

Evento: XXV Jornada de Pesquisa
ODS: 9 - Indústria, Inovação e Infra-estrutura

O MOTOR DE EXECUÇÃO DE PLATAFORMAS DE INTEGRAÇÃO BASEADO EM TAREFAS E A TEORIA DAS FILAS¹

THE TASK-BASED INTEGRATION PLATFORM EXECUTION SYSTEM AND THE QUEUE THEORY

Diogo Izequiel Rüdell², Fabricia Carneiro Roos Frantz³, Rafael Zancan Frantz⁴

¹ Projeto de Pesquisa Realizado no Programa de Pós-Graduação em Modelagem Matemática, Grupo de Pesquisa em Computação Aplicada (GCA), Departamento de Ciências Exatas e Engenharias, UNIJUI, Ijuí-RS

² Mestrando do Programa de Pós-Graduação em Modelagem Matemática, diogo.rudell@sou.unijui.edu.br, UNIJUI, Ijuí-RS

³ Professora Doutora do Departamento de Ciências Exatas e Engenharias, Orientadora, frfrantz@unijui.edu.br, UNIJUI, Ijuí-RS

⁴ Professor Doutor do Departamento de Ciências Exatas e Engenharias, Coorientador, rzfrantz@unijui.edu.br, UNIJUI, Ijuí-RS

Resumo

Atualmente é comum encontrar uma empresa que tenha um ecossistema de software composto por diversos aplicativos, que foram criados com diferentes tecnologias, modelos de dados, sistemas operacionais e, na maioria dos casos, não foram projetados para trocar dados e compartilhar funcionalidades. Essa heterogeneidade impulsionou o campo de estudo da Integração de Aplicações Empresariais que busca fornecer metodologias e ferramentas para projetar e implementar uma solução para integrar um ecossistema de software heterogêneo. Existem diversas plataformas que foram desenvolvidas a fim de dar suporte para a integração de aplicações, fazendo com que as aplicações trabalhem de forma integrada. O motor de execução das plataformas de integração é responsável pela execução das soluções de integração e está diretamente relacionado ao tempo de execução de uma solução, o que torna seu desempenho uma questão extremamente importante. Portanto modelar e analisar o motor de execução das plataformas de integração é essencial. Nesse sentido, sabe-se que a simulação envolve a criação de um modelo que represente os comportamentos do sistema real, para posteriormente modelar e simular o funcionamento. Este artigo propõem o uso da Teoria das Filas para modelar o motor de execução de plataformas de integração baseado em tarefas.

Abstract:

Currently, it is common to find a company that has a software ecosystem composed of several applications, which were created with different technologies, data models, operating systems and, in most cases, were not designed to exchange data and share functionality. This heterogeneity boosted the field of study of the Integration of Business Applications that seeks to provide methodologies and tools to design and implement a solution to integrate a heterogeneous software ecosystem. There are several platforms that have been developed in order to support the integration of applications, making the applications work in an integrated manner. The runtime system of the integration platforms is responsible for the execution of the integration solutions and is directly related to the execution time of a solution, which makes its performance an extremely important issue. Therefore, modeling and analyzing the integration platform runtime system is essential. In this sense, it is known that the simulation involves the creation of a model that represents the behaviors of the real system, to later

Evento: XXV Jornada de Pesquisa
ODS: 9 - Indústria, Inovação e Infra-estrutura

model and simulate the functioning. This article proposes the use of Queue Theory to model the runtime system of task-based integration platforms.

Palavras-chave: Plataformas de integração; Motor de execução baseado em tarefas; Redes de Petri; Teoria das Filas.

Keywords: Integration platforms; Task-Based Runtime System; Petri nets; Queuing theory.

INTRODUÇÃO

A crescente utilização de aplicações (*softwares*) pelas empresas ocorre devido a necessidade de competitividade e a necessidade de agilidade em seus processos de negócios, sendo que uma aplicação tem como finalidade organizar um conjunto de dados e informações de forma integrada. As aplicações buscam atender as necessidades da empresa como coletar, organizar, distribuir e disponibilizar informações para os seus processos de negócios (FREIRE et al., 2019).

