
MODELAGEM E SIMULAÇÃO DE UMA SOLUÇÃO DE INTEGRAÇÃO PARA IDENTIFICAÇÃO DE GARGA- LOS DE DESEMPENHO BASEADAS EM FORMALISMO MATEMÁTICO



UMA ABORDAGEM ORIENTADA À TEORIA DAS FILAS

ARLÉTE KELM WIESNER
UNIVERSIDADE REGIONAL DO NOROESTE DO
ESTADO DO RIO GRANDE DO SUL

DISSERTAÇÃO DE MESTRADO

ORIENTADOR:

DR. RAFAEL ZANCAN FRANTZ

COORIENTADORA:

DRA. FABRICIA CARNEIRO ROOS FRANTZ



Applied
Computing
Research Group

MARÇO, 2016

First published in February 2016 by
Applied Computing Research Group - GCA
Department of Exact Sciences and Engineering
Rua Lulu Ilgenfritz, 480 - São Geraldo
Ijuí, 98700-000, Brazil.

Copyright © MMXII Applied Computing Research Group
<http://www.gca.unijui.edu.br>
gca@unijui.edu.br

In keeping with the traditional purpose of furthering science, education and research, it is the policy of the publisher, whenever possible, to permit non-commercial use and redistribution of the information contained in the documents whose copyright they own. You however are *not allowed* to take money for the distribution or use of these results except for a nominal charge for photocopying, sending copies, or whichever means you use redistribute them. The results in this document have been tested carefully, but they are not guaranteed for any particular purpose. The publisher or the holder of the copyright do not offer any warranties or representations, nor do they accept any liabilities with respect to them.

Financiamento: O presente trabalho foi realizado com apoio da FAPERGS, Fundação de Amparo à Pesquisa do Estado do Rio Grande do Sul - Brasil.

Universidade Regional do Noroeste do Estado do Rio Grande do Sul

A Comissão Examinadora, abaixo assinada, _____ a dissertação intitulada “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”, elaborada por Arléte Kelm Wiesner, como requisito parcial para a obtenção do título de Mestre em Modelagem Matemática.

Dr. Rafael Zancan Frantz
UNIJUÍ
(Orientador)

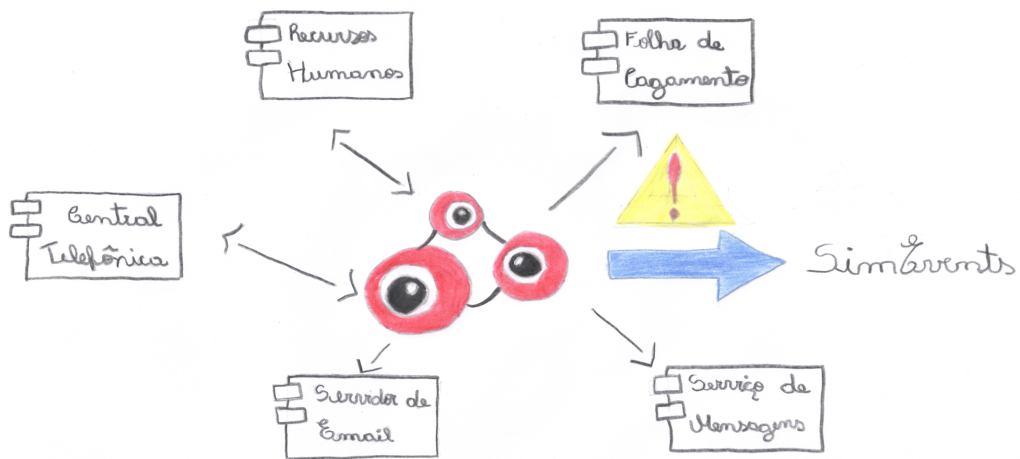
Dra. Fabricia Carneiro Roos Frantz
UNIJUÍ
(Co-orientadora)

Dr. Bruno Zatt
UFPel

Dr. Manuel Martín Pérez Reibold
UNIJUÍ

Dr. Sandro Sawicki
UNIJUÍ

Ijuí, ____ de _____ de _____.



Simulação de Modelos Conceituais de Integração de Aplicações por Karine, 11 anos de idade.

Ao meu marido, Ricardo Wiegner, pelo apoio incondicional e constante incentivo.

Conteúdo

Agradecimentos	ix
Resumo	xi
Abstract	xiii
1 Introdução	1
1.1 Contexto da Pesquisa	1
1.2 Motivação	6
1.3 Objetivos	8
1.3.1 Geral	8
1.3.2 Específicos	8
1.4 Metodologia	9
1.5 Resumo das Contribuições	10
1.6 Estrutura dessa Dissertação	11
2 Referencial Teórico	13
2.1 Tecnologia Guaraná	14
2.1.1 A Linguagem de Domínio Específico	14
2.1.2 Sintaxe Concreta	16
2.1.3 Runtime System	23
2.2 Sistemas e Simulação de Modelos	25
2.2.1 Passos na Formulação de um Modelo de Simulação	28
2.2.2 Elementos da Simulação	30
2.2.3 Classificação dos Modelos de Simulação	32
2.3 Distribuições de Probabilidade	34
2.3.1 Distribuição de Poisson (α)	34
2.3.2 Distribuição Exponencial (α)	37

2.4	Teoria das Filas	40
2.4.1	Definição	40
2.4.2	Elementos de um Sistema de Filas	41
2.4.3	Clientes e Tamanho da População	43
2.4.4	Processo de Chegada	43
2.4.5	Processo de Atendimento	44
2.4.6	Mecanismo de Atendimento e Número de Servidores ...	44
2.4.7	Fila e Disciplina da Fila	45
2.4.8	Notação de um Sistema de Filas	45
2.4.9	Medidas de Desempenho de um Sistema de Filas	47
2.4.10	Relações Básicas e Fórmulas de Little	50
2.4.11	Rede de Filas	51
2.5	Resumo do Capítulo	52
3	Trabalhos Relacionados	53
3.1	Teoria das Filas e Simulação	54
3.2	Resumo do Capítulo	60
4	Caso de Estudo	61
4.1	Ecosistema de Software	61
4.2	Modelo Conceitual da Solução de Integração	62
4.3	Resumo do Capítulo	64
5	Modelagem e Simulação	65
5.1	Equivalência do Guaraná com o Sistema de Filas	66
5.1.1	Variáveis da Solução de Integração	68
5.2	Modelo Matemático	70
5.2.1	Processo de Nascimento e Morte	71
5.2.2	Modelo M/M/1	73
5.3	Modelo de Simulação	80
5.3.1	Ferramenta SimEvents	81
5.3.2	Descrição do Modelo	87
5.4	Experimentação	92
5.4.1	Descrição dos Cenários da Simulação	93
5.4.2	Interpretação e Análise dos Resultados	95
5.5	Resumo do Capítulo	117

6	Verificação e Validação	119
6.1	Definição de Verificação e Validação	120
6.1.1	Técnicas de Verificação	121
6.1.2	Técnicas de Validação	123
6.2	Verificação do Modelo de Simulação	125
6.3	Validação do Modelo de Simulação	128
6.4	Resumo do Capítulo	129
7	Conclusões e Trabalhos Futuros	131
A	Resultados da verificação dos cenários 2 – 7	135
	Bibliografia	141

Índice de figuras

1.1	Ambiente de integração de aplicações (de Frantz et al. [15])	4
2.1	Mapa Conceitual da tecnologia Guaraná (de Frantz et al. [14])	15
2.2	Categorias das tarefas (de Frantz et al. [14])	17
2.3	Modelo de execução do Guaraná (de Frantz et al. [13])	23
2.4	Recorte para exemplificar o processo de execução	25
2.5	Abordagens para estudar um sistema (de Law e Kelton [26])	26
2.6	Etapas em um estudo envolvendo modelagem e simulação	29
2.7	Taxonomia de modelos (de Law e Kelton [26])	32
2.8	Função de densidade probabilística (de Hillier e Lieberman [18])	38
2.9	Elementos de uma fila (de Prado [32])	42
2.10	Localização das variáveis (baseado em Prado [32])	48
4.1	Modelo conceitual	63
5.1	Modelo de um sistema de serviço (a) Guaraná DSL (b)	67
5.2	Relação entre os elementos que compõe um sistema discreto	67
5.3	Localização das variáveis em uma solução de integração	70
5.4	Diagrama do processo de nascimento e morte	72
5.5	Diagrama de transição de estados do modelo M/M/1	73
5.6	Biblioteca do software SimEvents	82
5.7	Tarefa <i>Filter</i> – T0, T7 e T9 – modelada no SimEvents	88
5.8	Tarefa <i>Replicator</i> - T4 e T5 – modelada no SimEvents	88
5.9	Tarefa <i>Translator</i> - T2, T6, T8 e T10 – modelada no SimEvents	89
5.10	Tarefas T3 e T4 modelada no SimEvents	89
5.11	Modelo de simulação proposto	91
5.12	Número médio de mensagens nos <i>slots</i>	103
5.13	Tempo médio de permanência das mensagens nos <i>slots</i>	109

5.14	Taxa de utilização das tarefas	115
5.15	Taxa ociosa das tarefas	116

Índice de tabelas

2.1	Simbologia da sintaxe concreta (de Frantz et al. [14])	17
2.2	Tarefas da categoria <i>Routers</i> (de Frantz et al. [14])	19
2.3	Tarefas da categoria <i>Modifiers</i> (de Frantz et al. [14])	20
2.4	Tarefas da categoria <i>Transformers</i> (de Frantz et al. [14])	21
2.5	Tarefas da categoria <i>Stream Dealers</i> (de Frantz et al. [14])	21
2.6	Tarefas da categoria <i>Mappers</i> (de Frantz et al. [14])	22
2.7	Tarefas da categoria <i>Communicators</i> (de Frantz et al. [14])	22
5.1	Fórmulas do modelo M/M/1	80
5.2	Blocos da biblioteca <i>Generators</i>	83
5.3	Bloco da biblioteca <i>Queues e Servers</i>	84
5.4	Blocos da biblioteca <i>Routing</i>	85
5.5	Bloco da biblioteca <i>Entity Management e SimEvents Sinks</i>	86
5.6	Estatísticas da variável NF no cenário 1	97
5.7	Estatísticas da variável NF no cenário 2	97
5.8	Estatísticas da variável NF no cenário 3	98
5.9	Estatísticas da variável NF no cenário 4	99
5.10	Estatísticas da variável NF no cenário 5	99
5.11	Estatísticas da variável NF no cenário 6	100
5.12	Estatísticas da variável NF no cenário 7	101
5.13	Estatísticas da variável TF no cenário 1	104
5.14	Estatísticas da variável TF no cenário 2	105
5.15	Estatísticas da variável TF no cenário 3	106
5.16	Estatísticas da variável TF no cenário 4	106
5.17	Estatísticas da variável TF no cenário 5	107
5.18	Estatísticas da variável TF no cenário 6	107
5.19	Estatísticas da variável TF no cenário 7	108
5.20	Estatísticas da variável taxa de utilização no cenário 1	111

5.21	Estatísticas da variável taxa de utilização no cenário 2	111
5.22	Estatísticas da variável taxa de utilização no cenário 3	112
5.23	Estatísticas da variável taxa de utilização no cenário 4	112
5.24	Estatísticas da variável taxa de utilização no cenário 5	113
5.25	Estatísticas da variável taxa de utilização no cenário 6	113
5.26	Estatísticas da variável taxa de utilização no cenário 7	114

Agradecimentos

As pessoas felizes lembram o passado com gratidão, alegram-se com o presente e encaram o futuro sem medo.

Epicuro de Samos, filósofo grego (342 a.C. - 271 a.C.)



o Grande Arquiteto do Universo, por estar sempre iluminando meu caminho e guiando minhas escolhas. A minha família, em especial ao meu marido Ricardo Wiesner. Sou muito grata pela compreensão, mas principalmente pelo apoio e incentivo durante o período em que estive envolvida nesta pesquisa. Obrigada por acreditar em mim, mesmo quando eu não acreditava.

Ao meu orientador professor Dr. Rafael Zancan Frantz, pela confiança referente ao presente trabalho. Agradeço as cobranças, exigências, dinamismo, confiança e por acreditar em meu potencial.

A minha co-orientadora professora Dra. Fabrícia Carneiro Roos Frantz, seus conhecimentos e incentivos foram fundamentais para a concretização deste projeto.

Ao professor Dr. Sandro Sawicki que sempre me ajudou muito para que o desafio fosse vencido com êxito.

Ao grupo de professores do Mestrado em Modelagem Matemática, que contribuíram na construção do meu conhecimento acadêmico, dando-me ensinamentos que levarei por toda a minha existência.

Aos professores coordenadores do Mestrado em Modelagem Matemática, Dra. Airam Sausen e Dr. Paulo Sausen.

Aos meus colegas de mestrado, pelo convívio, amizade e estudo.

A FAPERGS pelo aporte financeiro que recebi, tornando possível a realização deste sonho, tão significativo para mim.

Resumo

O início de todas as coisas é pequeno.

Marcus T. Cícero, Filósofo Romano (106 a.C. - 43 a.C.)

Frequentemente as empresas adquirem ou desenvolvem aplicações para apoiar a tomada de decisões e aperfeiçoar seus processos de negócio. Estas aplicações compõem o seu ecossistema de software, que geralmente é heterogêneo, e foram desenvolvidas sem levar em conta sua possível integração, dificultando assim a sua reutilização. A área de integração de aplicações empresariais proporciona metodologias, técnicas e ferramentas para as empresas desenvolverem soluções de integração. A tecnologia Guaraná possibilita aos engenheiros de software projetar soluções de integração com um alto nível de abstração, utilizando uma abordagem centrada em modelos. A análise de soluções de integração, para prever seu comportamento e encontrar possíveis gargalos de desempenho, é uma importante atividade que contribui para aumentar a qualidade das soluções desenvolvidas. A abordagem, geralmente adotada pelos engenheiros de software, consiste na construção e execução da solução de integração. No entanto, o desenvolvimento da solução envolve custos e riscos inerentes, que geralmente são elevados. Soluções de integração podem ser classificadas como sistemas estocásticos, dinâmicos e discretos. Nesse contexto, esta dissertação propõe uma nova abordagem para identificar possíveis gargalos de desempenho, utilizando o modelo conceitual da solução de integração, por meio do desenvolvimento de um modelo formal de simulação, baseado na Teoria das Filas. Utiliza-se, como estudo de caso, uma solução de integração projetada na tecnologia Guaraná, para o gerenciamento de telefonemas pessoais realizados por funcionários na Unijuí. Os resultados experimentais do modelo de simulação, permitem avaliar o comportamento da solução de integração em diferentes taxas de trabalho e identificar possíveis gargalos de desempenho na sua estrutura. O modelo de simulação proposto foi verificado e validado por meio de técnicas formais descritas na literatura.

Palavras-chave: Integração de aplicações empresariais. Simulação. Teoria das Filas. Gargalos de desempenho.

Abstract

The beginnings of all things are small.

Marcus T. Cicero, Roman philosopher (106 BC - 43 BC)

Over and over again companies get and develop applications to support decision making and improve their business processes. These applications make up their own software ecosystem, which is generally heterogeneous, and developed without taking into account their possible integration, thus making it difficult to reprocess. The enterprise application integration area provides methodologies, techniques and tools for companies to develop integration solutions. The Guaraná technology enables software engineers to design integration solutions with a high level of abstraction, using a model-centric approach. The analysis of integration solutions, to predict their behavior and find possible performance bottlenecks is an important activity that contributes to increasing the quality of the solutions developed. The approach generally adopted by software engineers, consists of the construction and implementation of the integration solution. However, solution development involves inherent risks and costs that are generally higher. Integration solutions can be classified as stochastic, dynamic and discrete systems. In this context, this work proposes a new approach to identify potential performance bottlenecks by using the conceptual model of integration solution through the development of a formal simulation model based on the Theory of Queues. It is used as a case study, an integration solution designed in Guaraná technology for managing personal phone calls made by employees at Unijuí. The experimental results of the simulation model to assess the behavior of the integration solution in different labor rates and identify potential performance holdups in their structure. The proposed simulation model was verified and validated through formal techniques described in the literature.

Keywords: Enterprise application integration. Simulation. Theory of Queues. Performance holdups.

Capítulo 1

Introdução

Tudo é loucura ou sonho no começo.
Nada do que o homem fez no mundo teve início
de outra maneira - mas já tantos sonhos se realizaram
que não temos o direito de duvidar de nenhum.

Monteiro Lobato, Escritor brasileiro (1882-1948)

1.1 Contexto da Pesquisa

Atualmente, a utilização de sistemas de informação nas empresas tornou-se imprescindível para a prática de negócios. Segundo Laudon e Laudon [25], esta ferramenta possibilita altos níveis de eficiência e produtividade, como também criar novos produtos, serviços e modelos de negócios. Possibilita, ainda, um relacionamento mais estreito com clientes e fornecedores, uma melhor tomada de decisões, vantagem competitiva e sobrevivência no mundo dos negócios.

Um sistema de informação pode ser definido como um conjunto no qual os componentes estão conectados visando armazenar, tratar e fornecer informações, com o objetivo de dar suporte aos processos (planejamento, controle, coordenação, análise) de uma empresa de negócio. Em um sistema de informação o elemento principal é a informação, obtida através de um conjunto de dados organizados.

Sistemas de informação podem ter diferentes níveis de automatização [41]. Para tratar a informação de forma automatizada, interconectando os processos da empresa, é necessário que estes sistemas sejam desenvolvidos para executar em sistemas computacionais. Portanto, esta dissertação considera que um sistema de informação é um software desenvolvido para apoiar as ações dos processos de negócio das organizações.

O software tornou-se uma tecnologia indispensável no nosso cotidiano, pois tudo está direta ou indiretamente ligado aos sistemas informatizados. Os softwares são utilizados em diversas áreas, cita-se, engenharias e gestão empresarial, com a finalidade de dar suporte a execução das tarefas do dia-a-dia.

O termo software costuma ser confundido por algumas pessoas com programas de computador. Contudo, a definição de software é mais ampla, pois engloba também toda a documentação associada e os dados de configuração, indispensáveis para que estes programas funcionem corretamente.

O desenvolvimento do software passa por várias fases e, para que aconteça dentro do prazo e com qualidade, deve seguir um processo da área de engenharia de software. Essa ocupa-se de todos os aspectos do desenvolvimento do software e, tipicamente, seus processos incluem as fases de especificação, projeto, implementação, teste e evolução.

A fase de especificação é fundamental para o desenvolvimento do software, sendo definido o que se espera como resultado do sistema. Para isso, são utilizadas técnicas de modelagem e especificação de sistemas, as quais possibilitam a detecção de erros na fase inicial do projeto. A fase de projeto envolve o desenvolvimento de vários modelos do sistema, em diferentes níveis de abstração. É nessa fase que são criados os modelos conceituais do software, sendo realizada uma descrição detalhada da estrutura de software a ser implementado, dos dados que são parte do sistema e das interfaces entre os componentes do sistema. A implementação é a fase do desenvolvimento do software, na qual tudo o que foi definido na fase de especificação e no projeto se torna um sistema executável. Os testes são realizados com o software, de maneira a revelar erros e validar o pedido do cliente e a fase de evolução atende as necessidades mutáveis do cliente [39].

De acordo com Pressman [33], atualmente existem sete grandes categorias de software: software de sistema, software de aplicação, software científico/de engenharia, software embutido, software para linhas de produtos, aplicações para a Web, software de inteligência artificial. Considerando o contexto da pesquisa, será abordado apenas a categoria software de aplicação,

que é definida como programas que solucionam uma necessidade específica de negócio. As aplicações processam dados comerciais, com a finalidade de facilitar tomadas de decisão administrativas. São utilizadas para controlar e auxiliar nos processos de negócio em tempo real, tais como o processamento de transações em pontos de venda, controle de estoque, entre outros.

As aplicações são utilizadas nas empresas para agilizar, qualificar e dar suporte aos seus processos de negócios. Segundo Rezende [34], as aplicações empresariais reestruturam os dados para facilitar as operações comerciais e as tomadas de decisões administrativas.

No cenário atual de negócios, as empresas adquirem ou desenvolvem aplicações para apoiar a tomada de decisões, a coordenação, o controle e o aperfeiçoamento de seus processos. Essas aplicações compõem o ecossistema de software da empresa, que geralmente é heterogêneo. Um ecossistema de software, segundo Messerschmitt e Szyperski [29], envolve aplicações distintas, adquiridas de fornecedores diferentes ou desenvolvidas na própria empresa.

Normalmente, as aplicações que compõem os ecossistemas de software, foram projetadas sem levar em conta a possibilidade de serem reutilizadas. A ausência de integração normalmente acaba gerando redundância de dados, como também a ausência. Por exemplo, informações que estão sendo tratadas em uma aplicação e são necessárias em outra. Assim, a área para Integração de Aplicações Empresariais – do inglês *Enterprise Application Integration* (EAI), está relacionado à elaboração de soluções que resolvam estes problemas, sendo definido por Linthicum [27] como o compartilhamento irrestrito de dados e processos de negócios entre todas as aplicações conectadas e fontes de dados na empresa.

A área de estudos conhecido como EAI tem tido um papel muito importante, à medida que proporciona metodologias, técnicas e ferramentas para que as empresas consigam desenvolver soluções de integração. Segundo Hohpe e Woolf [19], EAI é a tarefa de fazer aplicações distintas trabalharem juntas, para produzir um conjunto único de funcionalidades.

O objetivo de uma solução de integração é manter em sincronia os dados e as funcionalidades das aplicações ou desenvolver novas funcionalidades a partir daquelas existentes, de tal forma que as aplicações não sejam alteradas pela solução [15]. A Figura §1.1 representa uma solução de integração que sincroniza as aplicações do ecossistema de software. A solução visa reutilizar as aplicações existentes no ecossistema de software da empresa e dar suporte às novas demandas que surgem com a evolução dos processos de negócio.

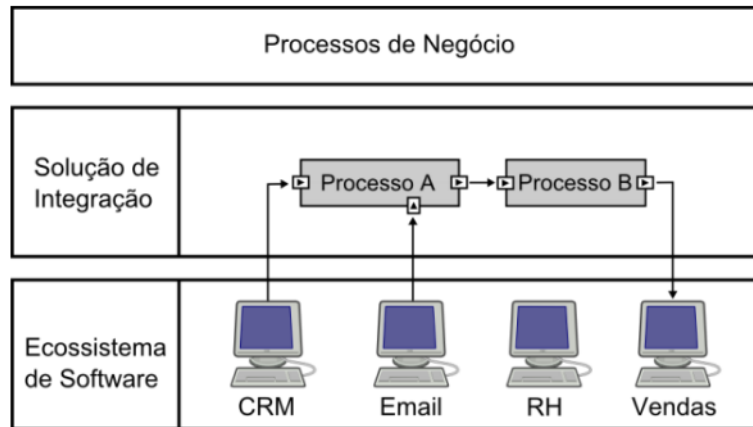


Figura 1.1: Ambiente de integração de aplicações (de Frantz et al. [15])

De acordo com Hohpe e Woolf [19], a EAI pode ser realizada por meio de dois estilos de integração: compartilhamento de dados e compartilhamento de funcionalidades. A integração através do compartilhamento de dados é caracterizada pelo compartilhamento de informações entre as aplicações do ecossistema e pode acontecer por meio da transferência de arquivos, ou por meio da utilização de um Sistema de Gerenciamento de Banco de Dados (SGBD).

O compartilhamento de dados por meio de arquivos ocorre quando uma aplicação escreve em arquivos informações que outras aplicações precisam acessar para fazer a leitura das informações nele contidas. A grande vantagem deste estilo de integração é que os desenvolvedores da solução de integração não precisam conhecer ou acessar o código fonte das aplicações, sendo necessário apenas o conhecimento do conteúdo e o formato do arquivo para a utilização deste arquivo na troca e leitura de informações com outras aplicações. Desta forma, tem-se aplicações independentes entre si, sendo possível a aplicação realizar mudanças no seu código fonte, sem interferir em outras aplicações. Porém, um fator negativo é o grande esforço computacional, por isso é executado periodicamente e quanto maior for esse período, maior é a possibilidade da solução estar trabalhando com dados obsoletos [19].

A característica do compartilhamento de dados utilizando SGBD é a utilização de um banco de dados, que pode ser acessado pelas aplicações para realizar os processos de escrita, leitura e alterações. A vantagem deste estilo é

a comunicação assíncrona entre as aplicações e dados consistentes. A transferência assíncrona significa que é possível enviar mensagens mesmo quando as aplicações de destino não estejam sendo executadas. Hohpe e Woolf [19] citam, como pontos negativos, a perda de desempenho (provocada pelo acesso simultâneo ao banco de dados por vários processos) e a dificuldade de criar e manter um esquema de dados único, considerando a evolução dos processos de negócios de uma empresa que demanda alterações e novas funcionalidades em suas aplicações.

A integração de aplicações, por meio do compartilhamento de funcionalidades, é utilizada quando as funcionalidades de uma aplicação precisam ser acessadas pelas outras aplicações. Esse estilo de integração pode ser realizado através da técnica de Chamada de Procedimento Remoto ou utilizando um sistema de Filas de Mensagens [19].

Uma chamada de procedimento remoto possibilita que aplicações acessem e executem funcionalidades disponibilizadas por uma outra aplicação, através de uma API. Assim, quando uma aplicação precisa informações que estão disponíveis em outra aplicação, é possível realizar uma solicitação diretamente entre as aplicações e receber uma resposta, de forma semelhante a chamada de procedimento local, que consiste em uma chamada de procedimento dentro da própria aplicação. A desvantagem desta técnica é o alto acoplamento porque as aplicações dependem uma da outra para realizar as suas operações sendo necessário que ambas estejam em funcionamento no mesmo instante de tempo [19].

O compartilhamento de funcionalidades, baseado em um sistema de Filas de Mensagens, acontece quando as aplicações realizam a transferência de informações, enviando por meio da troca de mensagens entre elas. Essa técnica possibilita o compartilhamento e a transferência de dados, porque a transferência de mensagens é feita através de um canal assíncrono. Nesse caso, a mensagem aguarda no canal até o momento que a aplicação de destino seja executada.

A integração de aplicações não é uma tarefa trivial e para reduzir os riscos de erros, é aconselhável usar padrões de integração. Os padrões consistem em soluções que foram implementadas e aprovadas pelos engenheiros de software. Permitem que o desenvolvedor escolha as estratégias mais adequadas para a solução do problema de integração e podem ser utilizados em problemas recorrentes, reduzindo dessa forma, o árduo trabalho para realizar uma nova implementação.

A presença de ecossistemas de softwares heterogêneos nas empresas constituiu uma demanda por soluções de integração. Atualmente, existem várias

tecnologias baseadas em mensagens e nos padrões de integração que oferecem suporte a concepção e implementação de soluções de integração. Camel [20], Spring Integration [9], Mule [7] e Guaraná [14], são algumas das tecnologias com suporte para projeto e desenvolvimento de soluções de integração. Possuem uma linguagem específica de domínio – do inglês *Domain-Specific Language* (DSL). Além da linguagem própria, possuem uma API para implementar as soluções.

Para a realização da pesquisa desta dissertação, foi utilizada a tecnologia Guaraná, que possibilita projetar, implementar e executar soluções de integração. Além disso, a tecnologia foi criada por pesquisadores que atuam no Grupo de Pesquisa em Computação Aplicada (GCA), grupo pelo qual a pesquisa foi desenvolvida. Em função desta proximidade, que permitiu o acesso a documentação, destacou-se a tecnologia Guaraná com o objetivo de contribuir para o aperfeiçoamento e evolução, no contexto de identificação de possíveis gargalos de desempenho.

Nesse sentido, propõe-se no presente trabalho analisar o comportamento e identificar possíveis gargalos de desempenho em soluções de EAI modeladas na tecnologia Guaraná, antes de sua implementação e execução, por meio da criação/desenvolvimento de um modelo matemático de simulação, utilizando como base a Teoria das Filas.

1.2 Motivação

O sucesso das empresas em seus processos de negócios, atualmente, depende da execução correta e eficiente da solução de integração. Portanto, a análise de soluções de integração de aplicações empresariais, para prever seu comportamento com diferentes cargas de trabalho e identificar possíveis gargalos de desempenho é considerada uma importante atividade para melhorar a qualidade das soluções entregues.

Uma solução de integração consiste em uma nova aplicação. Sendo assim, seu desenvolvimento deve seguir um processo da área de engenharia de software que, tipicamente, incluem as fases de especificação, projeto, implementação, teste e evolução.

Normalmente, a abordagem adotada pelos engenheiros de software para análise do comportamento frente a cenários críticos de funcionamento e recolher dados consiste na construção e execução da solução de integração. Nesse sentido, a abordagem que permite analisar o comportamento e identificar possíveis gargalos de desempenho, ainda na fase de projeto, a

partir do modelo conceitual, poderá reduzir custos, riscos e tempo de desenvolvimento [36].

A simulação é um método que utiliza um modelo matemático para possibilitar o estudo e a análise do comportamento e desempenho do sistema, sem que seja necessário realizar alterações no sistema real podendo, assim, prever um comportamento futuro. Para realizar a simulação da solução de integração é necessário, ter conhecimento sobre a representação do modelo. Uma solução de integração pode ser caracterizada como um sistema, cujo modelo é classificado como estocástico, dinâmico e discreto [36].

Modelos estocásticos utilizam a probabilidade para modelar sistemas reais, nos quais não é possível determinar o seu comportamento, sendo considerados bons modelos para representar sistemas reais. Soluções de integração podem ser caracterizadas como sistemas estocásticos porque a taxa de chegada dos dados de entrada (mensagens) não pode ser prevista, sendo essencialmente aleatória e dependente das aplicações integradas. Modelos dinâmicos representam sistemas que alteram seu estado ao longo do tempo. A alteração de estado em uma solução de integração, pode ser entendida como o número de mensagens que estão sendo processadas em determinado tempo de execução e as operações executadas nas mensagens. Modelos discretos são orientados a eventos e usados para modelar sistemas que mudam seu estado em momentos específicos no tempo, a partir da ocorrência de eventos. Soluções de integração podem ser caracterizadas como sistemas discretos, porque quando ocorre um evento todos os componentes envolvidos na solução consomem um tempo específico de execução. Assim, a ocorrência de um evento altera o estado da solução. Nesse sentido, a presente dissertação é motivada pela possibilidade de analisar o comportamento e identificar possíveis gargalos de desempenho ainda na fase de projeto, a partir do modelo conceitual transcrito em um modelo de simulação equivalente.

Considerando o contexto apresentado, formula-se a seguinte hipótese:

Soluções de integração de aplicações empresariais podem ser classificadas como sistemas estocásticos, dinâmicos e discretos. Portanto, é possível utilizar técnicas de simulação de eventos discretos e modelos matemáticos computacionais para analisar o comportamento e identificar possíveis gargalos de desempenho que podem surgir nas soluções de integração quando submetidas a cenários críticos de funcionamento, tendo como base seus modelos conceituais. A simulação pode ser eficiente para detectar onde ocorrem os gargalos e possibilitar uma tomada de decisão para melhorar a solução.

1.3 Objetivos

A análise do comportamento e a identificação de possíveis gargalos de desempenho de soluções de integração, ainda na fase de projeto, poderá reduzir custos computacionais e riscos inerentes. Nesse sentido, esta dissertação tem os seguintes objetivos:

1.3.1 Geral

Desenvolver um modelo de simulação com base no formalismo matemático da Teoria das Filas para analisar o comportamento e identificar possíveis gargalos de desempenho em uma solução de integração, ainda na fase de projeto.

1.3.2 Específicos

- Destacar a importância dos sistemas de informação nas organizações empresariais;
- Contextualizar que os sistemas de informação nas organizações empresariais são formados por distintas aplicações que compõe o ecossistema de software, afim de apresentar os ambientes onde a integração é necessária;
- Destacar a importância das tecnologias voltadas a integração de aplicações nas organizações empresariais;
- Apresentar as especificidades e os elementos da linguagem de domínio específico da tecnologia Guaraná;
- Apresentar os conceitos de sistema, modelo e classificação de modelos, afim de aplicá-los na caracterização de uma solução de integração como um sistema de eventos discretos;
- Estudar, a partir da literatura, os elementos, as características e as medidas de desempenho de um sistema de filas;
- Demonstrar a equivalência entre os elementos da solução de integração projetada na tecnologia Guaraná com os elementos de um sistema de filas;

- Desenvolver um modelo de simulação baseado na Teoria das Filas, equivalente ao modelo conceitual projetado na tecnologia Guaraná;
- Demonstrar a equivalência entre o modelo conceitual do Guaraná e o modelo de simulação desenvolvido no SimEvents;
- Realizar uma análise de tempos e quantidade de mensagens nos *slots* por meio do modelo de simulação, a fim de identificar possíveis gargalos de desempenho;
- Prever o comportamento de soluções de integração de aplicações em cenários críticos de funcionamento;
- Apresentar técnicas de verificação e validação, presentes na literatura, para que sejam caracterizadas em relação ao contexto;
- Verificar e validar o modelo matemático de simulação por meio de técnicas de verificação e validação formais existentes na literatura.

1.4 Metodologia

A metodologia empregada nesta pesquisa baseia-se no *framework* de referência fornecido pelo *Unified Process* (UP). A escolha da mesma foi motivada pela experiência que o Grupo de Pesquisa em Computação Aplicada (GCA) possui em utilizá-la em pesquisas. Além do que, seu ciclo de vida iterativo e incremental é apropriado ao desenvolvimento de projetos que visam alcançar um grande dinamismo, incorporando, a todo o momento, o *feedback* de outros grupos de trabalho e/ou projetos relacionados, mas mantendo sob controle os riscos que podem ocorrer no dia-a-dia.

Considerando-se este *framework*, a pesquisa foi dividida nas seguintes fases:

INÍCIO: Trata-se de compreender o que será feito, identificar os principais pontos do projeto, compreender o problema de pesquisa desta dissertação e decidir sobre o processo a ser seguido. O trabalho foi realizado através de reuniões do tipo *brainstorming*, com os professores e colegas mestrandos do GCA, presencialmente, para que cada pesquisador pudesse contribuir com suas ideias e conhecimentos, promovendo um compartilhamento de saberes e não apenas para que esta dissertação tivesse êxito.

ELABORAÇÃO: Parte dos objetivos dessa fase consiste em identificar e descrever os principais blocos de trabalho e suas atividades. Este

trabalho também foi desenvolvido por meio das reuniões do tipo “*modelstorming*” e os resultados foram a elaboração do plano de atividades e o cronograma. Durante o desenvolvimento desta dissertação, foi necessário refinar os blocos de trabalhos, priorizar algumas atividades e incrementar novos pontos não considerados no início do trabalho.

CONSTRUÇÃO: Essa fase dividiu as atividades dos blocos de trabalho em pequenas etapas, que foram abordadas de forma iterativa, buscando ciclos rápidos de análise detalhada do caso de estudo, bibliografia, proposta de solução do problema e verificação, para que fosse possível obter, rapidamente, *feedbacks* úteis. A forma de abordar, com êxito, essa fase foi mediante uma reunião inicial do tipo “*task planning game*”, reuniões periódicas para apresentação de resultados e seminários de pesquisa, nas quais eram apresentados os progressos alcançados, sejam eles resultados ou conclusões sobre a bibliografia estudada. As reuniões e seminários foram realizados sempre de forma presencial. Durante essa fase, participou-se de eventos científicos e tecnológicos relacionados com os temas da dissertação, com o objetivo de apresentar os resultados e obter *feedback*.

TRANSIÇÃO: O principal objetivo dessa fase é a transferência de resultados obtidos para o grupo de pesquisa e a apresentação desta dissertação. Os resultados obtidos serão utilizados para melhorar soluções de integração projetadas na tecnologia Guaraná, como também publicados em conferências para consolidar e ampliar a rede de colaborações acadêmicas nacionais e internacionais.

1.5 Resumo das Contribuições

Esta dissertação é parte integrante do projeto de pesquisa sobre a simulação para predição do comportamento e desempenho de soluções de integração de aplicações empresariais projetadas na tecnologia Guaraná e busca contribuir para melhorar a qualidade das soluções desenvolvidas pelos engenheiros de software. Trata-se de um passo inicial da simulação no contexto de integração de aplicações empresariais. Nesse sentido, as principais contribuições deste trabalho são listadas a seguir:

- Análise do comportamento e identificação de possíveis gargalos de desempenho em soluções de integração de aplicações empresariais a partir da simulação de seus modelos conceituais, utilizando o formalismo matemático da Teoria das Filas.

- Identificação de uma técnica e ferramenta de simulação para analisar o comportamento e identificar possíveis gargalos de desempenho em soluções de integração de aplicações empresariais.
- Desenvolvimento de um modelo de simulação formal, equivalente ao modelo conceitual projetado na tecnologia Guaraná.
- Verificação do modelo de simulação utilizando técnicas de verificação formal existentes na literatura.

O desenvolvimento desta dissertação tem outras duas contribuições importantes:

- Durante a revisão bibliográfica, buscou-se conhecer o contexto de integração de aplicações empresariais e identificou-se uma técnica matemática que apresenta elementos equivalentes aos de uma solução de integração. Alguns resultados desta pesquisa foram publicados no III Seminário de Formação Científica e Tecnológica (SFCT), promovido pelo Grupo de Pesquisa em Computação Aplicada no dia 15 de junho de 2015, Ijuí, RS, Brasil [43].
- O estudo detalhado sobre a tecnologia Guaraná possibilitou a extração de informações que foram utilizadas na caracterização de uma solução de integração como um sistema de eventos discretos. Os resultados foram publicados no Salão do Conhecimento, evento realizado pela Unijuí para divulgação da produção em pesquisa e extensão [44].

1.6 Estrutura dessa Dissertação

Essa dissertação está organizada conforme a estrutura apresentada a seguir. O Capítulo §1 compreende essa introdução. O Capítulo §2 proporciona para o leitor uma revisão da literatura técnica e científica relacionadas à pesquisa desenvolvida nessa dissertação. Inicialmente aborda-se os conceitos relacionados com a tecnologia Guaraná, em seguida para a simulação de sistemas e por último a definição de Teoria das Filas, os elementos e as características de um sistema de filas.

No Capítulo §3, aborda-se os trabalhos relacionados, identificados ao longo dessa pesquisa. Contemplou-se os trabalhos que utilizaram a Teoria das Filas para analisar e estudar sistemas de eventos discretos.

No Capítulo §4, apresenta-se o caso de estudo que será objeto dessa pesquisa. Trata-se de uma solução de integração projetada na tecnologia Guaraná para automatizar o faturamento de ligações pessoais realizadas por funcionários usando os telefones da Central Telefônica de uma universidade.

No Capítulo §5 é apresentado o trabalho desenvolvido, para analisar uma solução de integração e encontrar possíveis gargalos de desempenho, por meio da simulação do modelo conceitual. O capítulo aborda a equivalência dos elementos do Guaraná DSL com os da Teoria das Filas, as fórmulas de um modelo de filas M/M/1, o modelo de simulação desenvolvido, os cenários e análise dos resultados.

