with particular cases of stochastic processes and thus is utilized for discrete events, allowing an identification and analysis of possible performance bottlenecks during the modeling of a system. The process is initiated when there is a message in the queue. Thus, in simulations with discrete events, the variables can be equated in a model, whose state changes occur at discrete time points. The events generate state transitions, represented by a transition matrix. In addition to the discrete formalism, this type of probabilistic model has ample support in the form of software tools, which aid in the proposal to simulate integration solutions, in terms of formal verification of the model and performance analysis.

3.4. Queueing Theory

Queueing Theory is an analytical method that utilizes mathematical equations to understand system problems and aims to determine and evaluate performance measures that express productivity or operationality of the system. A queueing system can be represented by different models, but the basic process has common elements. Briefly, the process begins with clients seeking a service and arriving randomly over time, forming a queue. At a given moment, one of the clients is called to be served following a rule, denominated queue discipline. The client is served using the service mechanism and then leaves the queue system[24].

Specification of a queue model usually requires that performance characteristics be explicit. These characteristics are: arrival process, service process, number of servers, queue capacity, population size and queue discipline. With these data, it is possible to calculate and estimate results for system performance, based on properties such as service time and number of clients served in a given time period. By obtaining inadequate values, it is possible to make the queueing system more efficient, avoiding performance problems.

The functional dynamics of an integration solution are similar to those adopted by Queueing Theory. For example, a client can be seen as a message that can wait in the queue (slot) for a given amount of time to be processed by the service (task). Since the characteristics of individual components of an integration solution are known, a simulation based on discrete events is a good strategy for obtaining quantitative and qualitative measures of the system.

4. Equivalence of conceptual model elements

In general, behavior analysis and identification of performance bottlenecks in application integration solutions involve implementation for subsequent execution and testing against critical scenarios designed by the software engineers. However, this can be expensive, risky and time-intensive[23]. The conceptual models developed in Guaraná feature a high level of detail in the integration solutions and can be classified as stochastic, dynamic and discrete. A discrete system can be translated to a simulation model and simulated in specific environments, in order to obtain key information about system operation, allowing an analysis of various aspects of the system, without needing to implement it, thus reducing the resources used in the integration solution development.

Fig. 1 shows the set of elements that makes up the graphical notation of the Guaraná DSL. The processes are blocks that aggregate a given set of tasks. They also possess four ports through which a process can share information with integrated applications. These are: entry port, exit port, solicitor port and responder port. The slots are temporary storage units that connect the ports with tasks or tasks with tasks. Slots provide an independence between tasks through asynchronous processing of the messages that flow through the process. The tasks are responsible for processing and modifying the messages. They operate from a received message from the input slot. When a task receives a message, it reads the content, processes the message and writes on the next slot, leaving it available for the next task. Possible tasks in Guaraná include translation, filtering and delivery of messages.

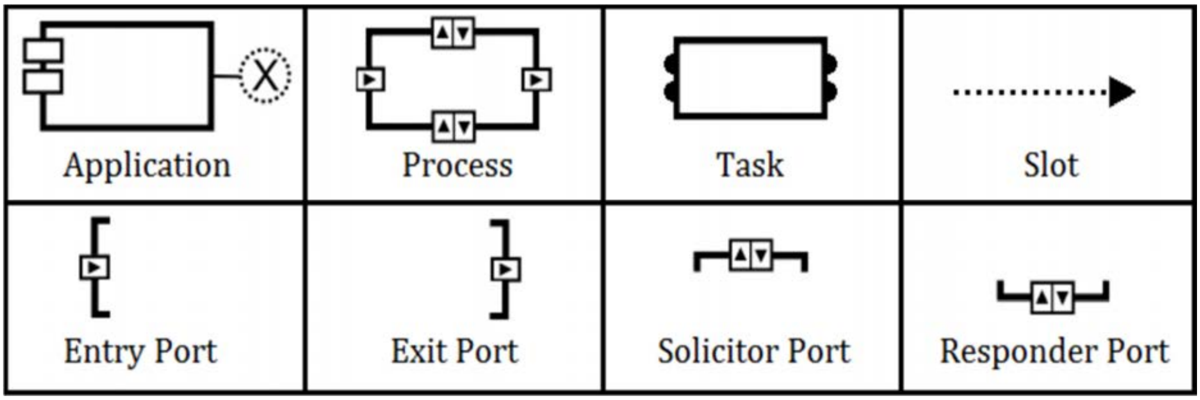

Fig. 1. Graphical notation of Guaraná DSL.