O desenvolvimento e a aquisição das aplicações de forma desordenada acarretou em uma ampla gama de aplicações, o que ficou conhecido como ecossistema de *software*. As aplicações são desenvolvidas com linguagem de programação variadas, rodam em plataformas diferentes e possuem finalidades específicas. Dessa forma torna-se um desafio quando dois ou mais aplicativos precisam colaborar para apoiar um processo de negócio (FREIRE et al., 2019).

O campo de pesquisa conhecido como Integração de Aplicações Empresariais (EAI) busca oferecer metodologias, técnicas e ferramentas para a concepção e a implementação de soluções de integração (HOHPE & WOOLF, 2004). Nos últimos anos, muitas plataformas de integração foram criadas, as quais fornecem suporte para projetar, implementar, executar e monitorar as soluções de integração, fazendo com que as aplicações trabalhem de forma sincronizada. Dentre as várias plataformas de integração criadas estão a *Apache Camel* (IBSEN & ANSTEY, 2018), *Mule* (DOSSOT et al., 2014), *Spring Integration* (FISHER et al., 2012), *Petals* (SURHONE et al., 2010) e *Guaraná* (FRANTZ, 2012).

Essas tecnologias suportam os padrões de integração documentados por Hohpe & Woolf (2004), tornando-se referência na criação de plataformas de integração. Além disso, seguem o estilo de arquitetura *Pipes-and-Filters* (HOHPE & WOOLF, 2004). Os *pipes* representam os canais onde as mensagens da solução de integração transitam e os *filters* representam as tarefas que implementam os padrões de integração, utilizados para processar os dados encapsulados nas mensagens. Permitindo



Evento: XXV Jornada de Pesquisa
ODS: 9 - Indústria, Inovação e Infra-estrutura

desta forma dessincronizar as tarefas que compõem a solução de integração (ALEXANDER, 1977).

Criada por pesquisadores que atuam no Grupo de Pesquisa em Computação Aplicada (GCA) da Universidade Regional do Noroeste do estado do Rio Grande do Sul - UNIJUI, grupo pelo qual esta pesquisa é desenvolvida, a tecnologia Guaraná possibilita projetar, implementar e executar soluções de integração. A maximização do desempenho de uma plataforma está diretamente relacionada ao motor de execução, o qual pode ser estudado através de modelos matemáticos, simulação computacional e técnicas de modelação. O artigo propõe o uso da Teoria das Filas para modelar o funcionamento do motor de execução de plataformas de integração baseado em tarefas, utilizando o formalismo de Redes de Petri.

BACKGROUND

A plataforma Guaraná oferece uma linguagem de domínio específico (DSL), que permite modelar com alto nível de abstração (FRANTZ, 2012). A Guaraná DSL possui uma sintaxe gráfica concreta que possibilita aos engenheiros de *software* uma visão de todos os processos presentes em uma solução de integração. Os modelos obtidos com essa tecnologia são independentes de plataforma, o que permite que as soluções projetadas possam ser implementadas em qualquer tecnologia. Dessa forma, os engenheiros de software podem manter o foco na criação de modelos, sem preocupar-se com detalhes da implementação da solução.

O Guaraná segue o estilo de integração baseado em mensagens e dá suporte ao uso dos padrões de integração (HOHPE & WOOLF, 2004). A plataforma Guaraná permite a modelagem, implementação e o monitoramento de soluções de integração, a qual é composta por alguns construtores principais, tais como: mensagem, tarefa, *slot*, porta e recursos. A mensagem é a informação transmitida e transformada na solução de integração. As tarefas são as unidades que executam um determinado processamento nas mensagens que fluem na solução de integração, como filtro, cópia, transformação, separação, reagrupamento. Os *slots* fazem a conexão entre tarefas ou entre tarefas e portas. As portas são usadas para que as tarefas interajam com os recursos. E os recursos representam as fontes de informação que são tipicamente aplicativos ou bancos de dados.