No Capítulo §6 são apresentados os conceitos de verificação e validação e realiza-se a verificação do modelo de simulação desenvolvido. As técnicas de verificação abordadas procuram garantir que o modelo esteja ausente de erros.

No Capítulo §7 são apresentadas as conclusões deste estudo e as possibilidades de trabalhos futuros.

Capítulo 2

Referencial Teórico

É a teoria que decide o que podemos observar.

Albert Einstein, físico teórico alemão (1879-1955)

Para a predição da análise do comportamento e a identificação de possíveis gargalos de desempenho em soluções de integração de aplicações empresariais, é necessário conhecer os conceitos envolvidos nesta nova abordagem. Nesse sentido, este capítulo apresenta na Seção §2.1 o estudo realizado sobre a tecnologia Guaraná com uma descrição da linguagem de domínio específico e do funcionamento do motor de execução. Na Seção §2.2 é dada a fundamentação teórica para a análise de um sistema, por meio da simulação de modelos. A Seção §2.3 apresenta as distribuições teóricas de probabilidade que caracterizam o processo de chegada e atendimento de um sistema de filas. A Seção §2.4 apresenta os conceitos e elementos da Teoria das Filas, para avaliar a eficiência de um sistema por meio do estudo e análise das medidas de desempenho da fila. Essa Seção aborda, ainda, as redes de filas que representam sistemas de filas, onde os clientes recebem atendimento em uma rede de instalações de atendimento. A última Seção §2.5 apresenta o resumo do capítulo.

2.1 Tecnologia Guaraná

Uma solução de integração de aplicações empresariais precisa manter os dados e as informações do ecossistema de software da empresa em sincronia, além de permitir que novas funcionalidades sejam desenvolvidas a partir daquelas existentes, sem que as aplicações tenham que ser modificadas.

Nesse contexto, a tecnologia Guaraná é utilizada para projetar soluções de integração de aplicações empresariais, de forma que os engenheiros de software possam manter o foco na criação de modelos para a solução do problema, sem precisar se preocupar com detalhes técnicos de sua implementação. Dentre os principais recursos dessa tecnologia estão uma Linguagem de Domínio Específico (do inglês *Domain-Specific Language* – DSL) e um motor de execução (do inglês *runtime system*), que permite a implementação e execução da solução de integração. A versão da tecnologia Guaraná considerada nessa dissertação é a 1.2.

2.1.1 A Linguagem de Domínio Específico

De acordo com Fowler [11], não existe ainda uma definição exata para o termo Linguagem de Domínio Específico (DSL), mas atribui uma definição simples: é uma linguagem de programação com expressividade limitada com foco em um domínio específico. Uma DSL é utilizada para solucionar um determinado conjunto de problemas, através de uma linguagem própria em um domínio concreto.

A tecnologia Guaraná proporciona uma linguagem de domínio específico que permite projetar soluções de integração a um alto nível de abstração utilizando uma sintaxe concreta gráfica e conceitos de modelagem intuitivos. Esta linguagem de modelagem é baseada nos padrões de integração documentados por Gregor Hohpe e Bobby Woolf [19].

De acordo Frantz et al. [15], os modelos conceituais projetados na tecnologia Guaraná são independentes de plataforma, de modo que os engenheiros de software não dependem de conhecimentos específicos em tecnologias de integração de baixo nível para construir soluções de integração. Esta característica permite que os engenheiros centralizem seus esforços em projetar modelos que visam a solução do problema, reduzindo, assim, os custos envolvidos, à medida que o engenheiro não precisa aprender a usar as distintas e, geralmente complexas, tecnologias de implementação.

Os modelos conceituais são desenvolvidos na tecnologia Guaraná utilizando uma linguagem gráfica. A transformação dos modelos em código executável é obtida por meio da Engenharia Dirigida por Modelos [37]. Dessa forma, modelos projetados na tecnologia podem ser reutilizados para gerar, automaticamente, soluções de integração para serem executadas em diferentes tecnologias. O Guaraná fornece um conjunto de transformações por meio das quais o engenheiro de software traduz os modelos projetados em código Java, para serem executados [14].

Os conceitos necessários para projetar modelos de soluções de integração de aplicações empresariais na tecnologia Guaraná são apresentados na Figura 2.1. As funcionalidades e a estrutura de uma solução de integração são completamente definidas, utilizando blocos construtores: portas, processos, tarefas, *slots* e *links*. As tarefas são fornecidas em caixas de ferramentas e são classificadas de acordo com a sua semântica. A Seção 2.1.2 contempla uma descrição completa das principais categorias de tarefas ilustradas na Figura 2.1.

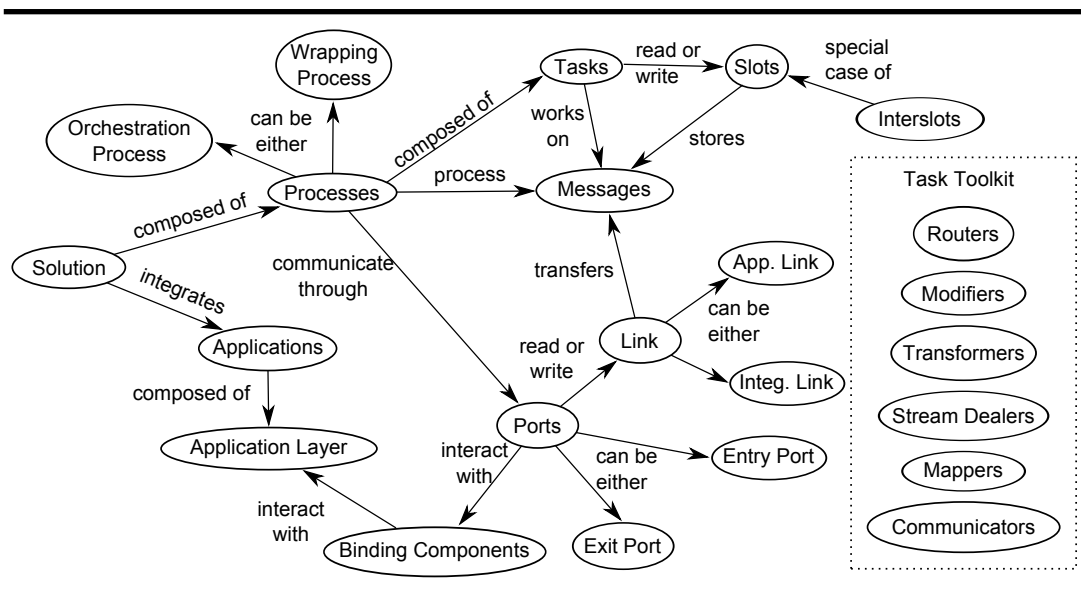


Figura 2.1: Mapa Conceitual da tecnologia Guaraná (de Frantz et al. [14])

Segundo Frantz et al. [14] o ponto inicial dos conceitos é a solução de integração que representa um conjunto de processos que cooperam para integrar várias aplicações. Os processos podem ser logicamente divididos em processos que contêm a lógica de comunicação com as aplicações integradas e

processos que contém a lógica de integração da solução. Os processos dependem de tarefas para executar a comunicação com as aplicações ou suas atividades de orquestração. De forma simplificada, um processo pode ser visto como um processador de mensagens.

Uma mensagem é uma abstração de uma parte da informação que é trocada e transformada, através da solução de integração, sendo composta por um cabeçalho e um corpo. O cabeçalho contém propriedades pré-definidas, como identificador de mensagens, identificador de correlação e prioridade da mensagem. A estrutura das mensagens depende completamente das soluções de integração em que estão envolvidas.

As tarefas executam processamento sobre mensagens e podem ter uma ou mais entradas, por meio das quais recebem as mensagens e uma ou mais saídas, através das quais as mensagens resultantes do processamento são despachadas. As tarefas podem ser executadas em paralelo, e se comunicam através de *slots*, que atuam como um *buffer* entre duas tarefas, ligando uma entrada de uma tarefa com a saída de uma outra tarefa, permitindo o processamento assíncrono de mensagens. Um *slot* pode seguir diferentes disciplinas para organizar as mensagens que serão executadas pela tarefa, como por exemplo *First-In, First-Out* ou baseadas em prioridades.

Os processos utilizam portas para comunicar-se uns com os outros ou com as aplicações envolvidas em uma solução de integração. Uma porta é responsável pela abstração dos detalhes necessários para interagir com um componente de comunicação (tipo especial de tarefa nomeado de *communicator task*). Esse por sua vez, abstrai os detalhes necessários para interagir com um aplicativo do ecossistema de software ou com um processo.

Na versão da linguagem, utilizada nessa dissertação, existem dois tipos de portas: porta de entrada, porta de saída. As funções das portas em uma solução são: a primeira permite que um processo receba informações de um aplicativo. A segunda permite que um processo envie informações para um aplicativo.

As portas geralmente, transformam as mensagens para posterior transferência. Isso implica que sejam compostas por tarefas. Portanto, uma porta também precisa de *slots*. Já a comunicação de uma tarefa da porta com uma tarefa do processo ocorre por meio de um *slot* especial conhecido como *interslot*.

2.1.2 Sintaxe Concreta

A Tabela §2.1 apresenta a sintaxe concreta utilizada para representar os conceitos da tecnologia Guaraná. Essas representações gráficas expressam

todos os conceitos necessários para projetar soluções de integração: *Application*, *Process*, *EntryPort*, *ExitPort*, *IntegrationLink*, *ApplicationLink*, *Slot* e *Task*.

Notação	Conceito	Notação	Conceito
	<i>Application</i>		<i>IntegrationLink</i>
	<i>Process</i>		<i>ApplicationLink</i>
	<i>EntryPort</i>		<i>Slot</i>
	<i>ExitPort</i>		<i>Task</i>

Tabela 2.1: Simbologia da sintaxe concreta (de Frantz et al. [14])

A representação gráfica das tarefas (*Task*) na Tabela §2.1 é genérica porque as tarefas são fornecidas em caixas de ferramentas especiais, conhecidos como *toolkit*. Portanto, elas não fazem parte do núcleo da linguagem. Nas laterais do ícone, pode-se observar saliências que representam as entradas e saídas da tarefa. Tais entradas e saídas são ligadas a *slots* permitindo, assim, a conexão e a comunicação das tarefas.

A representação relativa a *Process*, *EntryPort* e *ExitPort* é abreviada porque a estrutura interna é composta de outros conceitos. Segundo Frantz et al. [14], pode-se representar estes conceitos no modelo conceitual, omitindo ou apresentando a especificação da estrutura interna.

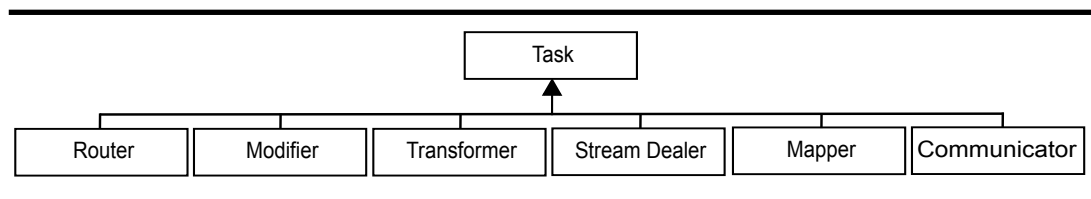


Figura 2.2: Categorias das tarefas (de Frantz et al. [14])

A Figura §2.2 fornece uma visão geral da caixa de ferramentas (*Task Toolkit*) que fornece um conjunto de várias tarefas, as quais contemplam a maioria dos padrões de integração documentados por Hohpe e Woolf [19]. Na tecnologia Guaraná, as tarefas são classificadas de acordo com a sua semântica:

- *Router*: Tarefas roteadoras não alteram o estado das mensagens que processam, mas encaminham as mensagens por meio de um processo. As tarefas desta categoria estão listadas na Tabela §2.2.
- *Modifier*: Tarefas modificadoras adicionam ou removem dados da mensagem, de forma que isso não resulte em mensagens com um esquema diferente. Um exemplo desta funcionalidade é a tarefa que adiciona dados na mensagem correlacionada. As tarefas desta categoria estão listadas na Tabela §2.3.
- *Transformer*: Tarefas transformadoras ajudam a traduzir uma ou mais mensagens em uma nova mensagem, com um esquema diferente. Essas tarefas são listadas na Tabela §2.4.
- *Stream Dealer*: Tarefas que trabalham com um fluxo de *bytes* e ajudam a comprimir/descomprimir, criptografar/descriptografar ou codificar/decodificar mensagens. Detalhes desta categoria estão na Tabela §2.5.
- *Mapper*: Tarefas mapeadoras mudam o formato das mensagens que processam, por exemplo, a partir de um fluxo de *bytes* em um documento XML. Detalhes destas tarefas estão na Tabela §2.6.
- *Communicator*: Tarefas do tipo comunicadoras são utilizadas nas portas, para interagir com componentes de comunicação, geralmente conhecidos também como adaptadores. As tarefas desta categoria estão descritas na Tabela §2.7.










Notação	Tarefa	Descrição
	<i>Correlator</i>	Analisa mensagens de entrada e produz, como saída, um conjunto de dados correlacionados.
	<i>Merger</i>	Direciona as mensagens que chegam de diferentes <i>slots</i> de entrada em um único <i>slot</i> de saída
	<i>Resequencer</i>	Reordena as mensagens em sequências, por meio de uma ordem pré-estabelecida.
	<i>Filter</i>	Filtra as mensagens indesejadas.
	<i>Idempotent Transfer</i>	Remove mensagens duplicadas.
	<i>Dispatcher</i>	Envia uma mensagem a exatamente um <i>slot</i> de saída.
	<i>Distributor</i>	Distribui mensagens para dois ou mais <i>slots</i> .
	<i>Replicator</i>	Realiza cópias da mensagem de acordo com o número de <i>slots</i> de saída.
	<i>Semantic Validator</i>	Valida a semântica de uma mensagem.

Tabela 2.2: Tarefas da categoria Routers (de Frantz et al. [14])









Notação	Tarefa	Descrição
	<i>Slimmer</i>	Remove o conteúdo de uma mensagem de acordo com uma política estática.
	<i>Context Based Slimmer</i>	Remove o conteúdo de uma mensagem base de acordo com uma política dinâmica, que é fornecida por uma mensagem contexto.
	<i>Content Enricher</i>	Adiciona conteúdos estáticos na mensagem.
	<i>Context Based Content Enricher</i>	Adiciona conteúdos dinâmicos de uma mensagem contexto em uma mensagem base.
	<i>Header Enricher</i>	Adiciona conteúdos estáticos para o cabeçalho de uma mensagem.
	<i>Context Based Header Enricher</i>	Adiciona conteúdo dinâmico de uma mensagem contexto para o cabeçalho de uma mensagem base.
	<i>Header Promoter</i>	Promove uma parte do corpo de uma mensagem para o cabeçalho.
	<i>Header Demoter</i>	Retira uma parte do cabeçalho de uma mensagem e coloca no corpo da mensagem.

Tabela 2.3: Tarefas da categoria *Modifiers* (de Frantz et al. [14])

Notação	Tarefa	Descrição
	<i>Translator</i>	Traduz o conteúdo da mensagem para outro formato.
	<i>Splitter</i>	Divide uma mensagem que contém elementos repetidos em várias mensagens.
	<i>Aggregator</i>	Constrói uma nova mensagem, a partir de várias mensagens produzidas anteriormente, através do <i>splitter</i> .
	<i>Chopper</i>	Quebra uma mensagem em duas ou mais mensagens.
	<i>Assembler</i>	Constrói uma nova mensagem, a partir de duas ou mais mensagens.
	<i>Cross Builder</i>	Constrói uma nova mensagem contendo o produto cartesiano de todas as mensagens de entrada.

Tabela 2.4: Tarefas da categoria Transformers (de Frantz et al. [14])

Notação	Tarefa	Descrição
	<i>Zipper</i>	Compacta uma mensagem.
	<i>Unzipper</i>	Descompacta uma mensagem.
	<i>Encrypter</i>	Criptografa uma mensagem.
	<i>Decrypter</i>	Descriptografa uma mensagem
	<i>Encoder</i>	Codifica uma mensagem.
	<i>Decoder</i>	Decodifica uma mensagem.

Tabela 2.5: Tarefas da categoria Stream Dealers (de Frantz et al. [14])



Notação	Tarefa	Descrição
	<i>XML2Stream</i>	Mapeia um fluxo de bytes para uma mensagem XML.
	<i>Stream2XML</i>	Mapeia uma mensagem XML para um fluxo de bytes.

Tabela 2.6: Tarefas da categoria Mappers (de Frantz et al. [14])



Notação	Tarefa	Descrição
	<i>In Communicator</i>	Usado em portas para ler mensagens.
	<i>Out Communicator</i>	Usado em portas para escrever mensagens.

Tabela 2.7: Tarefas da categoria Communicators (de Frantz et al. [14])

2.1.3 Runtime System

O motor de execução (em inglês *Runtime System*) da tecnologia Guaraná é representado na Figura 2.3. O motor de execução apresenta conceitos altamente técnicos. Assim, esta dissertação priorizou o estudo dos conceitos necessários para a realização desta pesquisa: *work unit*, *work queue* e *thread*. Uma discussão mais detalhada do motor de execução pode ser obtida em [13].

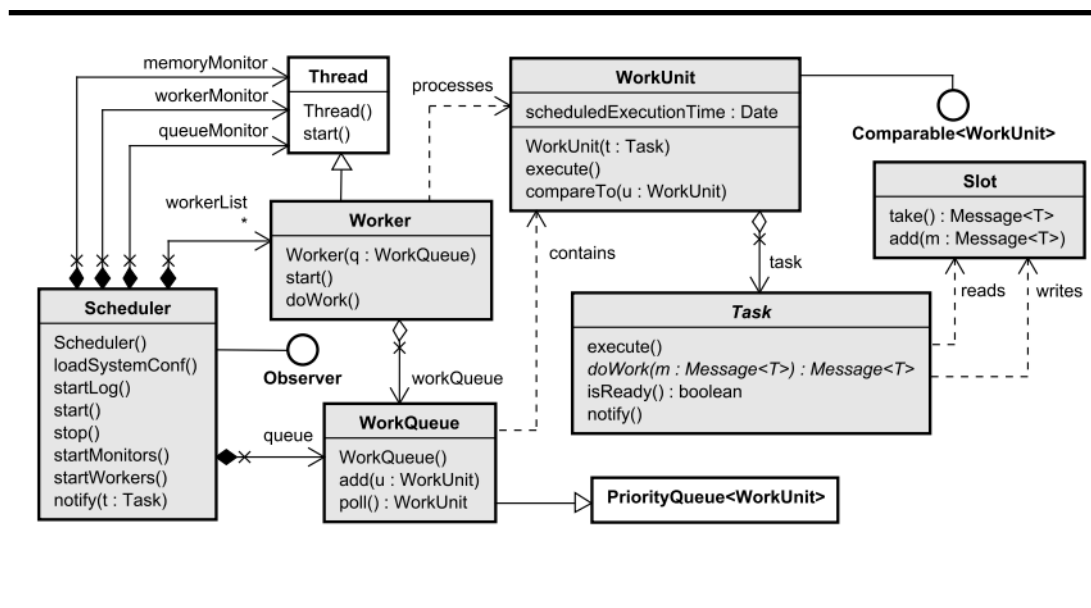


Figura 2.3: Modelo de execução do Guaraná (de Frantz et al. [13])

O Guaraná fornece uma implementação do motor de execução baseado em tarefas, o que o diferencia das outras tecnologias de integração, que utilizam um modelo de execução baseado em processos. No modelo baseado em tarefas, *threads* são alocadas a nível de tarefa, permitindo o processamento paralelo de distintas mensagens por um mesmo processo. Já no modelo baseado em processo, as *threads* são alocadas ao processo, que quando inicia o processamento de uma mensagem bloqueia as demais até que essa seja finalizada.

No modelo de execução baseado em processos, quando uma tarefa de uma porta realiza uma solicitação à uma aplicação integrada, a *thread* atribuída permanece bloqueada e inativa até que esta tarefa receba a resposta.

Por não ter controle sobre as aplicações integradas, o tempo de resposta pode ser longo, mantendo assim a *thread* bloqueada e desperdiçando recursos do sistema.

O modelo de execução da tecnologia Guaraná, baseado em tarefas, permite que as tarefas sejam executadas assim que tiver uma mensagem disponível em todos os *slots* de entrada. Sendo assim, considerada pronta para ser executada.

Uma fila de trabalho (*work queue*) armazena as unidades de trabalho (*work units*) em uma fila de prioridade, para serem processadas. Nessa fila, são anotadas as tarefas que podem ser executadas. A unidade de trabalho é composta por uma tarefa para ser executada e um tempo programado para a execução. Normalmente, este tempo é definido como o tempo atual. Isso significa que a tarefa pode ser prontamente executada. No entanto, algumas tarefas possuem um tempo agendado, conseqüentemente a execução é adiada até o transcorrer do tempo. A *thread* no motor de execução é responsável por retirar uma unidade de trabalho da fila e executar. O Guaraná permite que sejam utilizadas várias *threads* para executar uma solução de integração.

A Figura §2.4 é um recorte da solução de integração apresentada no Capítulo §4, para entender melhor o funcionamento do motor de execução. Na Figura, é possível observar que nos *slots* S0, S1, S2 e S4 mensagens aguardam para serem executadas pelas tarefas T0, T1, T2 e T3. No entanto, a tarefa T3 não está apta para ser executada porque não apresenta mensagens no *slot* S5.

Para cada tarefa apta, o motor criará uma *work unit* e a incluirá na fila de prioridade *work queue* para serem executadas pelo motor. As *threads* irão consumir de forma concorrente as *work units* dessa fila e processar a tarefa associada.

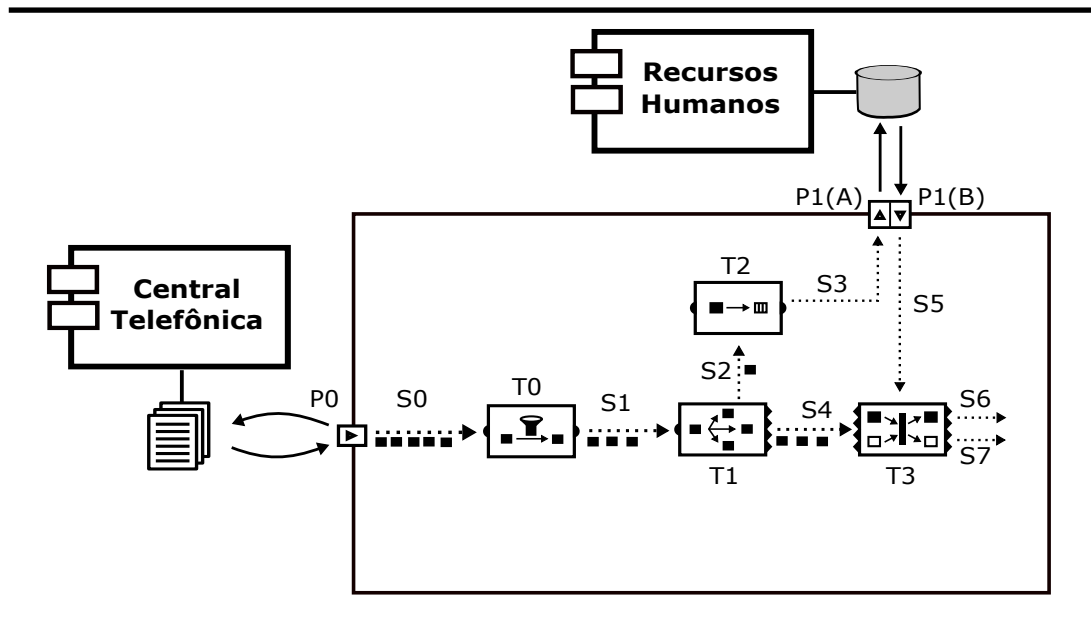


Figura 2.4: Recorte para exemplificar o processo de execução

2.2 Sistemas e Simulação de Modelos

Nesta pesquisa, o termo simulação refere-se à simulação de sistemas. O termo sistema é utilizado em muitas situações, o que dificulta a elaboração de uma definição suficientemente abrangente e, da mesma forma, sucinta. De acordo com Law e Kelton [26], na prática, a definição de sistema depende dos objetivos de um particular estudo. Para Prado [32], uma definição simples seria: “Sistema é uma agregação de objetos que têm alguma interação ou interdependência”.

Sommerville [39] também aborda o assunto, sendo que para ele uma definição útil para sistema seria: “Um sistema é uma coleção significativa de componentes inter-relacionados, que trabalham em conjunto para atingir um objetivo”. Em outras palavras, sistema pode ser definido como um conjunto de objetos entre os quais pode-se encontrar algum tipo de relação, que atuam, interagem e cooperam para alcançar um objetivo ou um propósito lógico [16].

A Figura §2.5 ilustra as diferentes formas de estudar um sistema. Inicialmente, existem duas possibilidades: experimento com o sistema real ou experimento com um modelo do sistema.

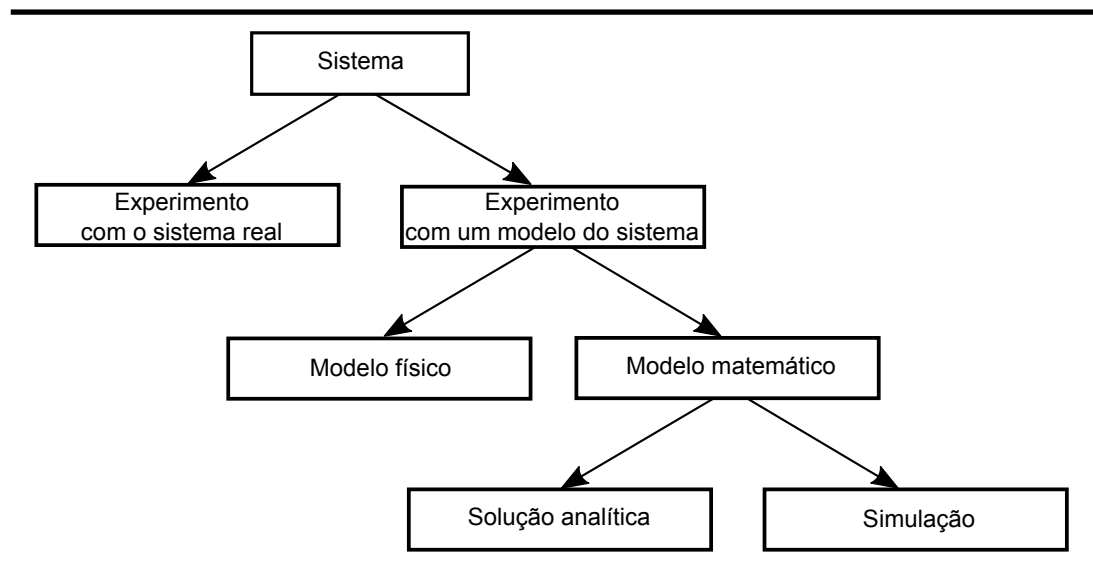


Figura 2.5: Abordagens para estudar um sistema (de Law e Kelton [26])

Sistemas reais podem ser estudados por meio de experimentos e medições diretamente no sistema. No entanto, isto pode demandar tempo, alto custo financeiro e causar perturbações no sistema. A outra opção é realizar experiências por meio de um modelo do sistema.

Um modelo pode ser definido como uma representação do sistema real e possibilita estudar e analisar um sistema sem a necessidade da implementação real. O modelo pode ser suficientemente detalhado ou “válido” para permitir ao analista tomar as mesmas decisões que tomaria utilizando o sistema real. Sua principal vantagem encontra-se na possibilidade de realizar alterações no sistema, para fins de estudo e análise dos resultados, sem que se tenha a necessidade de alterar o sistema real [26].

Os modelos do sistema podem ser físicos ou matemáticos. Modelos físicos são empregados para projetar protótipos em escala. Por exemplo, para estudar a aerodinâmica de um automóvel.

O modelo matemático é uma representação do sistema real. De acordo com Hillier e Lieberman [18], um dos principais pontos fortes do modelo matemático é a abstração da essência do problema e revela sua estrutura subjacente, fornecendo as relações causa-efeito contidas no sistema. Em uma solução analítica, fórmulas matemáticas são utilizadas para representar os sistemas. Esses modelos possuem um alto grau de abstração e, portanto, de simplificação em relação aos sistemas que representam [16].

Do ponto de vista de Law e Kelton [26], as soluções analíticas utilizam técnicas tradicionais de matemática e de estatística para analisar e obter informações exatas sobre o sistema. No entanto, muitos sistemas são tão complexos que os modelos matemáticos que representam estes sistemas são igualmente complexos, inviabilizando a utilização de soluções analíticas porque as relações matemáticas ficam extensas e o cálculo complexo. Nesse caso, procura-se utilizar a simulação, porque consegue modelar as características complexas do sistema, inclusive seu aspecto estocástico (ou probabilístico) e dinâmico, sem tornar o modelo difícil de ser estudado.

A simulação é um método que utiliza um modelo matemático para possibilitar o estudo e a análise do comportamento de um sistema sem que seja necessário realizar alterações no sistema real. Pode-se assim, prever o comportamento e ajudar no processo de tomada de decisão.

Na literatura, encontram-se várias definições do termo simulação. Na visão de Schriber [38], simulação implica na modelagem de um sistema, de tal forma que o modelo imite as respostas do sistema real em uma sucessão de eventos, que ocorrem ao longo do tempo. Pegden et al. [31] afirmam que simulação é o processo de projetar um modelo computacional de um sistema real e conduzir experimentos com o modelo para entender seu comportamento e/ou avaliar estratégias para sua operação.

A simulação pode ser efetuada de forma manual. No entanto, a execução da simulação normalmente exige a geração e processamento de uma grande quantidade de dados. Isso, possivelmente, irá demandar muito tempo, além de estar propensa a erros, pelas inúmeras repetições das mesmas operações matemáticas. Assim, esses experimentos simulados devem ser, inevitavelmente, realizados em um computador [18].

O advento do computador e o desenvolvimento de softwares de simulação tornou esta técnica popular, à medida que a simulação se popularizou nas últimas décadas. Atualmente, simulação é sinônimo de simulação computacional. A simulação computacional executa um modelo computacional, o qual imita o comportamento do sistema, à medida que as variáveis apresentam o mesmo comportamento dinâmico e estocástico do sistema real.

De acordo com Hillier e Lieberman [18], um modelo de simulação sintetiza o sistema simulado construindo-o, componente por componente, e evento por evento. Em seguida, o modelo é executado para obter, por meio dos diversos eventos gerados aleatoriamente, as estatísticas do desempenho do sistema.

A simulação é uma excelente ferramenta de análise de sistemas e a principal vantagem é que os sistemas reais estudados não precisam ser alterados. No entanto, pode apresentar algumas desvantagens:

- A formulação e experimentação com modelos de simulação consomem muitos recursos, principalmente tempo para desenvolver/depurar os programas de computador necessários para executar a simulação e, por vezes são necessários vários processamentos longos para obter dados confiáveis.
- Todos os dados e resultados obtidos da simulação devem ser cuidadosamente analisados, interpretados, devendo representar a realidade.

Hillier e Lieberman [18] afirmam que a simulação não deve ser utilizada quando existe a possibilidade de um procedimento menos oneroso, capaz de fornecer as mesmas ou melhores informações sobre o sistema. Geralmente a simulação apenas é utilizada quando o sistema estudado é muito complexo para ser analisado por modelos matemáticos, de forma satisfatória.

De acordo com Freitas Filho [16], as razões mais comuns pelas quais tem-se clareza da importância dos modelos de simulação são:

- **O sistema real ainda não existe:** nesse caso a simulação é utilizada para planejar o futuro sistema. Um novo hospital, uma nova fábrica ou um novo ambiente de suporte a negócios na Internet, por exemplo.
- **Experimentar com o sistema real é dispendioso:** com um custo menor o modelo poderá indicar quais os benefícios de se investir em um novo equipamento, por exemplo.
- **Experimentar com o sistema real não é apropriado:** um exemplo típico é o planejamento do atendimento de situações de emergência de um desastre aéreo em um aeroporto. Não se pode provocar um desastre deste tipo para testar planos de emergência.

2.2.1 Passos na Formulação de um Modelo de Simulação

Para que a análise de um sistema, por meio da modelagem e simulação, seja bem sucedido, é necessário seguir algumas etapas. De acordo com Paul e Balmer [30], o desenvolvimento de um modelo de simulação é composto de três grandes etapas, ilustradas na Figura §2.6:

- i. Formulação do modelo;
- ii. Implementação do modelo;
- iii. Análise dos resultados do modelo.

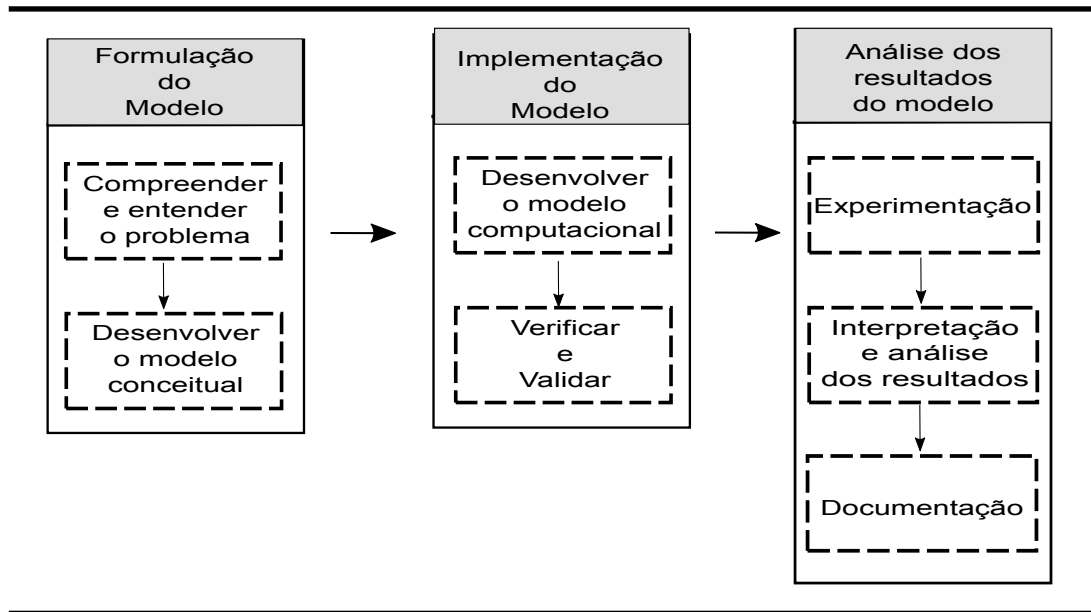


Figura 2.6: *Etapas em um estudo envolvendo modelagem e simulação*

Na primeira etapa, deve-se entender e compreender o sistema que será simulado. Os objetivos do estudo e os dados de entrada devem ser definidos por meio da discussão do problema com especialistas. Todas as hipóteses devem ser claramente estabelecidas e realizada uma descrição dos vários cenários que serão investigados.

A formulação do modelo conceitual também é realizada na primeira fase. É traçado um esboço do sistema de forma gráfica ou algorítmica, definindo os componentes e a estratégia de modelagem discreta ou contínua, descrevendo as variáveis e interações lógicas que constituem o sistema [16].

A segunda etapa é constituída da conversão do modelo conceitual em um modelo computacional por meio da implementação no computador, utilizando uma linguagem de simulação ou um simulador comercial apropriado. Nessa fase também é realizada a verificação e validação do modelo computacional, a partir de alguns resultados gerados pela execução do modelo de simulação.

A verificação é importante para analisar se o modelo realiza o que é pretendido e está livre de erros da implementação computacional, por meio da aplicação de testes de consistência. Já a validação é necessária para determinar se o modelo de simulação representa de maneira satisfatória o sistema real para realizar as experimentações. A validação é realizada por meio da análise dos resultados obtidos da simulação, observando se estes são representativos dos resultados do sistema real, sendo possível apenas quando existe um sistema para referências ou comparações.

Após a verificação e validação, o modelo computacional está pronto para realizar os experimentos. Nessa terceira, etapa o modelo computacional é executado várias vezes para geração de dados que são interpretados e analisados estatisticamente. De acordo com Freitas Filho [16], estimativas para as medidas de desempenho do sistema nos cenários planejados são efetuadas. As análises poderão resultar na necessidade de um maior número de execuções (replicações) do modelo de simulação, para alcançar precisão estatística dos resultados.

Realizada a análise dos resultados é necessário, documentar o modelo de simulação. Uma documentação adequada serve para auxiliar o entendimento do estudo, possibilitar que outras pessoas utilizem o modelo e os resultados obtidos, bem como dar credibilidade aos resultados e facilitar futuras modificações, quando necessárias.

Para estudar e analisar um sistema por meio da simulação, é preciso conhecer e compreender as definições dos principais elementos do sistema.

2.2.2 Elementos da Simulação

Uma série de termos é utilizada para conceituar os elementos básicos envolvidos no processo de estudar e analisar um sistema por meio da modelagem e simulação de sistemas. Kelton et al. [23] e Freitas Filho [16] sugerem os seguintes elementos:

- **Entidade:** Representa um objeto de interesse do sistema. É dinâmica, movendo-se dentro do sistema, sendo processada nos recursos e armazena informações e atributos. As entidades podem representar objetos, pessoas, carros, peças, documentos, etc.
- **Atributo:** Propriedade característica de cada entidade. Permite caracterizar e individualizar entidades. É possível caracterizar ou dar nome a peças, clientes ou tarefas, por meio de um atributo simples como um

código, um número ou tipo. Atributos também possibilitam obter estatísticas importantes para analisar o comportamento dos sistemas, à medida que armazena valores individuais para cada entidade, como por exemplo, o armazenamento do momento exato (tempo no relógio da simulação) da entrada da entidade no sistema.

- **Variável:** Permite a personalização do modelo de simulação em termos de estatística e lógica. Armazena valores e informações, que ficam disponíveis para todo modelo. As variáveis pertencem ao sistema geral, enquanto um atributo está vinculado a uma entidade específica. Como exemplo de variável, cita-se o tempo total da entidade no sistema.
- **Variável de Estado:** Variáveis cujos valores determinam o estado do sistema. Constituem o conjunto de informações necessárias à compreensão do que está acontecendo no sistema (ou modelo), em um determinado instante no tempo, em relação aos objetos de estudo. Cita-se, como exemplo, o número de clientes esperando na fila de um banco, em um determinado instante da simulação.
- **Evento:** Acontecimentos que ocorrem em um instante de tempo e que provocam uma mudança de estado em um sistema. Toda mudança de estado é provocada pela ocorrência de um evento, porque sempre que ocorre um evento, pelo menos uma variável é alterada. Normalmente, três tipos de eventos podem acontecer: chegada, início do atendimento e fim do atendimento.
- **Atividade** Execução de um serviço, procedimento ou ação, em um determinado período de tempo. Na simulação, corresponde a um período de tempo predeterminado e a duração da atividade pode ser um valor aleatório, baseado em uma distribuição de probabilidade. Todo início e final de uma atividade é causada por um evento (mudança de estado).
- **Processo:** Consiste nas ações realizadas sobre as entidades, ao longo da simulação. É uma sequência de eventos, em ordem cronológica, separados por intervalos de tempos que descrevem o ciclo de vida de cada entidade. Durante cada intervalo de tempo, uma entidade estará recebendo serviço ou esperando por serviço.
- **Recurso:** Objeto que fornece os serviços requisitados pelas entidades. Um recurso processa uma entidade por um tempo e pode ter vários estados, tais como: ocupado, livre, bloqueado e indisponível. São exemplos, uma máquina, um atendente e um operador.