In order to simulate an integration solution, it is necessary to use a simulation tool, which consequently requires the creation of a simulation model based on the conceptual model[25]. The Guaraná model and the translated models are shown in Fig. 2.

The use of different mathematical theories is based on the specificity of each problem. In general, Markov Chains, and more specifically their decision processes, are not recommended for large systems, as they are difficult to build, despite offering more general applicability. Stochastic Petri Nets do not possess an explicit structure, as Queueing Theory does, but they have a graphical notation system and the models have a formal interpretation. Although Queueing Theory facilitates the construction of the model, it has significant limitations, notably its lack of a formal interpretation[13].

These theories were chosen as objects of study due to the various techniques and tools they possess for simulating discrete events. Integration solutions developed in Guaraná possess characteristics like those of a discrete-event system through the relationship between its elements and the functional structure. In discrete-event system modeling, as in integration solutions, the identification of and relationship to performance variables and the way they interact with each other and with other elements aids to understand the system.

Fig. 2 demonstrates that the proposal for translating integration solution conceptual models into Petri Nets, Markov Chains and Queueing Theory models is possible due to the analogy between their components and the similarity to the execution model. The main elements in Guaraná conceptual models are messages, slots and tasks, which are equivalent, respectively, to: tokens, places and transitions in Petri Nets; processes and steps in Markov Chains; and customers, queues and services in Queueing Theory.

In Petri Nets, the semantics of Guaraná tasks cannot be translated and, therefore, its functionality is maintained and represented in an abstract form. The arcs are components without functional values and, in addition to connecting places and transitions, they also indicate the message flow direction. In order to better understand how the analogy between these approaches functions, we can consider an example of two elements of Guaraná graphical notation, the entry port and exit port, which make up the integration solution. The entry port inserts new messages into the integration solution to be processed. Its representation in Petri Nets is equivalent to a transition (task), which upon firing, inserts a token (message) into a place (slot). On the other hand, the messages that are in the slots exit the process through a transition, in this case the exit port.

The discrete-time Markov Chain is a conceptual model of a system as a discrete set of states, where the transitions between states occur in discrete intervals of time. In its graphical notation, the states are circles and the transitions are arrows, labelled with their associated probabilities. The atomic propositions used to label the states are derived from the set of atomic propositions. In Markov Chains, the process begins when a messages arrives and waits in the system. The steps correspond to both slots and tasks in Guaraná. After one step of time, the process enters the state from which the message is sent to the next step with a probability between 0 and 1, followed by another step of time, again with a probability between 0 and 1, and so on throughout the process.

In Queueing Theory, customers arrive, form a queue and wait for a given time to be served. After being served, the customer leaves the queueing system. Service is given in accordance with a rule, called queue discipline, which may be, among others: First-In, First-Out (FIFO), Last-In, First-Out (LIFO), priority-based service and random service. The similarity to Guaraná integration solutions can be seen in that the messages (customers) arrive and wait in the slot (queue) to be processed by the task (service). The messages are selected for execution following a given message processing order (queue discipline). After processing, the messages exit the task, following the integration flow.

A mathematical formalism is present in these theories. In Petri Nets, the formal representation for a simulation model is expressed by the sequence $MdS = (S, T, A, TD)$, where S represents the set of places, T represents the set of transitions and A represents the mapping of arcs, subdivided later into subsets A_e and A_s, sequences that represent, respectively, input arcs and output arcs. The set TD represents the value of transition firing rates with respect to the order of the set T [13].

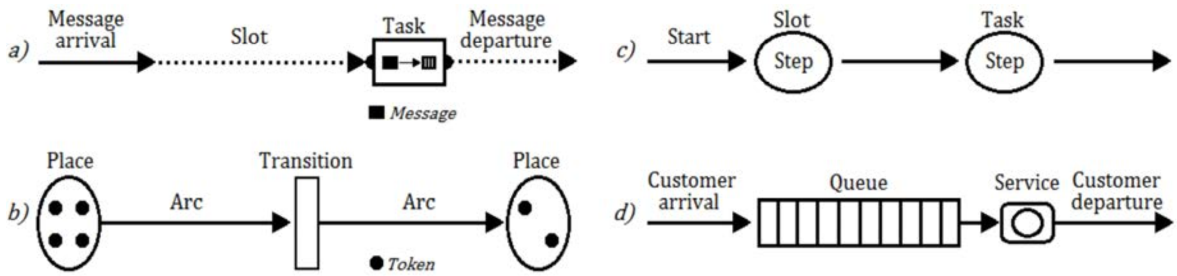

Fig. 2. Execution models in: (a) Guaraná, (b) Petri Nets (c) Markov Chains (d) Queueing Theory.