As Redes de Petri são uma técnica de modelagem baseada em formalismo matemático que permite a representação de sistemas paralelos, concorrentes e não-determinísticos (MURATA, 1989). Como ferramenta gráfica, podem ser usadas como um auxílio na comunicação visual semelhante a gráficos

Evento: XXV Jornada de Pesquisa
ODS: 9 - Indústria, Inovação e Infra-estrutura

de fluxo, diagramas de blocos e redes (PETRI, 1962), permitindo uma visualização estrutural e comportamental dos processos do sistema (LAW & KELTON, 2000).

Composta por dois elementos, um passivo (lugar) e outro ativo (transição), as Redes de Petri são grafos bipartidos, os lugares representam uma condição, atividade ou recurso, sendo representados por círculos, e as transições representam um evento (ação) do sistema, cuja representação gráfica é um traço, barra ou retângulo. A união desses elementos formam os vértices do grafo associado às Redes de Petri (MARSAN, 1988), (CHWIF, 1999). Desta forma, uma Redes de Petri é um grafo direcionado com dois tipos de nós, não havendo arcos entre nós do mesmo tipo (DESEL & ESPARZA, 2005).

Os vértices são interligados por arcos dirigidos, que interligam lugares às transições, que correspondem à relação entre as condições verdadeiras, que em dado momento, possibilitam a execução das ações, como pode-se observar na Figura 1. Enquanto os arcos que interligam transições aos lugares representam a relação entre as ações e as condições que se tornam verdadeiras com a execução das ações (PETERSON, 1977).

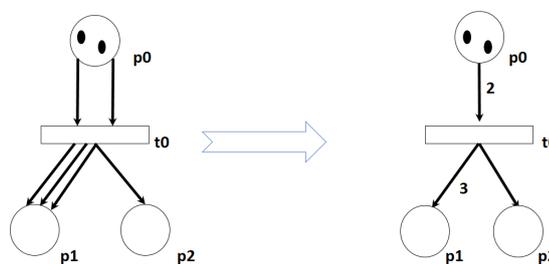


Figura 1: Representação de arco valorado (DESEL; ESPARZA, 2005).

Os lugares podem armazenar marcações (*tokens*), os quais são representados por pontos pretos. A distribuição de *tokens* está relacionada ao estado da Redes de Petri. Além disso, é possível que os vértices de um grafo sejam interligados por múltiplos arcos. Entretanto por conveniência os múltiplos arcos são substituídos por um único arco valorado, onde o numeral associado corresponde ao número de arcos que interligam os vértices.



Evento: XXV Jornada de Pesquisa
ODS: 9 - Indústria, Inovação e Infra-estrutura

PROPOSTA DE SOLUÇÃO

A plataforma Guaraná é construída usando tecnologia Java, que é uma linguagem de programação (MCMMASTER et al., 2017) e fornece uma linguagem de domínio específico, um toolkit de desenvolvimento, um ambiente para realização de testes, uma ferramenta de monitoramento e um motor de execução (FRANTZ, 2012). O motor de execução utiliza uma fila, que segue a disciplina de quem primeiro chega, primeiro sai (*first-in first-out*), também conhecida como FIFO e um conjunto de recursos computacionais (*threads*).

A execução de uma solução de integração é de responsabilidade do motor de execução (*Runtime System*), o qual possui papel fundamental no desempenho da solução. A literatura apresenta dois principais modos de funcionamento dos motores de execução, um baseado em processo (*Process-Based*) e outro baseado em tarefas (*Task-Based*) (FRANTZ, 2012), (BLYTHE et al., 2005), (ALKHANAK et al., 2016), sendo essa classificação oriunda da forma com que as tarefas são executadas.

