
MODELAGEM E SIMULAÇÃO DE UMA SOLUÇÃO DE INTEGRAÇÃO DO SIS- TEMA DE CURRÍCULOS DA AGÊNCIA NACIONAL DE CIÊNCIA E TECNOLO- GIA DE PORTUGAL PARA A REDUÇÃO DE GARGALOS DE DESEMPENHO

ALEXSANDRO QUEIROZ LENCINA
UNIVERSIDADE REGIONAL DO NOROESTE DO
ESTADO DO RIO GRANDE DO SUL

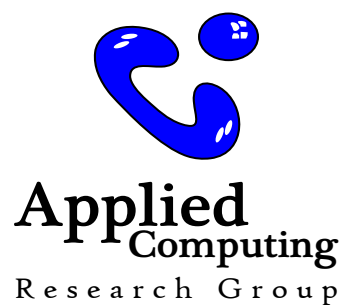
DISSERTAÇÃO DE MESTRADO

ORIENTADORA:

DRA. FABRICIA CARNEIRO ROOS FRANTZ

COORIENTADOR:

DR. VITOR MANUEL BASTO FERNANDES



MARÇO, 2017

First published in March 2017 by
Applied Computing Research Group - GCA
Department of Exact Sciences and Engineering
Rua Lulu Ilgenfritz, 480 - São Geraldo
Ijuí, 98700-000, Brazil.

Copyright © MMXVII 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: Bolsa de mestrado concedida pela CAPES. A pesquisa desenvolvida neste trabalho também teve o apoio dos projetos do Grupo de Pesquisa em Computação Aplicada.

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 do sistema de currículos da agência nacional de Ciência e Tecnologia de Portugal para a redução de gargalos de desempenho: ”, elaborada por Alexsandro Queiroz Lencina, como requisito parcial para a obtenção do título de Mestre em Modelagem Matemática.

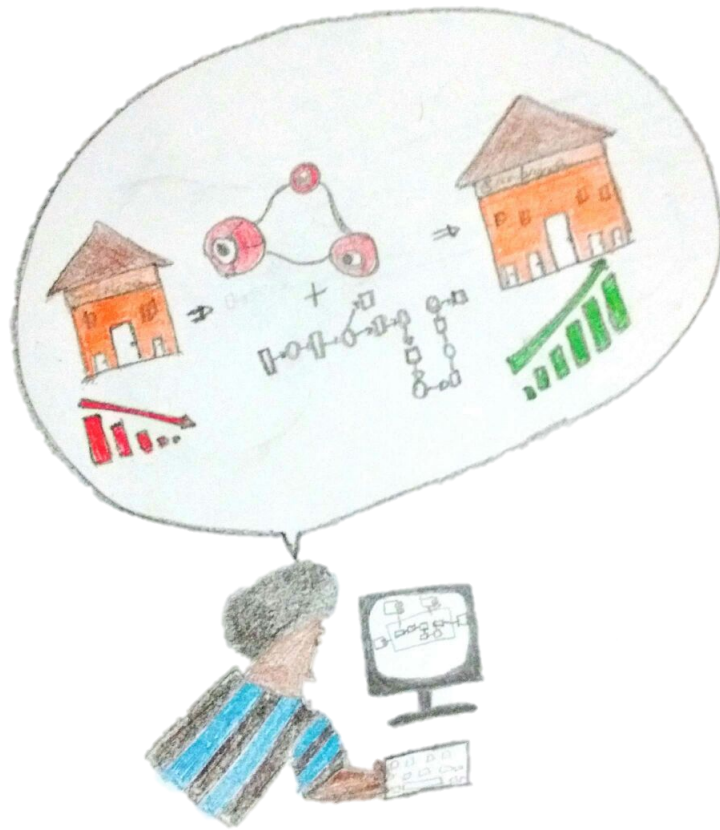
Dra. Fabricia Carneiro Roos Frantz
UNIJUÍ
(Orientadora)

Dra. Luciana de Oliveira Rech
UFSC

Dr. Luiz Antonio Rasia
UNIJUÍ

Dr. Rafael Zancan Frantz
UNIJUÍ

Ijuí, ____ de _____ de _____.



Entendimento da pesquisa por Henrique, 11 anos de idade.

Dedico este trabalho a minha esposa, amiga, incentivadora, Daniele Bonapace Lencina, por sempre estar me apoiando, mesmo nas horas mais difíceis e dando todo suporte para que eu ficasse tranquilo, principalmente com relação a pouca atenção que destinei neste processo para os nossos filhos.

Conteúdo

Agradecimentos	vii
Resumo	ix
Abstract	xi
1 Introdução	1
1.1 Contexto da Pesquisa	1
1.2 Motivação	3
1.3 Objetivos	3
1.3.1 Geral	3
1.3.2 Específicos	3
1.4 Metodologia	4
1.5 Resumo das Contribuições	5
1.6 Estrutura dessa Dissertação	6
2 Revisão da Literatura	9
2.1 Integração de Aplicações Empresariais	9
2.1.1 Estilos de Integração	10
2.1.2 Topologias	14
2.1.3 Plataformas de Integração	18
2.2 Plataforma Guaraná	19
2.2.1 Guaraná DSL	19
2.2.2 Notação Gráfica	20
2.2.3 Exemplo de Solução de Integração	21
2.3 Simulação de Eventos Discretos	22
2.3.1 Sistemas	22
2.3.2 Modelos de Sistemas	24

2.3.3	Simulação	27
2.4	Redes de Petri	28
2.4.1	Formalismo Algébrico	29
2.4.2	Representação Gráfica	30
2.4.3	Redes de Petri Estocásticas	33
2.4.4	Redes de Petri Estocásticas Generalizadas	34
2.5	Distribuição de Probabilidade	34
2.6	Trabalhos Relacionados	35
2.6.1	Redes de Petri e Simulação	35
2.7	Resumo do Capítulo	37
3	Modelagem	39
3.1	Caso de Estudo	39
3.1.1	Ecosistema de Software	40
3.1.2	Modelo Conceitual de Integração	42
3.2	Equivalência entre redes de Petri e Guaraná	44
3.3	Modelo de Simulação Proposto	47
3.4	Resumo do Capítulo	51
4	Experimentação	53
4.1	Experimento	53
4.1.1	Variáveis Observadas	54
4.1.2	Apresentação da Ferramenta	55
4.2	Resultados e Discussão	57
4.3	Verificação do Modelo de Simulação	63
4.4	Resumo do Capítulo	69
5	Conclusões e Trabalhos Futuros	71
	Bibliografia	73

Índice de figuras

1.1	Solução de Integração. [14].	2
2.1	Transferência de Arquivos.	12
2.2	Banco de dados.	12
2.3	Chamada de Procedimento Remoto.	13
2.4	Sistema de Mensagens.	14
2.5	Point-to-point.	16
2.6	BUS.	17
2.7	Hub and Spoke.	17
2.8	Solução de Integração do Problema do Café. [15]	23
2.9	Possibilidade de Estudo dos Sistemas ([25])	24
2.10	Modelo Taxonômico [25].	26
2.11	Elementos das redes de Petri[8].	30
2.12	Exemplo das redes de Petri [8].	31
2.13	Representação de arco valorado [8].	31
2.14	Exemplo de disparo de uma transição [8].	32
3.1	Esquema DeGois®CV XML.[42]	40
3.2	Esquema Pesquisadores.XML. [42].	41
3.3	Modelo Conceitual. [42].	43
3.4	Semelhança entre troca de estados [38].	45
3.5	Modelo de Simulação das redes de Petri Estocásticas.	48
3.6	Modelo de Simulação das redes de Petri Estocásticas Generalizadas.	50
3.7	Tarefa Enquirer	51
4.1	Interface Gráfica PIPE2	56
4.2	Aba de Sensibilização das Transições Estocásticas e Estocásticas Generalizadas do PIPE2	57

4.3	Interface Simulation PIPE2	58
4.4	Gráfico de Formação de Filas nos Slots	62
4.5	Gráfico de Formação de Filas nos Slots	63
4.6	Diagrama de Aplicação da Técnica de Sargent na SPN	65
4.7	Diagrama de Aplicação da Técnica de Sargent na GSPN	66

Índice de tabelas

2.1	Notação Gráfica do Guaraná DSL (adaptada de Frantz et al. [14]) ...	20
3.1	Comparação entre os Elementos do Guaraná e redes de Petri [38]. ..	44
3.2	Tarefas Comunicadores	45
3.3	Tarefas Mapeadores	46
3.4	Tarefas Roteadoras	46
3.5	Tarefas Modificadoras	47
3.6	Tabela de Equivalência entre Guaraná DSL e redes de Petri.	49
4.1	Tabela de Resultados das Simulações	59
4.2	Resultados Experimentais das SPN.	60
4.3	Resultados Experimentais das GSPN.	61
4.4	Aplicação da Técnica de Verificação na SPN.[40]	67
4.5	Aplicação da Técnica de Verificação na GSPN.[40]	68
4.6	Aplicação da Técnica de Validade de Eventos.[40]	68

Agradecimentos

Agradeço todas as dificuldades que enfrentei;
não fosse por elas, eu não teria saído do lugar.
As facilidades nos impedem de caminhar.
Mesmo as críticas nos auxiliam muito.

Chico Xavier

Agradeço primeiramente a DEUS pela vida, pela família, pela proteção nas estradas, pela companhia, pela saúde, pela paciência enfim por tudo que me proporcionaste, para eu chegar até aqui.

Ao meu PAI Perseverando, exemplo, sempre me incentivando, sempre deixando claro que qualquer decisão que eu tomasse elas teriam consequências, tanto positivas quanto negativas.

A minha MÃE Rejeine, por toda força, todas as orações, todas as palavras de incentivo, por estar sempre do meu lado em todas as horas da minha vida e sempre participando das minhas decisões.

Aos meus irmãos Alessandro, Alisson e Henrique, que todas as vezes em que precisei, pude contar com eles e eles sabem que podem contar sempre comigo.

Aos meus filhos Laura e Alex Filho, pois todos os meus passos tem um rumo, e o rumo é poder proporcionar uma vida digna para eles, eles são o meu oxigênio.

À Profª Fabrícia, minha orientadora, pelas palavras firmes e corretas, pelas correções, pelos ensinamentos, por contribuir com a minha formação como pesquisador.

Ao Prof^o Rafael, Prof^o Sandro e Prof^o Vitor, pelas aulas, pelas reuniões, pelas metas impostas no desenrolar desta dissertação, com certeza o nosso grupo de pesquisa está bem alicerçado com os senhores no comando.

Aos colegas do mestrado, que dividiram o peso da pós-graduação ao longo desses dois anos, pelas amizades e companheirismo que levaremos para a vida.

À Geni, pessoa fundamental no processo de qualificação, pois é a parte doce, amiga, meiga, sempre com palavras certas para os momentos mais difíceis, sem falar na sua competência em conduzir a parte burocrática do programa.

À Direção do Instituto Estadual Osvaldo Cruz, na pessoa da Prof^o Edriane, que auxiliou no início do mestrado, fazendo diversas trocas de períodos e sempre me incentivando, que se dependesse dela eu seria mestre.

Ao Instituto Federal Farroupilha, pela oportunidade que dá aos seus docentes na busca de qualificação.

À CAPES, pela bolsa de estudos, investindo assim no desenvolvimento da ciência.

Resumo

*O sábio nunca diz tudo o que pensa,
mas pensa sempre tudo o que diz.*

Aristóteles



grande maioria das empresas possui um ecossistema de software composto por um conjunto de aplicações, para auxiliar na tomada de decisões ou na melhoria dos seus processos de negócio. O campo de estudo Integração de Aplicações Empresariais oferece ferramentas e metodologias para criar soluções de integração que possibilitam que as aplicações do ecossistema, geralmente heterogêneas, possam trabalhar de maneira conjunta e sincronizada. Esta dissertação utiliza como caso de estudo uma solução de integração projetada para o problema de integração do sistema de currículos da agência nacional e tecnologia de Portugal. Esta solução, projetada com uma linguagem de modelagem de alto nível de abstração, é caracterizada como um sistema de eventos discretos. Existem vários paradigmas para a modelagem de sistemas de eventos discretos, dentre eles aqueles baseados em modelos probabilísticos. Nesta dissertação, utiliza-se uma abordagem baseada em simulação com redes de Petri Estocásticas para a identificação de gargalos de desempenho em soluções de integração, caracterizando, assim, uma solução como um processo probabilístico. Para a redução desses gargalos utilizou-se as redes de Petri Estocásticas Generalizadas. A simulação de uma solução de integração, a partir do seu modelo conceitual, possibilita a análise do seu comportamento ainda na fase de projeto, reduzindo custos, riscos e tempo no desenvolvimento da solução. Os resultados da simulação mostram que os gargalos de desempenho podem ser identificados ainda na fase de projeto e que é possível reduzi-los dedicando recursos computacionais às tarefas mais críticas do modelo.

Palavras-chave: Integração de Aplicações Empresariais. Redes de Petri Estocásticas. Simulação. Gargalos de Desempenho.