The mathematical modeling of a discrete-time Markov Chain is described formally as a finite set of states (S), an initial state s_0, belonging to S, a transition probability matrix (P) de $S \times S \rightarrow [0-1]$, a set of atomic propositions (AP), and a labeling function, attributing to the states a set of atomic propositions $L: S \rightarrow 2^{AP}$. Formally, the Markov decision process is a sequence $(S, s_{init}, Passos, L)$, where S is a finite set of states, $s_{init} \in S$, and s_0 is the initial state. The steps $S \rightarrow 2^{Act \times Dist(S)}$ are a function of transition probability, Act is a set of actions, $Dist(S)$ is a set of discrete probability distributions over S, and $L: S \rightarrow 2^{AP}$ is a labeling with atomic propositions[13,25].

In Queueing Theory, some random variables for evaluating simulation model behavior are obtained by simple mathematical equations. Considering that λ is the mean arrival rate and μ is the mean service rate, one can determine, for instance, the mean number of messages in the system (S_n), the mean time that a message remains in the system (I_p), the mean interval between arrivals (A_r), the mean time lost by the message in the slot (T_q), the mean number of messages in the slot (N_q), the mean service time (S_t) and the mean number of processed messages (N_m) [23]. The equations are laid out in detail in (1).

$$S_n = \frac{\lambda}{\mu-\lambda}; I_p = \frac{1}{\mu-\lambda}; A_r = \frac{1}{\lambda}; T_q = \frac{\lambda}{\mu(\mu-\lambda)}; N_q = \frac{\lambda^2}{\mu(\mu-\lambda)}; S_t = \frac{1}{\mu}; N_m = \frac{\lambda}{\mu} \quad (1)$$

In dealing with a process with some level of uncertainty and without specificity as to the probability distribution of arriving messages in the integration solution, it is necessary to indicate one of a number of probabilistic distributions for this task. One example is the negative exponential distribution, in which $F_{Xi}(x) = 1 - e^{-\lambda_i x}$ is the probability of transition x firing at that instant, considering its firing rate λ_i.

5. Conclusions and future research

EAI seeks to develop methods and tools for designing and implementing integration solutions with a high level of abstraction using conceptual models. An integration solution can be simulated through a simulation model created from the translation of elements that make up the conceptual model it represents.

This work presents the translation of conceptual models based on Pipes and Filters for three mathematical models that enable its simulation, allowing an analysis of system behavior while still in the design phase, thus reducing integration solution implementation costs. In order to demonstrate this proposal, we presented a mapping of building blocks Slot and Task from Guaraná technology and its translated models for Petri Nets, Markov Chains and Queueing Theory.

Throughout the research undertaken in this work, it became clear that EAI still requires a great deal of investigation, mainly in the area of simulation of integration solutions. Thus, it is hoped that in future studies, it will be possible to provide tools and methods to support this important task in the simulation and modeling of a system, specifically to determine whether a proposed model faithfully portrays the system under investigation.

Acknowledgements

The research work on which we report in this paper is supported by CNPq, CAPES, FAPERGS, and the internal Research Programme at UNIJUI University. First author is also thanks the UFFS University for the support to the development of his research.

References

1. Hohpe G, Woolf B. *Enterprise integration patterns: designing, building, and deploying messaging solutions*. 1st ed. Boston: Addison-Welsey Professional; 2003.
2. Frantz RZ, Corchuelo R, Roos-Frantz F, Sawicki S. Modelling enterprise application integration solutions. In: *Advances in Engineering Research*. Nova Science Publishers; 2015. p. 67-96.
3. Fisher M, Partner J, Bogoevici M, Fuld I. *Spring integration in action*. 1st ed. Manning Publications Co.; 2012.
4. Ibsen C, Anstey J. *Camel in action*. 1st ed. Manning Publications Co.; 2011.
5. Dossot D, D'Emic J. *Mule in action*. 1st ed. Manning Publications Co.; 2009.
6. Frantz RZ, Reina-Quintero AM, Corchuelo R. A Domain-specific language to design enterprise application integration solutions. In: *International Journal of Cooperative Information Systems* 2011;**20(2)**:143-176.
7. Cargnin RS. *Modelagem e simulação de uma solução de integração para identificação de gargalos de desempenho baseadas em formalismo matemático - uma abordagem orientada a redes de Petri*. Dissertação de Mestrado. Mestrado em Modelagem Matemática, UNIJUI, 2016.
8. Horn MM. *Modelagem e simulação de uma solução de integração para identificação de gargalos de desempenho baseadas em formalismo matemático - uma abordagem orientada à cadeias de Markov*. Dissertação de Mestrado. Mestrado em Modelagem Matemática, UNIJUI, 2016.
9. Wiesner AK. *Modelagem e simulação de uma solução de integração para identificação de gargalos de desempenho baseadas em formalismo matemático - uma abordagem orientada à teoria das filas*. Dissertação de Mestrado. Mestrado em Modelagem Matemática, UNIJUI, 2016.