Com o objetivo principal de executar tarefas, o motor baseado em tarefas executa os padrões que foram programados, modificando, transformando e roteando se necessário. Onde as tarefas são implementadas pelas plataformas através dos padrões de integração, sendo que cada tarefa é única e possui seu código de execução particular. Em um processo existe uma ou mais tarefas, as quais se comunicam indiretamente através de canais (*slots*) e suas respectivas portas. Cada canal equivale a uma unidade de armazenamento temporário, onde existe a prioridade na memória que faz com que o processo ocorra de forma assíncrona. As portas tem a função de enviar e receber mensagens, ou seja, funciona como um ponto de conexão entre um canal e uma tarefa (FRANTZ, 2012).

A simulação foi uma das primeiras aplicações dos computadores, por apresentar-se de forma versátil, com poucas restrições e com resultados relativamente fáceis de interpretar (AALST, 2015). A simulação pode ser definida como uma técnica de solução de um problema, sendo realizada através da análise de um modelo que descreve o comportamento do sistema real. Desta forma pode-se definir o modelo como uma abstração da realidade, que visa se aproximar do verdadeiro comportamento do sistema. Ao modelar um sistema, o objetivo é capturar o que realmente é importante no sistema, sendo assim, o modelo deve ser mais simples que o sistema real (WAINER, 2017). Então para estudar e analisar o funcionamento do motor de execução é necessário abstrair os aspectos essenciais e selecionar os pressupostos básicos que a caracterizam, para então escolher uma técnica, modelar ou simular.

Evento: XXV Jornada de Pesquisa
ODS: 9 - Indústria, Inovação e Infra-estrutura

O processo de execução de uma solução de integração pelo motor se inicia quando uma mensagem chega na porta de entrada, e percorre o canal até a primeira tarefa do fluxo. Então cria-se uma anotação (*Work Unit*) que informa aos recursos computacionais que está pronta para ser executada, essa anotação vai para uma fila de trabalho (*Work Queue*) aonde aguarda até ser executada pelos recursos computacionais (*threads*) (ALKHANAK et al., 2016). Ao terminar a execução na primeira tarefa a mensagem segue o fluxo até a próxima tarefa, onde se repete a mesma situação, como se pode observar na Figura 2. Desta forma, o processo faz com que exista a relação de dependência durante a execução das tarefas, onde a tarefa posterior só pode ser executada quando a primeira já tenha sido executada.

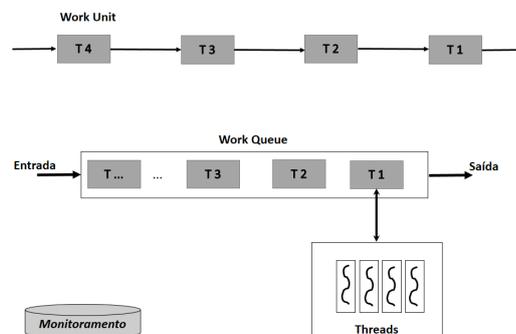


Figura 2: Representação do motor de execução baseado em tarefas das plataformas de integração (FRANTZ, 2012).

O processo de aguardar em uma fila até ser atendido não é agradável, e quando é necessário esperar por um longo período de tempo, causa transtornos para as pessoas. A partir disto, muitas vezes as pessoas tomam atitudes de não retornar mais ao estabelecimento que possui a longa fila de espera. Para as empresas, isso pode significar uma grande perda financeira. Em uma solução de integração, onde as mensagens esperam *threads* ficarem disponíveis para serem executadas, ocorre a geração de filas que pode causar ineficiência do sistema (HILLIER & LIEBERMAN, 2013). A execução das mensagens ou soluções de integração pelos *threads* do motor de execução, se inicia quando elas chegam na fila de trabalho, o que torna o estudo de Teoria das Filas de importante relevância, pois existe a relação de produtor e consumidor na fila de trabalho, como salientado por Aalst & Stahl (2011).

Evento: XXV Jornada de Pesquisa
ODS: 9 - Indústria, Inovação e Infra-estrutura

A Teoria das Filas é um método analítico que estuda a formação de filas por meio de fórmulas matemáticas e usa modelos de filas para representar os diversos tipos de sistemas de filas que surgem na prática. Conforme Fogliatti & Mattos (2007), um sistema de filas é qualquer processo onde clientes oriundos de uma população chegam para receber um serviço. Quando a demanda é maior do que a capacidade de atendimento, em termos de fluxo, os clientes esperam e, depois do atendimento, saem do sistema.