Abstract

*The wise never say everything he thinks,
but he always thinks everything he says.*

Aristoteles

The vast majority of companies have a software ecosystem composed of a heterogeneous set of applications to aid in making decisions or improve the performance of their business processes. Enterprise Application Integration research field offers tools and methodologies to build integration solutions that allows for applications, usually heterogeneous, work together in a synchronized way. This dissertation uses as case study an integration solution designed for the integration problem of the curriculum system of the national agency and technology of Portugal. This solution, designed with a high level modelling language, is characterized as a discrete event system. There are several paradigms for the modeling of discrete event systems, among them those based on probabilistic models. In this dissertation, we employ a simulation-based approach with Stochastic Petri nets to identify performance bottlenecks in integration solutions, thus characterizing a solution as a probabilistic process. We propose a reduction of these bottlenecks, by using Generalized Stochastic Petri nets. The simulation of an integration solution, from its conceptual model, enables performance analysis still in the design phase, reducing costs, risks and time during solution development. The simulation results show that performance bottlenecks can still be identified in the design phase of an integration solution and that it is possible to reduce these bottlenecks by devoting computational resources to the more critical tasks of the model.

Key-words: Enterprise Application Integration. Stochastic Petri Nets. Simulation. Performance Bottlenecks.

Capítulo 1

Introdução

*Se todos estão indo adiante juntos,
então o sucesso encarrega-se de si mesmo.*

Henri Ford

A introdução trata da problematização da pesquisa desenvolvida e está dividida em 6 Seções. A Seção §1.1 está relacionada com o contexto da pesquisa, voltado as aplicações empresariais, a Seção §1.2 aborda a motivação para o desenvolvimento da pesquisa, a Seção §1.3 comenta sobre os objetivos gerais e objetivos específicos da pesquisa, a Seção §1.4 aborda a metodologia utilizada no trabalho baseada no *framework* de referência divididas em 4 fases, início, elaboração, construção e transição, a Seção §1.5 apresenta um resumo das contribuições desta pesquisa para a modelagem de soluções de integração e a Seção §1.6 encerra o capítulo apresentando a estrutura de organização utilizada para a dissertação.

1.1 Contexto da Pesquisa

Atualmente, a maioria das organizações possui um conjunto de aplicações de software que dão suporte a suas atividades de negócio, o qual é denominado ecossistema de software. Grande parte dessas aplicações precisam evoluir ao longo do tempo para manter sua capacidade de suporte às demandas das organizações, e muitas vezes novas funcionalidades devem ser acrescentadas [12]. Esses ecossistemas são compostos, normalmente,

por aplicações heterogêneas, concebidas sem levar em conta sua possível integração, podendo apoiar campos diversos, tais como contabilidade, comunicação com clientes e gestão de recursos humanos.

Na área de Engenharia de Software, o campo de estudos conhecido como Integração de Aplicações Empresariais [21] busca proporcionar metodologias, técnicas e ferramentas para a concepção e a implementação de soluções de integração. Em termos gerais, uma solução de integração tem como objetivo orquestrar um conjunto de aplicações para mantê-las sincronizadas ou proporcionar novas funcionalidades que possam ser construídas a partir daquelas já existentes. Conforme a Figura §1.1, uma solução de integração está composta por processos que contém lógica de integração e portas de comunicação, que conectam processos ou aplicações do ecossistema à solução de integração.

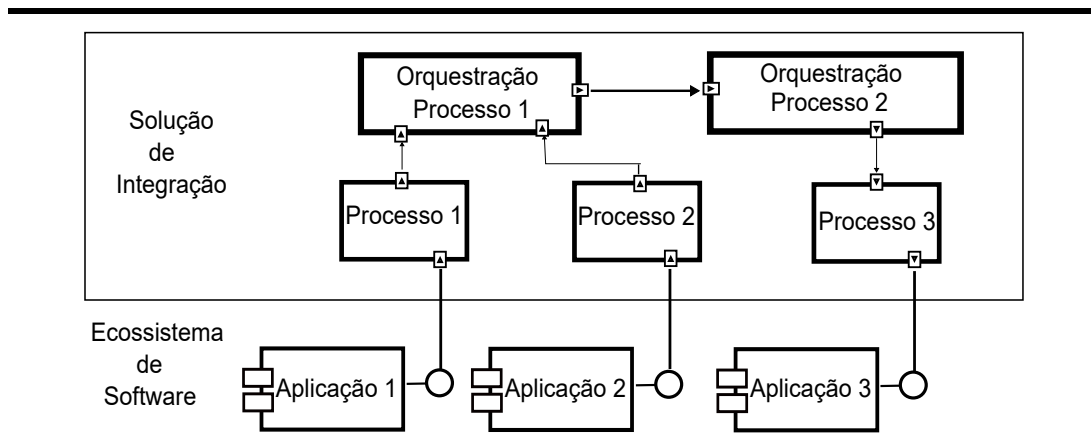


Figura 1.1: Solução de Integração. [14].

O objeto de estudo desta dissertação é a simulação da solução de integração proposta por Sequeira em [42], para conhecer e analisar seu comportamento, com o intuito de identificar possíveis gargalos de desempenho e propor um forma de redução destes gargalos. A solução integra aplicações no contexto do sistema de currículos da agência nacional de ciência e tecnologia de Portugal. A partir do modelo conceitual da solução, desenvolveu-se um modelo de simulação baseado em redes de Petri Estocásticas. Para a redução dos gargalos de desempenho propõe-se a extensão do modelo de simulação, utilizando redes de Petri Estocásticas Generalizadas (GSPN).

A simulação realizou-se na ferramenta *Platform Independent Petri Net Editor* (PIPE). Os resultados mostram que há possíveis gargalos de desempenho no comportamento da solução. O resultado da simulação do modelo

utilizando as GSPN mostra que quando as tarefas que geram gargalos são tratadas com prioridade, há a possibilidade de redução desses gargalos.

1.2 Motivação

A necessidade de integração fomentou o surgimento de várias tecnologias de integração de aplicações. Em um ecossistema de software, composto geralmente por aplicações heterogêneas, as tecnologias para a integração de aplicações mostram-se potencialmente efetivas.

A análise do comportamento e a identificação de gargalos de desempenho em soluções de integração de aplicações geralmente envolvem sua construção para posterior execução e teste frente à cenários críticos projetados pelos engenheiros de software. Por demandar que as soluções sejam construídas, tal abordagem traz consigo custos (tempo, recurso) e riscos (*bugs*) inerentes que costumam ser elevados.

A presente pesquisa é motivada pela possibilidade de análise do comportamento, identificação e redução de possíveis gargalos de desempenho ainda na fase de projeto. A partir de um modelo conceitual de uma solução de integração é possível definir uma rede de Petri equivalente. Esta linguagem de modelagem possui base formal e ferramentas automáticas de análise.

1.3 Objetivos

A seguir, apresenta-se os objetivos desta dissertação:

1.3.1 Geral

Analisar o comportamento, identificar e propor uma redução de gargalos de desempenho de uma solução de integração de aplicações a partir de seu modelo conceitual.

1.3.2 Específicos

- Dissertar sobre tecnologias de integração de aplicações, em especial sobre a linguagem de modelagem da tecnologia de integração Guaraná;
- Desenvolver um modelo de simulação para o caso de estudo apresentado;

- Prever o comportamento da solução do caso de estudo por meio da simulação baseada em Redes de Petri Estocásticas;
- Identificar gargalos de desempenho ainda na fase de projeto da solução de integração;
- Apresentar uma proposta para a redução dos gargalos de desempenho ainda na fase de projeto;
- Realizar a verificação do modelo de simulação.

1.4 Metodologia

Para o desenvolvimento desta pesquisa será utilizada como metodologia de trabalho o *framework* de referência fornecido pelo *Unified Process* (UP). O seu ciclo de vida iterativo e incremental é apropriado para o desenvolvimento de projetos que visam alcançar um grande dinamismo, pois a todo momento é possível ter o *feedback* de outros grupos de trabalho ou projetos relacionados, porém sempre mantendo os riscos que podem ocorrer no cotidiano sob controle.

De acordo com *framework UP*, a divisão deste trabalho de pesquisa será dividida em quatro fases:

INÍCIO: a etapa inicial deste trabalho, consiste em fazer uma revisão bibliográfica para assim compreender o que será realizado, identificar os trabalhos publicados na área, perceber quais são os pontos da pesquisa que apresentam maior relevância baseados em um estudo de caso, quais são os principais riscos que podem afetar o seu desenvolvimento e optar sobre qual o melhor processo a ser seguido.

ELABORAÇÃO: Parte dos objetivos desta fase consiste em identificar e descrever os principais blocos de trabalho e suas atividades de trabalho foram as seguintes: i) identificação das tarefas do modelo conceitual do caso de estudo; ii) identificação da equivalência entre as tarefas do modelo conceitual e os elementos da rede de Petri Estocásticas; iii) geração do modelo de simulação; iv) realizar simulações, v) identificar os possíveis gargalos de desempenho e vi) executar a verificação do sistema. A partir dos dados obtidos criar outro modelo de simulação com redes de Petri Estocásticas Generalizadas, executar a simulação, realizar a verificação dos modelos com base em técnicas e métodos disponíveis na literatura e realizar a análise e comparação do comportamento dos dois modelos.

CONSTRUÇÃO: Esta fase irá dividir as atividades dos blocos de trabalho em pequenas etapas que sejam abordáveis de forma iterativa, visando aprofundar cada etapa da elaboração dentro de um cronograma determinado. O número de etapas e os objetivos específicos em cada uma, bem como marcos intermediários, são acordados pelos pesquisadores envolvidos em cada atividade. A forma de abordar, com êxito, esta fase é mediante a uma reunião inicial onde os professores orientadores apresentam as instruções referente a fase seguinte ao cronograma seguido e são planejadas as tarefas, reuniões periódicas para apresentação de resultados e seminários de pesquisa nos quais os mestrandos apresentam os progressos alcançados nas fases anteriores, sejam eles resultados ou conclusões sobre a bibliografia estudada.

TRANSIÇÃO: Nesta fase os resultados serão incorporados às demais tecnologias já desenvolvidas pelos pesquisadores do grupo GCA envolvidos, procura-se aplicá-las em projetos reais em colaboração com empresas locais no nosso estado e/ou empresas parceiras em âmbito internacional, as quais o grupo já tem histórico de colaboração. Também busca-se participar de conferências e/ou fazer visitas a outros grupos para a divulgação dos resultados, consolidando e ampliando a rede de colaborações acadêmicas. Como objetivo desta etapa, obter informações que possam ser úteis para melhorar nossos métodos de trabalho e assim abordar melhor futuros projetos de pesquisa.

1.5 **Resumo das Contribuições**

Esta pesquisa faz parte de um projeto do Grupo de Pesquisa em Computação Aplicada (GCA), o qual propõe a simulação de soluções de integração com o objetivo de analisar o comportamento das soluções utilizando uma abordagem matemática computacional.

As principais contribuições da pesquisa são apresentadas a seguir:

- As redes de Petri como técnica para modelagem e representação formal dos modelos conceituais desenvolvidos na tecnologia Guaraná;
- Construção de um modelo de simulação equivalente ao modelo conceitual representado na linguagem Guaraná DSL;
- Simulação da rede de Petri para análise de comportamento, identificação e redução de possíveis gargalos de desempenho;
- Verificação do modelo de simulação proposto por meio de técnicas existentes na literatura.

As seguintes publicações apresentam parte dos resultados da pesquisa realizada para esta dissertação:

- As Redes de Petri como Instrumento para Modelagem Matemática de um Problema do Sistema de Currículos da Agência Nacional e Tecnologia de Portugal: O presente trabalho relata a modelagem dos modelos conceituais proposto na tecnologia Guaraná com base nas equivalências dos componentes das redes de Petri, para a obtenção de modelos de simulação de sistema de eventos discretos. A partir do modelo de simulação é possível submeter a uma ferramenta de simulação com o objetivo de obter resultados e assim ter a possibilidade de fazer a análise do comportamento do sistema não sendo necessária a implementação [28].
- Modelagem e simulação de uma solução de integração baseada no formalismo matemático redes de Petri: O referido trabalho relata sobre as redes de Petri e suas diferentes extensões, define o estudo de redes de Petri Estocásticas para o caso de estudo desta pesquisa e apresenta uma ferramenta de simulação e suas vantagens na a obtenção de resultados, apresenta a possibilidade de análise do comportamento por meio de dados matemáticos alicerçados por métodos probabilísticos, para facilitar a interpretação e dar credibilidade a pesquisa [29].

1.6 Estrutura dessa Dissertação

Essa dissertação está organizada da seguinte maneira:

Capítulo §1: Introdução. Compreende a introdução que está subdivida pela Seção §1.1 contexto da pesquisa, a Seção §1.2 motivação, a Seção §1.3 objetivos, a Seção §1.4 metodologia e a Seção §1.5 resumo das contribuições.

Capítulo §2: Revisão da Literatura. Proporciona para o leitor uma revisão da literatura técnica e científica relacionadas à pesquisa desenvolvida nessa dissertação. A Seção §2.1 comenta sobre integração de aplicações empresariais, a Seção §2.2 apresenta a plataforma guaraná, a Seção §2.3 introduz a simulação de eventos discretos, a Seção §2.4 apresenta as redes de Petri, a Seção §2.5 trata a distribuição de probabilidades, a Seção §2.6 discute os trabalhos relacionados e a Seção §2.7 encerra esse capítulo apresentando um breve resumo dos assuntos abordados.