- **Fila:** Acúmulo de entidades geradas pela espera de entidades que não consegue acessar o recurso solicitado imediatamente. O gerenciamento de uma fila depende da disciplina adotada no sistema, sendo a mais comum FIFO (sigla em inglês para *First-In, First-Out*).

Os elementos envolvidos na modelagem caracterizam o modelo, em função da maneira como provocam mudanças no sistema.

2.2.3 Classificação dos Modelos de Simulação

Para o propósito de modelar e simular sistemas, é necessário saber identificar se o sistema a ser analisado representa um modelo determinístico ou estocástico; estático ou dinâmico; discreto ou contínuo.

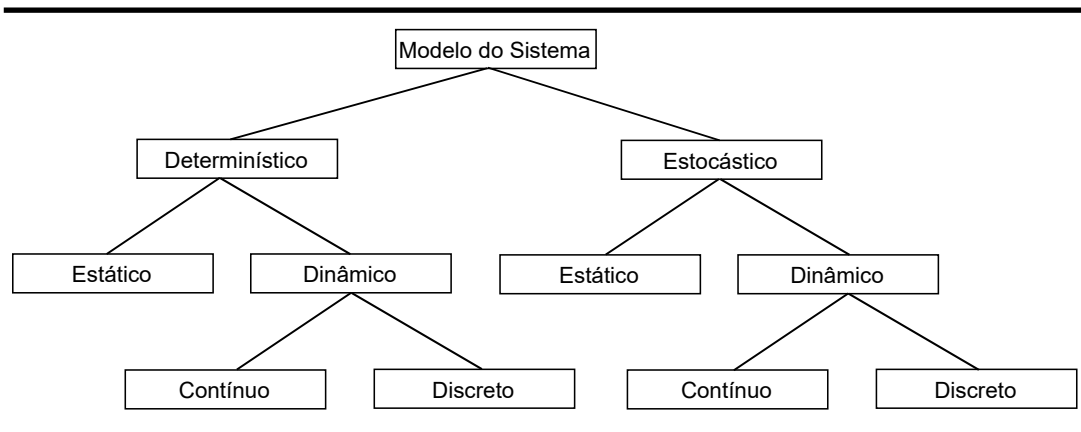


Figura 2.7: Taxonomia de modelos (de Law e Kelton [26])

A Figura 2.7 ilustra a Taxonomia de Modelos proposta por Law e Kelton [26], que classifica, inicialmente, os modelos em determinístico ou estocásticos. Um modelo é determinístico quando as variáveis não possuem nenhum componente probabilístico. Na simulação de um modelo determinístico, os dados de entrada são considerados constantes, gerando assim saídas constantes. De acordo com Sawicki et al. [36], um modelo determinístico para um conjunto conhecido de entrada de dados, produz um conjunto único de saída.

Um modelo estocástico ou probabilístico utiliza variáveis aleatórias como entrada que determina e produz saídas aleatórias. A saída de uma simulação estocástica deve ser analisada como uma estimativa estatística das

características reais de um sistema. Um modelo de simulação estocástico representa um sistema que evolui probabilisticamente ao longo do tempo. Para imitar o desempenho do sistema real, utiliza-se distribuições de probabilidade para gerar aleatoriamente, os eventos que acontecem no sistema [18]. Geralmente, os modelos estocásticos são mais complexos, mas representam melhor um sistema real do que os modelos determinísticos.

Segundo Law e Kelton [26], um modelo de simulação estático é uma representação de um sistema em um determinado momento. Neste tipo de simulação, a variável tempo não é um fator determinante, ou seja, não influencia no modelo. São exemplos de simulação estática os modelos de Monte Carlo [32].

Em relação ao modelo de simulação dinâmico, Law e Kelton [26] afirmam que este modelo é influenciado pelo tempo, pois representa um sistema que evolui ao longo do tempo. A simulação realizada com um modelo dinâmico considera as aleatoriedades e a interdependência das variáveis. Isso melhora a capacidade da simulação em prever o comportamento real do sistema.

De acordo Freitas Filho [16], para um modelo computacional evoluir dinamicamente na simulação de um sistema, é preciso utilizar programas de computadores orientados a eventos. Dessa forma, na medida em que o tempo de simulação evolui, determinados eventos provocam alterações nas variáveis do programa, as quais são responsáveis por informar a ocorrência de mudanças nos estados que envolvem o modelo.

O modelo dinâmico pode ser classificado em contínuo ou discreto. De acordo com Law e Kelton [26], os modelos de simulação discreto e contínuo são definidos de forma análoga aos sistemas discretos e contínuos. Nos sistemas contínuos, as variáveis de estado mudam continuamente no tempo. Nesse modelo, as variáveis que descrevem o sistema têm seus valores alterados de forma contínua em incrementos de tempos iguais, como por exemplo, o modelo de simulação de um voo de uma aeronave, onde o objetivo é conferir a cada segundo o valor da variável nível atual de combustível.

Segundo Freitas Filho [16], um modelo de simulação discreto mantém as variáveis de estado inalteradas ao longo de intervalos de tempos e mudam seus valores somente em pontos específicos, conhecidos como tempo de ocorrência do evento. Como exemplo de um evento discreto, cita-se uma fila de um supermercado, na qual pode-se considerar a ocorrência de três eventos: chegada, atendimento e saída.

A simulação é uma excelente ferramenta de estudo e análise de um sistema, porque o modelo de simulação pode ser quase tão detalhado quanto o

sistema real, tendo ainda a vantagem de não causar perturbações no sistema real. Diferentemente da modelagem analítica que utiliza fórmulas matemáticas a simulação tenta imitar o funcionamento do sistema real. Nos modelos de simulação este objetivo é alcançado através da utilização de distribuições de probabilidades.

2.3 Distribuições de Probabilidade

Os modelos de simulação computacionais devem ser capazes de imitar o comportamento estocástico do sistema real. Esse objetivo é alcançado pela utilização de distribuições de probabilidade para representar os eventos aleatórios.

Na modelagem de sistemas de filas, as distribuições de probabilidade são utilizadas para representar o comportamento aleatório no processo de chegada e atendimento. A distribuição de probabilidade comumente utilizada para representar a taxa média de chegada, é a distribuição de Poisson e para o tempo médio entre chegadas sucessivas e tempo médio de atendimento, é a distribuição exponencial.

Segundo Hillier e Lieberman [18], na Teoria das Filas para formular um modelo de filas representativo do sistema real, é necessário especificar a forma assumida das distribuições probabilísticas do processo de chegada e atendimento. A forma assumida deve ser suficientemente realista, cujo modelo forneça previsões razoáveis sendo, ao mesmo tempo, simples a fim de que modelo seja matematicamente tratável. Baseado nessas considerações, essa seção aborda as distribuições probabilísticas mais importantes na Teoria das Filas; a distribuição de Poisson e a distribuição exponencial.

2.3.1 Distribuição de Poisson (α)

A distribuição de Poisson consiste em uma distribuição discreta de probabilidade, comumente utilizada para descrever as probabilidades do número de ocorrências de sucesso de um determinado evento aleatório, em um intervalo especificado.

Uma característica da distribuição de Poisson consiste no fato de permitir a contagem de sucessos e não a de falhas. Por exemplo, em uma análise no número de clientes que entram em uma loja, pode-se determinar o número de pessoas que entraram na loja durante certo intervalo. Porém, não se

pode determinar o número de pessoas que deixaram de entrar. Por essa razão, a distribuição de Poisson é uma distribuição que envolve somente estados contáveis [2].

Os exemplos do emprego da distribuição de Poisson são diversos:

- Número de clientes que chegam ao caixa de um supermercado em determinado intervalo de tempo;
- Número de requisições feitas a um servidor em determinado intervalo de tempo;
- Número de componentes de um sistema que falham em um intervalo de tempo.

Muitos experimentos consistem em observar os tempos de ocorrência de chegadas aleatórias. Os exemplos incluem as chegadas de clientes para o serviço, as chegadas de chamadas a uma central telefônica, entre outras. A distribuição de Poisson é utilizado para modelar o número destas chegadas que ocorrem em um dado período de tempo. Como é aplicável a inúmeros processos de chegada, é muito utilizada na modelagem de filas.

Considerando a probabilidade de ocorrência de sucessos em um determinado intervalo, para aplicar a distribuição de Poisson, as seguintes hipóteses devem ser válidas:

- O número de ocorrências de um evento, em qualquer intervalo, é independente do número de eventos ocorrendo em outros intervalos;
- A probabilidade de mais de uma ocorrência simultânea nesse intervalo é de aproximadamente zero;
- A probabilidade da ocorrência de um sucesso no intervalo é igual em intervalos de mesmo comprimento.

Definição: Uma variável aleatória discreta X tem distribuição de Poisson com parâmetro $\alpha > 0$ se:

$$P(X = k) = \frac{e^{-\alpha} \alpha^k}{k!}, \quad \text{para } k = 0, 1, 2, \dots \quad (2.1)$$

Onde:

e = constante cujo valor aproximado é 2, 71828

α = número médio de ocorrências em um determinado intervalo de tempo ou espaço

k = número de ocorrências de sucesso no intervalo desejado

O valor esperado e variância de X são respectivamente,

$$E(X) = \alpha$$

$$\text{Var}(X) = \alpha$$

A distribuição de probabilidade representa a probabilidade da variável aleatória X assumir um certo valor k : $P(X = k)$. A soma de todas as possibilidades que k pode assumir é 1. Assim, para verificar que a expressão §2.1 representa uma legítima distribuição de probabilidade, basta observar que usando o resultado da expansão em série de Taylor da função exponencial, sabe-se que para todo x real:

$$e^x = \sum_{k=0}^{\infty} \frac{x^k}{k!} \quad (2.2)$$

Portanto,

$$\sum_{k=0}^{\infty} P(X = k) = \sum_{k=0}^{\infty} \frac{e^{-\alpha} \alpha^k}{k!} = e^{-\alpha} \sum_{k=0}^{\infty} \frac{\alpha^k}{k!} = e^{-\alpha} e^{\alpha} = e^0 = 1 \quad (2.3)$$

Para determinar a probabilidade de um dado número de ocorrências na distribuição de Poisson, é necessário somente o valor de α . A média do número de eventos em um intervalo de tempo ou espaço, representada por α , é chamada de parâmetro da distribuição de Poisson.

2.3.2 Distribuição Exponencial (α)

A distribuição exponencial consiste em uma distribuição contínua de probabilidade e desempenha papel importante na descrição de uma grande classe de fenômenos.

A distribuição exponencial é muito utilizada na modelagem de tempos decorridos entre dois eventos, como por exemplo o tempo entre duas chegadas consecutivas de entidades em um sistema ou entre falhas de um equipamento [16].

Definição: Uma variável aleatória contínua T , que tome todos os valores não-negativos, tem uma distribuição exponencial com parâmetro $\alpha > 0$ se sua função de densidade probabilística for dada por

$$f_T(t) = \begin{cases} \alpha e^{-\alpha t} & \text{para } t \geq 0 \\ 0 & \text{para } t < 0, \end{cases} \quad (2.4)$$

conforme mostrado na Figura §2.8. Nesse caso, as probabilidades acumulativas são

$$\begin{aligned} P\{T \leq t\} &= 1 - e^{-\alpha t} \quad t \geq 0, \\ P\{T > t\} &= e^{-\alpha t} \end{aligned} \quad (2.5)$$

O valor esperado e a variância de T são, respectivamente,

$$E(T) = \frac{1}{\alpha}$$

$$\text{Var}(T) = \frac{1}{\alpha^2}$$

Suponha que uma variável aleatória T represente tempos entre chegadas ou tempos de atendimento e as ocorrências que marcam o final desses tempos (chegadas ou finalizações de atendimentos) denominados de eventos. Por que é possível supor que T possui uma distribuição exponencial para um modelo de filas?

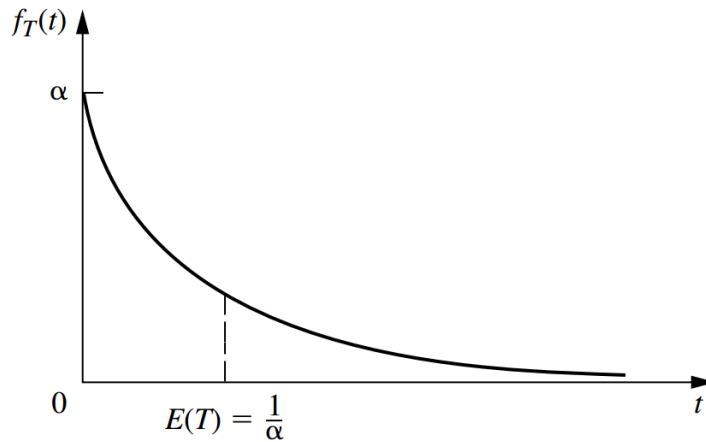


Figura 2.8: Função de densidade probabilística (de Hillier e Lieberman [18])

Para responder a essa pergunta, as justificativas a seguir estão baseadas nas interpretações das propriedades da distribuição exponencial apresentadas por Hillier e Lieberman [18].

Propriedade 1: $f_T(t)$ é uma distribuição estritamente decrescente de t ($t \geq 0$).

Consequência da Propriedade 1 é a grande probabilidade de tempos entre chegadas pequenos, e tempos de atendimento curtos e a pequena probabilidade para tempos entre chegadas grandes e tempos de atendimento longos. Entretanto, um padrão irregular como este faz parte da aleatoriedade. Assim, a distribuição exponencial pode ser utilizada para tempos de atendimento em sistemas onde a maioria dos clientes é atendida de forma rápida e o atendimento de alguns clientes é demorado.

Considere um sistema real, onde as tarefas solicitadas ao servidor são diferentes de um cliente para outro. A natureza geral do atendimento pode ser a mesma, mas difere o tipo e o tempo de atendimento. Por exemplo, caixas de bancos e caixas de lojas podem atender aos clientes de forma rápida, contudo, eventualmente, o atendimento de um cliente pode ser mais demorado. Assim, para esse tipo de situação pode-se admitir que o tempo de atendimento tem distribuição exponencial.

Se T representar tempos entre chegadas, a Propriedade 1 descarta as situações em que clientes tendem a adiar a entrada no sistema de filas, caso vejam outro cliente entrar à frente deles.

Propriedade 2: Falta de memória.

A distribuição probabilística do tempo até o evento chegada ou término do atendimento ocorrer é sempre a mesma, independente de quanto tempo já tenha passado. Em outras palavras, a probabilidade de um evento é independente do instante de tempo em que ocorreu o evento anterior. Isso significa que o conhecimento prévio do tempo da ocorrência do último evento não ajuda na previsão do tempo de ocorrência do próximo evento. De fato, o processo se “esquece” de seu passado, sendo essa a razão da grande aplicabilidade da distribuição exponencial para modelagem de tempos decorridos entre dois eventos.

A Propriedade 2 aplicada ao processo de chegada a um sistema de filas significa que o tempo até a próxima chegada não sofre nenhuma influência do instante de tempo da última chegada. Já se for aplicada para tempos de atendimento, em que as operações de atendimento diferem entre os clientes, nesse caso, um longo período de atendimento não significa que o serviço deve estar quase concluído, mas sim que esse cliente requer um atendimento mais longo.

Propriedade 3: Relação com a distribuição de Poisson.

Suponha que o tempo entre ocorrências consecutivas de um evento, por exemplo, chegadas ou termos de atendimento, tenha uma distribuição exponencial com parâmetro α . A Propriedade 3, tem relação com a implicação sobre a distribuição do número de vezes que esse tipo de evento ocorre ao longo do tempo especificado.

Particularmente, façamos que $X(t)$ seja o número de ocorrências no instante t ($t \geq 0$), onde o tempo 0 designa o instante de início da contagem. A implicação é que

$$P\{X(t) = n\} = \frac{(\alpha t)^n e^{-\alpha t}}{n!}, \quad \text{para } n = 0, 1, 2, \dots ;$$

isto é $X(t)$ possui uma distribuição de Poisson com parâmetro αt . Por exemplo, com nenhum cliente no sistema $n = 0$,

$$P\{X(t) = 0\} = e^{-\alpha t},$$

é a probabilidade da distribuição exponencial de que o primeiro evento ocorra após o tempo t . A média dessa distribuição de Poisson é

$$E\{X(t)\} = \alpha t,$$

de modo que o número esperado de eventos por unidade de tempo seja α . Portanto, α é a taxa média da ocorrência de eventos. O processo de contagem $\{X(t); t \geq 0\}$ de eventos em uma forma contínua é um processo de Poisson com parâmetro α , que representa a taxa média da ocorrência desses eventos.

Para tempos de atendimento com distribuição exponencial de parâmetro μ , a Propriedade 3 fornece informações úteis sobre os termos do atendimento. Essa informação é obtida definindo $X(t)$ como o número de conclusões de atendimentos de um servidor ocupado, no tempo decorrido t , onde $\alpha = \mu$.

Sistemas de filas em que os tempos entre chegadas sucessivas possuem uma distribuição exponencial de parâmetro λ , a propriedade é muito útil para descrever o comportamento probabilístico das chegadas. Nesse caso, $X(t)$ é o número de chegadas no tempo decorrido t , onde a taxa média de chegada é $\alpha = \lambda$. Portanto, as chegadas ocorrem de acordo com um processo de entrada de Poisson com parâmetro λ . Estes modelos são descritos com uma taxa de chegada e tempo entre chegadas sucessivas de acordo, respectivamente, com a distribuição de Poisson e distribuição exponencial.

2.4 Teoria das Filas

As filas fazem parte do nosso cotidiano e, certamente, qualquer pessoa sabe exatamente o que são filas porque, em algum momento, deve ter esperado para pagar as compras em um supermercado, fazer um depósito bancário, comprar um ingresso no cinema, pagar o pedágio, entre outras situações. As filas são dispendiosas e causam ineficiências. O estudo das filas, por meio da Teoria das Filas, tem como objetivo dimensionar os sistemas de forma que não ocorra a formação de filas.

2.4.1 Definição

Aguardar por um serviço em uma fila não é nada agradável e, quando é necessário esperar por muito tempo, causa aborrecimento. Diante dessas amargas experiências, geralmente as pessoas adotam a atitude de não mais comprar em um supermercado, mudar a conta bancária para outra agência ou banco, entre outras. Para as empresas, essas atitudes significam perda de negócios. O ideal seria chegar ao local de serviço e ser atendido imediatamente.

De acordo com Hillier e Lieberman [18], as filas de espera não se resumem a transtornos pessoais. O tempo perdido nas filas é um importante fator na qualidade de vida e na eficiência da economia de uma nação. A formação de outras filas esperas, além de pessoas, podem causar grandes ineficiências. Por exemplo, máquinas esperando por reparação resultam em perdas na produção. Veículos que aguardam para serem descarregados atrasam embarques seguintes. Aviões aguardando para decolar ou pousar atrasam voos posteriores. Ordens de produção esperando para serem realizadas pode afetar a produção de lotes seguintes.

Filas são dispendiosas e segundo Prado [32], quando as filas de um sistema recebem valores além dos adequados, passam a constituir gargalos de desempenho. O ideal seria dimensionar os sistemas de modo que não existissem filas. Para minimizar as implicações e operar um sistema de filas de forma mais eficiente, utiliza-se a Teoria da Filas porque permite calcular e estimar resultados relacionados à performance dos sistemas, com base em propriedades mensuráveis tais como os tempos de serviço e/ou a quantidade de clientes servidos, durante um determinado período de observação [16].

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. Fogliatti e Mattos [10] define a Teoria das Filas como sendo a modelagem analítica de sistemas de filas e tem como objetivo determinar e avaliar medidas de desempenho, que expressam a produtividade/operacionalidade desse sistema. Entre essas medidas, pode-se citar: número de clientes na fila, tempo de espera pelo atendimento e tempo ocioso dos servidores.

Algumas características devem ser conhecidas em um sistema de filas para fazer a escolha do modelo analítico que melhor representa o sistema sob análise. Para tanto, a seguir apresenta-se o estudo realizado sobre os elementos que compõem o sistema.

2.4.2 Elementos de um Sistema de Filas

De acordo com Fogliatti e Mattos [10], 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.

Um sistema de filas pode ser representado por diferentes modelos. No entanto, possuem elementos comuns, que fazem parte do processo básico, conforme representado na Figura §2.9.

De acordo com Hillier e Lieberman [18], os elementos que compõem o processo são: população de onde provêm as chegadas dos **clientes** que necessitam de **atendimento**. Os clientes 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, deixa o sistema de filas.

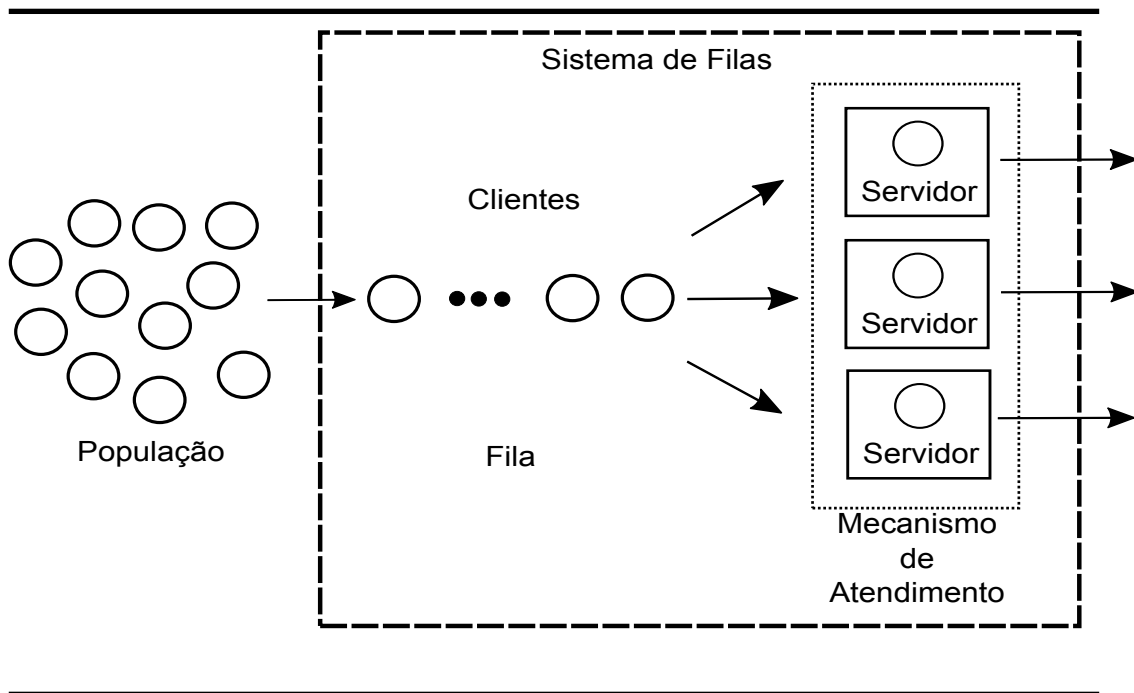


Figura 2.9: Elementos de uma fila (de Prado [32])

O termo cliente é usado de forma genérica e, como sinônimo, usa-se o termo transação ou entidade. O atendimento pode ser constituído de um ou mais servidores que também podem ser chamados de atendentes ou canais de serviços. Isso porque os clientes não são necessariamente pessoas, podem ser, por exemplo, peças ou carros. Da mesma maneira, os atendentes não precisam ser pessoas. Um atendente pode ser por exemplo, uma máquina, um veículo, um dispositivo eletrônico, etc.

A Teoria das Filas pode ser aplicada a vários tipos diferentes de sistemas de filas. O tipo mais frequente está representado na Figura 2.9. Nesse tipo de sistema, a fila de espera é única, formada na frente de uma única instalação de atendimento, que pode ter um ou mais atendentes.

A especificação de um modelo de fila normalmente requer que as características de desempenho sejam declaradas. Essas características são: processo de chegada, processo de atendimento, número de servidores, capacidade da fila, tamanho da população e disciplina da fila. Na modelagem de sistemas de filas, é comum a utilização de uma notação para representar, de maneira resumida, estas seis características. Nas seções a seguir apresenta-se esta notação e as definições das seis características.

2.4.3 Clientes e Tamanho da População

Um cliente é oriundo de uma população. Segundo Hillier e Lieberman [18], o tamanho da população é o número total de possíveis clientes distintos que necessitariam de atendimento, em determinados períodos de tempo.

Para Hillier e Lieberman [18], o tamanho da população pode ser suposto como infinito ou finito. Quando a população é grande, para efeitos práticos, propõe-se que o tamanho é infinito porque os cálculos realizados são considerados mais fáceis. Conforme Prado [32], a população é infinita quando a chegada de um novo cliente a uma fila não afeta a taxa de chegada de clientes subsequentes. Dessa forma, as chegadas são definidas como independentes.

Entretanto, quando a população é finita (pequena), a solução analítica é considerada difícil. Nesse caso, a qualquer momento, o número de clientes que estão no sistema de filas pode afetar o número de possíveis clientes fora do sistema. Nessa situação, toda a população está na fila. Assim, durante um determinado período, não chegará nenhum cliente ao sistema [18].

2.4.4 Processo de Chegada

De acordo com Fogliatti e Mattos [10], o processo de chegada é definido pelo comportamento do fluxo de chegadas dos clientes ao sistema, podendo ser determinístico ou estocástico. O processo é definido como determinístico quando são conhecidos o número de chegadas e os instantes de tempo em que ocorrem. Nesse caso, o processo de chegada é regular, não havendo variação entre os valores do tempo entre chegadas sucessivas. Isso significa que se o tempo entre chegadas for 5 segundos, rigorosamente, a cada 5 segundos chegará um novo cliente. Esse comportamento é muito raro em um sistemas de filas.

Normalmente, o fluxo de chegadas apresenta um comportamento aleatório, constituindo um processo estocástico caracterizado por uma distribuição

de probabilidade. Para essa distribuição, é necessária a especificação de um parâmetro denominado taxa de chegadas, que representa o número médio de clientes que chegam ao sistema por unidade de tempo. Para quantificar essa variável, é usada a letra grega λ , que significa taxa média de chegada e IC para intervalo médio entre chegadas, também denominado de tempo médio entre chegadas sucessivas [32].

Segundo Prado [32], utiliza-se uma distribuição de probabilidade para caracterizar um processo de chegada aleatório porque não basta fornecer valores médios, mas é necessário mostrar como os valores se distribuem em torno da média. Geralmente, a chegada de clientes ocorre de acordo com um processo de Poisson. Isso significa, que o número de clientes que chegam até dado momento é expresso pela distribuição de Poisson. Nesse caso, as chegadas ocorrem de forma aleatória, porém a uma taxa média fixa, independentemente de quantos clientes já estão no sistema de filas [18].

Uma hipótese equivalente é que a distribuição de probabilidade do tempo entre chegadas consecutivas é uma distribuição exponencial. Prado [32] afirma que ao analisar um processo de chegada e verificar que a taxa de chegadas segue a distribuição de Poisson, pode-se então afirmar que os tempos entre chegadas seguirão a distribuição exponencial.

2.4.5 Processo de Atendimento

O processo de atendimento é descrito pelo tempo de atendimento por cliente e, geralmente, ocorre de forma estocástica caracterizado por uma distribuição de probabilidade. O modelo de um sistema de filas deve especificar a distribuição de probabilidade dos tempos de atendimento do servidor.

Para a distribuição, é necessário especificar o parâmetro tempo de atendimento, que representa a duração média do atendimento do cliente. Segundo Hillier e Lieberman [18], a distribuição de tempos de atendimento com maior frequência, na prática, é a distribuição exponencial, por que é mais fácil de ser tratada.

Prado [32] afirma que o processo de atendimento também é quantificado por uma importante variável randômica. A letra grega μ significa taxa média de atendimento e TA para o tempo médio de atendimento.

2.4.6 Mecanismo de Atendimento e Número de Servidores

O mecanismo de atendimento é formado por uma ou mais instalações de atendimento, cada uma das quais contendo um ou mais servidores [18]. O

número de servidores pode ser finito ou infinito, mas a maioria dos modelos parte do pressuposto de uma instalação de atendimento com um servidor ou com um número finito. Entretanto, existem sistemas de filas com mais de uma instalação de atendimento, onde os clientes devem receber atendimento em algumas ou todas as instalações (servidores em série), constituindo redes de filas. Redes de filas são abordadas de forma mais detalhada na Seção §2.4.9.

Um modelo de filas deve especificar o número de servidores. O sistema mais simples considerado, é aquele de um único servidor e, geralmente, admite-se que apenas um cliente pode ser atendido por um servidor no mesmo instante. Na Figura §2.9, o mecanismo de atendimento tem uma instalação de atendimento com 3 servidores.

2.4.7 Fila e Disciplina da Fila

Fila é o local onde os clientes esperam antes de serem atendidos. A fila de espera é caracterizada pelo espaço disponível definido pelo número máximo de clientes que pode conter. As filas são chamadas finitas ou infinitas de acordo com a capacidade (tamanho) limitada ou ilimitada, respectivamente. Os modelos de filas, geralmente, admitem por padrão capacidade ilimitada (fila infinita), mesmo quando é limitada, mas este limitante é relativamente grande.

A disciplina da fila define a ordem pela qual os clientes são selecionados da fila para o atendimento. A disciplina mais utilizada é FIFO (*First-In, First-Out*), na qual o atendimento é realizado de acordo com a ordem de chegada, o primeiro cliente que chegar é o primeiro a ser atendido.

Outros tipos de disciplina de fila também podem ser utilizados como: o último cliente que chegar é o primeiro a ser atendido (LIFO - *Last-In, First-Out*), o atendimento segue um ordem aleatória (SIRO - *service in random order*) e seleção para serviço por ordem de prioridade.

2.4.8 Notação de um Sistema de Filas

Para classificar um sistema de filas e especificar de forma resumida o modelo de filas é utilizada a notação proposta por Kendall [24], que possui a forma $A/B/s/K/m/Z$, onde.

- A representa a distribuição dos tempos entre chegadas sucessivas;
- B representa a distribuição do tempo de atendimento;

- s é o número de servidores em paralelo;
- K é a capacidade do sistema (tamanho da fila);
- m é o tamanho da população que fornece os clientes;
- Z é a disciplina da fila.

Os tipos de distribuições para os tempos entre chegadas sucessivas e tempo de atendimento costumam ser representadas pela seguinte nomenclatura:

- M: Exponencial (ou marcoviana ou Poisson);
- E_k : Erlang com parâmetro k ;
- H_k : Hiperexponencial com parâmetro k ;
- D: Determinística;
- G: Geral.

Segundo Fogliatti e Mattos (2007), alguns autores usam uma simplificação da notação, omitindo as letras K , m e Z quando a capacidade do sistema e o tamanho da população for infinita (∞) e a disciplina da fila é do tipo o primeiro a chegar, primeiro a ser atendido (FIFO).

Por exemplo, a notação $M/M/1/\infty/\infty/\text{FIFO}$ ou apenas $M/M/1$ indica um sistema de filas com as seguintes características:

- M: tempos entre chegadas sucessivas com distribuição exponencial ;
- M: tempo de atendimento com distribuição exponencial ;
- 1: um único servidor;
- ∞ : a capacidade do sistema é infinita ;
- ∞ : o tamanho da população é infinita;
- FIFO: a disciplina da fila é do tipo “ o primeiro a chegar é o primeiro a ser atendimento”.

2.4.9 Medidas de Desempenho de um Sistema de Filas

Para estabelecer as medidas de desempenho por meio da Teoria das Filas, um sistema de filas deve estar em uma condição de estado estável.

De acordo com Hillier e Lieberman [18], no início da operação, o estado do sistema (número de clientes no sistema) é afetado enormemente pelo estado inicial e pelo tempo que passou. Diz-se que o sistema se encontra em uma condição transitória. Entretanto, após ter decorrido um tempo, o estado do sistema se torna, basicamente, independente do estado inicial e do tempo decorrido e atinge uma condição de estado estável. A Teoria das Filas concentrou-se na condição de estado estável porque o caso transitório é mais difícil analiticamente.

A abordagem analítica de sistemas de filas exige estabilidade no fluxo de chegada e no processo de atendimento. Os valores da taxa média de chegada (λ) e taxa média de atendimento (μ) devem se manter constantes no tempo. Em um sistema que se encontra em estado estável, todas as características randômicas se mantêm estáveis durante todo tempo. Isso significa que oscilam apenas em torno do valor médio.

Outra exigência para o estado estável é a taxa média de atendimento ser maior que a taxa média de chegada ($\mu > \lambda$). Isso implica que a taxa de utilização dos atendentes seja menor que 1 ($\rho < 1$). Os atendentes devem ser capazes de atender o fluxo de chegada. De outro modo, o tamanho da fila aumentará infinitamente. De acordo com Prado [32], um sistema de filas que não atinge o estado estável deve ser estudado por meio da simulação.

Em sistemas de filas onde a taxa média de atendimento é maior que a taxa média de chegada ($\mu > \lambda$) ocorre a formação de filas porque os processos de chegada e atendimento não são regulares, e a aleatoriedade ocasiona tanto filas, quanto longos períodos de inatividade para o servidor. Isso acontece porque sistemas de filas, geralmente, são representados por processos randômicos (não são regulares), nos quais μ e λ representam valores médios. Como exemplo da razão da ocorrência de filas quando $\mu > \lambda$ cita-se:

- Em determinado instante de tempo podem chegar mais clientes que a capacidade de atendimento nesse momento.
- O atendimento de um cliente pode demorar mais que a média.

Considerando um sistema de filas em estado estável, a abordagem matemática pela Teoria das Filas permite avaliar a eficiência do sistema por meio da análise de suas características, representadas por variáveis randômicas e, assim, obter as medidas de desempenho. Essas variáveis estão descritas a seguir e sua localização no sistema de filas é mostrado na Figura §2.10.

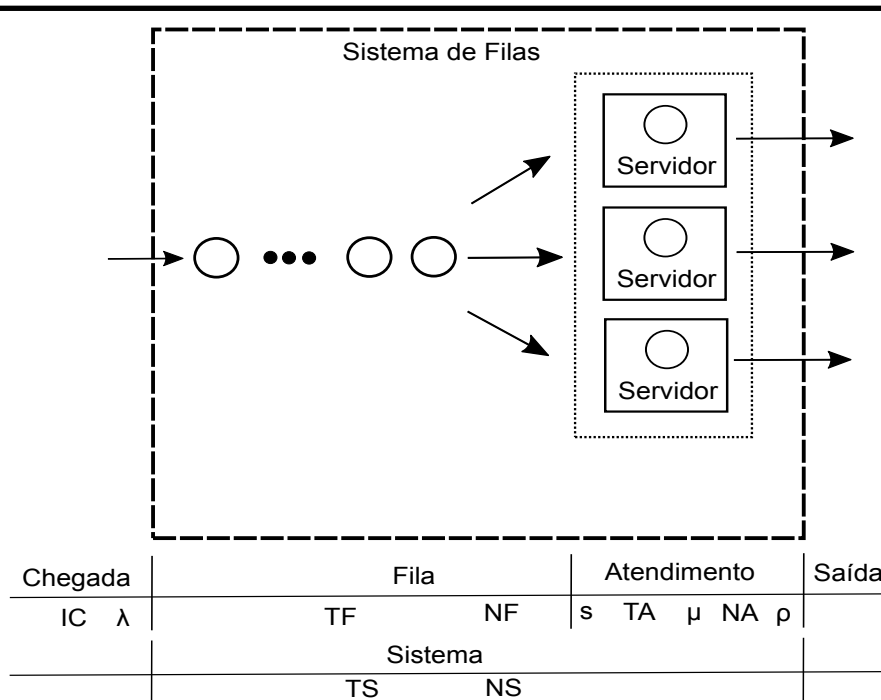


Figura 2.10: Localização das variáveis (baseado em Prado [32])

Variáveis referentes ao processo de chegada

- λ = Taxa média de chegada
- IC = Tempo entre chegadas sucessivas ou intervalo médio entre chegadas

Variáveis referentes à fila

- TF = Tempo médio de permanência na fila
- NF = Número médio de clientes na fila

Variáveis referentes ao processo de atendimento

- μ = Taxa média de atendimento
- TA = Tempo médio de atendimento ou de serviço
- NA = Número médio de clientes que estão sendo atendidos
- s = Número de atendentes

Variáveis referentes ao sistema

- TS = Tempo médio de permanência no sistema
- NS = Número médio de clientes no sistema
- P_n = Probabilidade de haver n clientes no sistema

Por definição, o intervalo médio entre duas chegadas sucessivas e o tempo médio de atendimento, podem ser descritas matematicamente por:

$$IC = \frac{1}{\lambda} \quad (2.6)$$

$$TA = \frac{1}{\mu} \quad (2.7)$$

Da mesma forma, a taxa de utilização dos atendentes para o sistema de uma fila e um atendente pode ser descrita matematicamente por :

$$\rho = \frac{\lambda}{\mu} \quad (2.8)$$

No caso de uma fila e vários atendentes a expressão matemática é :

$$\rho = \frac{\lambda}{s\mu} \quad (2.9)$$

A taxa de utilização representa a fração de tempo esperada em que cada atendente se encontra ocupado. De forma análoga, pode-se expressar matematicamente a taxa ociosa do atendente como: $1 - \rho$.

De acordo com Hillier e Lieberman [18], em um sistema que se encontre em uma condição de estado estável, o número médio de clientes no sistema e o número médio de clientes na fila é dado, respectivamente, por:

$$NS = \sum_{n=0}^{\infty} nP_n \quad (2.10)$$

2.4.10 Relações Básicas e Fórmulas de Little

Existem algumas relações óbvias entre as variáveis randômicas mostradas na Figura §2.10. O número médio de clientes no sistema é igual ao número médio de clientes na fila, somado ao o número médio de clientes que estão sendo atendidos:

$$NS = NF + NA \quad (2.11)$$

O tempo médio de permanência no sistema é igual a soma do tempo médio de permanência na fila com o do atendimento.

$$TS = TF + TA \quad (2.12)$$

O número médio de clientes que estão sendo atendidos é igual a taxa de utilização dos atendentes.

$$NA = \rho \quad (2.13)$$

John D. C. Little [28] demonstrou, em 1961, que em um sistema de filas de estado estável tem-se:

$$NS = \lambda TS \quad (2.14)$$

$$NF = \lambda TF \quad (2.15)$$

Quando o tempo médio de atendimento é constante no tempo, segue então que:

$$TS = TF + \frac{1}{\mu} \quad (2.16)$$

Isolando TS de §2.14, TF de §2.15 e substituindo em §2.16 obtêm-se

$$NS = NF + \frac{\lambda}{\mu} \quad (2.17)$$

De acordo com Hillier e Lieberman [18], as relações de Little são extremamente importantes, pois permitem que possam ser determinadas todas as quatro variáveis (NF, NS, TF e TS), que expressam as medidas de desempenho de um sistema de filas estável, assim que uma delas for encontrada analiticamente.