Na Teoria das Filas, os sistemas são representados por modelos, de acordo com suas características, possuindo elementos comuns a todos que fazem parte do processo básico. Na Figura 3, os elementos que compõem o processo são os clientes que necessitam de atendimento, os quais chegam ao longo do tempo, entram no sistema de filas e, devido a indisponibilidade de atendimento imediato, formam uma fila. Os clientes são selecionados em certos momentos para o atendimento por uma regra conhecida como disciplina da fila. Depois que o cliente é atendido pelos servidores, deixa o sistema de filas (COLUMBO et al., 1994).



Figura 3: Modelo de execução baseado na Teoria das Filas (AALST & STAHL, 2011).

Desta forma pode-se comparar o modelo conceitual do motor de execução baseado em tarefas da plataforma Guaraná com modelos matemáticos baseados na Teoria das Filas. Analogamente pode-se dizer que os clientes são equivalentes as mensagens ou soluções de integração a serem processadas, a disciplina da fila de trabalho do motor de execução baseado em tarefas é FIFO e os servidores são equivalentes aos threads. O que fundamenta a proposta de modelagem analítica do motor de execução de plataformas de integração baseado em tarefas com a Teoria das Filas.

Para simular uma solução de integração faz-se necessário a utilização de ferramentas de simulação, que por consequência exige a criação de um modelo de simulação a partir do modelo conceitual. Sendo que para construir modelos de simulação existem diversas linguagens ou formalismos matemáticos para representar sistemas discretos, como por exemplo, as Redes de Petri, que permitem representar sistemas concorrentes, paralelos, assíncronos e não determinísticos.



Evento: XXV Jornada de Pesquisa

ODS: 9 - Indústria, Inovação e Infra-estrutura

As Redes de Petri oportunizam uma visualização de alta abstração e acompanhamento dinâmico das diferentes atividades do sistema que possuem características predominantemente discretas (AALST, 2015). Pode-se afirmar que o modelo de fila criado e modelado com Redes de Petri por Columbo et al. (1994), chamada de fila estática representa de forma satisfatória a fila de trabalho (Work Queue) dos motores de execução de plataformas de integração baseados em tarefas.

O modelo de fila criado e modelado por Columbo et al. (1994), é chamada de estática porque os itens transferidos para ela não têm deslocamento. O modelo produz a posição na fila para o próximo item a ser carregado nela, que é a posição da fila do próximo item a ser descarregado. Ou seja, a cada item carregado cria-se uma nova posição, a qual será descarregada sucessivamente, respeitando a ordem de chegada FIFO.

CONSIDERAÇÕES FINAIS

O sucesso das empresas em seus processos de negócios depende da execução correta e eficiente das soluções de integração. Para analisar e simular o comportamento do motor de execução baseado em tarefas frente a cenários críticos de funcionamento, é necessário verificar como acontece a execução da solução e o recolhimento de dados a partir da execução. A estas atividades estão relacionados custos, riscos e tempo de desenvolvimento. Sendo que a proposta deste artigo é um passo importante para a modelagem do motor de execução porém não é integra e conclusa.

Com relação a proposta deste artigo têm se muitas variáveis a serem consideradas, como processo de chegada, número de soluções na fila de trabalho, capacidade da fila, tempo de espera para processar, tempo ocioso dos *threads* e o número de *threads*.

REFERÊNCIAS

AALST, W. M. V. D. Business process simulation survival guide. In: Handbook on Business Process Management 1. [S.l.]: Springer, p. 337–370, 2015.

AALST, W. Van der; STAHL, C. Modeling business processes: a petri net-oriented approach. [S.l.]: MIT press, 2011.

ALEXANDER, C. A pattern language: towns, buildings, construction. [S.l.]: Oxford university press, 1977.