Capítulo §3: Modelagem. a proposta de modelagem matemática para o modelo conceitual de solução de integração de um problema do sistema de currículos da agência nacional e tecnologia de Portugal. A Seção §3.1 descreve o caso de estudo, o ecossistema de software e o modelo conceitual de integração. A Seção §3.2 apresenta a equivalência entre os componentes da linguagem Guaraná e as redes de Petri. A Seção §3.3 apresenta o modelo de simulação proposto. A Seção §3.4 encerra o capítulo apresentando um resumo dos temas abordados.

Capítulo §4: Experimentação. Apresenta o experimento realizado. A Seção §4.1 descreve o cenário utilizado, a ferramenta de simulação, as variáveis observadas e os resultados experimentais. Na Seção §4.2 é realizada a análise e interpretação dos resultados obtidos a partir da simulação da solução de integração modelada. A Seção §4.3 apresenta a verificação do modelo de simulação proposto. A Seção §4.4 encerra o capítulo apresentando um resumo dos temas abordados.

Capítulo §5: Conclusões e Trabalhos Futuros. São apresentadas as conclusões a partir da pesquisa desenvolvida nessa dissertação e propostos alguns trabalhos futuros para continuidade nesta pesquisa.

Capítulo 2

Revisão da Literatura

Buscar e aprender, na realidade,
não são mais do que recordar.

Platão

A revisão da literatura trata sobre os assuntos que deram suporte para esta pesquisa e está dividido em 7 seções. A Seção §2.1 comenta sobre integração de aplicações empresariais, seus estilos de integração, topologias e plataformas de integração. A Seção §2.2 aborda a plataforma guaraná, o guaraná DSL e sua notação gráfica. A Seção §2.3 comenta sobre a simulação de eventos discretos e sua importância para o desenvolvimento de sistemas, os modelos como representação abstrata de sistemas e sua classificação. A Seção §2.4 aborda as redes de Petri como formalismo matemático para representação de sistemas, apresenta seu formalismo algébrico e representação gráfica, redes de Petri Estocásticas e redes de Petri Estocásticas Generalizadas. A Seção §2.5 trata sobre distribuição de probabilidade e sua especificação nas redes de Petri. A Seção §2.6 apresenta trabalhos que abordam simulação de sistemas de eventos discretos utilizando o formalismo das Redes de Petri. A Seção §2.7 encerra esse capítulo apresentando um breve resumo dos assuntos abordados.

2.1 Integração de Aplicações Empresariais

Muitas empresas adquirem ou desenvolvem aplicações para apoiar a tomada de decisões e aperfeiçoar seus processos de negócio. Estas aplicações

compõe o seu ecossistema de *software*, que geralmente é heterogêneo, e foram desenvolvidas sem levar em conta sua possível integração, dificultando assim a sua reutilização. Assim, a área de Integração de Aplicações Empresariais, do inglês *Enterprise Application Integration* (EAI), está relacionada à elaboração de soluções que resolvam estes problemas, sendo definido por Linthicum [30] como o compartilhamento irrestrito de dados e processos de negócios entre todas as aplicações conectadas e fontes de dados na empresa. Segundo Hohpe e Woolf [21], 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 já existentes, de tal forma que as aplicações não sejam alteradas pela solução.

2.1.1 Estilos de Integração

O estilo compartilhamento de dados é caracterizado pela troca de informações entre aplicações. Esta troca pode ocorrer através da transferência de arquivos [27] conforme apresentado na Figura §2.1, ou através da utilização de um gerenciador de banco de dados conforme apresentado na Figura §2.2.

Algumas características são importantes e precisam ser consideradas para o compartilhamento de informações através da utilização de arquivos. Como aplicações distintas acessam o arquivo, seja para escrever no mesmo ou para fazer a leitura dos dados nele contidos, é necessário que se conheça todos os atributos deste arquivo, como nome, extensão e localização, além do formato do arquivo [21].

Uma grande vantagem considerada por [21], é que neste estilo de integração, não é necessário que se tenha acesso ao código fonte das aplicações envolvidas, sendo apenas necessário que se conheça o conteúdo e o formato dos arquivos. Estes arquivos são utilizados pela aplicação interessada sem sofrer ou causar interferência para as demais aplicações, com isso, têm-se aplicações independentes entre si utilizando o mesmo arquivo de dados.

De acordo com Hohpe e Woolf [21], 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 e 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 Mensagens.

Transferência de Arquivos

A Transferência de Arquivos (*File Transfer*), apresenta um mecanismo da integração por arquivos, é composto basicamente de uma aplicação com a função de criar exportar arquivos de dados a outra que irá importar o arquivo, interpretar e fazer uso de suas informações. A troca de arquivos é o recurso mais comum nas soluções de integração, pela versatilidade de implantação em diferentes plataformas, linguagens e conceitos de negócio. A simplicidade estrutural garante as aplicações integradas por arquivos uma independência devido ao baixo acoplamento, permitindo que qualquer aplicação envolvida possa ser alterada desde que mantenha os arquivos compartilhados em mesmo formato local e nomenclatura. A singela estrutura que facilita a integração por arquivos perde sua eficácia quando se trata de manter padrões de nomenclatura para arquivos bem como localização, devido ao grande fluxo de arquivos. Podem ser gerados arquivos incompletos. Este estilo de integração tem a tendência de não se adequar a cenários complexos e de grande tráfego, e pode se tornar instável e apresentar problemas como, por exemplo, duplicidade de dados e outros difíceis de identificar.

Uma das abordagens mais simples, seria a de integrar de alguma forma essas aplicações que utilizam arquivos, cada aplicativo de arquivos produzem informações que outros aplicativos precisam e que possam consumir. Integradores devem assumir a responsabilidade de transformar os arquivos em diferentes formatos. Produzir os ficheiros em intervalos regulares de acordo com a natureza do negócio.

Uma decisão importante com arquivos é o formato a utilizar, muito raramente será a saída de um aplicação ou ter exatamente o que é necessário para outro, assim você terá que fazer um pouco de processamento de arquivos ao longo do caminho, não somente fazer todas as aplicações que utilizam um arquivo consigam lê-lo, mas você também tem que ser capaz de usar ferramentas de processamento sobre ele. Como resultado, formatos de arquivo padrão tem crescido ao longo do tempo.

Banco de Dados

O estilo de integração por banco de dados (*shared database*), prevê a utilização de um único banco de dados pelas aplicações integradas. Partindo das premissas que os processos de negócio precisam ser integrados já

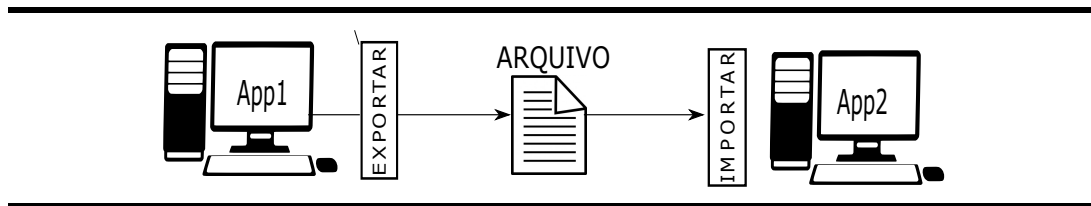


Figura 2.1: *Transferência de Arquivos.*

que muito destes necessitam de uma carga de dados semelhante, para haver um banco de dados para cada programa em execução seria necessário uma forma de sincronizar os dados para evitar inconsistência. A estratégia para estas situações pode ser o compartilhamento do mesmo banco de dados pelas aplicações integradas. O compartilhamento de um banco de dados único extingue problemas de sincronização, e há uma análise dos formatos dos arquivos, então é preciso determinar um esquema de funcionamento que atenda a todas as aplicações, tendo a segurança da solidez de seus dados.

Na prática, a principal dificuldade é um banco de dados que atenda a todas as aplicações, devido a heterogeneidade de sistemas em funcionamento, adquiridos ou construídos com estruturas diferentes (mais modernas ou então de fornecedores diferentes).

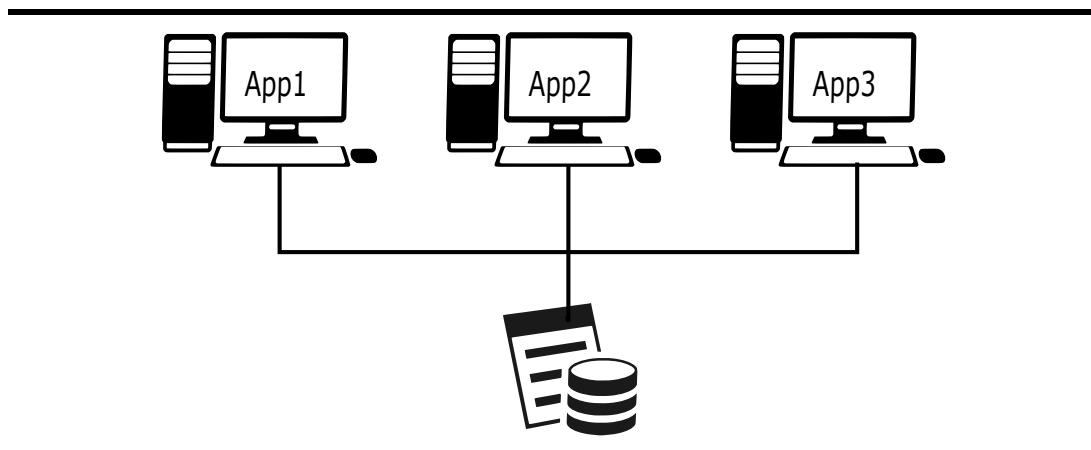


Figura 2.2: *Banco de dados.*

Chamada de Procedimento Remoto

A *Remote Procedure Invocation* (RPI), propõe uma integração bem mais além de mensagens e arquivos, e sugere unificar processos, com o diferen-

cial de neste estilo de integração ser integrados os serviços. A aplicação (app) app1 envia uma série de pedidos de consultas à aplicação integrada app2 que realiza as consultas, reúne resultados e responde a app1 que continua suas atividades. A sincronização e alteração dos dados são realizadas conforme realização dos processos. Integrações baseadas em chamada de procedimento remoto (RPC) precisam de implementação cuidadosa, devido ao aumento do acoplamento e dependência entre as aplicações, torna instável e frágil caso uma das aplicações para de funcionar corretamente. A mudança do mecanismo de negócio de uma aplicação integrada por RPC acarreta na alteração de todas as demais.

Muitos dos desenvolvedores de software usam chamadas de procedimento remoto, pois se encaixam com o que eles já estão acostumados. Na verdade, esta é mais uma desvantagem do que uma vantagem, pois há grandes diferenças no desempenho e confiabilidade entre chamadas de procedimento remoto e local. O *Remote Procedure Invocation* pode levar a sistemas lentos e pouco confiáveis. Embora a encapsulação ajuda a reduzir o acoplamento das aplicações, por eliminação de uma grande estrutura de dados compartilhada, as aplicações são ainda bastante fortemente acopladas.

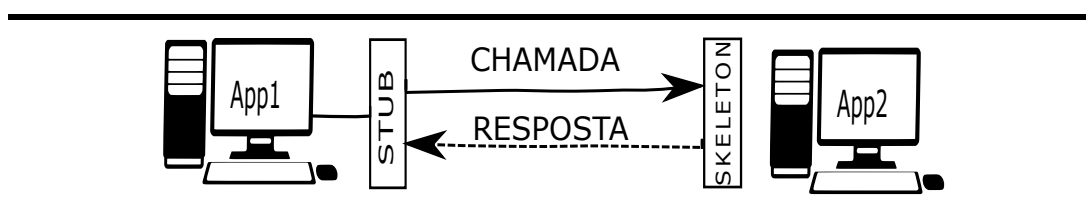


Figura 2.3: Chamada de Procedimento Remoto.

Sistemas de Mensagens

No sistema de mensagens (*Messaging*) o fluxo de mensagens é gerenciado por um *middleware*, ou seja, um terceiro componente independente, que trabalha como elo entre as aplicações. Com isso as aplicações podem trocar informações com diferentes aplicações, e a intermediação do *middleware* é responsável pela adequação do formato das mensagens, possibilitando integrar com aplicações integradas com os demais estilos de integração. A principal tarefa de um sistema de mensagem é manter uma conexão segura para mover mensagens da aplicação remetente para a receptora de uma maneira confiável. O sistema de mensagens garante a entrega da mensagem, repetindo a operação até que a mesma seja entregue. Integrações baseadas em mensagens são ideais para ambientes que exigem flexibilidade, por

que geralmente as aplicações são desenvolvidas sem a intenção da integração e já possuem uma estrutura de software em funcionamento. Esse estilo de integração propõe integrar as aplicações sem gerar dependência entre elas, mantendo a fluidez e a coesão de cada etapa do processo.