2.4.11 Rede de Filas

Até agora foram apresentados conceitos e definições para analisar sistemas de filas, contendo apenas uma única instalação de atendimento com um ou mais atendentes. Entretanto, sistemas de filas podem ser, na realidade, redes de filas.

Redes de filas são redes de instalações de atendimento onde clientes devem passar em algumas ou todas as instalações para receber atendimento e em frente de cada instalação pode-se formar filas. Assim, para obter as medidas de desempenho faz-se necessário estudar toda a rede [18]. Um exemplo simples de uma rede de filas é o sistema bancário, no qual muitas vezes o cliente solicita atendimento em várias instalações.

De acordo com Hillier e Lieberman [18], as pesquisas na área de rede de filas estão muito ativas em virtude de sua importância. Entretanto, esta dissertação limita-se a apresentar apenas as definições no contexto de redes de filas necessárias para o desenvolvimento da pesquisa.

Em uma rede de filas, o processo de entrada de clientes que chegam e o processo de saída de clientes que saem para certos sistemas de filas é expresso pela propriedade da equivalência [18].

Propriedade da equivalência: Suponha que em uma instalação de atendimento com s atendentes e uma fila infinita, as chegadas ocorram de acordo com um processo de Poisson com parâmetro λ e o tempo de atendimento com distribuição exponencial de mesmo parâmetro μ , para cada atendente, em que $s \mu > \lambda$. Então, a saída de estado estável dessa instalação também é um processo de Poisson, com parâmetro λ .

Independente da disciplina de fila usada, os clientes atendidos deixarão a instalação de atendimento de acordo com um processo de Poisson. Um resultado importante desta propriedade para redes de filas é que se esses clientes forem a outra instalação de atendimento, essa segunda também terá uma distribuição de Poisson. Com o tempo de atendimento de acordo com a mesma distribuição exponencial para todos atendentes, a propriedade de equivalência também será válida para essa instalação gerando, assim, uma entrada Poisson para a terceira instalação e assim, sucessivamente, até que os clientes deixem a rede de filas.

Uma rede de filas com servidores e filas infinitas em série pode ser estudada a partir das consequências da propriedade da equivalência.

Suponha que todos os clientes que entram em um sistema tenham que receber atendimento em uma série de m instalações com fila de capacidade infinita, em uma sequência fixa, de modo que a série de instalações forma um sistema de filas infinitas em série. Suponha que os clientes cheguem na primeira instalação de acordo com um processo de Poisson com parâmetro λ e que cada instalação i ($i = 1, 2, \dots, m$) tenha tempos de atendimento de acordo com a distribuição exponencial com parâmetro μ_i para os s_i atendentes, em que $s_i \mu_i > \lambda$. Decorre, então, da propriedade da equivalência que em um sistema sob condições de estado estável, cada instalação de atendimento tem uma entrada de Poisson com parâmetro λ . Portanto, as redes de filas podem ser analisadas como sistemas de filas $M/M/s$ independentes. Assim, cada instalação de atendimento é analisada independente das demais.

As medidas de desempenho das redes de filas são dadas pelas medidas de desempenho do modelo $M/M/s$, calculadas de forma independente para cada instalação de atendimento. Dessa maneira, o tempo médio, tempo médio total nas filas, número médio de clientes e o número médio total de clientes nas filas no sistema inteiro podem ser obtidos somando os valores correspondentes, obtidos nas respectivas instalações.

2.5 Resumo do Capítulo

Esse capítulo, abordou os principais conceitos e temáticas fundamentais para o desenvolvimento e entendimento desta dissertação. De acordo com a pesquisa bibliográfica realizada, a linguagem de domínio específico da tecnologia Guaraná possibilita aos engenheiros de software projetar soluções de integração com um alto nível de abstração, utilizando uma sintaxe gráfica intuitiva. Assim, blocos construtores definem as funcionalidades e a estrutura de uma solução de integração, sendo as tarefas fornecidas em caixas de ferramentas e classificadas de acordo com a sua semântica. No Guaraná as soluções de integração são executadas no motor de execução, no qual as tarefas aptas são armazenadas na *work queue* em uma fila de prioridade e a *thread* tem a função de retirar as tarefas da fila e executar. Em seguida, foram abordados os conceitos envolvidos na análise de um sistema por meio da simulação de modelos computacionais. Por último, apresentou-se os elementos e conceitos da Teoria das Filas, a qual utiliza modelos de filas para representar o sistema de filas e utiliza fórmulas para obter as medidas de desempenho. A maioria dos modelos de filas parte do pressuposto de uma instalação de atendimento com um ou mais atendentes. No entanto, sistemas de filas na realidade podem ser redes de filas, na qual o cliente poderá ser atendido por uma série de instalações de serviços.

Capítulo 3

Trabalhos Relacionados

*Que o teu trabalho seja perfeito para que,
mesmo depois da tua morte, ele permaneça.*

Leonardo da Vinci, Polímata (1452-1519)

Na literatura não encontrou-se muitos trabalhos relacionados com as técnicas utilizadas nesta pesquisa. Nos trabalhos identificados observou-se que a abordagem de sistemas por meio da Teoria das Filas e simulação ocorre em diferentes sistemas de eventos discretos. A Seção [§3.1](#) apresenta os trabalhos que utilizaram a Teoria das Filas e a simulação para obter as medidas de desempenho do sistema bancário, ambiente e-Business, tráfego de rede, terminal marítimo e serviço de correio eletrônico. A Seção [§3.2](#) apresenta o resumo deste capítulo.

3.1 Teoria das Filas e Simulação

Uma solução de integração pode ser caracterizada como um sistema cujo modelo é classificado como um sistema de eventos discretos. Diversos trabalhos utilizam a Teoria das Filas e a simulação de modelos para avaliar a eficiência de sistemas de eventos discretos. No entanto, não identificou-se, na literatura, trabalhos que analisaram soluções de integração de aplicações empresariais por meio da simulação e/ou Teoria das Filas. Assim, este trabalho é uma contribuição importante no desenvolvimento de pesquisas na área de EAI, à medida que pode ser considerado um passo inicial nessa abordagem. Considerando esse contexto, a pesquisa deteve-se em trabalhos que utilizaram a Teoria das Filas e a simulação para estudar e analisar medidas de desempenho de sistemas de eventos discretos.

Abensur et al. [1] propõe, oferecer condições para a formulação de estratégias consistentes para o autoatendimento bancário brasileiro, a partir da avaliação das variáveis de desempenho obtidas pela aplicação da teoria das filas sobre o caso estudado. As informações registradas no sistema de informática do banco foram utilizadas pelos autores, para estimar o fluxo de chegada dos usuários atendidos pelos caixas eletrônicos. Para estimar o tempo médio de ocupação dos equipamentos de autoatendimento, os autores utilizaram o tempo individual de processamento das principais transações utilizadas pelos usuários, acrescido de um tempo devido à permanência ociosa dos usuários em frente aos equipamentos.

Os autores desenvolveram um simulador de filas de autoatendimento em Visual Basic para Excel 2002. O modelo de simulação construído pelos autores está baseado na gestão da fila de autoatendimento. O modelo simula o atendimento para um ambiente com até cinco equipamentos, considerando um dia de trabalho de 24 horas. Durante intervalos de tempo de 15 segundos foram tiradas fotos do sistema para analisar a situação das variáveis de desempenho.

Os resultados das simulações, foram avaliadas para dois grupos distintos de usuários: com alta e baixa habilidade de uso de caixas eletrônicos, considerando o tempo médio de ocupação dos equipamentos e agrupadas em quatro categorias. De acordo, com o trabalho, o modelo de simulação baseado na Teoria das Filas mostrou que os ganhos de produtividade com a introdução de máquinas especializadas seriam significativos, melhorando a prestação dos serviços oferecidos aos clientes com sensíveis reduções do tempo médio de fila.

Changfu e Zhenyu [5] buscam, estabelecer modelos matemáticos e de simulação para analisar e avaliar o impacto da eficiência provocado pela variação de transações de requisição no sistema de filas do ambiente *e-Business*.

Segundo os autores, com o desenvolvimento da Internet nos últimos anos, mais e mais empresas têm adotado a tecnologia de *e-Business* para reestruturar seus processos de negócio. Entre eles, o pedido de transação e o processo de confirmação geralmente é informatizado em um ambiente de *e-Business*. Os clientes, bem como os prestadores de serviços, podem facilmente realizar pedidos de transação e confirmar atividades em seus próprios escritórios por intermédio de ligação EDI ou sistema *on-line*, sem a necessidade de documentação em papel.

De acordo com os autores, tradicionalmente os clientes devem submeter os formulários de pedidos de papel e esperar por uma confirmação dos representantes do serviço ao cliente, para realizar transações em um local específico do negócio. Um novo cliente que chega escolhe sempre a fila menor com seus formulários de pedidos de papel para esperar pelo serviço, podendo mudar, espontaneamente, para qualquer fila menor.

Em um ambiente *e-Business*, ao invés de ir para o site do serviço real com formulários de pedido de transação em papel, os clientes enviam mensagens de solicitação EDI para os prestadores de serviços correspondentes através do sistema EDI. As mensagens de transação EDI são recebidas por um sistema de pré-tratamento EDI de prestadores de serviços e, após, são colocadas numa fila para confirmação. Quando o representante de atendimento concluir uma transação, ele irá acionar um pedido de movimentação para a próxima transação. Se houver pelo menos um pedido de suspensão na fila de espera eletrônica, o sistema *e-Business* distribuirá o pedido de transação EDI para o representante baseada na regra *First-In, First-Out*. Em seguida, no ambiente *e-Business*, a manipulação do pedido de transação no sistema de filas foi alterada de um modelo com único servidor (M/M/1) para um multi-servidor (M/M/n) do sistema de filas e, ao mesmo tempo, a entrada e saída do tempo de documentação foi salvo devido ao compartilhamento de dados EDI.

Os autores, por meio da manipulação matemática das fórmulas da Teoria das Filas, afirmam que o ajuste estrutural na manipulação do pedido de transação no sistema de filas no ambiente de *e-Business* pode esgotar o tempo médio de espera. Indicam também que a simplificação da documentação de entrada e saída pode diminuir o tempo médio de espera e, com isso, espera-se diminuir o tempo de espera significativamente.

Para testar as conclusões descritas, os autores estudaram um negócio de reserva de exportação. Reserva de exportação é parte do negócio de transporte marítimo e para efetuar uma transação de reserva, o expedidor deverá submeter os formulários de solicitação de reserva de espaço à transportadora e obter confirmação com informações de envio da mesma.

Baseados nos dados estatísticos de reserva de uma companhia de agenciamento marítimo, os autores estabeleceram modelos de simulação do sistema de filas por meio do MATLAB e Excel, para a reserva de papel e sistemas de filas de reserva eletrônica, considerando e não considerando o tempo de documentação de entrada e saída.

De acordo com Changfu e Zhenyu [5], a conclusão teórica e os resultados da simulação mostram que o ajuste estrutural no sistema de filas no ambiente *e-Business* pode reduzir o tempo médio de espera do cliente, juntamente com a simplicidade de entrada e saída de documentos e, em seguida, a experiência de serviço ao cliente pode ser aperfeiçoada.

Kamali et al. [22] propõe , avaliar e estimar o monitoramento do tráfego de rede baseado na teoria das filas. Os autores afirmam que em um ambiente heterogêneo, o monitoramento do tráfego de rede é necessário para a avaliação da eficiência e confiança das constantes operações de rede. Examinam o desempenho e previsão de gerenciamento de tráfego de rede, dando uma sugestão para o controle do desempenho do tráfego com base na teoria das filas, simulando o ambiente de rede LAN, que pode ser um modelo para o mundo real (ambiente de rede LAN).

Segundo os autores, o monitoramento do tráfego de rede é um dos muitos aspectos de gerenciamento de rede e segurança que pode ter o maior número de soluções de desempenho e *lazing network* (rede preguiçosa). Algumas das vantagens do monitoramento da rede são: a estimativa da qualidade de serviço, o planejamento da largura de banda e o número de clientes da rede. A previsão do tráfego de rede pode ser baseada na sensibilidade de taxa de atraso no tráfego ou na alocação dinâmica de largura de banda sobre o tráfego.

De acordo com os autores, a engenharia de tele tráfego é a aplicação da teoria da engenharia de tráfego para as telecomunicações. Os engenheiros de tele tráfego utilizam conhecimentos básicos de estatística, teoria das filas, natureza do tráfego, modelos práticos, medições e simulações para fazer previsões e planejar redes de telecomunicações a um custo mínimo total.

Na rede de comunicações, a camada física e outras são exemplos de filas simples. Este processo de comunicações de rede pode ser considerado

como um modelo baseado na teoria de filas, sendo que o tipo de serviço transferido pode implementar um modelo M/M/1 de filas.

Os autores utilizaram variáveis e fórmulas da teoria das filas para a obtenção de parâmetros, tais como a capacidade de largura de banda, o tamanho do pacote e as distâncias entre dois computadores que determinam o tráfego de rede. Em seguida, realizaram a simulação do ambiente de rede heterogênea e usaram vários testes para o controle de parâmetros e valores.

Os autores utilizaram a ferramenta *Colasoft Capsa* para realizar a simulação de tráfego, considerando dois parâmetros, o número de usuários e o tamanho do pacote. Os resultados da simulação mostram que o modelo de simulação pode estimar e prever o tráfego de rede utilizada na rede real LAN.

De acordo com Kamali et al. [22], pode-se medir a eficiência do serviço heterogêneo e das diferentes tecnologias que chegam ao servidor de rede baseado em um modelo de simulação. O modelo de simulação para obter as informações sobre o desempenho de vários serviços é confiável por meio do laboratório e com base em testes.

Segundo os autores, pode-se monitorar a rede para gerenciar de forma eficiente. Para isso, a rede deve acompanhar e analisar a taxa de tráfego de rede que pode ter um papel importante no teste de estrutura e problemas de rede.

Camelo et al. [3] propõe, por meio da Teoria das Filas e da simulação, analisar características de atendimento aos navios no embarque de minério de ferro e manganês no Terminal Marítimo de Ponta da Madeira. A Teoria das Filas foi utilizada, no trabalho, para analisar o número médio de navios na fila e no sistema e o tempo médio de permanência de um navio na fila e no sistema no Píer I e Píer III e a técnica de simulação para simular a operação do Píer IV, que estava em fase de implantação.

Segundo os autores o embarque de minério de ferro e manganês é realizado em três berços, um no Píer I, com capacidade de carregamento de 16.000 ton/h e dois no Píer III (nominados de Norte e Sul), com capacidade de 8.000 ton/h cada. O terminal possui três canais de serviços com características de fila em paralelo, pois cada berço apresenta características únicas, sendo que o trabalho analisou, caso a caso a capacidade de atendimento aos navios.

Os autores coletaram dados das atracções e desatracções do Terminal Marítimo de Ponta da Madeira, durante os meses de junho, julho e agosto de 2009 e a partir destes dados conseguiram obter os valores das variáveis ritmo médio de chegada, intervalo médio entre chegadas, ritmo médio

de atendimento e tempo médio de atendimento do Píer I, Píer III – Norte e Píer III – Sul.

Com base nos dados obtidos, os autores definiram o sistema de filas como um modelo $M/M/1/\infty/\infty/FIFO$ e aplicaram os valores das variáveis anteriormente citadas nas fórmulas das variáveis utilizadas como medidas de desempenho da Teoria das filas, para observar o desempenho do sistema existente para carregamento de minério de ferro e manganês.

De acordo com Camelo et al. [3], os resultados analíticos obtidos evidenciam a eficiência do terminal, que apresenta uma taxa média de ocupação (ρ) dos berços superior a 80 %. No entanto, o terminal opera no limite de sua capacidade operacional, confirmado pela simulação computacional, que para o período de um ano, 661 navios passaram pelo sistema, resultando uma média de 55 navios por mês. Desta forma, pretende-se construir um novo píer para carregamento de minério de ferro e manganês, o Píer IV, para garantir a produção e transporte dos minerais, a médio e longo prazos.

Para simular a operação do novo píer para carregamento de minério de ferro e manganês, os autores utilizaram o software Arena 12.0, que permite o uso de diversos tipos de distribuição de frequência. Segundo os autores, o modelo $M/M/1$ não dimensiona corretamente um sistema de filas. Assim, para melhor dimensionar o sistema em análise, utilizou-se o modelo $M/E_k/1/\infty/\infty/FIFO$

Para simular a operação do Píer IV, os autores utilizaram os dados coletados de atracações e desatracações do Píer I, já que este possui características similares ao píer que será construído e a partir da análise dos resultados concluíram que para carregar o previsto de 53 navios por mês, terá que operar um máximo de 75 % da sua capacidade para navios grandes. Os resultados da simulação das principais medidas de desempenho do sistema em análise mostram que com a inclusão do Píer IV, haverá no Terminal uma redução no tamanho médio da fila de 4,2 para 3,9 navios e, no tempo médio na fila, de 7,1 para 5,8 navios, mantendo assim sua eficiência operacional, apresentando baixa ociosidade e taxas médias de ocupação superiores aos 80%.

Doy et al. [8] propõe estudar por meio da simulação de um modelo de filas, o serviço de correio eletrônico (*e-mail*) da rede de computadores do Instituto de Matemática e Estatística da Universidade de São Paulo, para identificar características estatísticas e avaliar seu desempenho.

Para a realização do estudo, os autores coletaram dados em um dia de semana, considerado típico, de março de 2000 e analisaram os dados referentes

às chegadas de mensagens no período identificado como de maior frequência de mensagens das 9 até 19. Segundo os autores, dado o grande volume diário de mensagens, as informações de um dia típico são suficientes para as tarefas de estimação envolvidas na obtenção do modelo a ser simulado.

De acordo com Doy et al. [8], uma mensagem de email tem sempre remetente e destinatário. Desta forma, existem 4 possíveis direções de tráfego de : LL, LR, RL e RR, com L e R indicando o usuário local e remoto, respectivamente. O tempo de permanência de uma mensagem no sistema depende de sua origem, destino e também de ter sido entregue (sucesso) ou devolvida (falha). As mensagens resultantes em sucesso tendem a demorar mais no sistema, pois são enviadas ao destinatário.

Doy et al. [8] afirmam que o serviço de correio eletrônico apresenta dificuldades para um estudo analítico. Para simular o serviço de correio eletrônico, os autores estabeleceram um sistema de filas com um Centro de Recepção e dois Centros de Serviço. No Centro de Recepção, a mensagem é classificada segundo a origem, o destino e o resultado de seu processamento. Nesse centro, são também desdobradas as mensagens originalmente enviadas a múltiplos usuários. Após passar pela recepção, a mensagem é enviada a um dos Centros de Serviço, de acordo com seu destino. Assim, todas as mensagens com destino local são processadas no Centro de Serviço 1 e as com destino remoto vão ao Centro 2, que possuem 10 e 20 servidores, respectivamente. O serviço em cada centro é, por ordem de chegada, a partir de uma fila única, com capacidade ilimitada de espera.

Os autores realizaram uma análise dos dados disponíveis do fluxo de mensagens, para obter informações sobre o processo de chegadas, o tempo de serviço e a proporção de sucessos na entrega, além de outras características para propor um modelo de filas. O sistema de filas foi simulado utilizando o software Arena versão 3.01 para avaliar os tempos de espera das mensagens nesse sistema. O tempo simulado correspondeu a um total de 10 horas de funcionamento do sistema (9 às 19 horas) e foi precedido de um aquecimento de 604 800 segundos (7 dias). Os resultados foram baseados em 30 réplicas.

Segundo os autores, os resultados mostraram que o sistema tem um bom espaço para absorver crescimento em várias frentes, mostrando-se praticamente insensível à duplicação no número de chegadas de mensagens. Por outro lado, as paradas no sistema têm um efeito importante nas medidas estudadas. O efeito dos desdobramentos foi também avaliado. Eles se originam em listas de interesse e têm um grande potencial multiplicador no tráfego de mensagens pela rede. No sistema estudado, o acréscimo aplicado ao tamanho das listas pode ser, em grande parte, absorvido pela capacidade existente no sistema.

3.2 Resumo do Capítulo

Neste capítulo, foram apresentados alguns trabalhos encontrados na literatura que utilizaram a Teoria das Filas e a simulação para avaliar a eficiência de sistemas de eventos discretos. Todos os trabalhos consideram as duas técnicas eficientes para obter as medidas de desempenho de sistemas de filas. Abensur et al. [1] utilizam a Teoria das Filas e a simulação para obter as medidas de desempenho dos serviços bancários oferecidos por caixas automáticos e, assim, oferecer condições para à formulação e avaliação de estratégias viáveis para a gestão do autoatendimento bancário. Changfu e Zhenyu [5] estabelecem, por meio da Teoria das Filas, modelos matemáticos e de simulação para analisar e avaliar o impacto do ajuste estrutural no sistema de filas do ambiente *e-Business*. Kamali et al. [22] propõe, baseado na Teoria das Filas, avaliar o monitoramento do tráfego de rede por meio de um modelo de simulação, medir a eficiência do serviço heterogêneo e das diferentes tecnologias que chegam ao servidor de rede. Camelo et al. [3] utilizam a Teoria das Filas para analisar características de atendimento aos navios no embarque de minério de ferro e manganês e obter as medidas de desempenho do Píer I e Píer III. A simulação foi utilizada para simular a operação do Píer IV, que será construído. Doy et al. [8] estudaram o serviço de correio eletrônico (e-mail) de uma rede de computadores por meio da simulação de um modelo de filas, para avaliar seu desempenho

Capítulo 4

Caso de Estudo

*O começo da sabedoria é encontrado na dúvida;
duvidando começamos a questionar,
e procurando podemos achar a verdade.*

Pierre Abélard, Teólogo e filósofo francês (1079-1142)

Uma abordagem que permita a análise do comportamento e a descoberta de possíveis gargalos de desempenho, ainda na fase de projeto das soluções de integração, a partir dos seus modelos conceituais, sem que as mesmas tenham de ser implementadas, requer o conhecimento detalhado sobre os modelos. Nesse sentido, este capítulo apresenta o caso de estudo desta dissertação, um modelo conceitual para a solução de integração de um problema real, projetado na tecnologia Guaraná. A Seção §4.1 apresenta a descrição do ecossistema de software da solução de integração. Na Seção §4.2 é realizada uma descrição detalhada do modelo conceitual, considerando o fluxo de trabalho das mensagens na solução. Por fim, a Seção §4.3 apresenta o resumo deste capítulo.

4.1 Ecossistema de Software

O problema de integração analisado nesta dissertação consiste em um problema real de uma Central Telefônica. A solução proposta tem como objetivo melhorar a funcionalidade da aplicação Central Telefônica e automatizar o faturamento das ligações pessoais realizadas por funcionários usando os telefones da Central Telefônica da Universidade Regional do Noroeste do

Estado do Rio Grande do Sul (Unijuí). A solução de integração possibilita que as ligações realizadas pelo funcionário identificadas por um código de acesso pessoal sejam deduzidas automaticamente na folha de pagamento.

A Figura §4.1, representa o modelo conceitual projetado com a tecnologia Guaraná para resolver o problema de integração Solução de Gerenciamento de Chamadas de Telefone da Universidade Unijuí, proposto por Frantz [12].

A solução integra cinco aplicações: Central Telefônica, Recursos Humanos, Folha de Pagamento, Servidor de E-mail e Serviço de Mensagens. Cada aplicação é executada em uma plataforma diferente; Recursos Humanos e Folha de Pagamento são sistemas legados desenvolvidos pela Universidade e as outras aplicações foram adquiridas. Além disso, o Servidor de E-mail provê interfaces POP3 e SMTP, e as outras aplicações foram projetadas sem levar em conta a possibilidade de integração. A Central Telefônica registra todas as chamadas que cada funcionário realiza dos telefones da Universidade; isso é possível por que cada funcionário tem um código de acesso pessoal, que deve ser digitado antes de discar o número que deseja ligar. Este código é utilizado para correlacionar as ligações com as informações da aplicação de Recursos Humanos e Folha de Pagamento. A aplicação de Recursos Humanos fornece informações pessoais sobre os funcionários e, todos os meses, a Folha de Pagamento calcula o salário de todos os funcionários, incluindo as deduções. O Servidor de E-mail e o Serviço de Mensagens executam, respectivamente, o e-mail da Universidade e serviços do sistema de mensagens, que são utilizados com a finalidade de notificação.

Todas as chamadas registradas pela Central Telefônica, exceto as chamadas gratuitas, devem ser transformadas em débito no sistema de Folha de Pagamento. Os funcionários podem ser notificados por e-mail ou por mensagem de texto sobre as chamadas realizadas, para que estejam cientes da dedução na folha de pagamento. A hipótese levantada é que a informação registrada na Central Telefônica não contenha outras informações, além dos dados das ligações e os códigos de acesso pessoal. É a aplicação de Recursos Humanos que fornece outras informações necessárias.

4.2 Modelo Conceitual da Solução de Integração

A solução de integração projetada no Guaraná é composta de um processo de orquestração que coordena, de forma exógena, as aplicações envolvidas na solução, conforme representado na Figura §4.1. Algumas portas usam arquivos de texto para comunicar-se, indiretamente, com as aplicações da Central Telefônica, Folha de Pagamento e Serviço de Mensagens. O

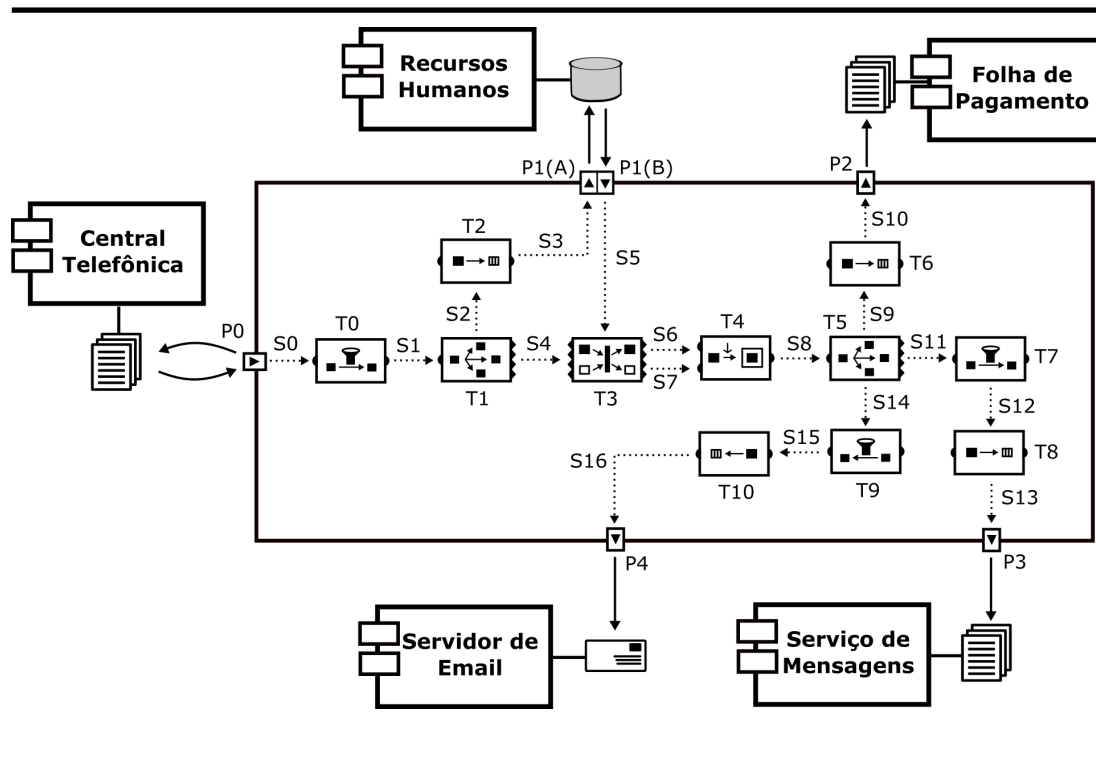


Figura 4.1: Modelo conceitual

acesso ao banco de dados da aplicação de Recursos Humanos ocorre por meio da combinação de uma porta de entrada e outra de saída. Uma tarefa *translator* é utilizada para traduzir o conteúdo da mensagem para um novo formato, utilizado pela aplicação integrada.

O fluxo de trabalho começa na porta de entrada P0 que, periodicamente, pesquisa e verifica a Central Telefônica, para encontrar novas chamadas. Cada ligação resulta em uma mensagem que é introduzida no processo de orquestração e contém as informações do telefonema. Dentro do processo, as mensagens são encaminhadas à tarefa T0, por meio do slot S0. Sempre que uma tarefa estiver executando as mensagens que chegam, aguardam nos slots formando uma fila, sendo posteriormente selecionadas para o processamento pela disciplina FIFO.

A tarefa T0 (*Filter*) filtra e descarta as mensagens de ligação gratuita e encaminha somente aquelas que tem custo para o slot S1, para serem executadas pela tarefa T1 (*Replicator*). Essa tarefa cria uma cópia da mensagem, que é encaminhada ao slot S2 para ser processada pela tarefa T2 (*Translator*)

que traduz o conteúdo da mensagem para o formato da aplicação Recursos Humanos, sendo encaminhada ao *slot* S3 para consultar a aplicação, por meio da porta P1.

As mensagens retornam por meio da porta P1 e o *slot* S5 a encaminha para a tarefa T3 (*Correlator*), que analisa mensagens de entrada e produz como saída um conjunto de dados correlacionados. Os *slots* S6 e S7 conduzem as mensagens correlacionadas para a tarefa T4 (*Context-based Content Enricher*), que adiciona na outra cópia correlacionada as informações que retornaram da aplicação Recursos Humanos. Em seguida, a tarefa T5 cria três cópias da mensagem enriquecida e envia para as aplicações de Folha de Pagamento, Servidor de E-mail e Serviço de Mensagens. A porta de saída P2 informa as ordens de débito na folha de pagamento e as cópias enviadas ao Serviço de Mensagens e Servidor de E-mail serão filtradas, caso as portas de saída P3 e P4 recebam mensagens sem informações suficientes dos destinatários, por exemplo número de telefone e endereço de e-mail.

4.3 Resumo do Capítulo

Nesse capítulo, foi apresentado o modelo conceitual de integração, projetado na tecnologia Guaraná, para resolver um problema real da Central Telefônica da Unijuí. De acordo com o estudo realizado, o objetivo da solução é automatizar a cobrança das ligações telefônicas realizadas pelos funcionários da Unijuí, sendo que as cinco aplicações, Central Telefônica, Recursos Humanos, Folha de Pagamento, Servidor de E-mail e Serviço de Mensagens, que compõe o ecossistema de software, são integradas através da tecnologia Guaraná. Por último, foi realizado um estudo detalhado do fluxo de trabalho das mensagens, durante a execução da solução de integração.

Capítulo 5

Modelagem e Simulação

No meio de qualquer dificuldade encontra-se a oportunidade.

Albert Einstein, físico teórico alemão (1879-1955)

Embora normalmente, a abordagem utilizada pelos engenheiros de software para análise do comportamento frente a cenários críticos de funcionamento consista na construção e execução da solução de integração, a abordagem que permite analisar o comportamento e identificar possíveis gargalos de desempenho, ainda na fase de projeto, a partir do modelo conceitual, poderá reduzir custos, riscos e tempo de desenvolvimento. Nesse sentido, este capítulo caracteriza uma solução de integração como um sistema de eventos discretos, possibilitando a identificação das variáveis que descrevem a solução e a simulação do modelo conceitual. O presente capítulo está organizado da seguinte forma: a Seção §5.1 aborda a caracterização de uma solução de integração e as variáveis que descrevem o comportamento da solução; a Seção §5.2 aborda a dedução das fórmulas matemáticas para obter as medidas de desempenho de um modelo M/M/1; na Seção §5.3 é realizada uma descrição do software e do modelo de simulação; na Seção §5.4 são descritos os cenários e realizada uma análise dos resultados da simulação. Por fim, a Seção §5.5, resume o capítulo.

5.1 Equivalência do Guaraná com o Sistema de Filas

Para caracterizar a solução de integração como um sistema de eventos discretos, esta dissertação demonstra a equivalência dos elementos do Guaraná (mensagem, *slot* e tarefa) com os elementos de um sistema de filas (cliente, fila e servidor).

A caracterização de uma solução de integração como um sistema de eventos discretos possibilita, por meio da simulação, prever o comportamento futuro do sistema, evitando os efeitos que seriam causados por alterações na solução real implementada. Assim, tem-se a possibilidade de construir hipóteses baseadas nas observações realizadas a partir dos modelos de simulação, como também o controle do tempo. Em outras palavras, o tempo pode ser expandido ou comprimido, permitindo a reprodução de fenômenos de maneira rápida ou lenta e, desta maneira, estudar e entender o comportamento da solução.

Para caracterizar a solução de integração como um sistema de eventos discretos, este trabalho utiliza a Teoria das Filas. Na Teoria das Filas, os sistemas são representados por modelos, de acordo com suas características. No entanto, os sistemas possuem elementos comuns a todos que fazem parte do processo básico. Os elementos que compõem o processo são clientes que necessitam de atendimento 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.

A Figura §5.1(a) ilustra, genericamente, um processo de um sistema de serviço que pode ser representado por um modelo de filas. Esse processo tem estrutura semelhante à do modelo conceitual de uma solução de integração projetada no Guaraná. Nessa estrutura os clientes chegam, formam uma fila e esperam um determinado tempo para serem atendidos. A Figura §5.1(b), ilustra uma parte extraída de uma solução de integração projetada no Guaraná, para demonstrar a equivalência entre os elementos da Teoria das Filas e os de uma solução de integração. As mensagens representam os clientes, os *slots* as filas e as tarefas os serviços.

Na solução de integração, as mensagens chegam e quando uma tarefa está executando, aguardam no *slot*. As mensagens serão selecionadas para serem executadas pelas tarefas seguindo uma determinada disciplina. Após o processamento, saem da tarefa, seguindo o fluxo de integração.

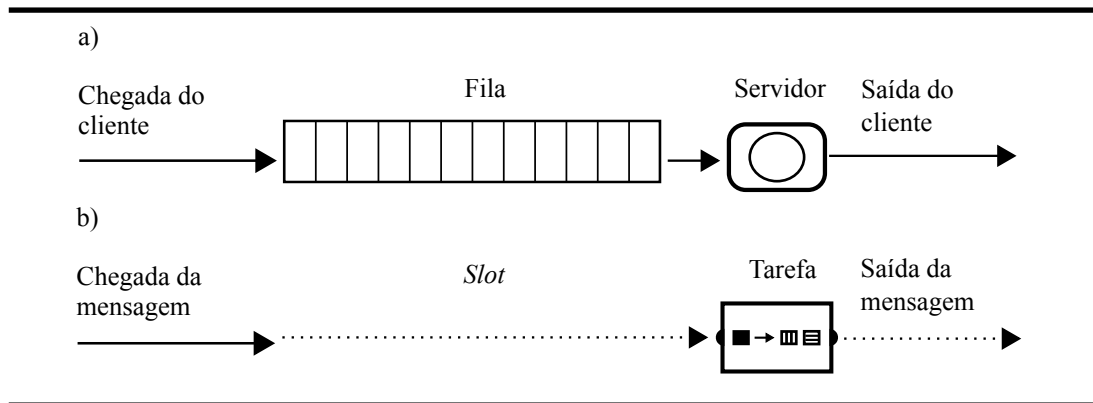


Figura 5.1: Modelo de um sistema de serviço (a) Guaraná DSL (b)

Outra característica comum entre uma solução de integração e um sistema de eventos discretos é a relação entre seus elementos e estrutura de funcionamento. A Figura 5.2 representa a relação entre os elementos conceituais de um sistema de eventos discretos.

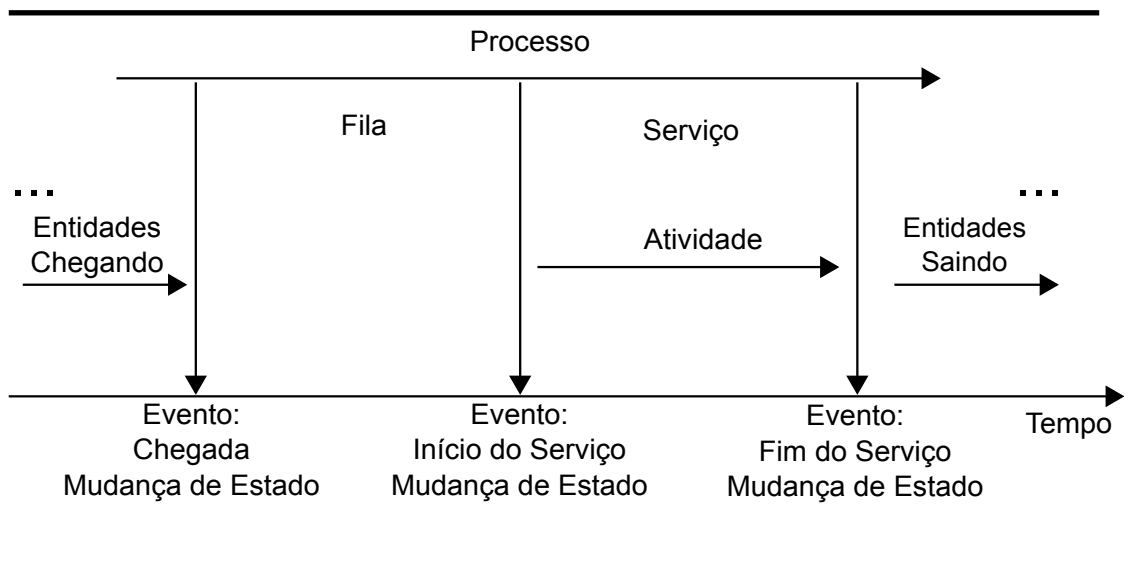


Figura 5.2: Relação entre os elementos que compõe um sistema discreto

A Figura 5.2 mostra entidades chegando ao sistema. Um processo consiste nas ações realizadas sobre as entidades ao longo do sistema. Uma atividade corresponde à execução de um serviço em um período predetermi-

nado. Uma atividade ocorre entre dois eventos sendo que a ocorrência de eventos muda o estado do sistema.

Essa relação é semelhante a dos elementos de uma solução de integração projetada na tecnologia Guaraná. Em uma solução de integração, as entidades são representadas por mensagens que chegam à solução e permitem a interação com o sistema. Um processo é a sequência de eventos, em ordem cronológica, que descrevem o ciclo de vida das mensagens. Uma atividade é a ação que uma mensagem pratica ou recebe durante o período de tempo entre a ocorrência de dois eventos. A execução de uma tarefa é uma atividade. A ocorrência de um evento pode ser caracterizada como a chegada de uma mensagem para ser processada por uma tarefa. Toda vez que a tarefa processa uma mensagem, altera o estado do sistema.