10. Petri, CA. *Communication with Automata.* Griffiss Air Force Base. Tech. Report. RADC-TR. 1 (suppl. 1). New York; 1966.
11. Brémaud, P. *Markov chains: gibbs fields, monte carlo simulation, and queues.* 1st ed. New York: Springer; 1999.
12. Kendall, DG. Stochastic processes occurring in the theory of queues and their analysis by the method of the imbedded Markov chain. In: *The Annals of Mathematical Statistics* 1953;**24(3)**:338-354.
13. Sawicki S, Frantz RZ, Basto-Fernandes V, Roos-Frantz F, Yevseyeva I, Corchuelo R. Characterising enterprise application integration solutions as discrete-event systems. In: *Handbook of Research on Computational Simulation and Modeling in Engineering*. IGI Global; 2015. p. 261-288.
14. Varela AM, Ramírez JAR, Gómez LHH, González AM, Reyes MYJ. Lean production system model with Petri nets to support for decision making. In: *Ingeniare. Revista Chilena de Ingeniería* 2015;**23(2)**:182–195.
15. Yamada MC, Porto AJV, Inamasu RY. Application of modeling and Petri net concepts in the productive process of the sugarcane industry. In: *Revista Pesquisa Agropecuária Brasileira. Brasília* 2002;**37(6)**:809–820.
16. Miyagi MM, Miyagi PE, Kisil, M. Modelagem e análise de serviços de saúde baseados em redes de Petri interpretadas. In: *Revista Produção* 2002;**11(2)**:23-39.
17. Gomes AVP, Wanke P. Modelagem da gestão de estoques de peças de reposição através de cadeias de Markov. In: *Gestão & Produção* 2008;**15(1)**:57–72.
18. Paulevé L, Magnin M, Roux O. Tuning temporal features within the stochastic π-calculus. Software Engineering. In: *IEEE Transactions on Software Engineering, Institute of Electrical and Electronics Engineers* 2011;**37(6)**:858–871.
19. Basagiannis S, Katsaros P, Pombortsis A, Alexiou N. A probabilistic attacker model for quantitative verification of DoS security threats. In: *Proceedings of the 32nd Annual IEEE International Conference on Computer Software and Applications (COMPSAC'08)*. Finland: 2008. p. 12–19.
20. Doy FE, Bressan G, Pereira GHdeA, Magalhães MN. Simulação do serviço de correio eletrônico através de um modelo de filas. In: *Pesquisa Operacional* 2006;**26(2)**:241–253.
21. Camelo GR, Coelho AS, Borges RM, de Souza RM. Teoria das filas e da simulação aplicada ao embarque de minério de ferro e manganês no terminal marítimo de ponta da madeira. In: *XXX Encontro Nacional de Engenharia de Produção*. São Carlos: 2010. p. 1-14.
22. Abensur EO, Brunstein I, Fischmann AA, Ho LL. Perspectives of the brazilian self-service banking: A strategic approach based on the queueing theory. In: *Revista de Administração Mackenzie* 2003;**4(2)**:39-59.
23. Roos-Frantz F, Binelo M, Frantz RZ, Sawicki S, Basto-Fernandes V. Using Petri nets to enable the simulation of application integration solutions conceptual models. In: *17th International Conference on Enterprise Information Systems (ICEIS)*. Barcelona: 2015. p. 87-97.
24. Hillier FS, Lieberman GJ. *Introduction operations research.* 8th ed. São Paulo: McGraw Hill; 2006.
25. Frantz RZ, Sawicki S, Roos-Frantz F, Yevseyeva I, Emmerich M. On using Markov decision processes to model integration solutions for disparate resources in software ecosystems. In: *17th International Conference on Enterprise Information Systems (ICEIS)*. Barcelona: 2015. p. 260-268.