Integrar aplicações requer troca de informações, há a necessidade de transportar dados de uma aplicação para a seguinte. Os dados podem ser um registro de endereço que precisa ser replicado, uma chamada para um serviço remoto ou um trecho de HTML. Independentemente da carga útil, este pedaço de dados precisa ser entendido por ambas as aplicações.

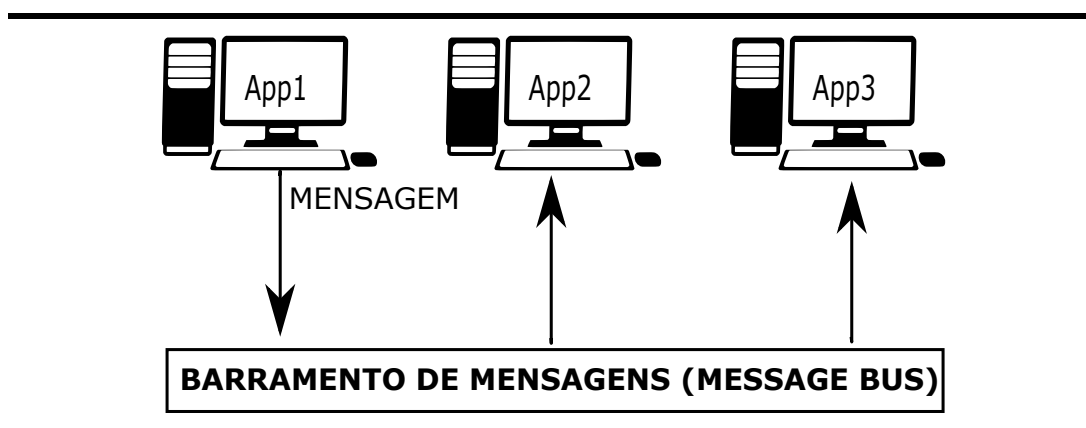


Figura 2.4: Sistema de Mensagens.

A integração de aplicações pode ser realizada de várias maneiras e ser utilizada de diversos recursos. Não é possível determinar o melhor a ser considerado para uma organização, pois cada um deles possui seus prós e contras, então é preciso analisar os estilos e determinar qual deles mais acrescentará benefícios à aplicação.

2.1.2 Topologias

Uma topologia para a integração de aplicações pode ser vista basicamente como uma arquitetura que determina a forma com que a solução será implementada. São consideradas as topologias ponto-a-ponto e *Enterprise Service Bus* (ESB). Uma integração baseada na topologia ponto-a-ponto se caracteriza basicamente por uma aplicação servir de cliente e servidor ao mesmo tempo.

Nesta topologia as aplicações compartilham serviços e dados entre si sem a necessidade de um servidor central. É uma abordagem simples, porém ultrapassada. Resulta em códigos de integração personalizados e espalhados

entre as aplicações de forma descentralizada dificultando o monitoramento dos processos [33].

De acordo com [14] e outros, evidenciam que uma ESB fornece a tecnologia necessária para implementar soluções de integração. Segundo os autores, grande parte das ESBs aderiram a especificação da arquitetura *pluggable* de serviços fornecido pela JBI (*Java Business Integration*), como por exemplo, *Open ESB*, *Petals ESB*, *ServiceMix* e *Fuse ESB*. Em JBI, adaptadores e conectores são tratados como componentes de ligação, e a linguagem utilizada é a orquestração *Business Process Execution Language* (BPEL).

Temos também que uma topologia ESB consiste em uma arquitetura mais elaborada que permite integrar diferentes aplicações através de um barramento de comunicação. Nesta topologia as aplicações se comunicam entre si através deste barramento, de forma independente e sem interferência entre as aplicações [33]. Conforme [4] destaca que esta topologia é utilizada em grande parte das soluções de integração de aplicações empresariais, destacaremos as topologias citadas com mais especificidades e representações gráficas, conforme [20]:

Point-to-Point

A topologia mais simples com um link dedicado entre dois pontos finais. Comutado ponto-a-ponto topologias são o modelo básico do convencional de telefonia. O valor de uma rede permanente ponto-a-ponto é comunicações sem entraves, entre os dois pontos finais. O valor de uma ligação *on-demand* ponto-a-ponto é proporcional ao número de possíveis pares de assinantes e foi expressa como a Lei de Metcalfe.

Mais fácil de entender, das variações de topologia ponto-a-ponto, é um ponto-a-ponto canal de comunicação que aparece, para o usuário, para ser permanentemente associado com os dois pontos finais. Infantil telefone de lata é um exemplo de um canal físico dedicado.

Dentro de muitos comutada sistemas de telecomunicações, é possível estabelecer um circuito permanente. Um exemplo pode ser um telefone no *lobby* de um edifício público, que está programado para tocar apenas o número de um despachante telefone. "Pregando down" uma conexão comutada economiza o custo de funcionamento de um circuito físico entre os dois pontos. Os recursos em uma conexão pode ser liberado quando não for mais necessário, por exemplo, um circuito de televisão a partir de uma rota do desfile de volta ao estúdio.

Enterprise Service Bus

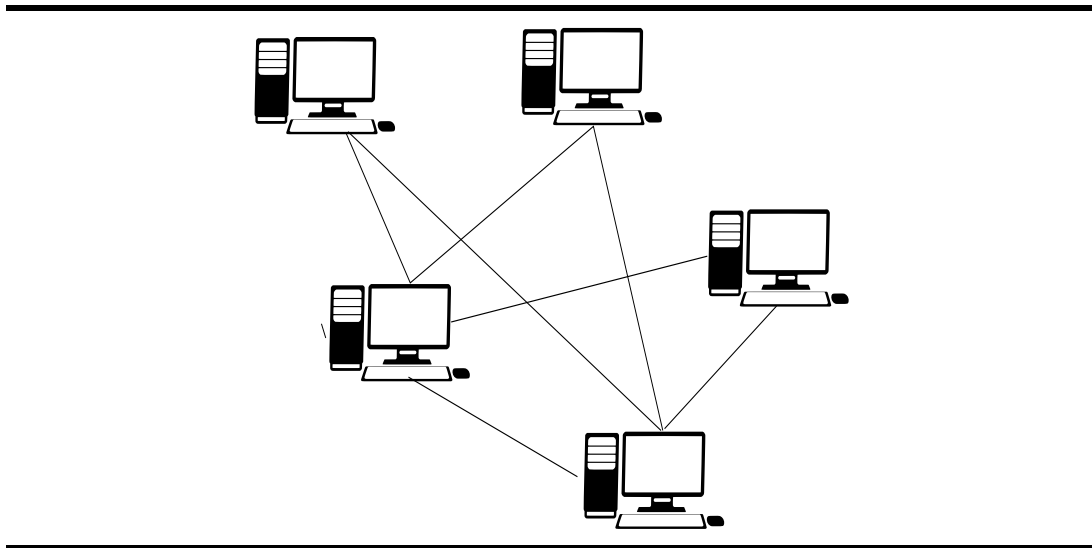


Figura 2.5: *Point-to-point*.

Uma topologia em ESB, é considerada a organização mais simples de uma rede, todos computadores estão ligados a uma mesma linha de transmissão através de cabo, geralmente coaxial. A palavra "ESB" designa a linha física que liga as máquinas da rede. Com a EAI baseados em *ESB* evoluíram, foram identificadas uma série de outras funcionalidades necessárias, tais como a segurança de processamento de transações, manipulação de erro, e muito mais, em vez de exigir embutir esses recursos para a lógica de integração central, como teria sido exigido por uma arquitetura de correção, a arquitetura de barramento permitiu estas funções serem locadas em componentes separados, conforme a Figura §2.6. O resultado final, soluções leves feitas à medida da integração com a confiabilidade garantida, que são completamente abstraídas da camada de aplicativos, seguem um padrão consistente, e pode ser projetado e configurado com código adicional mínimo com nenhuma modificação para os sistemas que precisam ser integrados.

Hub-and-Spoke

A integração através da topologia *hub-and-spoke* é implementada através da criação de um mecanismo central de comunicação (*hub*), do qual partem as ligações (*spokes*) que se conectam às diferentes aplicações, conforme a Figura §2.7. Um *hub* é definido como o ponto central de um sistema, através do qual os outros elementos se comunicam.

Essa forma de resolver o problema não está presente apenas na área

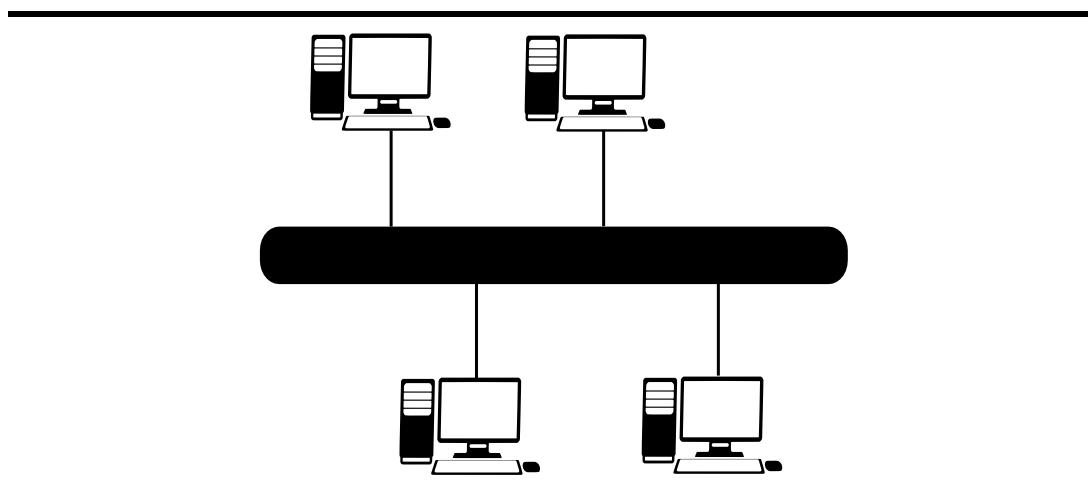


Figura 2.6: BUS.

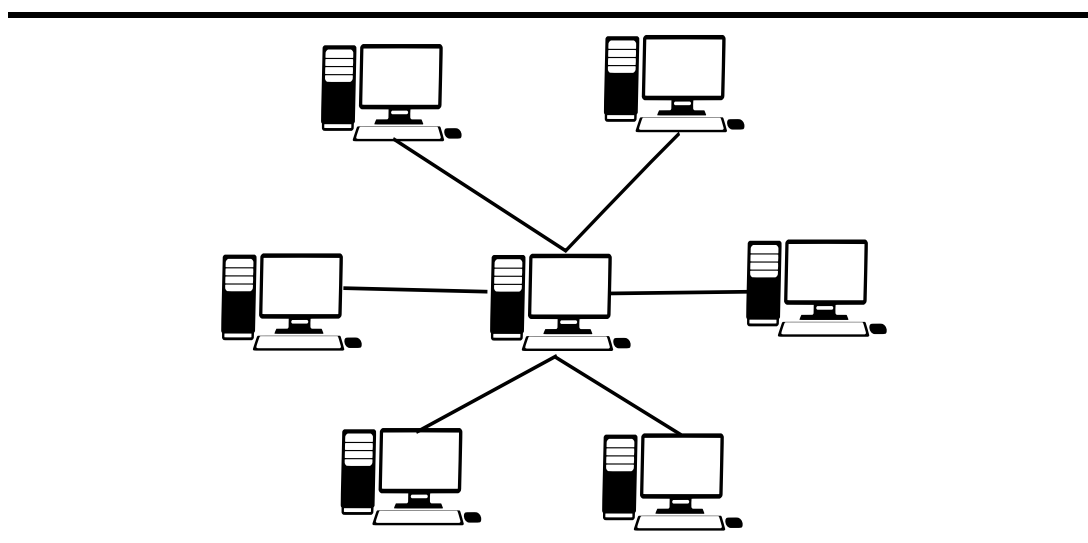


Figura 2.7: Hub and Spoke.

de EAI, mas também é usada em campos como a criação de linhas aéreas e a logística de sistemas de entregas. Conforme [21], essa forma de ligação reduz o número de ligações necessárias. Isso ocorre porque as aplicações não precisam ser ligadas diretamente entre elas, como ocorre na topologia *point-to-point*. É necessário apenas que cada uma delas seja ligada ao *hub* central através de um *spoke*. Dessa forma, o número de ligações

necessárias não possui um crescimento exponencial, mas linear.

O *hub* pode fazer modificações no formato dos dados e pode selecionar para onde enviá-los. Além disso, essa topologia pode simplificar o processo de integração, uma vez que ela pode conectar aplicações com diferentes formatos e meios de transmissão de dados. Esses fatores de comunicação entre aplicações com formatos de dados diferentes e de modificação no formato dos dados são apontados como importantes também por [21], que introduz o conceito de modelo canônico.

O que é sugerido pelo autor é que pode haver um formato canônico de dados, para o qual os dados enviados pelas aplicações são convertidos pelo *hub*. Em seguida, o *hub* faz uma nova conversão, dessa vez para o formato requerido pela aplicação de destino. Dessa forma, as aplicações podem se comunicar, independentemente do formato de dados usado por cada uma delas.