A disciplina do *slot* é outra característica importante em uma solução de integração, porque define a ordem de processamento das mensagens. Várias disciplinas podem ser utilizadas, tais como *First-In, First-Out*, *Last-In, First-Out*, serviço por ordem de prioridade, serviço randômico, entre outras. Entretanto, para o desenvolvimento desta pesquisa, assumiu-se que a disciplina do *slot* é FIFO (*First-In, First-Out*), dado que essa disciplina é a que predomina nas soluções de integração de aplicações empresariais.

O tamanho das filas de espera em um sistema discreto pode ser limitado ou ilimitado. Em uma solução de integração projetada no Guaraná, o tamanho da fila é ilimitado. Isso significa que o *slot* pode armazenar um número infinito de mensagens.

5.1.1 Variáveis da Solução de Integração

Um modelo de simulação de eventos discretos permite reproduzir variáveis relacionadas ao desempenho e à maneira como interagem entre si e com os outros elementos, facilitando a compreensão do comportamento do sistema. Segundo Fogliatti e Mattos [10], a utilização da Teoria das Filas permite avaliar a eficiência de um sistema por meio da análise de suas características. Essas características mudam ao longo do tempo e, por isso, são representadas por variáveis randômicas, cujos valores são utilizados como medidas de desempenho do sistema. Assim, para as principais variáveis têm-se um valor médio e uma distribuição de probabilidade.

O modelo conceitual da solução de integração, na realidade, é equivalente a uma rede de filas. As mensagens aguardam no slot (fila) para serem executadas pelas tarefas (instalações de atendimento). Decorre da propriedade de equivalência apresentada na Seção §2.4.11, que as redes de filas

podem ser analisadas como sistemas de filas M/M/s independentes. Assim, as medidas de desempenho da solução de integração, podem ser obtidas ao calcular as variáveis de forma independente para cada instalação de atendimento (slot-tarefa).

A especificação das variáveis em uma solução de integração inicia no processo de chegada das mensagens. Como não são conhecidos o número de chegadas de mensagens e os instantes de tempo em que elas acontecem, tem-se um comportamento randômico, constituindo um processo estocástico.

Para caracterizar corretamente o processo de chegada utiliza-se uma distribuição de probabilidade, sendo necessária a especificação de um parâmetro denominado taxa de chegadas, que representa o número médio de mensagens que chegam à solução por unidade de tempo. Nesse contexto, a taxa média de chegada de mensagens é uma variável randômica essencial. Para quantificar essa variável, na Teoria das Filas, utiliza-se, o símbolo λ e IC para o intervalo médio entre chegadas das mensagens.

As mensagens na solução de integração são processadas nas tarefas. Já as tarefas são executadas pelas *threads*. Baseado na Teoria das Filas, o processo de execução de uma mensagem na tarefa pode ser quantificado por uma variável randômica e acontece de forma análoga ao processo de chegada. Dessa forma, utiliza-se também uma distribuição de probabilidade. O símbolo μ indica taxa média de execução e TA o tempo médio de execução de uma tarefa. Pode-se, ainda, calcular o número médio de mensagens processadas na tarefa (NA), a taxa de utilização (ρ) e a taxa ociosa da tarefa ($1 - \rho$). Na Teoria das Filas, a quantidade de servidores é uma variável importante. Na solução de integração a quantidade de *threads* (s) utilizadas para executar uma tarefa é análoga a essa variável.

A análise da eficiência de uma solução de integração por meio da Teoria das Filas, deve também contemplar o estudo das características de um importante componente da solução, o *slot*. Uma solução possui vários *slots* e, quando algum deles assume valores além dos adequados, isso indica que a solução pode apresentar gargalos de desempenho.

Para identificar possíveis gargalos de desempenho, as variáveis do *slot* estão relacionadas à quantidade de mensagens e ao tempo de espera. As variáveis são, respectivamente, número médio de mensagens no *slot* (NF) e tempo médio de permanência no *slot* (TF). A solução de integração vista como um sistema também possui variáveis importantes que permitem observar e estudar o comportamento da solução. Essas variáveis são o número

médio de mensagens (NS) e o tempo médio de permanência das mensagens na solução (TS).

A Figura 5.3 apresenta a localização das variáveis identificadas em uma solução de integração.

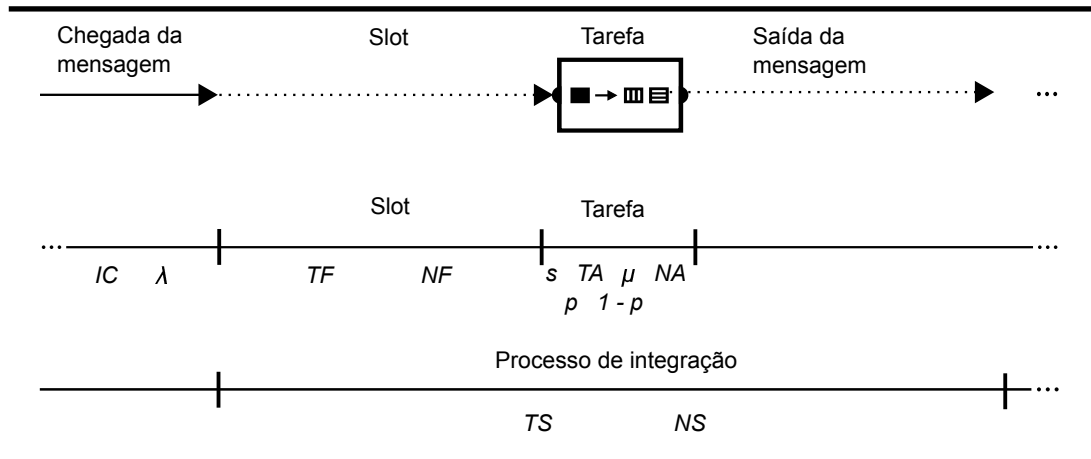


Figura 5.3: Localização das variáveis em uma solução de integração

5.2 Modelo Matemático

A Teoria das Filas permite avaliar a eficiência de um sistema por meio da análise de suas características e medidas de desempenho. Para representar os diversos tipos de sistema de filas que surgem, na prática, a Teoria das Filas utiliza modelos de filas. Os valores das variáveis utilizadas como medidas de desempenho são obtidas por meio das fórmulas do modelo e os resultados indicam como o sistema deve funcionar e operar, em uma série de circunstâncias ou cenários. A Seção 5.2.1 identificou as variáveis cujos valores podem ser utilizados como medidas de desempenho para analisar o comportamento e identificar possíveis gargalos de desempenho em uma solução de integração.

O modelo M/M/1 é o modelo de fila mais empregado. Nesse modelo, os tempos entre chegadas sucessivas e os tempos de atendimento seguem a distribuição exponencial, há um único servidor, a capacidade da fila e tamanho da população é infinita e a disciplina é FIFO. Essa pesquisa consiste em um passo inicial na abordagem de uma solução de integração, por meio da simulação de um modelo conceitual, utilizando a Teoria das Filas. Nesse sentido,

observando o processo de integração da solução e realizando algumas simplificações, identificou-se que uma solução pode ser representada por um modelo de filas M/M/1.

Os modelos de filas mais usados na Teoria das Filas baseiam-se no processo de nascimento e morte, um tipo especial de cadeia de Markov de tempo contínuo. Assim, a seguir aborda-se o processo de nascimento e morte e a dedução das fórmulas das medidas de desempenho de sistemas de filas do tipo M/M/1.

5.2.1 Processo de Nascimento e Morte

Segundo Hillier e Lieberman [18], os modelos de filas elementares partem da hipótese de que as entradas e saídas de clientes do sistema de filas ocorram de acordo com o processo de nascimento e morte. No contexto da Teoria das Filas, o termo nascimento corresponde à chegada de um cliente ao sistema de filas e o termo morte, à saída de um cliente atendido.

O processo de nascimento e morte descreve probabilisticamente como o estado do sistema muda à medida que t aumenta. Isso significa que nascimentos e mortes ocorrem aleatoriamente, e as suas taxas médias de ocorrência dependem apenas do estado atual do sistema (independentem do tempo) [18].

De acordo com Hillier e Lieberman [18], as hipóteses do processo de nascimento e morte são:

Hipótese 1: Dado estado n do sistema, a distribuição de probabilidade do intervalo de tempo do instante presente até o próximo nascimento (chegada) é exponencial, com parâmetro λ_n ($n = 0, 1, 2, \dots$).

Hipótese 2: Dado estado n do sistema, a distribuição de probabilidade do intervalo de tempo do instante presente até a próxima morte (término do atendimento) é exponencial, com parâmetro μ_n ($n = 0, 1, 2, \dots$).

Hipótese 3: As variáveis aleatórias do intervalo de tempo do instante presente até o próximo nascimento e do intervalo de tempo do instante presente até a próxima morte são independentes. As únicas mudanças permitidas, a partir de um determinado estado n do processo de nascimento e morte, é para os estados vizinhos. Assim, a próxima transição no estado do processo é

$$n \Rightarrow n + 1, \quad \text{que representa um nascimento}$$

ou então

$$n \Rightarrow n - 1, \quad \text{que representa uma morte}$$

Essas transições se processam com taxas λ_n e μ_n que em um sistema de filas representam, respectivamente, a taxa média de chegada de clientes e a taxa média de atendimento. Essas taxas indicam o número médio de clientes que chegam ou são atendidos por unidade de tempo, quando encontram-se n clientes no sistema. Supõe-se que essas taxas sejam constantes ao longo do tempo.

As hipóteses do processo de nascimento e morte podem ser esquematizadas por meio do diagrama de taxas apresentado na Figura 5.4. Os nós (círculos) representam os estados n (número de elementos no processo) e os arcos (setas), as únicas transições possíveis para cada estado do sistema e a taxa média para cada transição. Um exemplo típico de processo que pode ser representado dessa maneira é o da fila de um caixa de banco que se encontra no estado n . A chegada (nascimento) de um novo cliente altera o estado da fila de n para $n + 1$. O término do atendimento (morte) altera o estado da fila de n para $n - 1$.

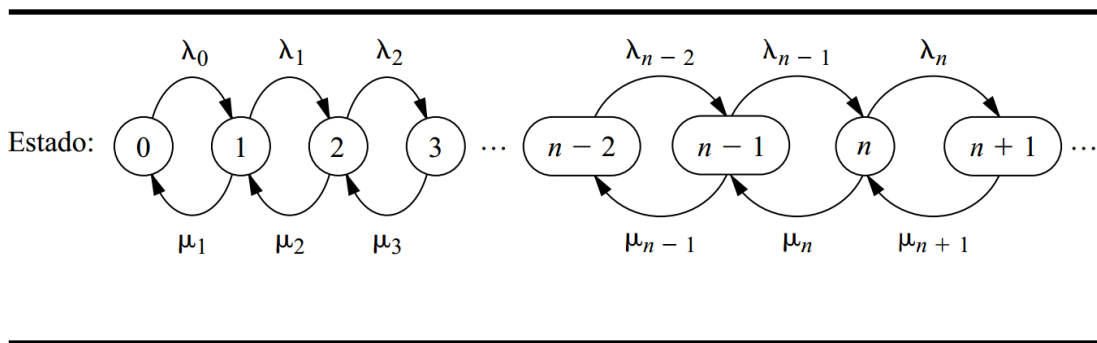


Figura 5.4: Diagrama do processo de nascimento e morte

De acordo com Hillier e Lieberman [18], a análise do processo de nascimento e morte para obter a distribuição probabilística do estado do sistema é considerada muito difícil quando o sistema se encontra em uma condição transitória. No entanto, é relativamente simples obter essa distribuição após o sistema ter atingido uma condição de estado estável. Essa obtenção pode ser feita diretamente do diagrama da Figura 5.4, ao verificar que o fluxo de entrada em um estado é igual ao fluxo de saída. Em sistemas estáveis, para qualquer estado n do sistema ($n = 0, 1, 2, \dots$), a taxa média de entrada é igual a taxa média de saída. Esse princípio é denominado de equação de equilíbrio.

A equação de equilíbrio será utilizada para obter as fórmulas do modelo de filas M/M/1. Isso é possível porque em um modelo de filas baseado

no processo de nascimento e morte, o estado n do sistema representa o número de clientes no sistema de filas. Assim, as medidas de desempenho podem ser obtidas após encontrar a probabilidade P_n de o sistema se encontrar em determinado estado n .

5.2.2 Modelo M/M/1

As chegadas e os términos do atendimento de um cliente no sistema de filas caracterizam um processo de nascimento e morte. Em virtude das hipóteses 1 e 2 e da Propriedade 3 da distribuição exponencial, pode-se dizer que o modelo possui entrada que segue a distribuição de Poisson e tempos de atendimento exponenciais.

No modelo M/M/1, os tempos entre chegadas sucessivas e os tempos de atendimento seguem a distribuição exponencial. A taxa média de chegada (λ) ao sistema de filas e taxa média de atendimento (μ) são constantes, independente do estado do sistema. Para esse sistema de filas de um único atendente essas taxas no processo de nascimento e morte são:

$$\lambda_n = \lambda, \quad \forall n \geq 0$$

e

$$\mu_n = \mu \quad \forall n \geq 1$$

A Figura 5.5 ilustra o diagrama de transição de estados para o modelo M/M/1. Os números nos círculos representam a quantidade de clientes no sistema de filas.

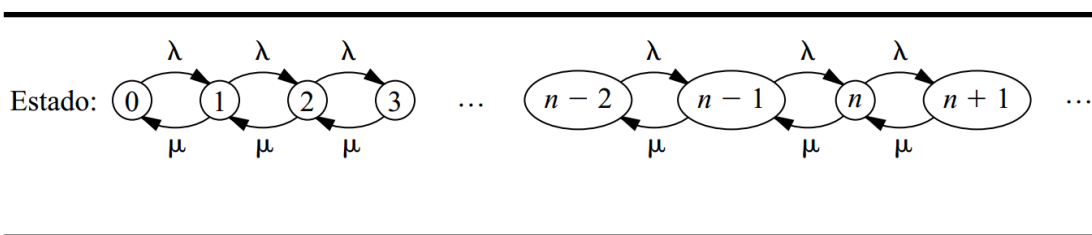


Figura 5.5: Diagrama de transição de estados do modelo M/M/1

Sabendo que a probabilidade de um sistema estável estar no estado n é representado por P_n , a probabilidade de o sistema estar no estado 0 será P_0 , no

estado 1 será P_1 , no estado 2 P_2 , e assim por diante. Dessa forma, as equações de equilíbrio para o processo de nascimento e morte da Figura §5.5 podem ser geradas considerando o estado $n = 0$.

Um sistema **entra** no estado 0 somente a partir do estado 1. Portanto, apenas por meio de uma saída (término do atendimento) do estado $n = 1$. Assim, P_1 é a probabilidade de estado estável no estado 1 e representa a proporção de tempo que o sistema permanece no estado 1. Dado que quando o sistema se encontra no estado 1, μ é a taxa média de atendimento e μP_1 é a taxa média à qual o sistema entra no estado 0. Por outro lado, considerando o estado $n = 0$, o sistema **sai** do estado 0 somente por meio de uma chegada. Dado que quando o sistema está no estado 0, λ é a taxa média de chegada de clientes, P_0 é a proporção de tempo que o sistema permanece no estado 0 e λP_0 é a taxa média à qual o sistema sai do estado 0.

A equação de equilíbrio para o **estado 0** é:

$$\mu P_1 = \lambda P_0 \quad (5.1)$$

Para solucionar as equações de equilíbrio, é preciso resolvê-las em relação a uma das variáveis, sendo a mais conveniente P_0 . Dessa forma, a equação de equilíbrio §5.1 para o estado 0, é usada para encontrar P_1 em termos de P_0 .

$$P_1 = \frac{\lambda}{\mu} P_0 \quad (5.2)$$

Para todos os outros estados ($n = 1, 2, \dots$), existem sempre duas transições, uma saindo e outra entrando no estado, conforme Figura §5.4. Portanto, cada lado das equações de equilíbrio (taxa que entra = taxa que sai) para esses estados é dada pela soma das taxas médias das respectivas transições envolvidas. No estado 1, estão entrando λP_0 e μP_2 ocorrendo, respectivamente, na transição vinda do estado 0 (chegada de um cliente) e do estado 2 (término de um atendimento). Estão saindo do estado 1, λP_1 e μP_1 que representam, respectivamente, a transição para o estado 2 (chegada de um cliente) e para o estado 0 (término de um atendimento). Assim, para o **estado 1**, a equação de equilíbrio é:

$$\lambda P_0 + \mu P_2 = \lambda P_1 + \mu P_1 \quad (5.3)$$

Isolando P_2 à esquerda têm-se:

$$\begin{aligned}
 P_2 &= \frac{\lambda P_1 + \mu P_1 - \lambda P_0}{\mu} \\
 &= \frac{\lambda}{\mu} P_1 + \frac{1}{\mu} (\mu P_1 - \lambda P_0)
 \end{aligned} \tag{5.4}$$

Da equação §5.1 sabe-se que $\mu P_1 = \lambda P_0$, substituindo em §5.4 obtêm-se:

$$\begin{aligned}
 P_2 &= \frac{\lambda}{\mu} P_1 + \frac{1}{\mu} (\lambda P_0 - \lambda P_0) \\
 &= \frac{\lambda}{\mu} P_1
 \end{aligned} \tag{5.5}$$

Para escrever P_2 em termos de P_0 , substitui-se §5.2 em §5.5.

$$\begin{aligned}
 P_2 &= \frac{\lambda}{\mu} \left(\frac{\lambda}{\mu} \right) P_0 \\
 &= \left(\frac{\lambda}{\mu} \right)^2 P_0
 \end{aligned} \tag{5.6}$$

A equação de equilíbrio do **estado 2** é dada por:

$$\lambda P_1 + \mu P_3 = \lambda P_2 + \mu P_2 \tag{5.7}$$

Isolando P_3 têm-se:

$$\begin{aligned}
 P_3 &= \frac{\lambda P_2 + \mu P_2 - \lambda P_1}{\mu} \\
 &= \frac{\lambda}{\mu} P_2 + \frac{1}{\mu} (\mu P_2 - \lambda P_1)
 \end{aligned} \tag{5.8}$$

Substituindo §5.5 em §5.8 obtêm-se:

$$\begin{aligned}
 P_3 &= \frac{\lambda \lambda}{\mu \mu} P_1 + \frac{1}{\mu} \left(\mu \frac{\lambda}{\mu} P_1 - \lambda P_1 \right) \\
 &= \left(\frac{\lambda}{\mu} \right)^2 P_1 + \frac{1}{\mu} (\lambda P_1 - \lambda P_1) \\
 &= \left(\frac{\lambda}{\mu} \right)^2 P_1
 \end{aligned} \tag{5.9}$$

Para escrever P_3 em termos de P_0 , substitui-se §5.2 em §5.9.

$$\begin{aligned} P_3 &= \left(\frac{\lambda}{\mu}\right)^2 \frac{\lambda}{\mu} P_0 \\ &= \left(\frac{\lambda}{\mu}\right)^3 P_0 \end{aligned} \quad (5.10)$$

Observando a probabilidade de o sistema se encontrar nos estados ($n = 0, 1, 2$), por indução conclui-se que:

$$P_n = \left(\frac{\lambda}{\mu}\right)^n P_0 \quad (5.11)$$

A taxa de ocupação de um servidor é dada por:

$$\rho = \frac{\lambda}{\mu} \quad (5.12)$$

Substituindo-se §5.12 na equação §5.11 obtêm-se:

$$P_n = \rho^n P_0 \quad (5.13)$$

Dado que a soma de todas as probabilidades deve ser igual a 1,

$$\sum_{n=0}^{\infty} P_n = 1 \quad (5.14)$$

implica que,

$$\begin{aligned} \sum_{n=0}^{\infty} \rho^n P_0 &= 1 \\ \frac{1}{P_0} &= \sum_{n=0}^{\infty} \rho^n \end{aligned} \quad (5.15)$$

O segundo termo da equação §5.15 é a soma de uma série geométrica obtida ao somar os infinitos termos de uma progressão geométrica e, converge, se e somente se, $\rho < 1$. Assim, para a existência de um sistema em estado estável (equilíbrio), a taxa média de atendimento deve ser maior que a taxa média de chegada ($\mu > \lambda$). Hillier e Lieberman [18] justifica, quando $\mu \leq \lambda$, a solução “estoura”, pois o somatório para calcular P_0

diverge. Ao iniciar a operação do sistema de filas sem nenhum cliente, o atendente seria bem-sucedido suportando os clientes que chegam em um curto período de tempo, mas isso é impossível a longo prazo e a fila crescerá sem limites. Mesmo quando $\mu = \lambda$, o número de clientes no sistema de filas cresce lentamente sem limites ao longo do tempo.

A soma da série geométrica é:

$$\sum_{n=0}^{\infty} \rho^n = \frac{1}{1-\rho} \quad \text{a série converge se } |\rho| < 1 \quad (5.16)$$

Assim, substituindo §5.16 em §5.15 é possível obter P_0 .

$$\begin{aligned} \frac{1}{P_0} &= \frac{1}{1-\rho} \\ P_0 &= 1-\rho \end{aligned} \quad (5.17)$$

Substituindo a equação §5.17 na equação §5.13 obtêm-se, a equação que permite calcular diretamente, a probabilidade de exatamente n clientes se encontrarem no sistema de filas.

$$P_n = \rho^n (1-\rho) \quad (5.18)$$

As medidas de desempenho para o sistema de filas podem agora ser obtidas a partir de P_n . A definição de NS dada na Seção §2.4.9 especifica,

$$NS = \sum_{n=0}^{\infty} n P_n$$

Para obter o número médio de clientes no sistema, inicialmente, substitui-se a equação §5.18 na equação §2.10.

$$\begin{aligned} NS &= \sum_{n=0}^{\infty} n \rho^n (1-\rho) \\ &= (1-\rho) \sum_{n=0}^{\infty} n \rho^n \end{aligned} \quad (5.19)$$

O somatório da equação §5.19 pode ser reescrito como:

$$\begin{aligned}
\sum_{n=0}^{\infty} n\rho^n &= \rho + 2\rho^2 + 3\rho^3 + 4\rho^4 + \dots \\
&= \rho(1 + 2\rho + 3\rho^2 + 4\rho^3 + \dots) \\
&= \rho \sum_{n=1}^{\infty} n\rho^{n-1}
\end{aligned} \tag{5.20}$$

Substituindo o somatório de §5.20 em §5.19 tem-se:

$$NS = (1 - \rho)\rho \sum_{n=1}^{\infty} n\rho^{n-1} \tag{5.21}$$

A derivada de ρ^n em relação a ρ é dada por:

$$\frac{d}{d\rho}\rho^n = n\rho^{n-1} \tag{5.22}$$

Assim, admitindo satisfeitas as condições para inverter o somatório e derivada, obtêm-se:

$$\begin{aligned}
NS &= (1 - \rho)\rho \sum_{n=1}^{\infty} n\rho^{n-1} \\
&= (1 - \rho)\rho \sum_{n=0}^{\infty} \frac{d}{d\rho}\rho^n \\
&= (1 - \rho)\rho \frac{d}{d\rho} \sum_{n=0}^{\infty} \rho^n \\
&= (1 - \rho)\rho \frac{d}{d\rho} \left(\frac{1}{1 - \rho} \right) \\
&= (1 - \rho)\rho \frac{1}{(1 - \rho)^2}
\end{aligned} \tag{5.23}$$

Simplificando §5.23 e substituindo ρ , tem-se:

$$\begin{aligned}
NS &= \frac{\rho}{(1 - \rho)} \\
&= \frac{\frac{\lambda}{\mu}}{1 - \frac{\lambda}{\mu}}
\end{aligned}$$

Assim, o número médio de clientes no sistema é dado pela fórmula:

$$NS = \frac{\lambda}{\mu - \lambda} \quad (5.24)$$

Considerando a fórmula de Little §2.14, apresentada na Seção §2.4.10 e isolando TS,

$$\begin{aligned} NS &= \lambda TS \\ TS &= \frac{NS}{\lambda} \end{aligned} \quad (5.25)$$

a fórmula do tempo médio de permanência no sistema pode ser obtida utilizando §5.24.

$$\begin{aligned} TS &= \frac{\frac{\lambda}{\mu - \lambda}}{\lambda} \\ &= \frac{\lambda}{\mu - \lambda} \left(\frac{1}{\lambda} \right) \\ &= \frac{1}{\mu - \lambda} \end{aligned} \quad (5.26)$$

Utilizando a relação §2.17, apresentada na Seção §2.4.10, isolando NF à esquerda,

$$\begin{aligned} NS &= NF + \frac{\lambda}{\mu} \\ NF &= NS - \frac{\lambda}{\mu} \end{aligned} \quad (5.27)$$

e substituindo NS por §5.24, é possível obter a fórmula do número médio de clientes na fila.

$$\begin{aligned} NF &= \frac{\lambda}{\mu - \lambda} - \frac{\lambda}{\mu} \\ &= \frac{\lambda\mu - \lambda(\mu - \lambda)}{\mu(\mu - \lambda)} \\ &= \frac{\lambda\mu - \lambda\mu + \lambda^2}{\mu(\mu - \lambda)} \\ &= \frac{\lambda^2}{\mu(\mu - \lambda)} \end{aligned} \quad (5.28)$$

Para obter o tempo médio de permanência na fila, utiliza-se a relação de Little §2.15, isolando TF,

$$\begin{aligned} \text{NF} &= \lambda \text{TF} \\ \text{TF} &= \frac{\text{NF}}{\lambda} \end{aligned} \quad (5.29)$$

e substituindo NF por §5.28, obtêm-se a fórmula:

$$\begin{aligned} \text{TF} &= \frac{\lambda^2}{\mu(\mu - \lambda)} \\ &= \frac{\lambda^2}{\mu(\mu - \lambda)} \left(\frac{1}{\lambda} \right) \\ &= \frac{\lambda}{\mu(\mu - \lambda)} \end{aligned} \quad (5.30)$$

A Tabela §5.1 apresenta o resumo das fórmulas do modelo M/M/1 que tratam as principais variáveis randômicas para obter as medidas de desempenho do sistema de filas.

Nome	Descrição	Fórmula
NS	Número médio de clientes no sistema	$\text{NS} = \frac{\lambda}{\mu - \lambda}$
TS	Tempo médio de permanência no sistema	$\text{TS} = \frac{1}{\mu - \lambda}$
NF	Número médio de clientes na fila	$\text{NF} = \frac{\lambda^2}{\mu(\mu - \lambda)}$
TF	Tempo médio de permanência na fila	$\text{TF} = \frac{\lambda}{\mu(\mu - \lambda)}$

Tabela 5.1: Fórmulas do modelo M/M/1

5.3 Modelo de Simulação

Os elementos da solução de integração e suas características apresentam equivalência com os elementos e as características de um sistema de filas. Portanto, o comportamento de uma solução de integração pode ser analisado por

meio da Teoria das Filas. No entanto, como o objetivo desta pesquisa é estudar a solução antes da implementação, não existem dados reais para utilizar o método analítico. Assim, faz-se necessário a utilização da simulação.

Para analisar o comportamento e identificar possíveis gargalos de desempenho em soluções de integração por meio da simulação utilizando a Teoria das Filas, o modelo conceitual da solução deve ser transformado em um modelo de simulação computacional equivalente, mantendo as características da mensagem, *slot* e tarefas. Nesse contexto, identificou-se o SimEvents, uma ferramenta de simulação de eventos discretos para a modelagem de sistemas de filas. O desenvolvimento de um modelo de simulação requer conhecimento sobre a ferramenta que será utilizada. Dessa forma, foi necessário um estudo detalhado do SimEvents para conhecer e compreender o funcionamento dos blocos.

5.3.1 Ferramenta SimEvents

O SimEvents é um software para a simulação de eventos discretos desenvolvido pela MathWorks, incorporado no Simulink da plataforma Matlab. O software proporciona um mecanismo para a simulação de eventos discretos de modo que gerencia e processa sequências de eventos assíncronos. A geração das entidades, o movimento e processamento provocam a ocorrência de eventos no sistema. Esses eventos modificam o estado do sistema. A simulação desse comportamento dos sistemas permite a análise de desempenho e a identificação de possíveis gargalos de desempenho.

Os blocos pré-definidos de construção gráfica no SimEvents estão disponíveis nas bibliotecas de domínio específico. Os modelos de simulação são desenvolvidos utilizando esses blocos que permitem criar, movimentar, processar e armazenar entidades, como também gerenciar os recursos do sistema.

O software possibilita a modelagem de redes simples, bem como complexas, de filas e servidores. As bibliotecas de blocos pré-definidos, como filas, servidores e *switches*, permitem representar com precisão o sistema de filas e personalizar o roteamento, atrasos de processamento, prioridades, e outras operações. Os blocos estão disponíveis de acordo com sua funcionalidade nas bibliotecas, conforme Figura §5.6. Por exemplo, o blocos para modelagem de filas, *FIFO Queue*, *LIFO Queue* e *Priority Queue*, estão disponíveis na biblioteca *Queues*.

Muitos parâmetros dos blocos no SimEvents podem ser configurados estatisticamente por meio de uma distribuição de probabilidade para modelar

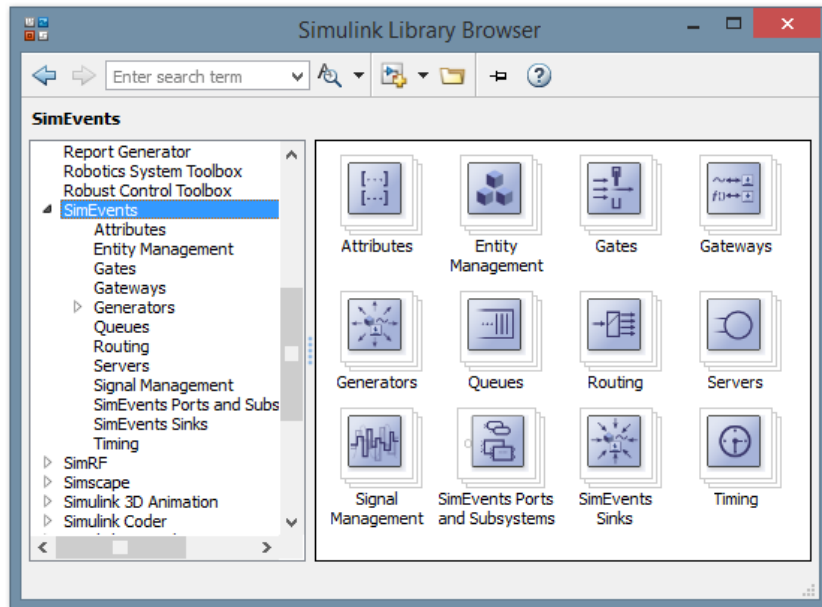


Figura 5.6: Biblioteca do software *SimEvents*

as variações probabilísticas do sistema real. Esses blocos utilizam números aleatórios e para definir automaticamente um novo valor para a semente inicial toda vez que o modelo de simulação será executado, pode-se usar uma função disponível na documentação do software. Por outro lado, as saídas estatísticas dos blocos permitem acompanhar e analisar as medidas de desempenho do sistema, tais como taxa de utilização do servidor, número de entidades na fila e tempo médio de permanência na fila.

Para o desenvolvimento de um modelo de simulação equivalente ao modelo conceitual da solução de integração apresentado no Capítulo 4, foi necessário identificar, nas bibliotecas do SimEvents, os blocos que podem ser utilizados para obter a equivalência com as funcionalidades do *slot* e das tarefas utilizadas na solução.

Para conhecer e compreender o funcionamento destes blocos e, assim, desenvolver um modelo de simulação equivalente, as tabelas a seguir apresentam a representação gráfica, o nome e a descrição das funcionalidades dos blocos identificados em cada biblioteca.


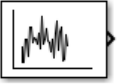
Ícone	Nome	Descrição
	<i>Time Based Entity Generator</i>	Gera entidades através da especificação de um tempo entre chegadas. O tempo entre chegadas pode ser configurado como constante ou uma distribuição de probabilidade e representa o intervalo de tempo entre dois eventos de geração sucessivos.
	<i>Event Based Random Number</i>	Gera números aleatórios baseado em eventos, inferindo a partir de um bloco subsequente quando gerará um novo número aleatório. Por exemplo, quando conectado à porta de entrada do bloco <i>Single Server</i> , aberta a partir do sinal, o bloco gera um novo número aleatório cada vez que uma entidade chega ao servidor. Os números randômicos são gerados a partir da especificação de uma distribuição de probabilidade, parâmetros e uma semente inicial.

Tabela 5.2: Blocos da biblioteca *Generators*



Ícone	Nome	Descrição
	<i>FIFO Queue</i>	Armazena até N entidades simultaneamente, formando uma fila, onde N é o valor da capacidade de armazenamento de entidades na fila, podendo ser um valor finito ou infinito. O bloco envia a entidade para a saída, mas se a porta estiver bloqueada retém a entidade até a porta ser liberada, saindo de acordo com a disciplina FIFO.
	<i>Single Server</i>	Apresenta a função de um servidor, presta o atendimento a uma entidade por um período de tempo e, após, está liberada para sair. O tempo de serviço é a duração do atendimento de uma entidade e pode ser especificado através de um parâmetro, atributo ou sinal. O sinal utilizado pode ser o bloco <i>Event-Based Random Number</i> , que permite que o tempo do serviço seja randômico de acordo com uma distribuição de probabilidade. O bloco determina o tempo de serviço para uma entidade após a sua chegada, sendo assumido em segundos.

Tabela 5.3: Bloco da biblioteca *Queues e Servers*

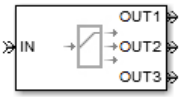
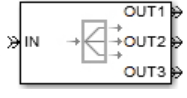
Ícone	Nome	Descrição
	<i>Output Switch</i>	Recebe entidades que saem através de uma das várias portas de saídas. A quantidade de portas de saídas pode ser configurada. O critério de chaveamento determina por qual porta de saída a entidade está selecionada para sair a qualquer momento.
	<i>Replicate</i>	Gera cópias das entidades que chegam. O número de cópias pode ser especificado através do parâmetro <i>Number of entity output ports</i> . Este bloco aceita uma entidade para replicar somente quando existe disponibilidade nas portas de entrada dos blocos subsequentes, de acordo com os critérios do parâmetro <i>Replicate entity when</i>

Tabela 5.4: Blocos da biblioteca Routing



Ícone	Nome	Descrição
	<i>Entity Combiner</i>	Gera uma nova entidade para cada conjunto de entidades que chegam, simultaneamente, nas várias portas de entrada. A operação de combinação acontece quando os blocos que estão conectados às portas de entrada do bloco <i>Entity Combiner</i> têm simultaneamente uma entidade pendente e as saídas deste estão disponíveis. Então, o bloco aceita a chegada de uma entidade em cada porta de entrada. Nos outros momentos, as portas de entrada estão indisponíveis. O bloco tem várias portas de entradas, que podem ser configuradas, e uma única de saída para as entidades.
	<i>Entity Sink</i>	Este bloco fornece uma maneira de terminar o caminho/ trajetória da entidade.

Tabela 5.5: Bloco da biblioteca *Entity Management* e *SimEvents Sinks*

5.3.2 Descrição do Modelo

Para obter um modelo de simulação equivalente ao modelo conceitual da solução de gerenciamento de chamadas de telefone da UNIJUÍ, foram consideradas as definições apresentadas na Seção §5.1. Observou-se, também, as características e funcionalidades do *slot* e das tarefas da solução e dos respectivos blocos do software de simulação, apresentados anteriormente.

Durante o estudo da biblioteca do SimEvents, percebeu-se que um bloco não seria suficiente para representar as funcionalidades que uma tarefa apresenta. Nesse contexto, essa seção aborda, inicialmente, as equivalências obtidas com o *slot* e cada tarefa do modelo conceitual e, em seguida, apresenta-se a descrição do modelo de simulação desenvolvido.

A caracterização de uma solução de integração como um sistema de eventos discretos, por meio da Teoria das Filas, demonstrou a equivalência do *slot* com uma fila. Um *slot* possui capacidade infinita e considera-se, nessa pesquisa, que organiza as mensagens de acordo com a disciplina FIFO. Assim, no modelo de simulação, foi utilizado o bloco *FIFO Queue*, de modo que a disciplina da fila é FIFO e a capacidade de armazenamento de entidades pode ser configurada como infinita.

Na caracterização, também foi demonstrada a equivalência de uma tarefa com um servidor. Assim, qualquer tarefa pode ser considerada equivalente ao bloco *Single Server*. As tarefas executam uma mensagem durante um período de tempo. O tempo de execução não é constante, e para torná-lo aleatório no modelo de simulação, o bloco *Single Server* receberá um sinal do bloco *Event Based Random Number*.

No entanto, o bloco *Single Server* apresenta a função de um servidor de forma geral e cada tarefa possui uma funcionalidade específica. Por exemplo, a tarefa *Filter*, ao executar o processamento da mensagem, realiza a função de filtrar as mensagens indesejadas e a tarefa *Replicator* realiza cópias da mensagem. Dessa forma, não foi possível representar a maioria das tarefas utilizando apenas os blocos *Single Server* e *Event Based Random Number*. Para obter a equivalência, além destes dois blocos, foi necessário utilizar os blocos *Output Switch*, *MATLAB Function*, *Entity Sink*, *Replicate* e *Entity Combiner*.

Para a tarefa *Filter*, foi necessário utilizar vários blocos, conforme a Figura §5.7. O bloco *Output Switch* permite determinar, pelo parâmetro

Switching criterion, por qual porta de saída a entidade deixará o bloco. O critério de chaveamento utilizado foi um sinal do bloco *MATLAB Function*. Nesse bloco, criou-se uma função para determinar, utilizando porcentagem, as entidades que seguirão no fluxo da simulação e as que serão enviadas para o bloco *Entity Sink*, representando as mensagens filtradas.