Evento: XXV Jornada de Pesquisa

ODS: 9 - Indústria, Inovação e Infra-estrutura

ALKHANAK, E. N. et al. Cost optimization approaches for scientific workflow scheduling in cloud and grid computing: A review, classifications, and open issues. *Journal of Systems and Software*, Elsevier, v. 113, p. 1–26, 2016.

BLYTHE, J. et al. Task scheduling strategies for workflow-based applications in grids. In: IEEE. CCGrid 2005. IEEE International Symposium on Cluster Computing and the Grid, 2005. [S.l.], v. 2, p. 759–767, 2005.

CHWIF, L. Redução de modelos de simulação de eventos discretos na sua concepção: uma abordagem causal. Tese (Doutorado) — Tese, Escola Politécnica da Universidade de São Paulo, 1999.

COLUMBO, A.; MARTINEZ, J.; KUCHEN, B. Formal specification and validation of a static fifo queue model using coloured petri nets. In: IEEE. Proceedings of 1994 IEEE International Symposium on Industrial Electronics (ISIE'94). [S.l.], p. 231–235, 1994.

D.L. Freire, R.Z. Frantz & F. Roos-Frantz. Ranking enterprise application integration platforms from a performance perspective: An experience report. *Software: Practice and Experience*, 49(5), 921–941, 2019.

DESEL, J.; ESPARZA, J. Free choice Petri nets. [S.l.]: Cambridge university press, v. 40, 2005.

DOSSOT, D.; D'EMIC, J.; ROMERO, V. Mule in action. [S.l.]: Manning Publications Co., 2014.

FISHER, M. et al. Spring integration in action. [S.l.]: Manning Publications Co., 2012.

FOGLIATTI, M. C.; MATTOS, N. M. C. Teoria de filas. Rio de Janeiro: Interciência, p.1–290, 2007.

FRANTZ, R. Z. Enterprise application integration: an easy-to-maintain model-driven engineering approach. Tese (Doutorado) — Universidad de Sevilla, 2012.

FREIRE, D. L.; FRANTZ, R. Z.; ROOS-FRANTZ, F. Ranking enterprise application integration platforms from a performance perspective: An experience report. *Software: Practice and Experience*, Wiley Online Library, v. 49, n. 5, p. 921–941, 2019.

HILLIER, F. S.; LIEBERMAN, G. J. Introdução à pesquisa operacional. [S.l.]: McGraw Hill Brasil, 2013.

HOHPE, G.; WOOLF, B. Enterprise integration patterns: Designing, building, and deploying messaging solutions. [S.l.]: Addison-Wesley Professional, 2004.

IBSEN, C.; ANSTEY, J. Camel in action. [S.l.]: Manning Publications Co., 2018.

LAW, A. M.; KELTON, W. D. Simulation modeling and analysis, mcgraw-hill. New York 2nd

Evento: XXV Jornada de Pesquisa
ODS: 9 - Indústria, Inovação e Infra-estrutura

edition is also OK, v. 2, 2000.

MARSAN, M. A. Stochastic petri nets: an elementary introduction. In: SPRINGER. European workshop on applications and theory in Petri nets. [S.l.], 1988.

MCMMASTER, K., Sambasivam, S., Rague, B., Wolthuis, S. Java vs. Python Coverage of Introductory Programming Concepts: A Textbook Analysis. Information Systems Education Journal, 15(3) pp 4-13, 2017.

MURATA, T. Petri nets: Properties, analysis and applications. Proceedings of the IEEE, IEEE, v. 77, n. 4, p. 541–580, 1989.

PETERSON, J. L. Petri nets. ACM Computing Surveys (CSUR), ACM New York, NY, USA, v. 9, n. 3, p. 223–252, 1977.

PETRI, C. A. Kommunikation mit automaten. 1962.

SURHONE, L.; TIMPLEDON, M.; MARSEKEN, S. Petals Enterprise Service Bus (Esb). [S.l.]: Betascript Publishing, 2010.

WAINER, G. A. Discrete-event modeling and simulation: a practitioner's approach. [S.l.]: CRC press, 2017.

Parecer CEUA: 3.069.588