2.1.3 Plataformas de Integração

As Plataformas de Integração são um conjunto de meios computacionais de hardware e software utilizados para viabilizar a interação direta, sem intervenção humana, entre sistemas de informação e dos participantes do mercado, por meio de serviços padronizados, contribuindo para o aumento da flexibilidade na execução de processos[37].

As plataformas de integração permitem representar uma solução de integração por meio de um modelo conceitual. Existem diversas plataformas de integração disponíveis no mercado, oferecem API de programação, implementação, um motor de execução e ferramentas de monitoramento, dentre elas podemos destacar o Apache Camel [23], Mule [10], *Spring Integration* [11] e o Guaraná DSL [15]. Destacaremos o Guaraná DSL que é uma das tecnologias que possibilita projetar modelos conceituais de soluções de integração, utilizando uma sintaxe concreta gráfica e intuitiva. Possibilita aos engenheiros de software projetar soluções de integração com um alto nível de abstração, utilizando uma abordagem centrada em modelos.

A ferramenta Guaraná se diferencia de outras semelhantes por possuir um mecanismo externo de controle de erros. Outro diferencial é o uso de um modelo de execução baseado em tarefas, não em processos. De acordo com [13], a diferença entre esses modelos está na granularidade da execução. No modelo de execução baseado em tarefas, o motor pode controlar tanto as instâncias de processos quanto as tarefas internas. No modelo baseado em processo, o motor controla as instâncias de processo como um

todo, não podendo interagir com suas tarefas internas. Dessa forma, ele requer o uso de um mecanismo para reunir as mensagens, correlacioná-las e decidir quando uma nova instância de processo pode ser iniciada.

Além disso, cada instância de processo requer a alocação de uma *thread* exclusiva, o que pode ter um impacto negativo no desempenho nos casos em que uma instância de processo envia uma chamada e espera por um longo tempo até obter a resposta.

Como o modelo baseado em tarefas lida com as tarefas que estão dentro de uma instância de processo, ele pode alocar as *threads* com maior eficiência: nenhuma *thread* deve permanecer parada enquanto houver uma tarefa pronta para ser executada, independentemente do processo a que ela pertencer. Devido a essas características do modelo baseado em tarefas, o Guaraná permite um uso mais eficiente dos recursos do sistema.

2.2 Plataforma Guaraná

A tecnologia guaraná nos dá toda a base científica a fim de realizar este trabalho, pois é uma ferramenta que oferece a possibilidade de organização de projetos de alto nível de abstração, com o auxílio de uma sintaxe gráfica concreta, de forma que os engenheiros de software possam ter uma visão de todos os processos presentes em uma solução de integração. Cabe ressaltar que os modelos obtidos e implementados na tecnologia Guaraná, são independentes de plataforma, isto permite que as soluções sejam implementadas em qualquer tecnologia [14] desta maneira os engenheiros não dependem de conhecimentos e habilidades em uma tecnologia de integração, a fim de projetar suas soluções.

2.2.1 Guaraná DSL

O emprego de uma linguagem de domínio específico (em inglês Domain-Specific Language - DSL) visa solucionar problemas específicos num domínio particular, por meio de uma linguagem própria. Os DSLs podem ser utilizados por usuários com conhecimento superficial em linguagens de programação, tendo em vista o nível de abstração e fácil interpretação. Com a simplicidade dos DSLs, é possível uma comunicação eficaz com os clientes, facilidade de manutenção, sintaxe legível e de fácil compreensão, tornando o tempo de desenvolvimento menor. De acordo com Frantz [14], o Guaraná DSL é uma linguagem de modelagem baseada nos padrões de integração.

2.2.2 Notação Gráfica

Uma linguagem de domínio específico permite que os engenheiros de software desenvolvam suas soluções de integração por meio de suporte gráfico e conceitos de modelagem muito intuitivos. De acordo com Hohpe [21], a estrutura Guaraná é baseada em padrões de integração de aplicações, o que possibilita o projeto da estrutura interna de todos os processos de integração e portas de comunicação.

O guaraná DSL é composto por cinco elementos gráficos principais: aplicações, processos, *slot*, tarefas e portas, como é apresentado na Tabela §2.1.

Notação	Conceito	Notação	Conceito
	Aplicação		Porta de Entrada
	Processo		Porta de Saída
	Tarefa		Porta de Resposta
	Slot		Porta de Solicitação

Tabela 2.1: Notação Gráfica do Guaraná DSL (adaptada de Frantz et al. [14])

Cada elemento tem a sua função dentro do sistema, como por exemplo:

Processos

Possui as portas de entrada, saída, solicitação e resposta, as tarefas que são interligadas pelos *slots*. É o responsável pelo agrupamento das tarefas, contém portas para a comunicação com as aplicações integradas.

Slots

São os responsáveis pelo processamento assíncrono das mensagens que fluem dentro de um processo, proporcionam independência entre as tarefas. São os *slots* que conectam as portas com tarefas ou aplicações e tarefas com tarefas.

Portas

São divididas em portas de entrada, saída, solicitação e resposta. Estão acopladas no processo com a função de enviar ou receber mensagens. Podem ser consideradas como uma abstração de detalhes dentro de um ecossistema de software.

Aplicações

AS aplicações contém as informações necessárias para o funcionamento da solução de integração. São responsáveis pela alimentação do sistema, solicitando ou disponibilizando os dados conforme a sua característica.

Tarefa

As tarefas são responsáveis pelo processamento e modificação das mensagens. Uma tarefa pode ter mais de uma entrada, assim como mais de uma saída, tudo dependendo de como ela será processada, pois de acordo com a sua semântica ela poderá dividir, replicar, agregar, traduzir, filtrar, combinar mensagens, entre outros. Elas são divididas em grupos de acordo com sua semântica, cita-se alguns exemplos: roteadoras, modificadoras, transformadoras, comunicadoras, mapeadoras, temporais, etc.

Pode-se exemplificar uma solução de integração pelo problema do café, proposto por [19] é um problema clássico utilizado para introduzir tecnologias e soluções de integração de um ponto de vista prático. Esse problema descreve como pedidos de bebidas quentes e frias são preparados por dois baristas diferentes em uma cafeteria. O cliente chega no caixa e faz um pedido, que é computado no sistema e então posto em uma fila. Cada pedido possui uma bandeja relacionada e quando todas as bebidas referentes a um pedido estão prontos, são postos na bandeja pelo caixa e entregues ao cliente por um garçom. O caixa vai recebendo pedidos e adicionando à fila sem a necessidade de aguardar que um barista finalize, pois cada barista retira os pedidos da fila. A integração desse problema, tem o desafio de pegar pedidos da fila, enviar aos respectivos baristas e avisar o garçom quando os pedidos estão prontos.

2.2.3 Exemplo de Solução de Integração

O modelo conceitual de uma solução de integração desenvolvida no Guaraná DSL está representado na Figura 2.8. Esta solução de integração, denominada Café, coordena, de forma exógena, quatro aplicações [15]. Essas

aplicações comunicam-se com a solução por meio de portas, que leem e escrevem mensagens para a solução e para as aplicações. Na aplicação Pedidos, os pedidos de bebidas são registrados e a porta P1 faz a leitura do *log* da aplicação e adiciona as informações em forma de mensagem no fluxo S1. A tarefa T1 divide cada mensagem em várias outras, cada uma correspondente a uma bebida. As mensagens, então, são encaminhadas pela tarefa T2 para a aplicação Barista Bebidas Frias ou para Barista Bebidas Quentes, conforme o tipo de bebida. Antes de ser enviada para a aplicação Barista Bebidas Frias, a mensagem é replicada pela tarefa T3 para as tarefas T4 e T5. A tarefa T4 traduz a mensagem para o esquema da aplicação Barista Bebidas Frias, que então é enviada por meio da porta P2 para a aplicação. Essa prepara a bebida e envia o resultado para a tarefa T5 por meio da mesma porta, P2. A tarefa T5 correlaciona a mensagem de espera vinda de T3 com a mensagem de resultado e as envia para a tarefa T6, que enriquece a mensagem de espera com a mensagem de resultado. O comportamento das mensagens entre as aplicações Barista Bebidas Frias e Barista Bebidas Quentes é o mesmo. A tarefa T7 recebe as mensagens vindas das duas aplicações e mescla-as em um único fluxo novamente. Essas mensagens, que anteriormente foram divididas pela tarefa T1, são reagrupadas nos respectivos pedidos e enviadas por meio da porta de saída P4 para a aplicação Garçon, que entrega os pedidos.

2.3 Simulação de Eventos Discretos

2.3.1 Sistemas

A definição de sistema está relacionada como um conjunto de elementos que se comunicam no desempenho de alguma função, com finalidade de alcançar um ou mais objetivos. Esta definição pode ser usada em uma variedade de contextos, como por exemplo: sistema econômico, sistema computacional, sistema de injeção eletrônica, sistema nervoso, sistema solar e assim por diante, o que unifica todos estes exemplos é o fato de que cada um deles possui um conjunto de elementos que tenham alguma relação, sejam eles componentes, subsistemas ou subunidades, e que é possível identificar alguma função desempenhada pelo sistema como um todo, como por exemplo, sistema econômico: mantém os recursos da economia em circulação, sistema computacional: os processamentos de informações de usuários, atendem a uma determinada necessidade geral. Dados os conceitos muito abrangentes de sistema, é preciso adotar categorias mais específicas, para que os sistemas sejam passíveis de análise e síntese.

Dentre os sistemas, os que evoluem de acordo com a ocorrência de eventos, dentre eles o sistema de eventos discretos (SED). Os eventos podem ser

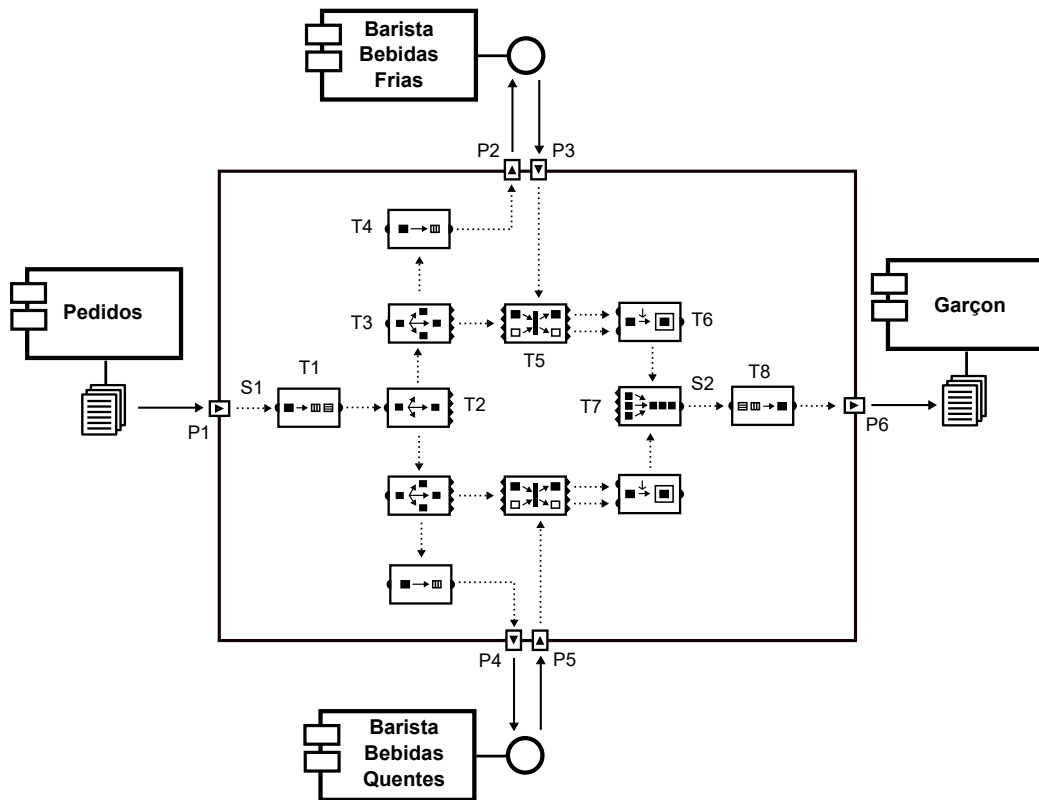


Figura 2.8: Solução de Integração do Problema do Café. [15]

propositais, espontâneos, ou o resultado de verificação de uma condição, mudando o estado do sistema, os quais geralmente ocorrem em momentos de tempo irregulares.

Os 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.

Os modelos podem apresentar-se como protótipos ou como modelos matemáticos, estes podem apresentar as soluções analíticas, como por exemplo, um modelo de regressão ou a simulação, permitindo assim, reconstituir a rotina funcional de um dado sistema real, conforme Figura §2.9. Para empregar os modelos de forma adequada deve-se na maioria das vezes, proceder a verificação e validação [6].