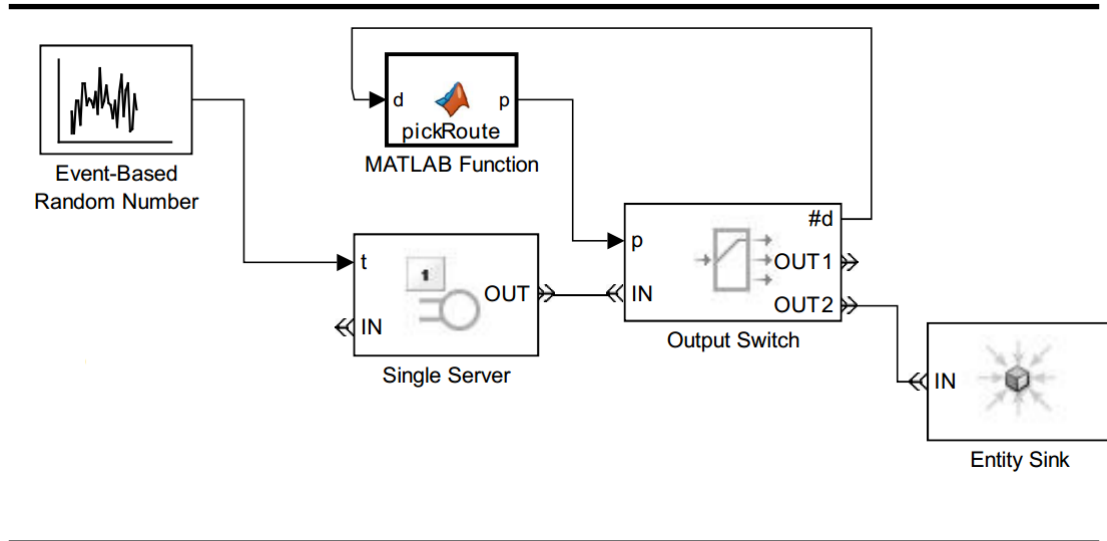


Figura 5.7: Tarefa Filter – T0, T7 e T9 – modelada no SimEvents

A Figura §5.8 mostra os blocos utilizados para obter as funcionalidades da tarefa *Replicator*. O bloco *Replicate* cria uma cópia da entidade. A quantidade de cópias da entidade é determinado através da configuração do número de portas de saída.

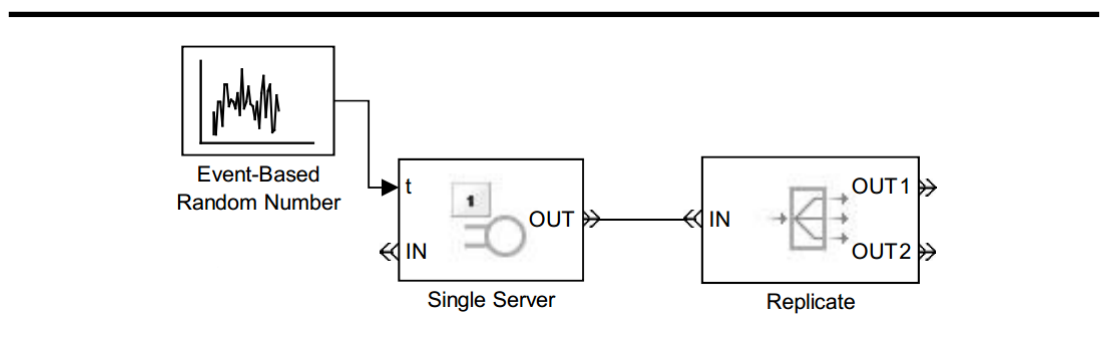


Figura 5.8: Tarefa Replicator - T4 e T5 – modelada no SimEvents

Para a tarefa *Translator* não identificou-se na biblioteca do SimEvents nenhum bloco que realize sua função. A ausência desta funcionalidade não interferirá na simulação e nos resultados, porque o bloco *Single Server* representa o tempo de processamento desta tarefa, conforme Figura §5.9.

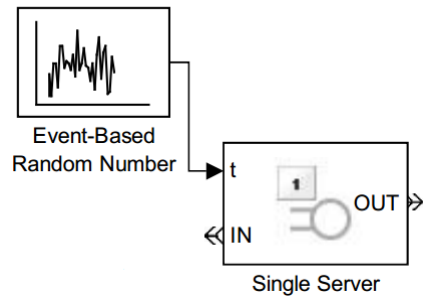


Figura 5.9: Tarefa *Translator* - T2, T6, T8 e T10 – modelada no SimEvents

O bloco *Entity Combiner* apresenta funcionalidades equivalentes às das tarefas *Correlator* e *Context-based Content Enricher*. O bloco detecta quando as entidades estão prontas para a operação de combinação e gera uma nova entidade para cada conjunto de entidades que chegam. Este bloco aceita entidades apenas quando a porta de entrada do bloco subsequente está disponível. Assim, para representar o tempo de execução das duas tarefas, será utilizado o bloco *Single Server*, para atrasar as operações de combinação, conforme a Figura §5.10.

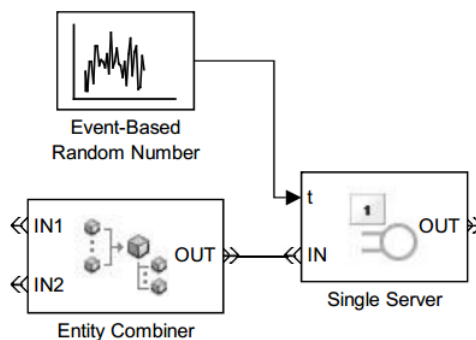


Figura 5.10: Tarefas T3 e T4 modelada no SimEvents

A Figura §5.11 apresenta o modelo de simulação desenvolvido. Para facilitar a compreensão do leitor, os blocos *FIFO Queue* que representam os *slots* foram identificados, com a mesma nomenclatura utilizada no modelo conceitual.

As portas da solução de integração são constituídas de tarefas e o processamento das mensagens consomem um determinado período de tempo. Assim, as portas foram modeladas como um servidor/tarefa.

As aplicações têm um banco de dados no qual são armazenadas as informações. Assim, modelou-se as aplicações da solução de integração com uma fila. Pode-se considerar essa equivalência porque tanto o banco de dados quanto a fila têm a função de armazenar objetos de interesse, informações e entidades, respectivamente.

Na solução de integração, as mensagens são enviadas para consultar a aplicação Recursos Humanos. A aplicação precisa processar esse pedido, demandando um período de tempo. Por isso, modelou-se como um servidor/tarefa. Já o Servidor de E-mail e o Serviço de Mensagens, no contexto da solução de integração, apenas recebem as mensagens. Assim, foi utilizado o bloco *Entity Sink*, que termina o trajetória da entidade.

No modelo de simulação, utilizou-se subsistemas para simplificar o modelo e tornar a representação gráfica mais semelhante possível com o modelo conceitual. Um subsistema é um conjunto de blocos agrupados em um único bloco *Subsystem*. Os subsistemas também ajudam entender e compreender a equivalência obtida, bem como o fluxo das entidades no modelo. Para a identificação, também utilizou-se a mesma nomenclatura do modelo conceitual.

A solução de integração integra cinco aplicações e é composta por um processo de orquestração. As aplicações integradas, os *slots* e as tarefas utilizadas no processo de orquestração são descritas a seguir, fazendo a relação com os subsistemas criados, os quais contém os blocos utilizados para obter a equivalência.

Os blocos *Time Based Entity Generator* e *FIFO Queue* representam a aplicação Central Telefônica. O bloco *Time Based Entity Generator* cria as entidades que representam as mensagens. As entidades criadas são armazenadas no bloco *FIFO Queue* e serão selecionadas para o atendimento no bloco P0. Após o atendimento, a entidade está liberada para sair, chegando assim, ao bloco S0.

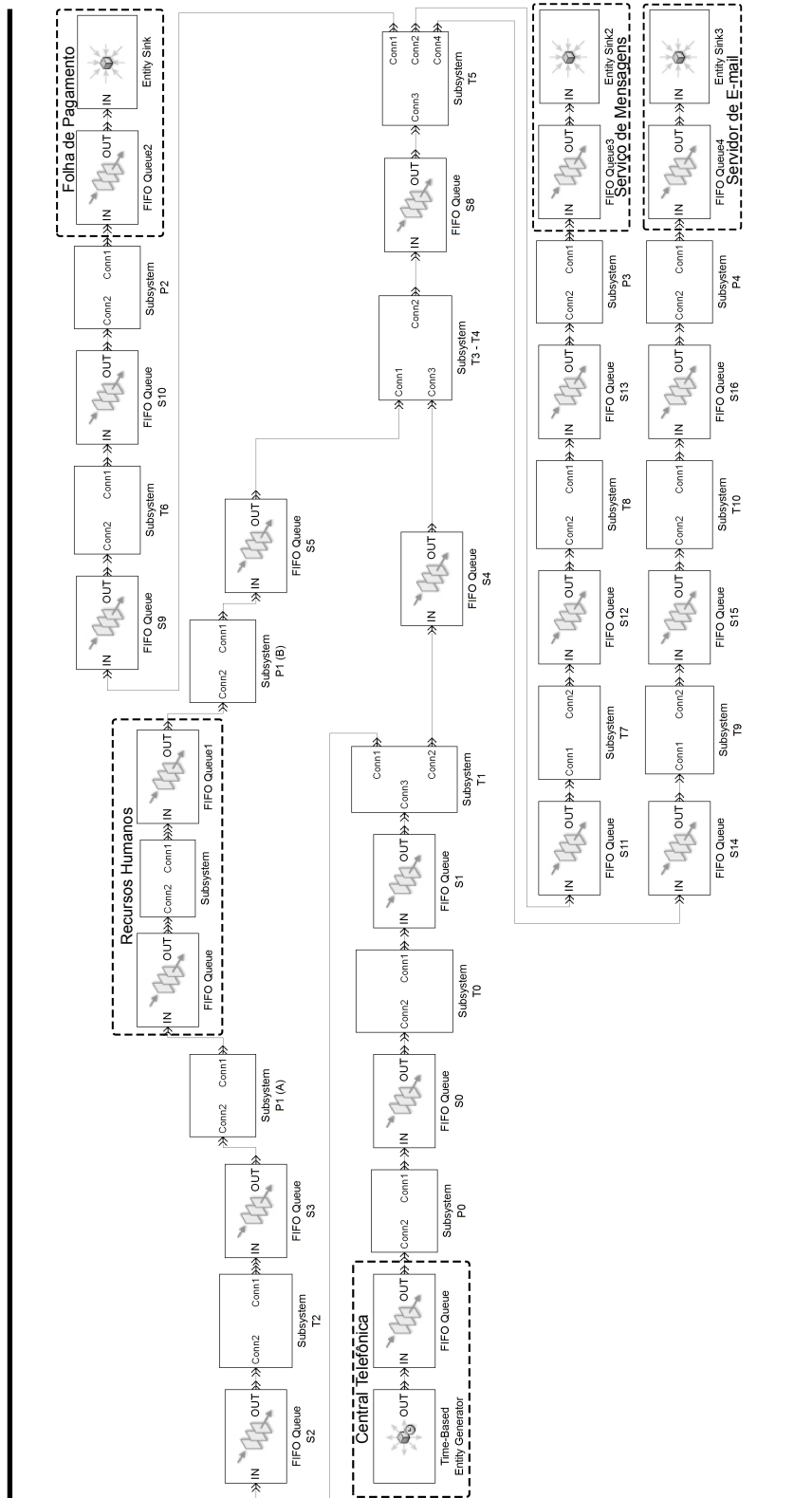


Figura 5.11: Modelo de simulação proposto

As entidades que saem do bloco S0 chegam ao bloco T0, onde são atendidas e, após, encaminhadas ao bloco *Output Switch*, que utiliza um sinal do bloco *MATLAB Function*, no qual foi criada uma função, para determinar a porcentagem de entidades filtradas. As entidades que seguem no fluxo são encaminhadas ao bloco S1, e selecionadas para o atendimento no bloco T1 onde, após o atendimento, são criadas duas cópias de cada entidade.

Uma cópia é enviada para o bloco S4, que armazena as entidades até que a outra cópia que foi enviada para consultar a aplicação Recursos Humanos chegue no bloco S5. Quando a cópia correspondente chegar no bloco S5 e no bloco T3-T4, o *Entity Combiner* estiver apto, será gerada uma nova entidade para cada conjunto de entidades que chegam. As entidades combinadas, depois de serem atendidas no bloco T3-T4, chegam ao bloco S8 e são selecionadas para o atendimento no bloco T5, onde após o atendimento são criadas três cópias de cada entidade. As cópias são enviadas para as aplicações Folha de Pagamento, Servidor de E-mail e Serviço de Mensagens.

Durante o desenvolvimento de um modelo de simulação, é preciso estar seguro de que a implementação está correta. Para isso, utilizam-se técnicas formais que procuram garantir que o modelo está livre de erros. A verificação do modelo de simulação desenvolvido é abordado no Capítulo §6. Após verificar que o modelo não apresenta erros de implementação computacional foram realizadas as simulações e coletados os dados.

5.4 Experimentação

De acordo com Freitas Filho [16], uma das principais etapas, no estudo de um sistema por meio da simulação, consiste na experimentação com o modelo, para que se possa observar seu comportamento sob determinadas condições e assim, cientificamente estudá-lo e entendê-lo. Nesse processo, procura-se imitar o desempenho do sistema real.

O processo de análise dos resultados das simulações trata os dados obtidos a partir dos experimentos. O objetivo da análise é inferir e prever o comportamento e o desempenho da solução, em situações reais de funcionamento.

Os modelos de simulação apresentam um comportamento estocástico, que imita o funcionamento do sistema real. Assim, pode-se usar o mesmo tratamento estatístico que seria empregado sobre o sistema real [16]. Para analisar o desempenho da solução de integração, por meio dos resultados obtidos a partir do modelo de simulação, observou-se o comportamento das variáveis relacionadas ao desempenho da solução durante os experimentos e nos distintos cenários de funcionamento.

5.4.1 Descrição dos Cenários da Simulação

O objetivo, nesta dissertação, é analisar o comportamento e identificar possíveis gargalos de desempenho na solução de integração, ainda na fase de projeto, tendo como entrada o modelo conceitual. Isso significa que a solução (sistema) real ainda não existe. Nesse caso, não é possível coletar dados para identificar a distribuição de probabilidade e estimar de forma precisa os parâmetros da distribuição do tempo entre chegadas e atendimento.

Na Teoria das Filas, o processo de chegada e atendimento são caracterizados por distribuições de probabilidade. Nessa pesquisa, a solução de integração é considerada equivalente a um sistema de filas representada pelo modelo M/M/1. Nesse modelo de filas, os tempos entre chegadas sucessivas e os tempos de atendimento seguem a distribuição exponencial, há um único atendente e a disciplina da fila é FIFO. Em vista disso, no modelo de simulação, o tempo entre chegadas e o tempo de atendimento foram configurados com distribuição exponencial e foram utilizados os blocos *Single Server* e *FIFO Queue*. A utilização de um bloco *Single Server* em cada tarefa é equivalente a uma thread ser alocada para executar uma tarefa.

Para configurar os parâmetros da distribuição exponencial do tempo de atendimento no modelo de simulação formal proposto, foram utilizadas estimativas do especialista de domínio. O bloco T3-T4, que representa duas tarefas (T3 e T4), foi configurado com um tempo de atendimento de 1 segundo. Nos blocos que representam as outras tarefas e as portas foi utilizado 0,5 segundo. Já na aplicação Recursos Humanos, foi configurado como sendo 2,5 segundos. A utilização de um parâmetro maior é justificada pelo fato da aplicação estar alocada em outro servidor. Assim, o pedido de consulta à aplicação é enviado por rede, consumindo mais tempo. Logo, para representar o tempo que leva para a aplicação receber o pedido, processar e mandar de volta à solução, utilizou-se um tempo de atendimento 5 vezes maior. Para filtrar as mensagens nos blocos T0 e T7, estimou-se que 5% das mensagens são filtradas e, para T9, utilizou-se 1% porque considerou-se que a maioria dos funcionários tenha cadastrado o endereço de email.

Os tempos entre chegadas foram estimados para obter diferentes taxas de chegadas de mensagens, para possibilitar a análise do desempenho da solução sob diferentes cargas de trabalho. Nesse contexto, a partir do modelo de simulação formal proposto, o comportamento da solução de integração foi analisado experimentalmente em sete cenários distintos. Em todos os

cenários, o tempo de simulação foi fixado em 24 horas e o tempo de atendimento nos valores apresentados, variando apenas o tempo entre chegadas de mensagens em 16, 8, 4, 2, 1, 0,5 e 0,25 segundos, respectivamente.

De acordo com a definição da Teoria das Filas, a taxa média de chegada de mensagens por segundo para o tempo entre chegadas utilizado na simulação em cada cenário é:

- Cenário 1: 0,0625
- Cenário 2: 0,125
- Cenário 3: 0,25
- Cenário 4: 0,5
- Cenário 5: 1
- Cenário 6: 2
- Cenário 7: 4

Pode-se observar que, enquanto de um cenário para outro o tempo entre chegadas diminui, a taxa média de chegada aumenta. Isso significa que para os cenários nos quais a solução de integração se encontra em estado estável, a quantidade de mensagens que chegam será maior em relação ao cenário anterior. Essa variação no tempo entre chegadas foi utilizada para analisar o desempenho da solução sob diferentes cargas de trabalho. Em cada cenário experimentado, foram coletados os dados para quatro variáveis: número médio de mensagens nos *slots*, tempo médio das mensagens nos *slots*, taxa de utilização e taxa ociosa das tarefas.

Para alcançar precisão estatística sobre os resultados da simulação, é necessário repetir a execução do modelo várias vezes. Na modelagem de sistemas estocásticos, entradas aleatórias geram saídas aleatórias. Assim, os resultados gerados pelos modelos estocásticos são diferentes a cada replicação, porque as variáveis de entrada são aleatórias.

De acordo com Chwif e Medina [6], a replicação é uma repetição da simulação do modelo, com a mesma configuração, a mesma duração e com os mesmos parâmetros de entrada, mas com uma semente de geração dos números aleatórios diferente. Apesar dos parâmetros de entrada serem os mesmos, como os números aleatórios gerados são diferentes, cada replicação terá resultados diferentes para as variáveis de desempenho.

Em estatística, quando um experimento é repetido um grande número de vezes com os mesmos dados, seguindo a Lei dos Grande Números [17], conforme o número de repetições se incrementa, a média amostral das variáveis do experimento se aproxima, cada vez mais, da média populacional, também conhecida como média teórica ou esperança matemática. Empiricamente, para a análise de sistemas que ainda não existem, onde para obter os dados é executado um experimento artificial, a média populacional costuma ser obtida com, aproximadamente, 25 replicações. Desta forma, foram realizadas 25 replicações de cada um dos sete cenários para obter resultados com precisão estatística do modelo de simulação formal proposto.

Toda e qualquer experimentação está sujeita a apresentar dados extremos. Na estatística, os dados que apresentam um grande afastamento dos demais (valores muito altos ou muito baixos) são conhecidos como outliers ou valores extremos. Geralmente, os dados que representam outliers são retirados da amostra, para obter dados mais homogêneos.

Para encontrar os outliers, os resultados da simulação obtidos das 25 replicações, foram submetidos ao método de Tukey [40]. O método consiste em encontrar valores extremos usando o intervalo interquartil (IQR) – diferença entre o terceiro quartil e o primeiro quartil ($Q3 - Q1$) – para filtrar números muito grandes ou muito pequenos. As fórmulas são as seguintes:

$$LI = Q1 - 1.5(Q3 - Q1) = Q1 - 1.5(IQR) \quad (5.31)$$

$$LS = Q3 + 1.5(Q3 - Q1) = Q3 + 1.5(IQR) \quad (5.32)$$

Os valores que estão abaixo do limite inferior (LI) ou acima do limite superior (LS) estabelecidos são considerados outliers. Os outliers encontrados nos resultados da simulação foram removidos e, depois, calculou-se as médias dos dados das 25 repetições e construiu-se os gráficos das variáveis para analisar as medidas de desempenho da solução de integração.

5.4.2 Interpretação e Análise dos Resultados

Para analisar os resultados da simulação do modelo, são realizadas inferências sobre os resultados obtidos e estimativas para as medidas de desempenho, nos diferentes cenários. De acordo com Freitas Filho [16], a simulação de sistemas que tem variáveis aleatórias gera resultados aleatórios. Para fazer qualquer tipo de inferência sobre os resultados que medem o desempenho do sistema modelado, é necessário realizar análises apropriadas.

A média apresenta a característica de ser sensível aos valores extremos. Como a média é muito afetada por esses valores, conseqüentemente também, o desvio padrão e o coeficiente de variação. Para evitar que os outliers interfiram na análise dos resultados, os valores extremos encontrados foram excluídos. No entanto, situações reais podem apresentar outliers e são caracterizados como valores atípicos. Assim, não é apropriado excluir e esquecer esses valores. O indicado é citar, analisar e inferir sobre os possíveis motivos da ocorrência desses valores.

Nesse contexto, calculou-se a média (\bar{x}) o desvio padrão (s) e o coeficiente de variação (CV), com e sem outliers, nos sete cenários, dos resultados das variáveis número médio de mensagens nos *slots*, tempo médio das mensagens nos *slots* e taxa de utilização para inferir se a medida de posição central e as medidas de dispersão são muito afetados pelos valores extremos. Para a taxa ociosa, não realizou-se essas inferências porque é expressa matematicamente como $1 - \rho$. Assim, os resultados dessa variável não são obtidos diretamente da simulação, mas a partir da taxa de utilização.

Os resultados da média, desvio padrão e coeficiente de variação com e sem outliers, para cada variável, nos sete cenários, são apresentados em forma de tabela. Nos resultados que foram encontrados outliers, estão identificados nas tabelas com o símbolo de asterisco (*). Inicialmente, apresentam-se os resultados da variável número médio de mensagens nos *slots* (NF) dos sete cenários. Após, é realizada a análise do gráfico dessa variável para estimar as medidas de desempenho da solução de integração.

No cenário 1, foram encontrados valores que estão abaixo do limite inferior no *slot* S4 e acima do limite superior em S0, S5, S12 e S13. Comparando os resultados da Tabela §5.6, é possível perceber que esses valores não afetam de maneira acentuada a média, a medida que não existe uma grande diferença entre a média com e sem outliers. No desvio padrão, a diferença também é pequena, o que demonstra que existe pouca dispersão nos valores do conjunto em torno da média. O coeficiente de variação menor que 30% comprova que a amostra é homogênea, existe pouca variação. Isso significa que a média é uma boa medida de tendência central. A amostra do *slot* S4 é mais homogênea do grupo, porque possui o coeficiente de variação mais baixo.

O cenário 2 apresentou valores abaixo do limite inferior no *slot* S2 e acima do limite superior no S0, S3 e S4. Analisando os resultados expressos na Tabela §5.7, percebe-se que não há muita diferença entre as estatísticas com e sem outliers. No entanto, a eliminação dos valores extremos resulta em uma amostra com dados mais homogêneos. O *slot* S4 nesse cenário também possui a amostra mais homogênea.

Slots	Com outliers			Sem outliers		
	\bar{x}	s	CV (%)	\bar{x}	s	CV (%)
S0*	0,00100	0,00011	10,88408	0,00097	0,00008	7,79131
S1	0,00089	0,00011	12,43839	0,00089	0,00011	12,43839
S2	0,00089	0,00011	11,90574	0,00089	0,00011	11,90574
S3	0,00094	0,00015	15,73158	0,00094	0,00015	15,73158
S4*	0,26947	0,00418	1,55169	0,26998	0,00337	1,24969
S5*	0,00374	0,00036	9,74989	0,00370	0,00032	8,58258
S8	0,00093	0,00010	11,10110	0,00093	0,00010	11,10110
S9	0,00090	0,00011	12,30599	0,00090	0,00011	12,30599
S10	0,00089	0,00010	11,36067	0,00089	0,00010	11,36067
S11	0,00088	0,00010	11,45787	0,00088	0,00010	11,45787
S12*	0,00079	0,00010	12,84668	0,00078	0,00009	11,62736
S13*	0,00085	0,00013	15,26382	0,00084	0,00011	13,09171
S14	0,00089	0,00013	14,05391	0,00089	0,00013	14,05391
S15	0,00086	0,00010	11,68624	0,00086	0,00010	11,68624
S16	0,00092	0,00011	12,37015	0,00092	0,00011	12,37015

Tabela 5.6: Estatísticas da variável NF no cenário 1

Slots	Com outliers			Sem outliers		
	\bar{x}	s	CV (%)	\bar{x}	s	CV (%)
S0*	0,00425	0,00031	7,31142	0,00421	0,00028	6,55419
S1	0,00375	0,00019	5,03787	0,00375	0,00019	5,03787
S2*	0,00376	0,00024	6,50162	0,00380	0,00020	5,30961
S3*	0,00376	0,00032	8,39788	0,00371	0,00026	7,01539
S4*	0,62515	0,00943	1,50813	0,62402	0,00773	1,23849
S5	0,01588	0,00072	4,55512	0,01588	0,00072	4,55512
S8	0,00375	0,00019	4,97307	0,00375	0,00019	4,97307
S9	0,00381	0,00030	7,75060	0,00381	0,00030	7,75060
S10	0,00375	0,00028	7,42410	0,00375	0,00028	7,42410
S11	0,00372	0,00023	6,20130	0,00372	0,00023	6,20130
S12	0,00333	0,00017	5,05838	0,00333	0,00017	5,05838
S13	0,00341	0,00017	4,99683	0,00341	0,00017	4,99683
S14	0,00370	0,00019	5,04457	0,00370	0,00019	5,04457
S15	0,00373	0,00024	6,45425	0,00373	0,00024	6,45425
S16	0,00369	0,00025	6,80510	0,00369	0,00025	6,80510

Tabela 5.7: Estatísticas da variável NF no cenário 2

Os dados obtidos da simulação do cenário 3 apresentaram valores abaixo do limite inferior no *slot* S5 e acima do limite superior no S1, S13 e S15. Nesse cenário, a média também não é sensível aos valores extremos à medida que percebe-se que não há uma variação significativa com a média sem outliers. O coeficiente de variação dos dados é baixo, mostrando que os resultados são homogêneos. O *slot* S4 apresenta o coeficiente de variação mais baixo da série de dados, conforme Tabela §5.8.

Slots	Com outliers			Sem outliers		
	\bar{x}	s	CV (%)	\bar{x}	s	CV (%)
S0	0,01781	0,00075	4,21905	0,01781	0,00075	4,21905
S1*	0,01608	0,00071	4,41246	0,01601	0,00063	3,90970
S2	0,01583	0,00069	4,37151	0,01583	0,00069	4,37151
S3	0,01601	0,00054	3,36893	0,01601	0,00054	3,36893
S4	1,93626	0,04948	2,55567	1,93626	0,04948	2,55567
S5*	0,07439	0,00228	3,06727	0,07481	0,00182	2,43669
S8	0,01602	0,00059	3,68765	0,01602	0,00059	3,68765
S9	0,01612	0,00051	3,14089	0,01612	0,00051	3,14089
S10	0,01585	0,00051	3,18916	0,01585	0,00051	3,18916
S11	0,01580	0,00054	3,43452	0,01580	0,00054	3,43452
S12	0,01437	0,00050	3,45448	0,01437	0,00050	3,45448
S13*	0,01433	0,00060	4,18448	0,01426	0,00050	3,52198
S14	0,01608	0,00036	2,22699	0,01608	0,00036	2,22699
S15*	0,01585	0,00056	3,50409	0,01574	0,00043	2,75294
S16	0,01564	0,00063	4,03079	0,01564	0,00063	4,03079

Tabela 5.8: Estatísticas da variável NF no cenário 3

O cenário 4 apresentou valores abaixo do limite inferior nos *slots* S5 e S13 e acima do limite superior nos *slots* S5, S9, S10, S13 e S14. Os dados são homogêneos, pois apresentam um coeficiente de variação baixo, conforme Tabela §5.9. Os valores extremos não afetam a média, pois não há uma diferença significativa comparada a média sem outliers.

Os *slots* S2, S5, S8, S9 S14 e S16 apresentaram valores abaixo do limite inferior no cenário 5. Já os *slots* S0, S3, S5, S10, S11, S12 e S16 apresentaram valores acima do limite superior. Há pouca variação entre as médias com e sem outliers. Assim, os valores extremos não afetam a média. O coeficiente de variação dos dados é baixo, sendo o S4 o conjunto de dados mais homogêneo, conforme Tabela §5.10.

Slots	Com outliers			Sem outliers		
	\bar{x}	s	CV (%)	\bar{x}	s	CV (%)
S0	0,08320	0,00196	2,35129	0,08320	0,00196	2,35129
S1	0,07433	0,00150	2,01860	0,07433	0,00150	2,01860
S2	0,07397	0,00187	2,52907	0,07397	0,00187	2,52907
S3	0,07413	0,00157	2,11905	0,07413	0,00157	2,11905
S4	3276,87273	158,04968	4,82319	3276,87273	158,04968	4,82319
S5*	0,26864	0,00918	3,41868	0,26729	0,00484	1,81052
S8	0,05038	0,00140	2,77255	0,05038	0,00140	2,77255
S9*	0,05025	0,00131	2,59838	0,05011	0,00114	2,27083
S10*	0,04979	0,00153	3,07869	0,04945	0,00101	2,04342
S11	0,04981	0,00112	2,25842	0,04981	0,00112	2,25842
S12	0,04488	0,00132	2,95050	0,04488	0,00132	2,95050
S13*	0,04477	0,00144	3,21065	0,04482	0,00081	1,79903
S14*	0,04995	0,00159	3,18708	0,04979	0,00140	2,80628
S15	0,04887	0,00141	2,88965	0,04887	0,00141	2,88965
S16	0,04871	0,00107	2,18697	0,04871	0,00107	2,18697

Tabela 5.9: Estatísticas da variável NF no cenário 4

Slots	Com outliers			Sem outliers		
	\bar{x}	s	CV (%)	\bar{x}	s	CV (%)
S0*	0,49642	0,01192	2,40022	0,49514	0,01026	2,07291
S1	0,43063	0,00686	1,59349	0,43063	0,00686	1,59349
S2*	0,42796	0,00908	2,12246	0,42888	0,00801	1,86676
S3*	0,42930	0,00720	1,67663	0,42794	0,00569	1,32916
S4	23817,55498	170,89275	0,71751	23817,55498	170,89275	0,71751
S5*	0,26767	0,00569	2,12696	0,26760	0,00393	1,46749
S8*	0,04935	0,00088	1,77644	0,04953	0,00064	1,29554
S9*	0,04974	0,00076	1,53276	0,04983	0,00063	1,25830
S10*	0,04956	0,00140	2,82625	0,04942	0,00124	2,50489
S11*	0,04998	0,00127	2,53569	0,04981	0,00099	1,98170
S12*	0,04436	0,00080	1,79567	0,04427	0,00068	1,53279
S13	0,04462	0,00111	2,47677	0,04462	0,00111	2,47677
S14	0,04971	0,00150	3,02445	0,04971	0,00150	3,02445
S15*	0,04896	0,00106	2,16605	0,04907	0,00094	1,92050
S16*	0,04865	0,00087	1,79518	0,04865	0,00070	1,43053

Tabela 5.10: Estatísticas da variável NF no cenário 5

No cenário 6, foram identificados valores abaixo do limite inferior nos *slots* S1, S2, S5 e S8 e acima do limite superior no S0, S1, S2, S5, S13 e S14, conforme a Tabela §5.11. O slot S0 apresenta dois valores acima do limite superior e percebe-se que há uma variação significativa nas médias. Nesse caso, os valores extremos afetam a média. No cenário 6 foi utilizado um tempo entre chegadas de mensagens de 0,5 segundos, tempo igual ao da execução das tarefas. Nessa configuração, o sistema não se encontra em estado estável o que pode ser o motivo dessa maior dispersão dos valores em torno da média. A amostra do S0 é heterogênea ($CV > 30\%$), já o S4 é o mais homogêneo do conjunto de dados.

Slots	Com outliers			Sem outliers		
	\bar{x}	s	CV (%)	\bar{x}	s	CV (%)
S0*	215,02272	106,70019	49,62275	192,46142	69,88046	36,30882
S1*	16,57355	1,35846	8,19658	16,56789	0,96230	5,80824
S2*	16,94826	1,30329	7,68982	16,94231	1,07467	6,34311
S3	16,65230	1,31071	7,87104	16,65230	1,31071	7,87104
S4	64312,91719	172,96660	0,26895	64312,91719	172,96660	0,26895
S5*	0,26849	0,00577	2,14836	0,26929	0,00364	1,35140
S8*	0,04992	0,00145	2,89947	0,05008	0,00125	2,48908
S9	0,05020	0,00136	2,71543	0,05020	0,00136	2,71543
S10	0,05000	0,00125	2,50805	0,05000	0,00125	2,50805
S11	0,05000	0,00150	3,00617	0,05000	0,00150	3,00617
S12	0,04463	0,00099	2,21054	0,04463	0,00099	2,21054
S13*	0,04520	0,00138	3,05563	0,04505	0,00119	2,63150
S14*	0,04962	0,00126	2,54201	0,04949	0,00113	2,27380
S15	0,04936	0,00084	1,70549	0,04936	0,00084	1,70549
S16	0,04910	0,00129	2,62538	0,04910	0,00129	2,62538

Tabela 5.11: Estatísticas da variável NF no cenário 6

No cenário 7, nenhum resultado apresentou valores abaixo do limite inferior, somente valores acima do limite superior nos *slots* S0, S9, S10 e S13. O slot S0 apresentou um valor acima do limite superior e percebe-se que há uma diferença significativa entre a médias com e sem outliers. Nesse cenário, o sistema também não se encontra em estado estável, pois a taxa média de chegada de mensagens é maior que a taxa média de execução das tarefas, podendo ser esse o provável motivo da amostra do S0 ser heterogênea e da dispersão dos valores desse conjunto de dados.

Slots	Com outliers			Sem outliers		
	\bar{x}	s	CV (%)	\bar{x}	s	CV (%)
S0*	351,23918	232,35106	66,15180	319,97996	175,62374	54,88586
S1	17,28143	1,47990	8,56352	17,28143	1,47990	8,56352
S2	17,31500	1,30930	7,56164	17,31500	1,30930	7,56164
S3	17,10764	1,41563	8,27484	17,10764	1,41563	8,27484
S4	64544,05342	223,94345	0,34696	64544,05342	223,94345	0,34696
S5	0,26879	0,00559	2,07900	0,26879	0,00559	2,07900
S8	0,05086	0,00117	2,29064	0,05086	0,00117	2,29064
S9*	0,05025	0,00135	2,69622	0,05014	0,00125	2,49565
S10*	0,05010	0,00126	2,51132	0,04997	0,00110	2,19953
S11	0,05001	0,00113	2,26560	0,05001	0,00113	2,26560
S12	0,04483	0,00126	2,80214	0,04483	0,00126	2,80214
S13*	0,04506	0,00154	3,42222	0,04491	0,00136	3,03057
S14	0,05036	0,00105	2,08941	0,05036	0,00105	2,08941
S15	0,04940	0,00092	1,86265	0,04940	0,00092	1,86265
S16	0,04888	0,00119	2,44467	0,04888	0,00119	2,44467

Tabela 5.12: Estatísticas da variável NF no cenário 7

Analisando os outliers, pode-se inferir que o provável motivo da ocorrência desses valores é o fato da variável em análise depender do processo de chegada e de execução das mensagens. Dessa forma, os valores abaixo do limite inferior correspondem a situações que uma quantidade menor de mensagens chegaram ou que a maioria das tarefas foram executadas de forma rápida. Nessa situação, poucas mensagens precisam esperar a execução, gerando um valor pequeno para o número médio. Já os valores acima do limite superior podem ser decorrentes de situações em que a quantidade de mensagens que chegaram foi maior ou que a execução de algumas tarefas demandou mais tempo. Assim, mais mensagens precisam esperar, aumentando o número médio de mensagens. Portanto, esses podem ser os prováveis motivos pelos quais os valores ficaram afastados do centro geométrico do conjunto de dados.

A Figura §5.12 representa o desempenho da solução de integração quanto ao número médio de mensagens nos *slots*. Quando um *slot* assume valores além dos adequados, pode constituir um gargalo. Os resultados obtidos da variável estão expressos no gráfico e permitem analisar o comportamento e identificar possíveis gargalos de desempenho da solução de integração.

Nos cenários 1, 2 e 3 com um tempo entre chegadas, respectivamente, de 16, 8 e 4 segundos todas as instalações encontram-se em estado estável ($\mu > \lambda$). No cenário 1, o valor do número médio de mensagens nos *slots* é menor que 1. Logo, em média, menos de uma mensagem precisa esperar para ser executada nas tarefas. Isso não significa que as mensagens nunca esperem. Pode ocorrer momentos em que o número de mensagens que chegam seja superior à capacidade de processamento ou situações em que a execução de uma tarefa seja mais demorada.

Analisando os três cenários, percebe-se que os *slots* seguem uma tendência, apresentando o mesmo comportamento. Dessa forma, os valores do número médio de mensagens, nos *slots* do cenário 2 e 3, aumentam porque a taxa média de chegada de mensagens é maior em relação ao cenário anterior, mas não são significativos para constituírem gargalos, nesses cenários.

Considerando os cenários 1, 2 e 3, não é possível afirmar que S4 seja um gargalo, mas percebe-se essa tendência, visto que nos três cenários apresenta o maior valor, seguido de S5. No cenário 4, a instalação que representa a aplicação de recursos humanos no modelo de simulação não está mais em estado estável e percebe-se que isso influencia diretamente S4, à medida que o valor aumenta de aproximadamente 2 (cenário 3) para aproximadamente 3277. Esse aumento significativo justifica-se porque no *slot* S4 as mensagens “aguardam” a correlacionada que foi enviada pela solução para buscar

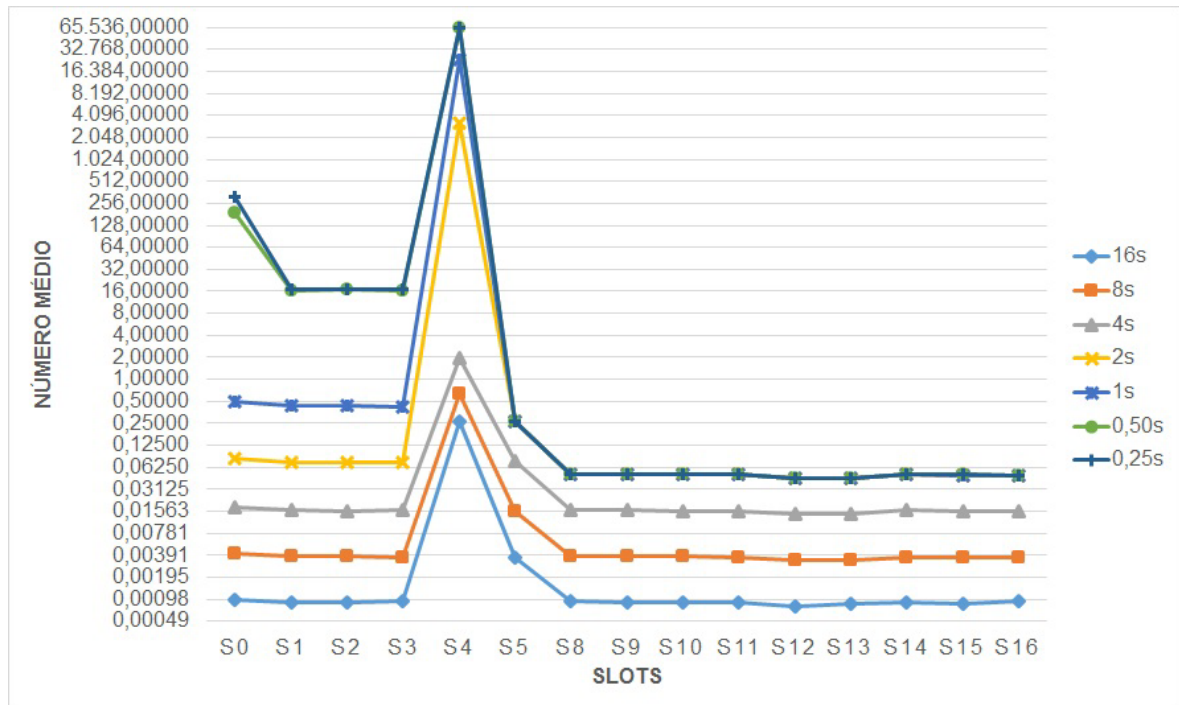


Figura 5.12: Número médio de mensagens nos slots

informações na aplicação Recursos Humanos. No cenário 5, o valor de S4 aumenta para aproximadamente 23818 e S5 apresenta aproximadamente o mesmo valor que no cenário 4.