Desse modo, a implementação de modelos matemáticos de simulação a

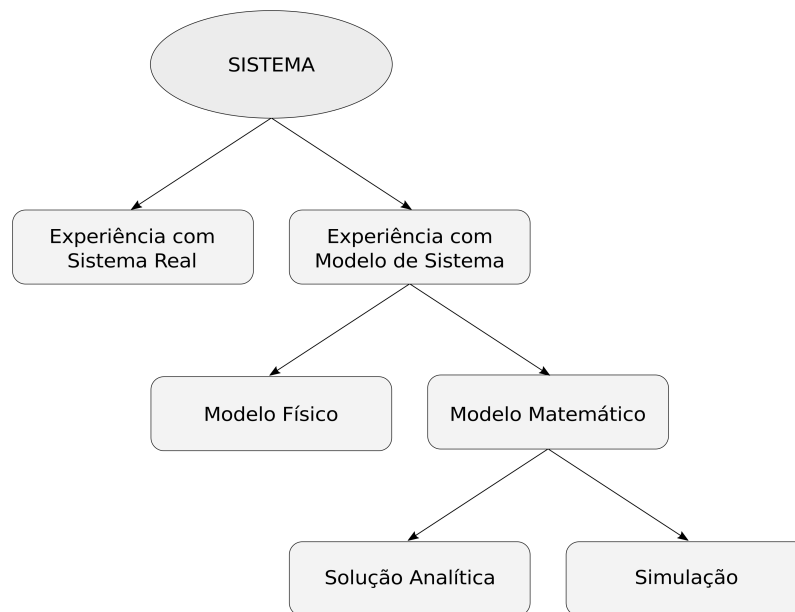


Figura 2.9: Possibilidade de Estudo dos Sistemas ([25])

serem utilizados, requerem o uso de algumas linguagens de programação como FORTRAN, C e PASCAL ou de linguagens de simulação como SLAM, GPSS, GASP, POWERSIM, ARENA, EXTEND e PIPE2 [25].

2.3.2 Modelos de Sistemas

A definição para um modelo pode ser como uma representação do sistema real, que 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 [25].

Os modelos do sistema podem ser físicos ou matemáticos. Modelos físicos são empregados para projetar protótipos em escala e são necessários para simular um sistema. Por exemplo, dado um sistema, é possível construir um modelo simplificado para representar as interações entre as suas partes. Neste contexto, o modelo é uma abstração da realidade, aproximando-se do verdadeiro comportamento do sistema, mas sempre mais simples do

que o real. Pode ser difícil de conseguir esta característica porque modelos realistas são raramente simples e modelos simples raramente são realistas.

Os modelos de simulação capturam melhor o comportamento real dos sistemas e permitem que sejam analisados em computadores. Eles são utilizados para obter respostas sobre a operação do sistema e podem ser capazes de analisar vários aspectos do mesmo, ou melhor, todos os que são de interesse. Sistemas atuais têm, geralmente, uma maior complexidade, principalmente devido à sua natureza dinâmica e aleatória.

Segundo Kelton [25], os modelos matemáticos de simulação, ou simplesmente modelos de simulação, podem ser classificados em Estáticos ou dinâmicos, determinístico, estocástico ou probabilístico, discretos ou contínuo. A decisão para usar um modelo discreto ou um modelo contínuo para um determinado problema depende dos objetivos específicos do estudo, representamos a divisão dos modelos, conforme a Figura §2.10:

Estáticos ou dinâmicos

Denominam-se modelos estáticos os que visam representar o estado de um sistema em um instante ou que em suas formulações não se leva em conta a variável tempo, portanto, podem ser utilizados em outros campos, tais como em métodos numéricos, problemas de otimização, teste de algoritmos, etc..., enquanto os modelos dinâmicos são formulados para representarem as alterações de estado do sistema ao longo da contagem do tempo de simulação.

Determinístico, estocástico ou probabilístico

No modelo probabilístico são utilizadas variáveis aleatórias como entrada, que determinam e produzem saídas aleatórias. A saída de simulação probabilística deve ser tratada como estimativa estatística das características reais de um sistema, se um modelo de simulação não contém nenhuma variável probabilística é chamado determinista. Em um modelo determinístico, para um conjunto conhecido de entrada de dados o resultado deverá ser um conjunto único de saída. Normalmente modelos probabilísticos são mais complexos e melhor representam sistemas reais do que os modelos deterministas.

Discretos ou contínuos

Modelos de simulação discretos ou contínuos podem ser definidos analogamente para o sistema de maneira discreta e contínua. Os modelos de simulação contínuos são utilizados para modelar sistemas em que as variáveis de estado mudam continuamente em função do tempo, isto é, o estado

do sistema muda continuamente de acordo com o tempo. Modelos de simulação de eventos discretos são utilizados para modelar sistemas que mudam de estado em momentos discretos no tempo, a partir da ocorrência de eventos. Em alguns casos, um modelo de simulação pode compreender, simultaneamente, aspectos contínuos e discretos, conhecido como modelo híbrido ou modelo de simulação combinada[36].

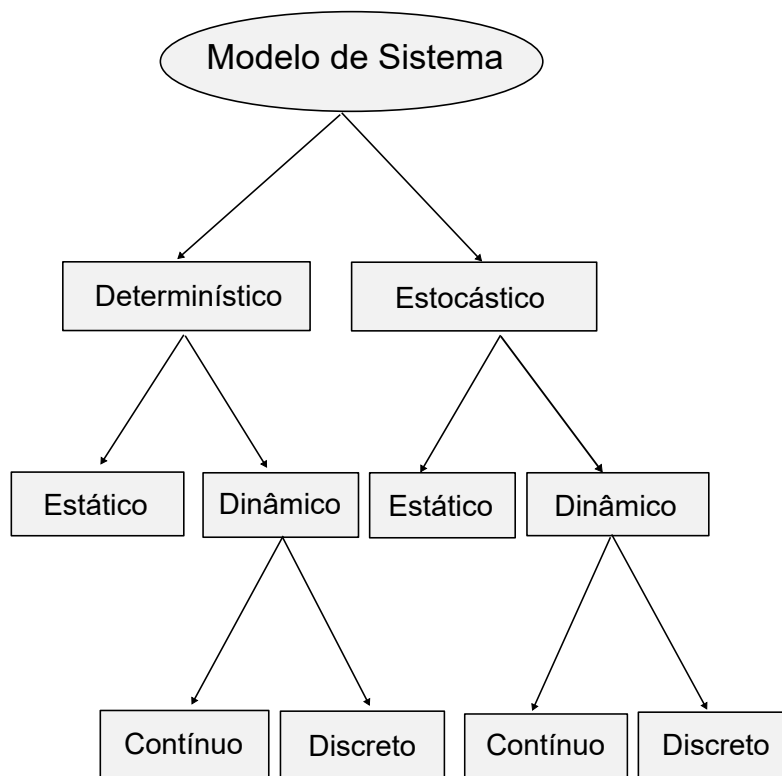


Figura 2.10: Modelo Taxonômico [25].

Há três elementos básicos que compõem um modelo de simulação de eventos discretos: entidade, atributo, e atividade. Qualquer objeto envolvido com o modelo é chamado entidade [39]. A propriedade desta entidade é chamada de chamado. Qualquer processo que gera uma alteração no modelo é chamado de atividade.

Um modelo de simulação de eventos discretos visa reproduzir as atividades das entidades envolvidas no âmbito de aplicação do sistema, a fim de saber algo sobre o seu comportamento e desempenho. Assim, é necessário definir os estados de um sistema e eventos que alteram o sistema a

partir de um estado para outro. O estado de um sistema pode ser definido como um conjunto de variáveis usadas para descrever o comportamento do sistema em um tempo particular.

2.3.3 Simulação

Existem diversas definições para a simulação, dentre elas podemos citar [7] que diz "a simulação é um processo de projetar um modelo computacional de um sistema real e conduzir experimentos com este modelo com o propósito de entender seu comportamento e/ou avaliar estratégias para sua operação". Desta maneira, podemos entender a simulação como um processo amplo que engloba não apenas a construção do modelo, mas todo o método experimental que se segue buscando:

- * Descrever o comportamento geral do sistema;
- * Construir teorias e hipóteses considerando as observações efetuadas;
- * Usar o modelo para prever o comportamento futuro, isto é, os efeitos produzidos por alterações no sistema ou nos métodos empregados em sua operação.

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.

Determinar se o modelo proposto retrata com veracidade o sistema em estudo, é uma das tarefas mais difíceis em se tratando de simulação. A verificação e validação são os processos mais recomendados para que isso seja possível. Para analisar se o modelo realiza o que é proposto e está livre de erros da implementação computacional, por meio da aplicação de testes de consistência usa-se a verificação e para determinar se o modelo de simulação representa de maneira satisfatória o sistema real para realizar as experimentações, utiliza-se a validação. A validação é realizada por meio da análise dos resultados do experimento obtidos da simulação, observando se estes são representativos dos resultados do sistema real, sendo possível apenas quando existe um sistema para referências ou comparações[43].

Simulação é um termo bastante genérico, uma vez que existem várias estratégias e mecanismos para se representar o comportamento de sistemas.

Segundo Silva [43], simulação é o estudo do comportamento real de sistemas de experimentação de modelos.

A simulação é dependente desses modelos que são a representação do sistema em questão, que reproduz (simula) seu funcionamento. 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 [7].

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

2.4 Redes de Petri

As Redes de Petri (ou simplesmente RdP) foram criadas a partir da tese de doutorado de Carl Adam Petri, intitulada *Kommunikation mit Automaten* (Comunicação com Autômatos), apresentada à Universidade de Bonn em 1962 [35]. Desde o princípio, objetivaram a modelagem de sistemas com componentes concorrentes.

Redes de Petri é uma técnica de especificação formal bem estabelecida, largamente difundida e adequada para a modelagem de sistema que tenham atividades paralelas, concorrentes, assíncronas e não determinísticas. Vários autores têm mostrado a abrangência e a aplicabilidade das redes de Petri nas mais diversas áreas, tais como: na ciência da computação, engenharias, eletrônica e química, administração de empresas, etc. No entanto, tem sido mais explorada na ciência da computação e na engenharia eletrônica, desde o estudo para a especificação de sistemas de hardware ou software, avaliação de desempenho, especificação de protocolos de comunicação, diagnóstico de falhas e no projeto de software/hardware, entre outros [31].

As ferramentas matemáticas de representação gráfica tem sido muito útil para permitir a visualização dos processos, oferecendo um ambiente propício para modelagem, análise e simulação de sistemas de eventos discretos possibilitando a visualização estrutural e comportamental de forma simultânea. O foco das redes de Petri é a modelagem específica de eventos e condições para que ocorram e as relações entre eles.

2.4.1 Formalismo Algébrico

Basicamente uma rede de Petri $RdP = (P, T, A, M_o)$ é um grafo composto de lugares (P), transições (T), arcos (A) e uma marcação ou *token* inicial M_o , conforme §2.1. Sua estrutura é formalmente especificada da seguinte maneira:

$$RdP = (P, T, A, M_o), \quad (2.1)$$

$$P = \{p_1, p_2, p_3, \dots, p_n\},$$

$$T = \{t_1, t_2, t_3, \dots, t_n\}, I \subseteq (P \times T),$$

$$O \subseteq (T \times P), A = (I \cup O),$$

$$M_o = \{M_o(p_1), M_o(p_2), \dots, M_o(p_n)\}$$

De acordo com as definições das redes de Petri, encontram-se as seguintes condições satisfeitas:

$$P \cap T = \text{Vazio} \quad (2.2)$$

e

$$P \cup T \neq \text{Vazio} \quad (2.3)$$

na condição da equação (2.2), os lugares e as transições são nós distintos (sua intersecção é vazia), o que determina o termo bipartido e pela condição da equação (2.3) entende-se que em uma rede de Petri existe pelo menos um lugar ou uma transição, sendo assim, uma rede de Petri é formada por uma estrutura que é formalizada pelos lugares e transições conectados pelos arcos e uma marcação inicial.

Uma estrutura de rede de Petri é uma rede sem marcação inicial, a qual é denotada por:

$$E_{rp} = (P, T, A, K, W) \quad (2.4)$$

Com essa definição a notação de uma rede de Petri é abreviado por;

$$RdP = \{E_{rp}, M_o\}. \quad (2.5)$$

2.4.2 Representação Gráfica

As redes de Petri, são denominadas grafos bipartidos, compostos por dois elementos: um passivo denominado de lugar e um ativo denominado transição. Os lugares representam uma condição, uma atividade ou um recurso e são representados graficamente por círculos. As transições representam um evento (ação) realizado pelo sistema; graficamente, são representados por um traço ou uma barra. A Figura §2.11 demonstra a representação gráfica dos elementos das redes de Petri.

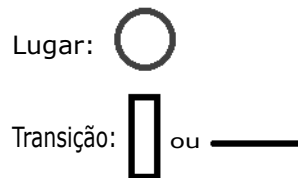


Figura 2.11: Elementos das redes de Petri[8].

Esses dois elementos são os vértices do grafo associado às redes de Petri. Os vértices são interligados por arcos dirigidos, como pode ser visto na Figura §2.12. Os arcos que interligam lugares às transições correspondem à relação entre as condições verdadeiras, que em um dado momento, possibilitam a execução das ações, enquanto os arcos que interligam transições aos lugares representam a relação entre as ações e as condições que se tornam verdadeiras com a execução das ações. Desta forma, uma rede de Petri é um grafo direcionado com dois tipos de nós, não havendo arcos entre nós do mesmo tipo [8].

Os lugares podem armazenar marcas ou *tokens*, representados graficamente por pontos pretos. Uma distribuição de *tokens* na rede é chamada de marcação e corresponde ao Estado da rede de Petri. Além dos *tokens*, ainda é possível que os vértices de um grafo sejam interligados por múltiplos arcos. A Figura §2.13 ilustra um exemplo, com que um lugar pode ser conectado a uma transição através de diversos arcos ou vice-versa. Por conveniência, os múltiplos arcos são substituídos por um único arco valorado, onde o numeral associado ao arco corresponde ao número de arcos que interligam os vértices.

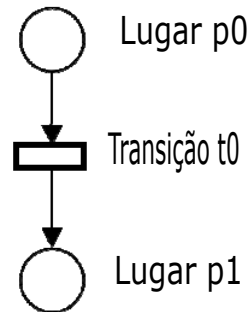


Figura 2.12: Exemplo das redes de Petri [8].

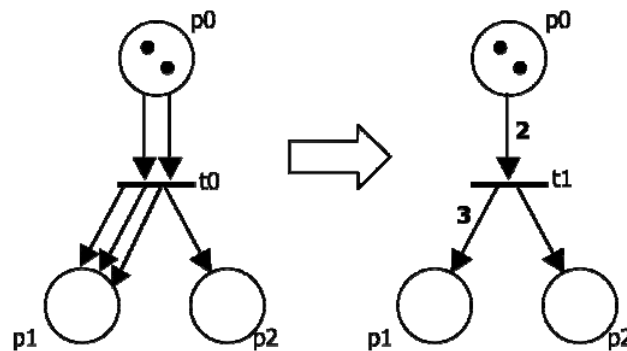


Figura 2.13: Representação de arco valorado [8].

Pode-se dizer que uma transição está habilitada se todos os lugares de entrada, ou seja, lugares de onde partem arcos em direção a esta transição têm ao menos o número de *tokens* igual ou maior que o número de arcos que ligam o lugar a esta transição. Uma transição habilitada pode disparar e seu disparo, ou realização de uma ação, está associado a algumas pré-condições, ou seja, existe uma relação entre os lugares e as transições, que possibilita a realização de uma ação. De forma semelhante, após a realização de uma ação, alguns lugares terão suas informações alteradas (pós-condições). A Figura §2.14(a) mostra uma rede de Petri com uma transição disponível, que, quando disparada, causa mudanças na marcação, deixando-a como a Figura §2.14(b).

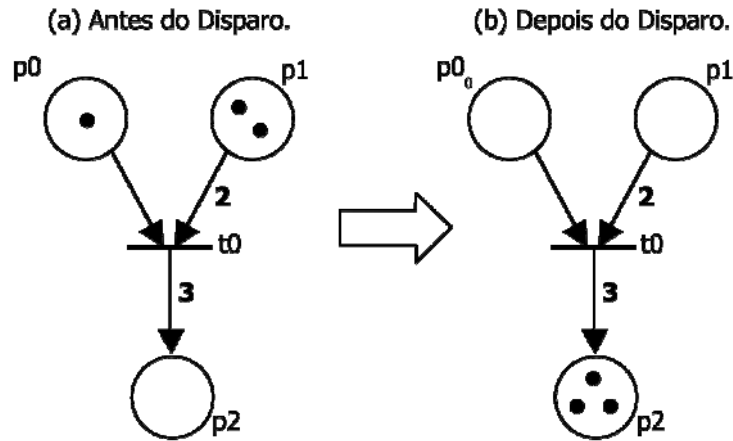


Figura 2.14: Exemplo de disparo de uma transição [8].

O Estado de uma rede, em um determinado instante, é representado pela quantidade de *tokens* em cada lugar da rede de Petri. O Estado da rede é alterado quando ocorre pelo menos um disparo. Como exemplo dessa representação, tem-se que o Estado da rede da Figura §2.14(a) é $M(1, 2, 0)$, que representa a quantidade de *tokens* dos lugares p_0 , p_1 e p_2 , respectivamente. Ao acontecer o disparo da transição t_0 , esse Estado é alterado para $M(0, 0, 3)$, que é o Estado da Figura §2.14(b).

Desde o trabalho original de Carl Adam Petri têm surgido diversas variantes ao seu modelo de redes de Petri. Pode-se afirmar que a maior parte destas variantes nasceu da necessidade de adaptação à especificidade da aplicação para as quais a sua utilização era desejada. O modelo original das redes de Petri, falha na representação de duas importantes características: aspectos funcionais complexos, tais como: condições que determinam o fluxo de controle, e os aspectos de temporização [31]. Para enfrentar essas limitações, extensões das redes de Petri foram desenvolvidas. Alguns exemplos delas são: as Redes de Petri Temporizadas Determinísticas [44], as Redes de Petri Hierárquicas [22], as Redes de Petri Coloridas [24] e as Redes de Petri Estocásticas [17], as Redes de Petri Estocásticas Generalizadas [32], entre outras. Esta pesquisa irá direcionar o estudo para as redes de Petri Estocásticas e redes de Petri Estocásticas Generalizadas.

2.4.3 Redes de Petri Estocásticas

As redes de Petri não foram desenvolvidas originalmente para prover avaliação de desempenho, apesar de toda a sua potencialidade para representar sistemas complexos, os quais naturalmente requerem cuidados a esse respeito. Esse panorama se modificou em 1982, quando M. K. Molloy [34] apresentou as redes de Petri Estocásticas (*Stochastic Petri Nets - SPN*) como uma técnica capaz de, além de especificar sistemas, também apresentar uma análise probabilística dos mesmos.

No seu formalismo as redes de Petri, possibilitam descrever apenas a estrutura lógica de sistemas, pois não consideram o tempo. Entretanto, o tempo interfere no funcionamento da maioria dos sistemas e tem um papel importante em sua estrutura, que precisa ser considerado na sua representação e determinam um atraso nas suas transições associando a probabilidade de disparo podendo alterar a sua frequência. Define-se redes de Petri Estocásticas assumindo como temporizadas todas as transições.

Uma definição formal de uma SPN, definida por Ajmone Marsan et al. [1], é a seguinte:

$$SPN = (P, T, A, M', R), \quad (2.6)$$

onde P, T, A, M' são definidos como em §2.1, e

$$R = \{r_1, r_2, r_3 \dots r_m\} \quad (2.7)$$

é o conjunto de taxas de disparo (possivelmente dependente da marcação ou *token*) associado as transições da rede de Petri.

O tempo de permanência de um *token* é uma variável aleatória distribuída exponencialmente com média, conforme §2.8:

$$\left[\sum_{i \in H} r_i \right]^{-1}, \quad (2.8)$$

onde H é o conjunto de transições que são habilitadas pela marcação.

A representação algébrica das redes de Petri Estocásticas é semelhante às redes de Petri, descritas por conjunto de transições, lugares, arcos e marcação inicial, anexando-se conjunto das taxas de disparo de cada transição. A representação gráfica também é semelhante, com alteração nas transições, que continuam sendo representados por retângulos, porém com interior branco para diferenciar das não estocásticas, também denominadas como imediatas.

2.4.4 Redes de Petri Estocásticas Generalizadas

Chiola [5], apresentou uma melhoria às SPN, denominada Redes de Petri Estocásticas Generalizadas (*Generalized Stochastic Petri Nets - GSPN*), cuja diferença fundamental está em admitir que as transições também podem ser não-estocásticas, isto é, uma transição também pode ser imediata, como nas redes de Petri convencionais. Chiola definiu que as transições imediatas deveriam ter retardo de disparo igual a zero, e que somente as transições estocásticas tinham retardos associados diferentes de zero.

Uma definição formal para as GSPN, assim como §2.6, contém apenas m' elementos, sendo m' o número de transições temporizadas na rede.

Nas GSPN, pode conter transições temporizadas e transições imediatas, quando isso ocorrer a prioridade de disparo sempre será da transição imediata, no caso de conter mais de uma transição imediata, adota-se o processo *Switching distribution*, o qual deve-se definir uma prioridade de disparo para as transições.

2.5 Distribuição de Probabilidade

A probabilidade ou taxa de disparo de cada transição estocástica de uma rede é determinada de acordo com sua frequência de disparo esperada. A distribuição da probabilidade de disparo é realizada através de uma distribuição exponencial, de acordo com a equação §2.9 que determina a probabilidade de disparo para cada transição apta a disparar.

$$F_{xi}(x) = 1 - e^{-\lambda x} \quad (2.9)$$

A função na equação §2.9, representa a probabilidade de uma transição x apta a disparar naquele momento, considerando sua taxa de disparo λ , a transição é disparada aleatoriamente.

Uma rede de Petri Estocástica Generalizada é uma extensão da rede de Petri Estocástica, onde cada transição pode disparar num tempo nulo ou com um tempo exponencialmente distribuído. Dessa forma as transições são divididas em dois conjuntos: transições imediatas (tempo zero) e transições temporizadas.

2.6 Trabalhos Relacionados

Nesta Seção, busca-se apresentar os trabalhos que contribuíram para o desenvolvimento desta pesquisa. A Seção 2.5.1, trata sobre os trabalhos que utilizaram as redes de Petri e a simulação em diferentes sistemas. A Seção 2.5.2 apresenta o resumo deste capítulo.

2.6.1 Redes de Petri e Simulação

Na literatura estão disponíveis, muitos trabalhos relacionados as redes de Petri, uma vez que a aplicação dessas redes abrangem diversas áreas do conhecimento, sendo possível a análise do comportamento de um sistema, associadas a um ambiente de simulação, porém no campo das soluções de integração interligadas com as redes de Petri, foram poucos os trabalhos que contemplam este estudo.

Os trabalhos relacionados apresentados nesta dissertação abordam o uso das redes de Petri para modelar, simular o comportamento de sistemas de eventos discretos.

O trabalho apresentado por Cargnin [3] propõe a simulação de uma solução de integração usando redes de Petri Estocásticas, com o objetivo de encontrar gargalos de desempenho no comportamento desta solução. A solução de integração consiste em uma aplicação para automatizar a cobrança de telefonemas pessoais que tenham custos realizados pelos funcionários com uso de linhas telefônicas da UNIJUÍ, oferecendo uma melhor funcionalidade da aplicação Central Telefônica na Universidade. Nesse trabalho, o autor toma como entrada do processo de simulação a rede de Petri modelada a partir do modelo conceitual da solução, e como saída um conjunto de variáveis, as quais são analisadas para determinar os possíveis gargalos de desempenho da solução. Nesta dissertação, propõe-se uma extensão da abordagem apresentada por Cargnin, na qual, por meio da utilização das redes de Petri Estocásticas Generalizadas, se busca reduzir os gargalos encontrados numa solução de integração. Cabe destacar que embora o processo inicial de identificação de gargalos seja o mesmo utilizado em [3], o modelo conceitual analisado pertence a outro domínio de aplicação.

O trabalho proposto por Roos-Frantz et al. [38] é o primeiro a propor o uso de Redes de Petri Estocásticas para a análise do comportamento de soluções de integração modeladas com a linguagem Guaraná. O objetivo do trabalho é mostrar que o modelo conceitual desenvolvido no Guaraná

pode ser traduzido em um modelo de simulação usando redes de Petri Estocásticas, assim pode ser simulado para obter informações sobre o desempenho da solução de integração, possíveis gargalos e formação de filas. Cada elemento do modelo conceitual do Guaraná é traduzido em redes de Petri, sendo realizada a verificação do modelo da equivalência de ambos comparando o funcionamento do sistema de execução do Guaraná com o processo de execução de uma rede de Petri. O estudo de caso é fundamentado num problema real de integração que propõe automatizar o processo de registro de novos usuários de um sistema, propõe integrar seis aplicações independentes, implementadas em plataformas diferentes.

A simulação é abordada segundo Roos-Frantz et al. [41] como forma de diminuir custos e minimizar riscos de falhas nos projetos de integração de aplicações, com a proposta de analisar características da solução de integração numa fase anterior a seu desenvolvimento. Esse trabalho contribuiu de forma significativa com esta dissertação, pois proporcionou a tradução dos componentes Guaraná e redes de Petri Estocásticas, possibilitando a identificação das suas características e o comportamento perante diferentes cenários.

Esse trabalho proposto por Marsan [1] propõe a aplicação das redes de Petri Estocásticas Generalizadas na avaliação de desempenho de sistemas multiprocessadores. Mostra-se que redes de Petri Estocásticas Generalizadas, são equivalentes a processos estocásticos de tempo contínuo, e métodos de solução para a derivação da distribuição de probabilidade de estado estacionário são apresentados. São apresentados três exemplos do processo de redução do GSPN. Eles começam a partir de uma descrição detalhada do GSPN de um sistema multiprocessador e levam à construção de três modelos compactos GSPN equivalentes. As propriedades do processo estocástico subjacente ao GSPN detalhado são preservadas nos modelos compactos que, na verdade, sempre produzem resultados exatos. Isso significa que os processos estocásticos subjacentes a esses modelos compactos correspondem a agregações exatas realizadas na descrição estocástica detalhada original. Estes exemplos de aplicação de modelos para a avaliação de desempenho de sistemas multiprocessadores mostram a utilidade e a eficácia desta ferramenta de modelagem.