No cenário 6 e 7, a instalação que compreende S0 não se encontra em estado estável e pode-se perceber que isso influencia nos resultados da variável, visto que apresenta um comportamento diferente dos demais cenários. Como a taxa média de chegada é maior que a de processamento, a quantidade de mensagens esperando tende a ser infinita.

Desta forma, pode-se concluir a análise dessa variável em estado estável, inferindo que apenas o slot S4 tende a ser um gargalo de desempenho e que a solução de integração apresenta o comportamento de que a uma taxa de trabalho maior, a quantidade de mensagens esperando nos slots tende a aumentar. Já em condições em que uma ou várias instalações não se encontram em estado estável, os resultados são influenciados por esse fator e também as instalações que apresentam relação ou dependência. Assim, é possível afirmar que na situação em que a taxa média de chegada de mensagens

for maior que à de execução das tarefas, não é possível realizar nenhuma inferência utilizando os resultados de um modelo de simulação baseado em um modelo da Teoria das Filas.

Os resultados da variável tempo médio de permanência nos *slots* (TF), dos sete cenários, são apresentados nas tabelas a seguir. Após, é realizada a análise do gráfico dessa variável para estimar as medidas de desempenho da solução de integração.

Os resultados do cenário 1 apresentaram apenas valores acima do limite superior. Os outliers foram encontrados nos *slots* S0, S4, S5 e S13. Esses valores extremos não afetam a média, a medida que não há uma variação significativa da média com e sem outliers. Os resultados de todos os *slots* são homogêneos, sendo S4 o conjunto mais homogêneo, porque apresenta o coeficiente de variação mais baixo, conforme Tabela §5.13.

Slots	Com outliers			Sem outliers		
	\bar{x}	s	CV (%)	\bar{x}	s	CV (%)
S0*	0,01600	0,00173	10,79945	0,01562	0,00120	7,66139
S1	0,01507	0,00184	12,21929	0,01507	0,00184	12,21929
S2	0,01511	0,00178	11,79992	0,01511	0,00178	11,79992
S3	0,01586	0,00248	15,62276	0,01586	0,00248	15,62276
S4*	4,55678	0,06121	1,34324	4,55008	0,05235	1,15045
S5*	0,06322	0,00612	9,67770	0,06258	0,00532	8,49515
S8	0,01570	0,00169	10,74817	0,01570	0,00169	10,74817
S9	0,01529	0,00184	12,01359	0,01529	0,00184	12,01359
S10	0,01512	0,00172	11,39639	0,01512	0,00172	11,39639
S11	0,01490	0,00173	11,60387	0,01490	0,00173	11,60387
S12	0,01414	0,00179	12,65711	0,01414	0,00179	12,65711
S13*	0,01516	0,00230	15,16255	0,01489	0,00193	12,96361
S14	0,01505	0,00210	13,96705	0,01505	0,00210	13,96705
S15	0,01474	0,00169	11,44518	0,01474	0,00169	11,44518
S16	0,01573	0,00192	12,18119	0,01573	0,00192	12,18119

Tabela 5.13: Estatísticas da variável TF no cenário 1

O cenário 2 também apresentou apenas valores acima do limite superior. Os outliers foram encontrados nos *slots* S0, S3 e S14. Percebe-se que esses valores não afetam a média, porque não há uma diferença significativa entre as médias com e sem outliers. Os resultados de todos os *slots* são homogêneos, sendo S4 o conjunto mais homogêneo porque apresenta o menor valor do coeficiente de variação, conforme Tabela §5.14.

Slots	Com outliers			Sem outliers		
	\bar{x}	s	CV (%)	\bar{x}	s	CV (%)
S0*	0,03406	0,00241	7,08501	0,03382	0,00214	6,33417
S1	0,03166	0,00151	4,77380	0,03166	0,00151	4,77380
S2	0,03170	0,00197	6,20971	0,03170	0,00197	6,20971
S3*	0,03175	0,00251	7,89230	0,03150	0,00224	7,12110
S4	5,27803	0,05799	1,09869	5,27803	0,05799	1,09869
S5	0,13407	0,00626	4,67125	0,13407	0,00626	4,67125
S8	0,03164	0,00147	4,63646	0,03164	0,00147	4,63646
S9	0,03215	0,00240	7,46548	0,03215	0,00240	7,46548
S10	0,03162	0,00232	7,32545	0,03162	0,00232	7,32545
S11	0,03142	0,00184	5,86402	0,03142	0,00184	5,86402
S12	0,02953	0,00147	4,98361	0,02953	0,00147	4,98361
S13	0,03023	0,00145	4,79122	0,03023	0,00145	4,79122
S14*	0,03127	0,00151	4,83777	0,03111	0,00132	4,25056
S15	0,03175	0,00196	6,16962	0,03175	0,00196	6,16962
S16	0,03144	0,00207	6,58665	0,03144	0,00207	6,58665

Tabela 5.14: Estatísticas da variável TF no cenário 2

No cenário 3, foram encontrados valores abaixo do limite inferior no *slot* S5 e acima do limite superior em S1, S8, S13 e S15. Percebe-se que não há uma variação significativa nas médias com e sem outliers, conforme Tabela §5.15. Os resultados de todos os *slots* apresentam o coeficiente de variação baixo, o que demonstra que o conjunto de dados é homogêneo.

No cenário 4, foram identificados valores abaixo do limite inferior nos *slots* S5 e S13 e acima do limite superior no S5, S10, S13 e S14. Os valores extremos não afetam a média porque existe pouca diferença entre as duas médias, conforme Tabela §5.16. Os resultados de todos os *slots* são homogêneos.

No cenário 5, foram encontrados valores abaixo do limite inferior nos *slots* S4, S5, S8 e S9 e acima do limite superior no S0, S3, S5, S10, S11 e S12. Analisando os resultados obtidos percebe-se que o conjunto é homogêneo para todos os *slots* e, existe, pouca diferença entre a média com e sem outliers, conforme Tabela §5.17.

Slots	Com outliers			Sem outliers		
	\bar{x}	s	CV (%)	\bar{x}	s	CV (%)
S0	0,07123	0,00294	4,12063	0,07123	0,00294	4,12063
S1*	0,06765	0,00288	4,25623	0,06736	0,00254	3,76455
S2	0,06659	0,00283	4,24588	0,06659	0,00283	4,24588
S3	0,06737	0,00225	3,34038	0,06737	0,00225	3,34038
S4	8,14529	0,18302	2,24698	8,14529	0,18302	2,24698
S5*	0,31297	0,00968	3,09136	0,31484	0,00753	2,39067
S8*	0,06740	0,00239	3,54996	0,06719	0,00219	3,25721
S9	0,06783	0,00208	3,07159	0,06783	0,00208	3,07159
S10	0,06668	0,00200	2,99223	0,06668	0,00200	2,99223
S11	0,06648	0,00228	3,43088	0,06648	0,00228	3,43088
S12	0,06353	0,00224	3,52954	0,06353	0,00224	3,52954
S13*	0,06336	0,00265	4,17761	0,06304	0,00217	3,43681
S14	0,06767	0,00160	2,35949	0,06767	0,00160	2,35949
S15*	0,06729	0,00232	3,44124	0,06682	0,00172	2,57486
S16	0,06641	0,00248	3,73434	0,06641	0,00248	3,73434

Tabela 5.15: Estatísticas da variável TF no cenário 3

Slots	Com outliers			Sem outliers		
	\bar{x}	s	CV (%)	\bar{x}	s	CV (%)
S0	0,16651	0,00379	2,27600	0,16651	0,00379	2,27600
S1	0,15636	0,00307	1,96084	0,15636	0,00307	1,96084
S2	0,15559	0,00371	2,38422	0,15559	0,00371	2,38422
S3	0,15593	0,00315	2,02324	0,15593	0,00315	2,02324
S4	6884,31031	338,98969	4,92409	6884,31031	338,98969	4,92409
S5*	0,67278	0,01979	2,94225	0,67000	0,01068	1,59437
S8	0,12619	0,00325	2,57195	0,12619	0,00325	2,57195
S9	0,12585	0,00298	2,36531	0,12585	0,00298	2,36531
S10*	0,12472	0,00380	3,04968	0,12389	0,00262	2,11558
S11	0,12475	0,00248	1,98916	0,12475	0,00248	1,98916
S12	0,11819	0,00341	2,88585	0,11819	0,00341	2,88585
S13*	0,11790	0,00354	3,00032	0,11811	0,00209	1,76566
S14*	0,12510	0,00365	2,91612	0,12477	0,00332	2,66384
S15	0,12364	0,00325	2,62887	0,12364	0,00325	2,62887
S16	0,12326	0,00264	2,13857	0,12326	0,00264	2,13857

Tabela 5.16: Estatísticas da variável TF no cenário 4

Slots	Com outliers			Sem outliers		
	\bar{x}	s	CV (%)	\bar{x}	s	CV (%)
S0*	0,49648	0,01081	2,17719	0,49531	0,00928	1,87440
S1	0,45305	0,00694	1,53113	0,45305	0,00694	1,53113
S2	0,45023	0,00859	1,90848	0,45023	0,00859	1,90848
S3*	0,45166	0,00740	1,63803	0,45026	0,00584	1,29783
S4*	25021,01283	143,48079	0,57344	25035,27918	127,17062	0,50797
S5*	0,67054	0,01260	1,87891	0,66863	0,00699	1,04507
S8*	0,12363	0,00209	1,69328	0,12409	0,00144	1,15697
S9*	0,12462	0,00206	1,65323	0,12483	0,00180	1,44093
S10*	0,12416	0,00321	2,58523	0,12387	0,00291	2,35074
S11*	0,12520	0,00307	2,45511	0,12479	0,00235	1,88475
S12*	0,11684	0,00214	1,82887	0,11660	0,00179	1,53395
S13	0,11751	0,00277	2,35595	0,11751	0,00277	2,35595
S14	0,12454	0,00353	2,83697	0,12454	0,00353	2,83697
S15	0,12391	0,00286	2,31179	0,12391	0,00286	2,31179
S16	0,12312	0,00215	1,74489	0,12312	0,00215	1,74489

Tabela 5.17: Estatísticas da variável TF no cenário 5

Slots	Com outliers			Sem outliers		
	\bar{x}	s	CV (%)	\bar{x}	s	CV (%)
S0*	107,68538	53,40441	49,59300	96,38353	34,93697	36,24787
S1*	8,76152	0,71254	8,13265	8,69324	0,41014	4,71788
S2*	8,96022	0,68127	7,60324	8,95560	0,56104	6,26467
S3	8,80448	0,68587	7,79007	8,80448	0,68587	7,79007
S4	33954,16799	160,56300	0,47288	33954,16799	160,56300	0,47288
S5*	0,67172	0,01308	1,94702	0,67334	0,01051	1,56066
S8*	0,12489	0,00340	2,71831	0,12526	0,00292	2,33063
S9	0,12559	0,00308	2,45015	0,12559	0,00308	2,45015
S10	0,12511	0,00299	2,38732	0,12511	0,00299	2,38732
S11	0,12510	0,00350	2,80062	0,12510	0,00350	2,80062
S12	0,11739	0,00244	2,07984	0,11739	0,00244	2,07984
S13*	0,11888	0,00344	2,89165	0,11853	0,00301	2,53883
S14	0,12414	0,00294	2,37174	0,12414	0,00294	2,37174
S15	0,12476	0,00218	1,74731	0,12476	0,00218	1,74731
S16	0,12409	0,00315	2,53550	0,12409	0,00315	2,53550

Tabela 5.18: Estatísticas da variável TF no cenário 6

O cenário 6 apresentou valores abaixo do limite inferior nos *slots* S1, S2, S5 e S8 e acima do limite superior no S0, S1, S2 e S13. O *slot* S0 apresentou dois valores acima do limite superior e percebe-se que esses valores extremos afetam a média, porque há uma variação significativa. Como existe uma dispersão no conjunto de dados a amostra de S0 é heterogênea. Os outros *slots* apresentam um conjunto de dados homogêneos, conforme Tabela §5.18.

No cenário 7, encontrou-se um valor abaixo do limite inferior no *slot* S4 e acima do limite superior em S0 e S5. Analisando os dados de S0 na Tabela §5.19, percebe-se que o valor extremo afeta a média, porque há uma diferença significativa entre a média com e sem outliers. O coeficiente de variação de S0 é maior que 30%, o que caracteriza uma amostra heterogênea.

Slots	Com outliers			Sem outliers		
	\bar{x}	s	CV (%)	\bar{x}	s	CV (%)
S0*	175,41430	116,02591	66,14393	159,81652	87,75306	54,90863
S1	9,11463	0,76701	8,41512	9,11463	0,76701	8,41512
S2	9,13390	0,68361	7,48430	9,13390	0,68361	7,48430
S3	9,02574	0,73997	8,19842	9,02574	0,73997	8,19842
S4*	34022,09480	129,22031	0,37981	34035,69429	112,24615	0,32979
S5*	0,67219	0,01432	2,13051	0,67078	0,01272	1,89598
S8	0,12719	0,00274	2,15484	0,12719	0,00274	2,15484
S9	0,12566	0,00322	2,56211	0,12566	0,00322	2,56211
S10	0,12530	0,00295	2,35294	0,12530	0,00295	2,35294
S11	0,12506	0,00278	2,22049	0,12506	0,00278	2,22049
S12	0,11785	0,00312	2,64550	0,11785	0,00312	2,64550
S13	0,11848	0,00408	3,44774	0,11848	0,00408	3,44774
S14	0,12594	0,00254	2,01407	0,12594	0,00254	2,01407
S15	0,12480	0,00226	1,81299	0,12480	0,00226	1,81299
S16	0,12347	0,00288	2,33416	0,12347	0,00288	2,33416

Tabela 5.19: Estatísticas da variável TF no cenário 7

Analisando as tabelas da variável, tempo médio de permanência das mensagens nos *slots*, pode-se inferir que o provável motivo da ocorrência dos outliers seja o fato dessa variável ser dependente dos processos de chegada e processamento das mensagens. Assim, os valores abaixo do limite inferior provavelmente correspondem a uma simulação na qual uma quantidade menor de mensagens chegou à solução, ou ainda em que a maioria das mensagens foram processadas de maneira rápida, se comparada às outras simulações. Dessa forma, as mensagens que terão que esperar será por pouco

tempo, ocasionando um tempo médio menor. Já os valores acima do limite superior correspondem, provavelmente, a uma simulação na qual uma quantidade maior de mensagens chega, ou ainda em que o processamento da maioria das mensagens foi demorado, aumentando assim, o tempo médio de permanência.

A Figura 5.13 representa o gráfico obtido da medida de desempenho, tempo médio de permanência das mensagens, na solução de integração, nos sete cenários. Analisando o gráfico, percebe-se que nos cenários de estado estável os valores da variável aumentam quando a taxa de chegada de mensagens aumenta. O *slot* S4 apresenta o tempo médio maior, sendo no cenário 3 de aproximadamente 8 segundos. Isso significa, que em média, as mensagens permanecem 8 segundos no *slot* até que sejam processadas pela tarefa. Os valores do S4 não constituem um gargalo nesses cenários, mas é possível perceber essa tendência, à medida que apresenta o maior valor e também uma diferença significativa comparando-se com os outros *slots*.

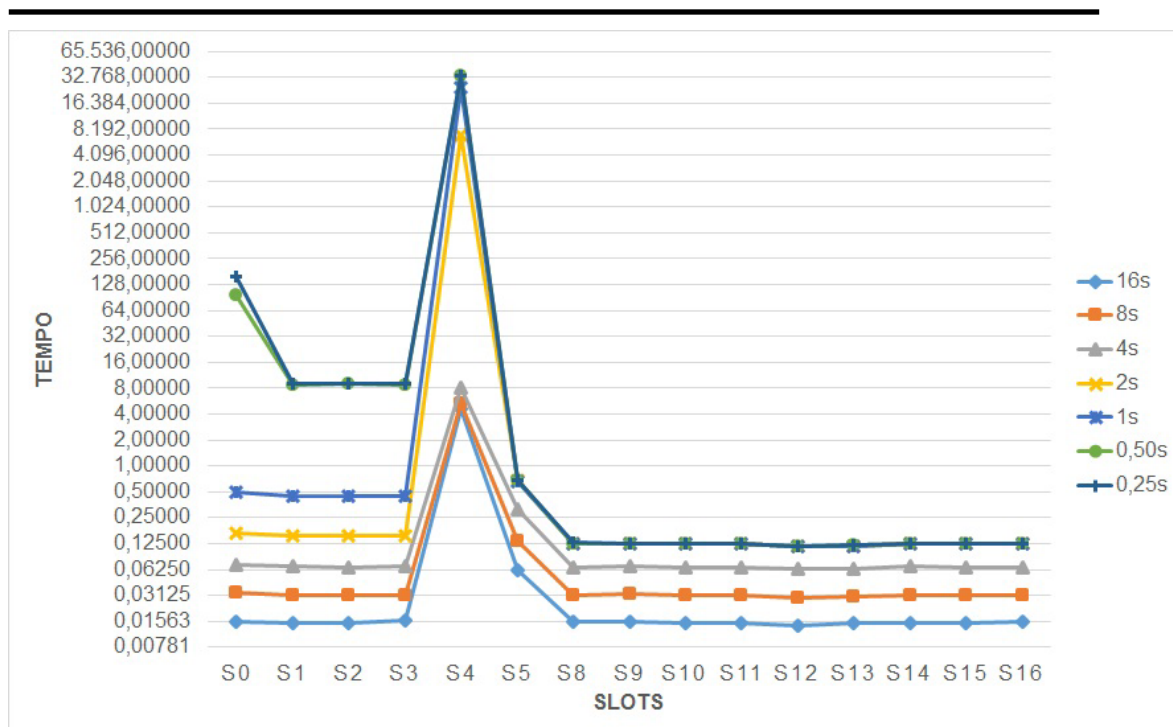


Figura 5.13: Tempo médio de permanência das mensagens nos slots

No cenário 4, a instalação que representa a aplicação Recursos Humanos

não está em estado estável ($\mu < \lambda$) e como as mensagens que chegam ao *slot* S4 dependem da mensagem correlacionada, que foi enviada para consultar a aplicação, o tempo médio de permanência aumentou consideravelmente para aproximadamente 6 884 segundos. No cenário 5, a taxa média de chegada é de uma mensagem por segundo, o que caracteriza um sistema em estado estável. Percebe-se que o tempo médio aumentou para aproximadamente 0,5 segundos no *slot* S0. Nesse contexto, é possível inferir que a solução, a uma taxa de trabalho maior, o tempo médio de permanência das mensagens nos *slots* irá aumentar e, considerando que a variável apresenta o valor maior em S4, esse *slot*, provavelmente constituirá um gargalo de desempenho.

Nos cenários 6 e 7, o processo de chegada de mensagens à solução de integração não ocorre de acordo com um sistema estável. Analisando o gráfico, percebe-se que isso influencia o comportamento da variável tempo médio nas primeiras instalações de atendimento, porque o gráfico em S0 apresenta um comportamento diferente. Dessa forma, não é adequado inferir sobre esses resultados, visto que as fórmulas da Teoria das Filas são para um sistema em estado estável.

Para analisar a variável taxa de utilização, os resultados estatísticos também foram organizados em forma de tabela. Inicialmente, aborda-se a análise sobre as estatísticas e, após, do gráfico para estimar o desempenho da solução de integração.

Nos resultados do cenário 1, foram encontrados outliers apenas abaixo do limite inferior nas tarefas T1, T7, T8 e T10. Analisando a Tabela §5.20, percebe-se que não há uma diferença significativa nas médias com e sem outliers. O coeficiente de variação é baixo para todas as tarefas, demonstrando que as amostras são homogêneas.

No cenário 2 foram encontrados valores abaixo do limite inferior nas tarefas T1, T2, T6, T7 e T10 e acima do limite superior em T2. Os valores extremos não afetam a média, à medida que não há uma diferença significativa das médias com e sem outliers. O coeficiente de variação é baixo, conforme Tabela §5.21. Isso significa que o conjunto de dados de cada tarefa é homogêneo.

O cenário 3, conforme Tabela §5.22, apresentou apenas um outlier na tarefa T5. O valor identificado estava abaixo do limite inferior.

A Tabela §5.23, apresenta as estatísticas do cenário 4, no qual foi encontrado um valor abaixo do limite inferior em T0 e um acima do limite superior na tarefa T6 e dois em T8. Já no cenário 5 foram identificados apenas valores abaixo do limite inferior, conforme Tabela §5.24.

Slots	Com outliers			Sem outliers		
	\bar{x}	s	CV (%)	\bar{x}	s	CV (%)
T0	0,03116	0,00049	1,57718	0,03116	0,00049	1,57718
T1*	0,02941	0,00046	1,55392	0,02946	0,00041	1,39086
T2	0,02967	0,00047	1,59111	0,02967	0,00047	1,59111
T3_T4	0,05921	0,00082	1,38885	0,05921	0,00082	1,38885
T5	0,02964	0,00050	1,69860	0,02964	0,00050	1,69860
T6	0,02955	0,00050	1,67879	0,02955	0,00050	1,67879
T7*	0,02959	0,00046	1,55362	0,02963	0,00042	1,41696
T8*	0,02811	0,00056	1,98415	0,02817	0,00047	1,65521
T9	0,02971	0,00050	1,69383	0,02971	0,00050	1,69383
T10*	0,02932	0,00060	2,03303	0,02939	0,00050	1,70916

Tabela 5.20: Estatísticas da variável taxa de utilização no cenário 1

Slots	Com outliers			Sem outliers		
	\bar{x}	s	CV (%)	\bar{x}	s	CV (%)
T0	0,06237	0,00072	1,14829	0,06237	0,00072	1,14829
T1*	0,05920	0,00090	1,52380	0,05930	0,00077	1,30442
T2*	0,05929	0,00085	1,42625	0,05928	0,00071	1,19511
T3_T4	0,11826	0,00128	1,08535	0,11826	0,00128	1,08535
T5	0,05918	0,00094	1,58766	0,05918	0,00094	1,58766
T6*	0,05947	0,00091	1,53106	0,05957	0,00077	1,28625
T7*	0,05925	0,00099	1,66689	0,05935	0,00087	1,46616
T8	0,05633	0,00065	1,15277	0,05633	0,00065	1,15277
T9	0,05910	0,00100	1,68986	0,05910	0,00100	1,68986
T10*	0,05886	0,00098	1,66423	0,05906	0,00073	1,22785

Tabela 5.21: Estatísticas da variável taxa de utilização no cenário 2

No cenário 6 foram encontrados valores abaixo do limite inferior nos resultados das tarefas T3_T4, T5 e T6 e acima do limite superior em T6 e T10, conforme Tabela §5.25. Já no cenário 7, as tarefas T3_T4, T5 e T9 apresentaram valores abaixo do limite inferior e T5 e T7 acima do limite superior, conforme Tabela §5.26.

Analisando as estatísticas obtidas percebe-se, que os valores extremos não afetam a média, porque não há uma variação significativa comparando as médias com e sem outliers. O coeficiente de variação é baixo. Isso

Slots	Com outliers			Sem outliers		
	\bar{x}	s	CV (%)	\bar{x}	s	CV (%)
T0	0,12491	0,00114	0,91414	0,12491	0,00114	0,91414
T1	0,11869	0,00105	0,88888	0,11869	0,00105	0,88888
T2	0,11876	0,00113	0,95324	0,11876	0,00113	0,95324
T3_T4	0,23746	0,00156	0,65488	0,23746	0,00156	0,65488
T5*	0,11892	0,00108	0,90612	0,11904	0,00090	0,75532
T6	0,11886	0,00115	0,96576	0,11886	0,00115	0,96576
T7	0,11868	0,00093	0,78041	0,11868	0,00093	0,78041
T8	0,11293	0,00094	0,83681	0,11293	0,00094	0,83681
T9	0,11875	0,00063	0,53256	0,11875	0,00063	0,53256
T10	0,11772	0,00088	0,74869	0,11772	0,00088	0,74869

Tabela 5.22: Estatísticas da variável taxa de utilização no cenário 3

significa que conjunto de dados é homogêneo.

O provável motivo para os valores abaixo do limite inferior, é uma simulação no qual uma quantidade menor de mensagens chegou à solução ou ainda em que a maioria foi processada de forma rápida. Já para os valores acima do limite superior, uma maior quantidade de mensagens chega ou o processamento foi mais demorado, se comparado as outras simulações.

Slots	Com outliers			Sem outliers		
	\bar{x}	s	CV (%)	\bar{x}	s	CV (%)
T0*	0,24972	0,00101	0,40415	0,24984	0,00082	0,32726
T1	0,23772	0,00137	0,57582	0,23772	0,00137	0,57582
T2	0,23784	0,00166	0,69787	0,23784	0,00166	0,69787
T3_T4	0,39922	0,00381	0,95323	0,39922	0,00381	0,95323
T5	0,19989	0,00159	0,79783	0,19989	0,00159	0,79783
T6*	0,19940	0,00165	0,82767	0,19920	0,00133	0,66785
T7	0,19955	0,00145	0,72691	0,19955	0,00145	0,72691
T8*	0,18984	0,00139	0,73096	0,18954	0,00095	0,49984
T9	0,19959	0,00153	0,76684	0,19959	0,00153	0,76684
T10	0,19752	0,00157	0,79236	0,19752	0,00157	0,79236

Tabela 5.23: Estatísticas da variável taxa de utilização no cenário 4

Slots	Com outliers			Sem outliers		
	\bar{x}	s	CV (%)	\bar{x}	s	CV (%)
T0*	0,50009	0,00277	0,55295	0,50041	0,00230	0,45978
T1	0,47549	0,00227	0,47795	0,47549	0,00227	0,47795
T2*	0,47465	0,00234	0,49229	0,47489	0,00205	0,43117
T3_T4	0,39891	0,00304	0,76134	0,39891	0,00304	0,76134
T5	0,19913	0,00122	0,61101	0,19913	0,00122	0,61101
T6	0,19958	0,00092	0,46234	0,19958	0,00092	0,46234
T7*	0,19949	0,00097	0,48376	0,19962	0,00075	0,37676
T8	0,18978	0,00090	0,47495	0,18978	0,00090	0,47495
T9*	0,19927	0,00148	0,74141	0,19942	0,00129	0,64539
T10*	0,19779	0,00103	0,52232	0,19801	0,00075	0,38008

Tabela 5.24: Estatísticas da variável taxa de utilização no cenário 5

Slots	Com outliers			Sem outliers		
	\bar{x}	s	CV (%)	\bar{x}	s	CV (%)
T0	0,99549	0,00200	0,20122	0,99549	0,00200	0,20122
T1	0,94490	0,00247	0,26150	0,94490	0,00247	0,26150
T2	0,94608	0,00270	0,28548	0,94608	0,00270	0,28548
T3_T4*	0,39948	0,00267	0,66824	0,39973	0,00242	0,60457
T5*	0,19980	0,00163	0,81392	0,19993	0,00153	0,76379
T6*	0,19956	0,00156	0,78120	0,19957	0,00125	0,62713
T7	0,19996	0,00161	0,80333	0,19996	0,00161	0,80333
T8	0,18987	0,00105	0,55228	0,18987	0,00105	0,55228
T9	0,19949	0,00153	0,76515	0,19949	0,00153	0,76515
T10*	0,19790	0,00147	0,74193	0,19778	0,00135	0,68443

Tabela 5.25: Estatísticas da variável taxa de utilização no cenário 6

A taxa de utilização representa a fração média do tempo em que cada tarefa está ocupada processando as mensagens, conforme Figura §5.14. No cenário 1, 2 e 3, todo o sistema encontra-se em estado estável e percebe-se que a tarefa T3_T4 apresenta a taxa de utilização maior. Isso acontece porque foi utilizado um tempo maior de processamento nessa tarefa, aumentando assim também a taxa de utilização.

No cenário 4, a taxa de utilização aumenta consideravelmente em relação ao cenário 3, de aproximadamente 12% em T0, T1 e T2 para aproxima-

Slots	Com outliers			Sem outliers		
	\bar{x}	s	CV (%)	\bar{x}	s	CV (%)
T0	0,99724	0,00209	0,21003	0,99724	0,00209	0,21003
T1	0,94790	0,00268	0,28311	0,94790	0,00268	0,28311
T2	0,94785	0,00252	0,26569	0,94785	0,00252	0,26569
T3_T4*	0,39991	0,00194	0,48624	0,40009	0,00176	0,44113
T5*	0,20027	0,00104	0,51725	0,20021	0,00062	0,30784
T6	0,19975	0,00127	0,63337	0,19975	0,00127	0,63337
T7*	0,19979	0,00120	0,60041	0,19966	0,00102	0,51032
T8	0,19017	0,00137	0,71927	0,19017	0,00137	0,71927
T9*	0,20014	0,00136	0,68057	0,20030	0,00113	0,56456
T10	0,19831	0,00115	0,57778	0,19831	0,00115	0,57778

Tabela 5.26: Estatísticas da variável taxa de utilização no cenário 7

mente 24% nessas tarefas. No cenário 5, as instalações de T0, T1 e T2 estão em estado estável. A taxa de ocupação da tarefa T0 é de 50%, caindo para aproximadamente 47% nas tarefas T1 e T2, isso porque nessas tarefas chegam menos mensagens, porque 5% das mensagens processadas em T0 são filtradas.

Analisando o gráfico dos cenários de estado estável percebe-se que, à medida que a taxa de entrada de mensagens na solução aumenta a taxa de utilização das tarefas aumenta. Isso significa que com um carga de trabalho maior, as tarefas ficam mais tempo ocupadas, processando as mensagens.

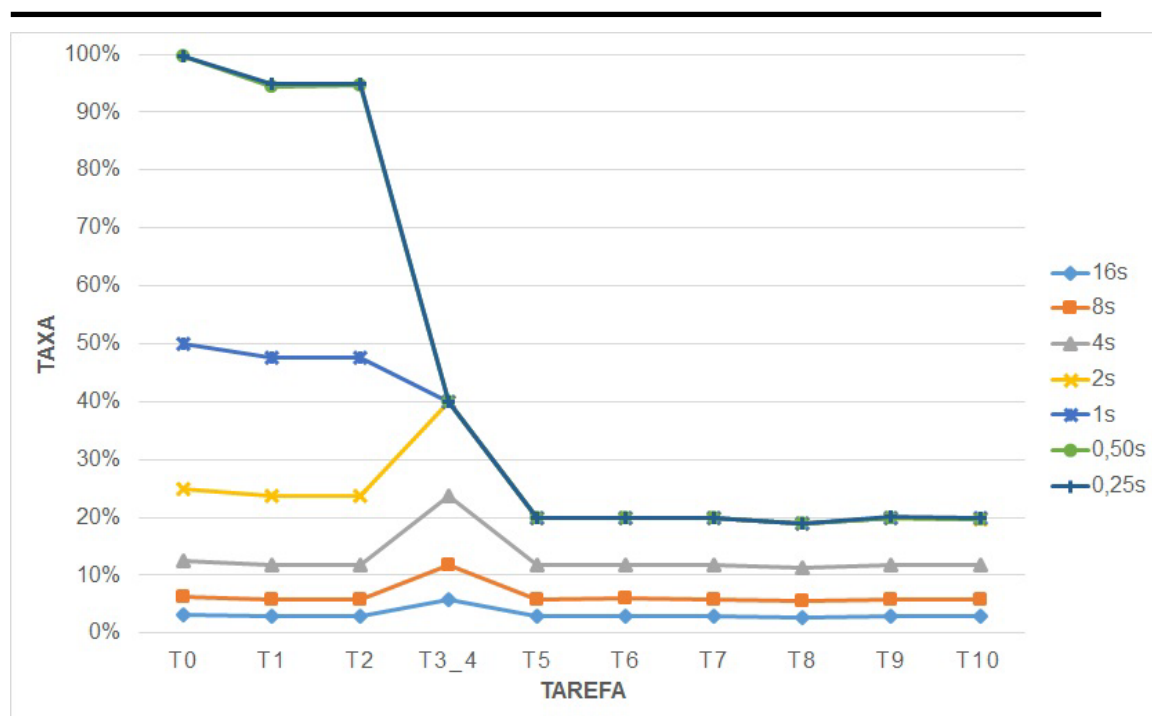


Figura 5.14: Taxa de utilização das tarefas

A taxa ociosa é obtida a partir da taxa de utilização e representa a fração média do tempo em que cada tarefa está desocupada. Isso significa o tempo em que não está processando nenhuma mensagem. Percebe-se que a taxa ociosa é o contrário da taxa de ocupação.

Analisando a Figura 5.15, que representa o gráfico da taxa ociosa das tarefas, percebe-se que nos três primeiros cenários as tarefas permanecem a maior parte do tempo ociosas. Nos outros cenários, percebe-se que as tarefas T5, T6, T7, T8, T9 e T10 apresentam aproximadamente o mesmo valor. Isso acontece porque algumas instalações não se encontram em estado estável. Assim, nessas instalações, a quantidade de mensagens nesses slots tendem a ser infinitas e muitas mensagens ficam retidas. Dessa forma, mesmo que a taxa de entrada de mensagens seja maior que nos cenários anteriores, essa quantidade de mensagens não chega nessas tarefas. Assim, não tendo mensagem para processar, a tarefa permanece o maior tempo ociosa.

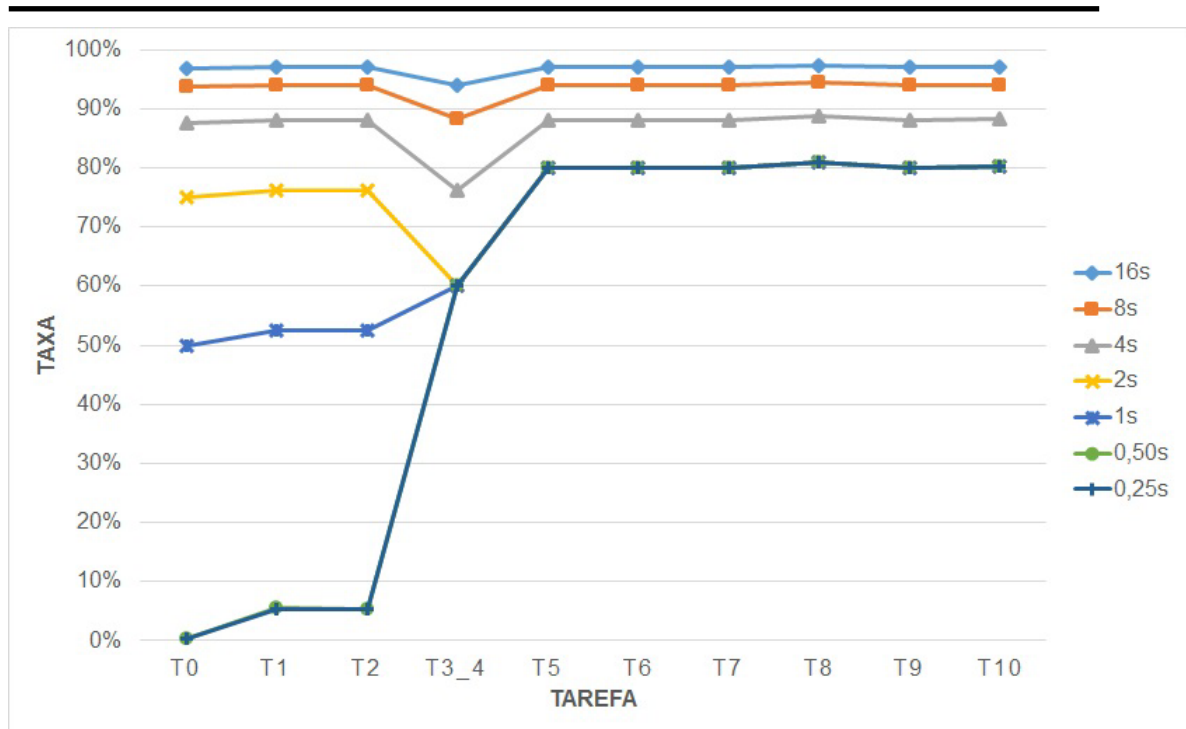


Figura 5.15: Taxa ociosa das tarefas

5.5 Resumo do Capítulo

Esse capítulo abordou o desenvolvimento da pesquisa desta dissertação. Para caracterizar a solução de integração como um sistema de eventos discretos, demonstrou-se a equivalência dos elementos do Guaraná com os elementos de um sistema de filas. A solução de integração, caracterizada como um sistema de eventos discretos, possibilita que os modelos conceituais sejam simulados para prever o comportamento da solução, sob diferentes cargas de trabalho. Nesse contexto, identificou-se as variáveis cujos valores podem ser utilizados para obter as medidas de desempenho da solução de integração. Um sistema de filas é classificado de acordo com suas características e a solução de integração pode ser representada por um modelo de filas $M/M/1$, cujas fórmulas foram deduzidas porque são utilizadas para obter as medidas de desempenho. Para desenvolver um modelo de simulação equivalente ao modelo conceitual da solução, estudou-se detalhadamente a biblioteca da ferramenta SimEvents e identificou-se os blocos para representar as funcionalidades do *slot* e das tarefas. O modelo de simulação formal proposto foi simulado para sete cenários diferentes, para obter as medidas de desempenho da solução sob diferentes cargas de trabalho. Isso possibilitou analisar o comportamento e identificar possíveis gargalos de desempenho.

Capítulo 6

Verificação e Validação

A dúvida é o princípio da sabedoria.

Aristóteles, filósofo grego (384 a.C. - 322 a.C.)

O desenvolvimento de um modelo de simulação computacional requer a segurança e a garantia de que o modelo esteja livre de erros da implementação computacional e que seja representativo do sistema real. Essas duas etapas são chamadas de verificação e validação do modelo, respectivamente. Para realizar a verificação e a validação de um modelo são utilizadas técnicas formais descritas na literatura. A Seção §6.1 apresenta a definição de verificação e validação e algumas técnicas que podem ser utilizadas para verificar e validar os modelos de simulação computacional. Na Seção §6.2, é apresentada a verificação e na Seção §6.3 a validação do modelo de simulação formal proposto. A Seção §6.4 apresenta o resumo do capítulo.

6.1 Definição de Verificação e Validação

Os modelos de simulação imitam o funcionamento do sistema e são utilizados para estudar o comportamento e prever a performance do sistema, em diferentes cenários de funcionamento. Durante o desenvolvimento de um modelo de simulação, é necessário estar seguro de que o mesmo está sendo implementado corretamente. Isso significa estar livre de erros de sintaxe e/ou lógica e ter acurácia suficiente para ser utilizado como substituto do sistema, para a análise e experimentação. Essas duas etapas são conhecidas como verificação e validação de um modelo.

O processo de verificação e validação dos modelos de simulação é fundamental para que a experimentação e análise dos resultados da simulação sejam bem-sucedidas. No desenvolvimento de um modelo de simulação, normalmente é realizada uma série de pressupostos e simplificações sobre o comportamento do sistema real. Portanto, qualquer tomada de decisão com base nos resultados da simulação deve ser precedida de uma avaliação de sua qualidade e apropriação [16].

A verificação ocorre com a finalidade específica de encontrar e corrigir erros de modelagem. Refere-se às técnicas utilizadas para assegurar que o modelo está correto e corresponde a qualquer simplificação e pressupostos realizados. Portanto, a verificação consiste em certificar que o modelo está livre de erros da implementação computacional.

A validação ocorre quando o desenvolvedor do modelo e pessoas conhecedoras do sistema real trabalham em conjunto para analisar e avaliar como o modelo funciona. Refere-se às técnicas utilizadas para assegurar que o modelo, apesar dos pressupostos e simplificações, representa o sistema real a um nível suficiente de precisão [4]. Nas fases de verificação e validação, muitas vezes são encontrados erros que exigem modificações significativas no modelo. Após a correção, o modelo deve ser novamente verificado e validado.

De acordo com Carson et al. [4], deve-se notar que nenhum modelo é 100% verificado ou validado. Um modelo é uma representação do sistema, e o comportamento do modelo é, na melhor das hipóteses, uma aproximação do comportamento do sistema real. A afirmação de que o modelo foi verificado ou validado refere-se a série de tarefas realizadas explicitamente para verificar e validar o modelo, com o grau necessário para os objetivos propostos.

Segundo Jain [21], um modelo de simulação computacional, após o desenvolvimento, pode se encontrar em uma das seguintes categorias:

- i. Não validado e não verificado;
- ii. Não validado e verificado;
- iii. Validado e não verificado;
- iv. Validado e verificado.

Para verificar e validar o modelo de simulação formal proposto, é necessário ter conhecimento das técnicas de verificação e validação descritas na literatura. Assim, inicialmente realizou-se uma revisão bibliográfica, com o objetivo de conhecer as técnicas e identificar quais poderiam ser utilizadas para garantir que o modelo está livre de erros e tem um comportamento semelhante à solução de integração.

As técnicas de verificação e validação, embora descritas de forma levemente diferente, são abordadas por autores como Sargent [35], Whitner e Balci [42], Jain [21] e Freitas Filho [16].

6.1.1 Técnicas de Verificação

De acordo com Freitas Filho [16], as técnicas de verificação baseiam-se nos métodos de correção de programas (*debugging*) presentes na literatura que trata do desenvolvimento de programas de computadores e nos procedimentos especialmente aplicáveis para modelos de simulação.

Algumas dessas técnicas para a verificação de modelos de simulação computacional são descritas a seguir:

- **Modelos determinísticos:** A variação dos resultados, gerados pelas variáveis aleatórias, utilizadas para representar os sistemas estocásticos, pode dificultar o trabalho de verificação dos modelos de simulação. A análise e a busca de erros em modelos determinísticos é, evidentemente, mais fácil do que naqueles com variáveis aleatórias, porque é mais fácil determinar os resultados das variáveis de saída. Assim, uma técnica bastante útil, consiste em atribuir valores determinísticos a essas variáveis, para fins de verificação e busca de erros.

- **Variações sobre os dados de entrada:** Essa técnica de análise bastante comum, consiste em variar os valores dos parâmetros de entrada e verificar se as respostas obtidas são adequadas e condizentes com as variações realizadas. Para o desenvolvedor do modelo, a utilização dessa técnica de verificação evidencia que o modelo está bem programado ao reagir às modificações de acordo com a forma esperada. Por exemplo, ao reduzir o tempo entre chegadas das entidades no sistema, o esperado é que o número médio de entidades na fila aumente.
- **Rotinas de rastreamento (Trace):** Uma rotina de rastreamento consiste em uma lista dos eventos ocorridos, ordenados pelo tempo de sua ocorrência. A lista apresenta, também, as modificações ocorridas nas variáveis associadas e atributos dependentes. Essa técnica é utilizada para rastrear o comportamento de uma entidade por meio do modelo, com o propósito de determinar se a lógica do modelo é correta e se a precisão necessária é obtida. Portanto, é útil para a verificação de erros, principalmente quanto à lógica do modelo.
- **Teste de continuidade:** Essa técnica consiste em executar a simulação várias vezes e a cada rodada de simulação realizar uma pequena alteração nos valores dos parâmetros de entrada. Para qualquer parâmetro, uma pequena alteração, geralmente deve produzir apenas uma pequena mudança nos resultados. Toda mudança mais radical nos resultados deve ser investigada, porque normalmente são provocadas por erros de modelagem.
- **Testes de degenerescência:** Esses testes consistem em verificar se o modelo funciona para valores extremos do sistema. Para testar o comportamento do modelo aplica-se nos parâmetros valores menores ou maiores do que os permitidos. Embora que esses casos extremos muitas vezes podem não ser representativos da realidade, eles ajudam na descoberta de erros que, em condições normais de funcionamento, não apareceriam. Por exemplo, quando a taxa média de chegada é maior que a taxa de atendimento, o número médio de entidades na fila de um único atendente aumenta continuamente ao longo do tempo?
- **Verificação de consistência:** Os testes de consistência são utilizados para verificar se o modelo produz resultados similares, quando em diferentes execuções são atribuídos valores equivalentes para os parâmetros de entrada. Quando os resultados do modelo apresentam diferenças significativas e não encontra-se justificativas consistentes para explicar essas diferenças, o programa contém erros de modelagem.

- **Independência de Sementes:** As sementes que são utilizadas na geração de números aleatórios não podem influenciar os resultados. Assim, para realizar a verificação, o modelo de simulação é executado com diferentes valores de sementes e deve produzir resultados semelhantes, nas diversas simulações realizadas.
- **Rotinas de verificação:** São elementos extras adicionados ao programa ou modelo para apontar erros, se existirem. Por exemplo, a contabilidade de entidades geradas e servidas. Nesse caso, a diferença entre o número de entidades geradas e o número de entidades que saíram, acrescidos daquelas nas filas, deve ser igual a zero. Um resultado diferente de zero indica erro de programação ou modelagem.

6.1.2 Técnicas de Validação

De acordo com Jain [21], a validação consiste em assegurar que os pressupostos e simplificações utilizados no desenvolvimento do modelo sejam razoáveis, à medida que, corretamente implementados, o modelo tem um comportamento e produz resultados semelhantes àqueles observados nos sistemas reais. O processo de validação depende dos pressupostos e, conseqüentemente do sistema que será modelado, pois, considera que, de alguma maneira, o modelador estará comparando o modelo com o sistema real.

Esse aspecto traz à tona alguns problemas. A impossibilidade de obter-se um modelo idêntico ao sistema real, considerando que ele exista. Aliás, normalmente o desenvolvimento de um modelo visa, justamente, à análise e a observação do sistema (existe ou não), em condições diferenciadas daquelas observadas na prática. A maioria das análises procura verificar o desempenho de inúmeras e diversificadas alternativas para um novo projeto (baseado em um sistema real existente ou não) ou, ainda, a simulação de um modelo de um sistema real sobre o qual se aplicam formas de operação e gerência totalmente diferenciadas. Nessas situações, mesmo que o sistema real exista, são poucas as possibilidades para a realização de comparações detalhadas para fins de validação [16].

Segundo Jain [21], a validação do modelo consiste em validar os três aspectos fundamentais do modelo:

- i. As simplificações e os pressupostos adotados;
- ii. Os valores utilizados nos parâmetros de entrada e as distribuições;

iii. Os resultados (valores de saída) e conclusões.

Cada um desses três aspectos podem ser submetidos a um teste de validade, utilizando algumas das técnicas descritas a seguir:

- **Intuição de especialistas:** Nessa técnica, são realizadas reuniões do tipo *brainstorming* com a participação das pessoas conhecedoras do sistema. Participam das reuniões as pessoas envolvidas no projeto, arquitetura, implementação, análise, marketing ou manutenção do sistema, dependendo, naturalmente, da fase em que está a modelagem do sistema. Os pressupostos, os valores de entrada e as distribuições, bem como os resultados (se disponíveis) são apresentados e discutidos na reunião. Na prática, é melhor validar os três aspectos – pressupostos, entrada e resultados – separadamente, conforme o desenvolvimento do modelo avança, em vez de esperar até o final. Os pressupostos do modelo devem ser analisados e discutidos em uma reunião *brainstorming*, assim que um projeto preliminar do modelo de simulação for desenvolvido. Os valores de entrada e as distribuições devem ser validados durante o desenvolvimento e os resultados tão logo um modelo executável verificado esteja disponível.
- **Teste de Turing:** Consiste na apresentação a especialistas dois conjuntos de resultados; um vindo do sistema real e outro do modelo de simulação. Caso o especialista não consiga distinguir os relatórios emitidos pelo modelo de simulação daqueles advindos do sistema real, então, o modelo poderá ser considerado válido.
- **Medições Obtidas em Sistema Reais:** A comparação com sistemas reais é a maneira mais confiável e preferencial para validar um modelo de simulação. Essa técnica de validação implica na comparação de dois conjuntos de observações: um proveniente do modelo e outro do sistema real. No entanto, na prática, a utilização dessa técnica é muitas vezes inviável, porque os sistemas reais podem não existir ou as medições podem ser dispendiosas para serem realizadas.
- **Comparação com outros modelos:** Essa técnica consiste em comparar vários resultados do modelo de simulação que está sendo validado com os de outros modelos válidos. Como por exemplo, casos simples de um modelo de simulação são comparados com resultados conhecidos de modelos analíticos, ou o modelo pode ser comparado com outros modelos de simulação validados.

Jain [21] salienta que “a validação completa de um modelo de simulação é um mito”. Na realidade, segundo o autor, é apenas possível demonstrar que o modelo não é inválido para algumas das situações comparadas. O objetivo principal da validação é aumentar a confiança nos resultados apresentados pelo modelo de simulação, para que se possa inferir com a máxima confiança sobre o comportamento do sistema real [16].

6.2 Verificação do Modelo de Simulação

Para a verificação do modelo de simulação formal proposto, foi utilizada a técnica de rotinas de verificação. De acordo com Freitas Filho [16], manter uma contabilidade sobre o número de entidades geradas e retiradas do modelo, durante a simulação, constitui uma rotina de segurança essencial. Jain [21] afirma que a técnica é muito útil, porque pequenas alterações no modelo, muitas vezes resultam em erros, que são descobertos pela contabilidade das entidades no final da simulação.

Essa técnica, possibilita que várias características do modelo sejam analisadas, de modo que são utilizados resultados de várias variáveis, contabilizados na verificação. Os dados do número de entidades geradas, nas filas, filtradas e que chegaram nos blocos que representam as aplicações Folha de Pagamento, Serviço de Mensagens e Servidor de E-mail, foram necessários para aplicar essa técnica. Esses dados foram obtidos a partir das saídas estatísticas dos blocos, no modelo de simulação.

Para facilitar a verificação em todos os cenários propostos, formulou-se equações. As variáveis utilizadas nas equações são:

NG = Número de entidades geradas na aplicação Central Telefônica

NF_{CT} = Número de entidades na fila da aplicação Central Telefônica

QF_{T0} = Quantidade de entidades filtradas em T0

NF_{RH1} = Número de entidades na fila para serem processadas na aplicação Recursos Humanos

NF_{RH2} = Número de entidades na fila da aplicação Recursos Humanos para retornar a solução

QF_{T7} = Quantidade de entidades filtradas em T7

QF_{T9} = Quantidade de entidades filtradas em T9

NF_{FP} = Número de entidades na fila da aplicação Folha de Pagamento

NF_{SM} = Número de entidades na fila da aplicação Serviço de Mensagens

NF_{SE} = Número de entidades na fila da aplicação Servidor de E-mail

NC_{FP} = Número de entidades que chegaram na aplicação Folha de Pagamento

NC_{SM} = Número de entidades que chegaram na aplicação Serviço de Mensagens

NC_{SE} = Número de entidades que chegaram na aplicação Servidor de E-mail

FI = Somatório das variáveis no fluxo inicial até T1

FC_{RH} = Somatório das variáveis no fluxo da cópia da entidade enviada para a aplicação Recursos Humanos

AC = Agrupa/junta as cópias oriundas de T1

Para o número de entidades nos blocos que representam os *slots* foram utilizadas, respectivamente, as seguintes variáveis: N_{S0} , N_{S1} , N_{S2} , N_{S3} , N_{S4} , N_{S5} , N_{S8} , N_{S9} , N_{S10} , N_{S11} , N_{S12} , N_{S13} , N_{S14} , N_{S15} , N_{S16} .

Para tornar a equação final mais simples e objetiva somou-se as variáveis do fluxo inicial das entidades até T1.

$$FI = NF_{CT} + N_{S0} + Q_{F_{T0}} + N_{S1} \quad (6.1)$$

O bloco T1 cria duas cópias de cada entidade que chega. Uma entidade é enviada para a aplicação Recursos Humanos, a outra para S4. Essas duas cópias são combinadas e resultam em apenas uma entidade em T3-T4. Assim, foi desenvolvida uma equação para que o número de entidades, nesse fluxo, não ficassem duplicadas no processo de verificação.

$$FC_{RH} = N_{S2} + N_{S3} + NF_{RH1} + NF_{RH2} + N_{S5} \quad (6.2)$$

$$AC = \frac{FC_{RH} + N_{S4}}{2} \quad (6.3)$$

A trajetória das entidades, no fluxo do modelo, termina em três pontos distintos; nas aplicações Folha de Pagamento, Serviço de Mensagens e Servidor de E-mail. Assim, para obter a garantia que o modelo está livre de erros

foi criada uma equação para verificar cada uma das três saídas do modelo. A equação para a verificação do modelo nas três saídas, aplicações Folha de Pagamento, Serviço de Mensagens e Servidor de E-mail, respectivamente, é :

$$V_1 = NG - (FI + AC + N_{S8} + N_{S9} + N_{S10} + NF_{FP} + NC_{FP}) \quad (6.4)$$

$$V_2 = NG - (FI + AC + N_{S8} + N_{S11} + QF_{T7} + N_{S12} + N_{S13} + NF_{SM} + NC_{SM}) \quad (6.5)$$

$$V_3 = NG - (FI + AC + N_{S8} + N_{S14}QF_{T9} + N_{S15} + N_{S16} + NF_{SE} + NC_{SE}) \quad (6.6)$$

Foi realizada uma verificação para cada cenário. Os resultados para o cenário 1 estão descritos a seguir, e dos outros cenários estão disponíveis no Apêndice [§A](#). Em todos os cenários, calculou-se, inicialmente FI e FC_{RH} para obter AC e substituir os valores nas equações V_1 , V_2 e V_3 .

$$FI = 0 + 0 + 269 + 0$$

$$FI = 269$$

$$FC_{RH} = 0 + 0 + 0 + 0 + 0$$

$$FC_{RH} = 0$$

$$AC = \frac{0 + 0}{2}$$

$$AC = 0$$

$$V_1 = 5.378 - (269 + 0 + 0 + 0 + 0 + 0 + 5.109)$$

$$V_1 = 5.378 - 5.378$$

$$V_1 = 0$$

$$V_2 = 5.378 - (269 + 0 + 0 + 0 + 257 + 0 + 0 + 0 + 4.852)$$

$$V_2 = 5.378 - 5.378$$

$$V_2 = 0$$

$$V_3 = 5.378 - (269 + 0 + 0 + 0 + 42 + 0 + 0 + 0 + 5.067)$$

$$V_3 = 5.378 - 5.378$$

$$V_3 = 0$$

A aplicação da técnica de verificação garante que o modelo está livre de erros, à medida que a diferença do número de entidades geradas com o número de entidades nas filas, filtradas e que saíram, em todos cenários, é zero.

6.3 Validação do Modelo de Simulação

Freitas Filho [16] afirma que, de acordo com vários autores, a análise e a avaliação de modelos por parte de especialistas é a melhor e a mais simples forma de validá-los. Para a validação do modelo de simulação formal proposto, utilizou-se a técnica da intuição de especialistas, porque as outras técnicas requerem resultados da solução de integração implementada e modelos validados, que ainda não existem.

O processo de validação ocorreu desde a fase de concepção até a fase de experimentação. Durante esse processo, foram realizadas várias reuniões *brainstorming*, nas quais foram avaliados, inicialmente, os pressupostos adotados. Como a solução de integração não está implementada, não é possível coletar dados da solução de integração para identificar a distribuição de probabilidade mais adequada a expressar o comportamento da variável aleatória tempo entre chegadas de mensagens à solução e processamento de mensagens nas tarefas.

A identificação da distribuição de probabilidade a ser empregada no modelo de simulação e a estimação de seus parâmetros são duas etapas muito importantes. Assim, após um estudo de algumas distribuições de probabilidade, a análise dos modelos da Teoria das Filas e o conhecimento do especialista de domínio, realizou-se os seguintes pressupostos:

- Distribuição exponencial para o tempo entre chegadas de mensagens;
- Distribuição exponencial para o tempo de processamento das mensagens nas tarefas;
- Definição de 16, 8, 4, 2, 1, 0,5 e 0,25 segundos para o tempo entre chegadas de mensagens, parâmetro da distribuição exponencial, nos sete cenários;
- Definição de 0,5 segundos para o tempo de processamento das mensagens, parâmetro da distribuição exponencial nas portas e em todas as tarefas, exceto T3-T4 no qual foi utilizado 1 segundo. Para a aplicação Recursos Humanos utilizou-se 2,5 segundos;

- Porcentagem de mensagens filtradas como sendo 5% nas tarefas T0 e T7 e de 1% em T9.

A distribuição exponencial e os parâmetros de entrada para o tempo de atendimento e tempo entre chegadas, nos sete cenários, foram discutidos e definidos após análises realizadas em várias reuniões, com a participação do especialista de domínio. Após a verificação do modelo, foram realizadas algumas simulações e os resultados examinados e analisados. Identificou-se que apenas o *slot* S4 provavelmente constituirá um gargalo de desempenho, atestando que os resultados correspondem ao que é esperado pelo especialista de domínio, tornando o modelo de simulação proposto válido.

6.4 **Resumo do Capítulo**

Nesse capítulo, abordou-se a definição de verificação e validação de modelos de simulação e também algumas das técnicas apresentadas na literatura. A aplicação de uma ou várias técnicas de verificação garante que o modelo está livre de erros da implementação computacional. A utilização de uma ou várias técnicas de validação procura garantir que o modelo tenha um comportamento semelhante ao do sistema modelado. A verificação e a validação do modelo de simulação formal proposto trouxe segurança para a realização da pesquisa desta dissertação.

Capítulo 7

Conclusões e Trabalhos Futuros

*Nenhum trabalho de qualidade pode ser feito
sem concentração e auto-sacrifício, esforço e dúvida.*

Max Beerbohm, Caricaturista e escritor inglês (1872-1956)



As empresas adquirem ou desenvolvem aplicações para apoiar a tomada de decisões e aperfeiçoar seus processos de negócio. Estas aplicações compõem o ecossistema de software da empresa, que geralmente é heterogêneo. No entanto, a maioria das aplicações foram projetadas sem a possibilidade de serem reutilizadas. Uma solução de integração visa reutilizar as aplicações, manter em sincronia os dados e as funcionalidades ou produzir novas funcionalidades a partir das existentes. A tecnologia Guaraná oferece um conjunto de ferramentas e metodologias para projetar, implementar e executar soluções de integração de aplicações empresariais.

O sucesso das empresas em seus processos de negócios depende da execução correta e eficiente da solução de integração. Atualmente, a abordagem adotada pelos engenheiros de software, para analisar o comportamento de uma solução de integração frente a cenários críticos de funcionamento, depende das atividades relacionadas com a construção, execução da solução e o recolhimento de dados a partir da execução. A estas atividades estão relacionados custos, riscos e tempo de desenvolvimento. A abordagem proposta, neste trabalho, tem como objetivo analisar a performance de uma solução de integração ainda na fase de projeto, por meio da simulação dos modelos conceituais.

Nesse contexto, essa dissertação propôs uma abordagem diferente por meio da simulação, utilizando o formalismo da Teoria das Filas para analisar o comportamento e identificar possíveis gargalos de desempenho de soluções de integração, projetadas na tecnologia Guaraná, tendo como base seus modelos conceituais. A Teoria das Filas é utilizada para estudar sistemas de filas, formulando modelos matemáticos. A equivalência dos elementos do Guaraná com os elementos de um sistema de filas demonstrou que pode-se utilizar o formalismo da Teoria das Filas para analisar uma solução de integração.

Na literatura técnica não identificou-se trabalhos que analisaram soluções de integração de aplicações empresariais por meio da simulação e/ou Teoria das Filas. De acordo com a revisão bibliográfica, uma solução de integração de aplicações empresariais pode ser caracterizada como um sistema cujo modelo é classificado como estocástico, dinâmico e discreto. A literatura dispõe de diversos trabalhos que utilizaram a Teoria das Filas e a simulação de modelos para avaliar a eficiência de sistemas de eventos discretos. Nesse contexto, esse trabalho trouxe importantes contribuições para o desenvolvimento de pesquisas na área de EAI.

A caracterização de uma solução de integração como sistema de eventos discretos, demonstrada por meio da equivalência dos elementos do Guaraná com os elementos de um sistema de filas e da relação entre seus elementos, tornou possível a utilização de técnicas de simulação de eventos discretos e modelos matemáticos computacionais para analisar o comportamento, identificar possíveis gargalos de desempenho e prever o comportamento futuro da solução.

Para a simulação do modelo conceitual, resultante de um problema real de integração da Central Telefônica da Unijuí, utilizou-se a ferramenta SimEvents, que fornece mecanismos para simulação de eventos discretos, sendo um componente da biblioteca do Simulink, no Matlab. A ferramenta possui várias bibliotecas com blocos pré-definidos, tais como filas e servidores que possibilitam a análise de desempenho de sistemas de filas, com base no formalismo matemático da Teoria das Filas. A partir do estudo dos blocos pré-definidos, disponíveis nas bibliotecas da ferramenta, conseguiu-se demonstrar que o modelo conceitual da solução de integração pode ser traduzido em um modelo de simulação equivalente, permitindo a simulação para identificar as características da solução.

Antes de realizar a simulação para coletar os dados, é preciso verificar e validar o modelo. A aplicação de técnicas de verificação e validação formais garantem e asseguram, respectivamente, que o modelo está livre de erros da

implementação computacional e que, apesar dos pressupostos e simplificações, representa a solução de integração a um nível suficiente de precisão. As saídas estatísticas dos blocos do modelo de simulação, permitiram acompanhar e analisar as medidas de desempenho da solução; número médio de mensagens nos *slots*, tempo médio das mensagens nos *slots*, taxa de utilização e taxa ociosa das tarefas. A análise destas medidas permitiu a identificação do comportamento da solução em diferentes taxas de trabalho e a ocorrência de possíveis gargalos de desempenho nessa solução de integração. A simulação de soluções de integração reduz os custos, tornando possível a identificação de possíveis falhas da solução, ainda na fase de projeto, a partir do modelo conceitual. Assim, não é necessário implementar e executar a solução para realizar testes.

A abordagem de simulação apresentada, nesse trabalho, para a análise do comportamento e a identificação de possíveis gargalos de desempenho de soluções de integração, pode melhorar a qualidade das soluções de integração desenvolvidas, utilizando a tecnologia Guaraná. Percebeu-se que a área de EAI é pouco estudada, utilizando o método da simulação. Portanto, esse trabalho permite a abertura de linhas de pesquisa no âmbito da simulação de soluções de integração de aplicações empresariais.

Esse trabalho consiste em um etapa inicial na linha de pesquisa que aborda a análise do comportamento e a identificação de possíveis gargalos de desempenho em soluções de integração projetadas na tecnologia Guaraná, por meio da simulação de eventos discretos, baseado na Teoria das Filas. Como trabalhos futuros, sugere-se implementar a solução de integração para a validação do modelo de simulação comparando com dados reais. A implementação, também irá possibilitar a coleta de dados para identificar as distribuições de probabilidades que caracterizam, respectivamente, o processo de chegada e o processamento das mensagens. A partir destas informações, será possível determinar qual modelo de filas representa a solução real implementada, podendo ser o modelo M/M/1 ou algum outro da Teoria das Filas. Esse processo consiste na validação dos parâmetros de entrada do modelo de simulação.

Para tornar a execução do modelo de simulação mais semelhante ao do motor de execução da solução de integração, sugere-se estudar se existe uma forma de todos os servidores não serem executados de forma paralela. A solução de integração é equivalente a uma rede de filas, e nesse trabalho, não conseguiu-se obter os dados das variáveis identificadas para a solução; número médio de mensagens (NS) e tempo médio de permanência das mensagens na solução (TS). Um trabalho futuro de simulação pode englo-

bar o estudo destas variáveis, como também realizar a simulação utilizando outras distribuições de probabilidade.

A abordagem da simulação, nesse trabalho, limitou-se ao estudo de uma solução de integração. O objetivo final dessa linha de pesquisa é a criação de um modelo de simulação para qualquer solução de integração, de forma que com essa automatização seja possível simular qualquer solução de integração de aplicações empresariais, projetada na tecnologia Guaraná.

Apêndice A

Resultados da verificação dos cenários 2–7

Para o cenário 2, obteve-se:

$$FI = 0 + 0 + 531 + 0$$

$$FI = 531$$

$$FC_{RH} = 0 + 0 + 0 + 0 + 0$$

$$FC_{RH} = 0$$

$$AC = \frac{0 + 0}{2}$$

$$AC = 0$$

$$V_1 = 10.723 - (531 + 0 + 0 + 0 + 0 + 0 + 10.192)$$

$$V_1 = 10.723 - 10.723$$

$$V_1 = 0$$

$$V_2 = 10.723 - (531 + 0 + 0 + 0 + 497 + 0 + 0 + 0 + 9.695)$$

$$V_2 = 10.723 - 10.723$$

$$V_2 = 0$$

$$V_3 = 10.723 - (531 + 0 + 0 + 0 + 88 + 0 + 0 + 0 + 10.104)$$

$$V_3 = 10.723 - 10.723$$

$$V_3 = 0$$

Para o cenário 3, obteve-se:

$$FI = 0 + 0 + 1.071 + 0$$

$$FI = 1.071$$

$$FC_{RH} = 0 + 0 + 5 + 0 + 0$$

$$FC_{RH} = 5$$

$$AC = \frac{5 + 5}{2}$$

$$AC = 5$$

$$V_1 = 21.694 - (1.071 + 5 + 0 + 0 + 0 + 0 + 20.618)$$

$$V_1 = 21.694 - 21.694$$

$$V_1 = 0$$

$$V_2 = 21.694 - (1.071 + 5 + 0 + 0 + 1.003 + 0 + 0 + 0 + 19.615)$$

$$V_2 = 21.694 - 21.694$$

$$V_2 = 0$$

$$V_3 = 21.694 - (1.071 + 5 + 0 + 0 + 191 + 0 + 0 + 0 + 20.427)$$

$$V_3 = 21.694 - 21.694$$

$$V_3 = 0$$

Para o cenário 4, obteve-se:

$$FI = 0 + 0 + 2.095 + 0$$

$$FI = 2.095$$

$$FC_{RH} = 0 + 0 + 6.026 + 0 + 0$$

$$FC_{RH} = 6.026$$

$$AC = \frac{6.026 + 6.026}{2}$$

$$AC = 6.026$$

$$V_1 = 43.104 - (2.095 + 6.026 + 0 + 0 + 0 + 0 + 34.983)$$

$$V_1 = 43.104 - 43.104$$

$$V_1 = 0$$

$$V_2 = 43.104 - (2.095 + 6.026 + 0 + 1 + 1.703 + 0 + 0 + 0 + 33.279)$$

$$V_2 = 43.104 - 43.104$$

$$V_2 = 0$$

$$V_3 = 43.104 - (2.095 + 6.026 + 0 + 0 + 349 + 0 + 0 + 0 + 34.634)$$

$$V_3 = 43.104 - 43.104$$

$$V_3 = 0$$

Para o cenário 5, obteve-se:

$$FI = 3 + 0 + 4.272 + 0$$

$$FI = 4.275$$

$$FC_{RH} = 0 + 0 + 48.076 + 0 + 0$$

$$FC_{RH} = 48.076$$

$$AC = \frac{48.076 + 48.076}{2}$$

$$AC = 48.076$$

$$V_1 = 86.571 - (4.275 + 48.076 + 0 + 0 + 0 + 0 + 34.220)$$

$$V_1 = 86.571 - 86.571$$

$$V_1 = 0$$

$$V_2 = 86.571 - (4.275 + 48.076 + 0 + 0 + 1.670 + 0 + 0 + 0 + 32.550)$$

$$V_2 = 86.571 - 86.571$$

$$V_2 = 0$$

$$V_3 = 86.571 - (4.275 + 48.076 + 0 + 0 + 345 + 0 + 0 + 0 + 33.875)$$

$$V_3 = 86.571 - 86.571$$

$$V_3 = 0$$

Para o cenário 6, obteve-se:

$$FI = 806 + 253 + 8.641 + 3$$

$$FI = 9.703$$

$$FC_{RH} = 25 + 8 + 129.030 + 0 + 0$$

$$FC_{RH} = 129.063$$

$$AC = \frac{129.063 + 129.063}{2}$$

$$AC = 129.063$$

$$V_1 = 173.268 - (9.703 + 129.063 + 0 + 0 + 0 + 0 + 34.502)$$

$$V_1 = 173.268 - 173.268$$

$$V_1 = 0$$

$$V_2 = 173.268 - (9.703 + 129.063 + 0 + 0 + 1.683 + 0 + 0 + 0 + 32.819)$$

$$V_2 = 173.268 - 173.268$$

$$V_2 = 0$$

$$V_3 = 173.268 - (9.703 + 129.063 + 0 + 0 + 346 + 0 + 0 + 0 + 34.156)$$

$$V_3 = 173.268 - 173.268$$

$$V_3 = 0$$

Para o cenário 7, obteve-se:

$$FI = 172.791 + 1.006 + 8.637 + 2$$

$$FI = 182.436$$

$$FC_{RH} = 7 + 3 + 128.821 + 0 + 0$$

$$FC_{RH} = 128.831$$

$$AC = \frac{128.831 + 128.831}{2}$$

$$AC = 128.831$$

$$V_1 = 345.883 - (182.436 + 128.831 + 0 + 0 + 0 + 0 + 34.616)$$

$$V_1 = 345.883 - 345.883$$

$$V_1 = 0$$

$$V_2 = 345.883 - (182.436 + 128.831 + 0 + 0 + 1.691 + 0 + 0 + 0 + 32.925)$$

$$V_2 = 345.883 - 345.883$$

$$V_2 = 0$$

$$V_3 = 345.883 - (182.436 + 128.831 + 0 + 0 + 347 + 0 + 0 + 0 + 34.269)$$

$$V_3 = 345.883 - 345.883$$

$$V_3 = 0$$

Bibliografia

- [1] E. O. Abensur, I. Brunstein, A. Fishmann e L. L. HO. *Tendências para o auto-atendimento bancário brasileiro: um enfoque estratégico baseado na teoria das filas*. *Revista de Administração Mackenzie*, 4(2), 2008.
- [2] A. L. Bruni. *Estatística aplicada à gestão empresarial*. São Paulo: Atlas, 4, 2007.
- [3] G. R. Camelo, A. S. Coelho, R. M. Borges e R. M. de Souza. *Teoria das filas e da simulação aplicada ao embarque de minério de ferro e manganês no terminal marítimo de ponta da madeira*. *Cadernos do IME-Série Estatística*, 29(2):1, 2010.
- [4] J. S. Carson et al.. *Model verification and validation*. Em *Simulation Conference, 2002. Proceedings of the Winter*, v. 1, 52–58. IEEE, 2002.
- [5] L. Changfu e L. Zhenyu. *Research of transaction request handling queuing system in the e-business environment based on queuing theory*. Em *Information Processing, 2009. APCIP 2009. Asia-Pacific Conference on*, v. 2, 589–592. IEEE, 2009.
- [6] L. Chwif e A. Medina. *Modelagem e simulação de eventos discretos, 4ª edição: Teoria e aplicações*, v. 4. Elsevier Brasil, 2014.
- [7] D. Dossot, J. D’Emic e V. Romero. *Mule in action*. Manning, 2014.
- [8] F. E. Doy, G. Bressan, G. H. d. A. Pereira e M. N. Magalhães. *Simulação do serviço de correio eletrônico através de um modelo de filas*. *Pesquisa Operacional*, 26(2):241–253, 2006.
- [9] M. Fisher, J. Partner, M. Bogoevici e I. Fuld. *Spring integration in action*. Manning Publications Co., 2012.
- [10] M. C. Fogliatti e N. M. C. Mattos. *Teoria de filas*. Rio de Janeiro: Interciência, 1–290, 2007.

- [11] M. Fowler. *Domain-specific languages*. Pearson Education, 2010.
- [12] R. Z. Frantz. *Enterprise application integration: an easy-to-maintain model-driven engineering approach*. Tese Doutoral, Universidad de Sevilla, 2012.
- [13] R. Z. Frantz, R. Corchuelo e F. Roos-Frantz. *On the design of a maintainable software development kit to implement integration solutions*. *Journal of Systems and Software*, 111:89–104, 2016.
- [14] R. Z. Frantz, A. M. Reina Quintero e R. Corchuelo. *A domain-specific language to design enterprise application integration solutions*. *International Journal of Cooperative Information Systems*, 20(02):143–176, 2011.
- [15] R. Z. Frantz, S. Sawicki, F. Roos-Frantz, R. Corchuelo, V. Basto-Fernandes e I. Hernández. *Desafios para a implantação de soluções de integração de aplicações empresariais em provedores de computação em nuvem*. Em *XIX Jornada de Pesquisa*, 1–11, 2014.
- [16] P. J. d. Freitas Filho. *Introdução à modelagem e simulação de sistemas: com aplicações em arena*. Visual Books, 2001.
- [17] C. M. Grinstead e J. L. Snell. *Introduction to probability*. American Mathematical Soc., 2012.
- [18] F. S. Hillier e G. J. Lieberman. *Introdução à pesquisa operacional*. McGraw Hill, 2010.
- [19] G. Hohpe e B. Woolf. *Enterprise integration patterns: Designing, building, and deploying messaging solutions*. Addison-Wesley Professional, 2004.
- [20] C. Ibsen e J. Anstey. *Camel in action*. Manning Publications Co., 2010.
- [21] R. Jain. *The art of computer systems performance analysis*, v. 182. John Wiley & Sons Chichester, 1991.
- [22] S. H. Kamali, M. Hedayati, A. S. Izadi e H. R. Hoseiny. *The monitoring of the network traffic based on queuing theory and simulation in heterogeneous network environment*. Em *Computer Technology and Development, 2009. ICCTD'09. International Conference on*, v. 1, 322–326. IEEE, 2009.

- [23] W. D. Kelton, R. P. Sadowski e D. A. Sadowski. *Simulation with arena*, v. 47. WCB/McGraw-Hill New York, 1998.
- [24] D. G. Kendall. *Stochastic processes occurring in the theory of queues and their analysis by the method of the imbedded markov chain. The Annals of Mathematical Statistics*, 338–354, 1953.
- [25] K. Laudon e J. Laudon. *Sistemas de informação gerenciais*. Pearson Prentice Hall, 2010.
- [26] A. M. Law e W. D. Kelton. *Simulation modeling and analysis*. McGraw Hill Boston, 2000.
- [27] D. S. Linthicum. *Enterprise application integration*. Addison-Wesley Professional, 2000.
- [28] J. D. Little. *A proof for the queuing formula: $L = \lambda w$* . *Operations research*, 9(3):383–387, 1961.
- [29] D. G. Messerschmitt e C. Szyperski. *Software ecosystem: Understanding an indispensable technology and industry (chapter 6: Organization of the software value chain)*. 424 pages, 2003.
- [30] R. J. Paul e D. W. Balmer. *Simulation modelling*. Chartwell-Bratt, 1993.
- [31] C. D. Pegden, R. P. Sadowski e R. E. Shannon. *Introduction to simulation using siman*. McGraw-Hill, Inc., 1995.
- [32] D. S. d. Prado. *Teoria das filas e da simulação*. Belo Horizonte, MG: Editora FALCONI, 2, 2014.
- [33] R. S. Pressman. *Engenharia de software*. McGraw Hill Brasil, 2011.
- [34] D. A. Rezende. *Engenharia de software e sistemas de informação*. Brasport, 2005.
- [35] R. G. Sargent. *Verification and validation of simulation models*. *Journal of simulation*, 7(1):12–24, 2013.
- [36] S. Sawicki, R. Z. Frantz, V. M. B. Fernandes, F. Roos-Frantz, I. Yevseyeva e R. Corchuelo. *Characterising enterprise application integration solutions as discreteevent system*. Em *Handbook of Research on Computational Simulation and Modeling in Engineering*, 255–282. IGI Global, 2015.
- [37] D. C. Schmidt. *Guest editor’s introduction: Model-driven engineering*. *Computer*, 39(2):0025–31, 2006.

- [38] T. J. Schriber. *Simulation using gpss*. Relatório técnico, DTIC Document, 1974.
- [39] I. Sommerville. *Engenharia de software*. Pearson Addison Wesley São Paulo, 2003.
- [40] J. Tukey. *Tukey exploratory data analysis*. Reading, MA: Addison, Weseley, 1977.
- [41] R. S. Wazlawick. *Análise e projeto de sistemas da informação orientados a objetos*. Elsevier Brasil, 2011.
- [42] R. B. Whitner e O. Balci. *Guidelines for selecting and using simulation model verification techniques*. Em *Proceedings of the 21st conference on Winter simulation*, 559–568. ACM, 1989.
- [43] A. K. Wiesner e R. Z. Frantz. *Modelagem matemática e computacional para identificação de gargalos de desempenho em soluções de integração*. Em *III Seminário de Formação Científica e Tecnológica*, 11–12, 2015.
- [44] A. K. Wiesner, G. Freytag, R. Z. Frantz, S. Sawicki e F. Roos-Frantz. *Teoria das filas como técnica matemática para representar o comportamento de uma solução de integração empresarial*. Em *Salão do Conhecimento*, 1–5, 2015.

This document was typeset on May 26, 2016 using class RC-BOR α 2.14 for L^AT_EX₂ ϵ . As of the time of writing this document, this class is not publicly available. Only members of [The Distributed Group \(TDG\)](#) and the [Applied Computing Research Group \(ACR\)](#) are allowed to typeset their documents using this class.