O trabalho proposto por Bonet [2], apresenta todas as propriedades e o aperfeiçoamento da ferramenta utilizada nesta dissertação PIPE2, onde dá suporte de simulação para as redes de Petri Estocásticas e redes de Petri Estocásticas Generalizadas, propõe a análise de desempenho, faz a comparação com outras ferramentas de simulação, destacando as potencialidades oferecidas pelo PIPE2.

Por fim ilustra as capacidades do PIPE2, segundo um estudo de caso onde uma rede é modelada pelas redes de Petri Estocásticas Generalizadas, onde são disponibilizados vários dados fundamentais para a análise do comportamento de um sistema, como: o número de *tokens* nos lugares, a densidade de probabilidade do *token*, o rendimento das transições temporizadas, entre outros. Os resultados são de certa forma monótonos, devido a simplicidade da rede. No entanto, utilizar um modelo mais realista, resultados semelhantes a estes poderiam ser utilizados para obter tempos médios de espera nos lugares, para diferentes valores do parâmetro de taxa de chegada.

2.7 Resumo do Capítulo

A integração de Aplicações Empresariais dá suporte aos novos processos de negócios, reutilizando funcionalidades presentes nas suas aplicações atuais ou para sincronizar duas ou mais aplicações dentro de seus ecossistemas de *software*. Geralmente essas empresas possuem aplicações que foram desenvolvidas de forma heterogênea, personalizadas ou adquiridas de terceiros, sem que tenham sido projetadas pensando em ser reutilizadas.

A linguagem Guaraná DSL possui cinco elementos básicos: aplicações, processos, tarefas, *slots* e portas. A tecnologia Guaraná atua sob um sistema baseado em mensagens que determina que uma tarefa esteja apta a disparar quando houver ao menos uma mensagem em cada *slot* de entrada.

A simulação tem o objetivo de compreender o funcionamento, prever o desempenho e o comportamento de um sistema. Os modelos são responsáveis por representar da forma equivalente o sistema real para realizar a simulação. Os modelos podem representar um sistema de forma simbólica estruturados com símbolos gráficos (retângulos e retas) com foco na descrição da mecânica de funcionamento, representar de forma analítica através de equações ou um conjunto destas e ainda representar por meio de um sistema de simulação.

As redes de Petri possibilitam modelar sistemas de eventos discretos e oportunizam uma visualização de alta abstração e acompanhamento dinâmico das diferentes atividades do sistema que possuem características predominantemente discretas. As redes de Petri propõem uma modelagem de sistemas direta, fácil e rápida. As redes de Petri possuem uma notação gráfica formada basicamente por lugares, arcos e transições. Também possui uma representação formal matemática que mapeia todos os elementos integrantes da rede e a conexão entre eles, resultando numa representação formal.

Ainda foram apresentados alguns trabalhos que contribuíram de forma significativa para a elaboração desta dissertação, sendo o trabalho proposto por [3], o qual utiliza as redes de Petri Estocásticas para a simulação de

uma solução de integração. Roos-Frantz et al. [41] apresenta a comparação dos elementos das redes de Petri com os elementos da plataforma Guaraná, para a obtenção de um modelo de simulação. Marsan [1] dispõe dos conceitos das redes de Petri Generalizadas e exemplos com os processos similares a esta dissertação, partindo das particularidades de uma rede de Petri Estocástica a correção ou melhoramento de desempenho de um sistema com a utilização das redes de Petri Estocásticas Generalizadas.

Capítulo 3

Modelagem

*Não é na ciência que está a felicidade,
mas na aquisição da ciência.*

Edgar Allan Poe

Neste capítulo é apresentada a proposta de modelagem matemática para o modelo conceitual de solução de integração de um problema do sistema de currículos da agência nacional e tecnologia de Portugal. A Seção §3.1 descreve o caso de estudo, o ecossistema de software e o modelo conceitual de integração. A Seção §3.2 apresenta a equivalência entre os componentes da linguagem Guaraná e as redes de Petri. A Seção §3.3 apresenta o modelo de simulação proposto. A Seção §3.4 encerra o capítulo apresentando um resumo dos temas abordados.

3.1 Caso de Estudo

O caso de estudo utilizado nesta dissertação aborda um problema de integração identificado no contexto do sistema de currículos da Agência Nacional de Ciência e Tecnologia de Portugal. Esse ecossistema de software envolve uma grande quantidade de fontes de dados e aplicações que dão suporte às políticas nacionais de domínio público e às decisões relacionadas com a produção científica nas instituições acadêmicas daquele país.

A quantidade e variedade de repositórios e fontes de dados disponíveis com os currículos dos pesquisadores, publicações científicas, projetos de pesquisa, eventos científicos, supervisão de treinamento avançada e ações de

disseminação da ciência, representam um grande e valioso recurso para pesquisadores, unidades de investigação, Instituições de Ensino Superior (IES), empreendedores e negócios orientados à inovação, explorarem de forma *ad-hoc*, com finalidades diferentes.

No entanto, aqueles que precisam ter uma perspectiva de resultados de pesquisa organizada, sistemática e concisa em unidades nacionais e de níveis de pesquisadores individuais, encontram dificuldades em recolher, harmonizar e sintetizar essas informações [42].

3.1.1 Ecossistema de Software

O problema de integração analisado, consiste em um problema real de integração do sistema de currículos da agência nacional e tecnologia de Portugal. A proposta tem como objetivo integrar quatro aplicações, referidas nesta dissertação como: aplicação local, aplicação DeGois, aplicação CMS e *WebofScience*, respectivamente. Como o próprio nome sugere, "A caracterização da unidade de pesquisa local", refere-se ao sistema de arquivos no computador que executa o processo de integração, contendo informações sobre a unidade de pesquisa para esta solução específica. A "PlataformaDeGois®" representa uma solução do repositório Curriculum Vitae (CV) de investigadores/pesquisadores portugueses, também pode ser fonte de consulta de um currículo de pesquisador no formato XML.

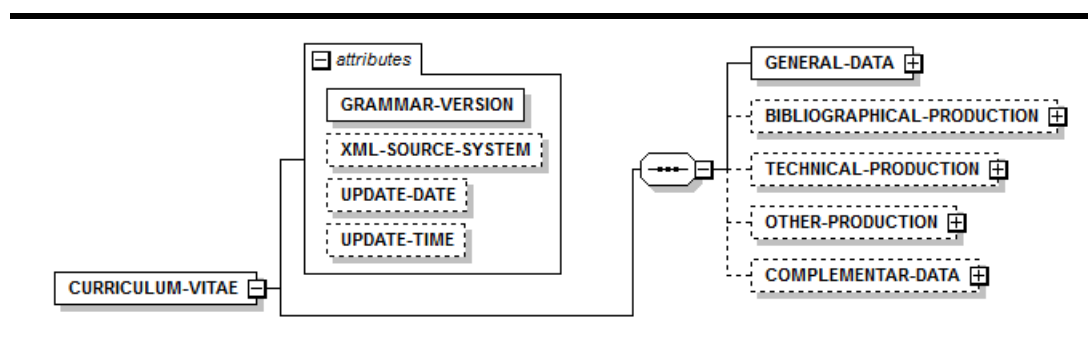


Figura 3.1: Esquema DeGois® CV XML.[42]

De acordo com Sequeira [42], o formato CV XML de um pesquisador, retornado por "PlataformaDeGois®" por meio de *SOAP Web Services* (soluções para apresentações empresarias) e solicitações e respostas *RESTful Web Services*, segue o esquema apresentado na Figura 3.1. A "Aplicação

CMS”, refere-se a um aplicativo de sistema de gerenciamento de conteúdos (Joomla®), os arquivos HTML são soluções de integração resultantes que devem ser armazenados.

A estrutura DeGois® CV XML abrange uma vasta gama de atividades e resultados de pesquisa, que são essenciais para uma análise completa e compreensiva de pesquisadores, unidades de investigação e análise e avaliação de resultados institucionais.

Outras fontes de informação são incorporadas na solução de integração para enriquecer a informação vinda da PlataformaDeGois®. Estas fontes incluem as informações recolhidas a partir da *Web of Science*, relacionado aos índices do *Journal Citation Reports (JCR)*, a plataforma ORCID® para gerenciamento de IDs de pesquisadores internacionais, SCOPUS® para coletar dados de citações específicas de pesquisadores através de dissertações, trabalhos, artigos específicos, etc.

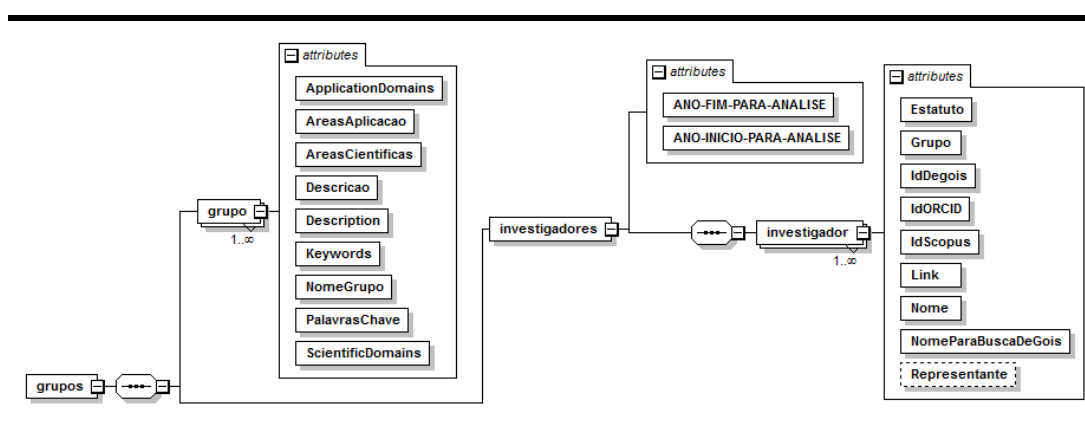


Figura 3.2: Esquema Pesquisadores.XML. [42].

A solução de integração recebe como entrada um documento XML. Este documento é disponibilizado pela aplicação Local e contém a estrutura das informações sobre os pesquisadores, conforme Figura 3.2. Este documento contém informações sobre pesquisadores (tag "Pesquisadores"), nomeadamente o seu estado (atributo "Status"), o grupo de pesquisa a que ele ou ela pertencem (atributo "Grupo"), o nome do pesquisador para fazer uma busca do seu CV em "PlataformaDeGois®" (atributo "NameForDeGoisSearch"), etc. Este documento é atualizado com dados da "PlataformaDeGois®", por exemplo, pesquisador CV última atualização em "PlataformaDeGois®".

3.1.2 Modelo Conceitual de Integração

O modelo conceitual da solução de integração está apresentado na Figura 3.3. A solução recebe como entrada o arquivo Pesquisadores.XML e produz como saída um conjunto de documentos HTML que são encaminhados para uma instância "Joomla®" CMS, sob a forma de artigos "Joomla®" [42]. A entrada da solução é determinada por um documento XML, o qual refere-se a interface da aplicação Local com a solução.

O fluxo de execução de uma mensagem na solução começa na entrada P1. Quando uma mensagem entra na porta P1, esta é enviada para a tarefa T1, a qual divide a mensagem em n mensagens, de acordo com o número de pesquisadores contidos no XML de entrada. A segunda tarefa, T2, filtra as mensagens que chegam a ela, deixando passar apenas aquelas que possuem o valor "Efetiva" no campo Posição. A tarefa T3 replica cada mensagem, fazendo duas cópias, uma é usada para construir um IDquestionário do pesquisador de "DeGois®" e obter o Currículo dos pesquisadores em XML da "PlataformaDeGois®" pela saída P2, a outra cópia permanece na tarefa T3 para ser enriquecida com informações retornadas da "PlataformaDeGois®" pela saída P2. A mensagem contendo o DeGoisId do pesquisador, retornada pela porta de entrada P2, é usada para enriquecer a cópia original da mensagem em T3, e então enviada para T4.

A tarefa T4 faz uma mudança de esquema de mensagem para a mensagem manter nova informação vinda da "Web of Science". Especificamente, o atributo XML "*Factor-de-Impacto-JCR*" é adicionado ao elemento XML "Artigo-Detalhado". As Tarefas T5 recuperam informações da "Web of Science®" ligado ao JCR fator de impacto do jornal, para ser associado, pelos meios de ISSNIds, para a produção bibliográfica dos pesquisadores, usando o atributo "*Factor-of-Impact-JCR*".

As informações sobre os pesquisadores não precisam ser tratados individualmente. A tarefa T6 reunifica as mensagens com informações sobre cada pesquisador para uma mensagem única, para o processamento gradual da unidade de pesquisa. A tarefa T7 replica esta mensagem em cinco cópias, cinco documentos de produção HTML diferentes, contendo indicadores de escala de unidade de pesquisa em um por categoria (projetos, trabalhos, eventos organizados, prêmios e treinamento avançado).

As tarefas T8, T9, T10, T11 e T12 (tarefas menores) realizam limpeza de mensagem, preservando apenas informações relatados de cada indicador de pesquisa específico para ser calculado/processado.

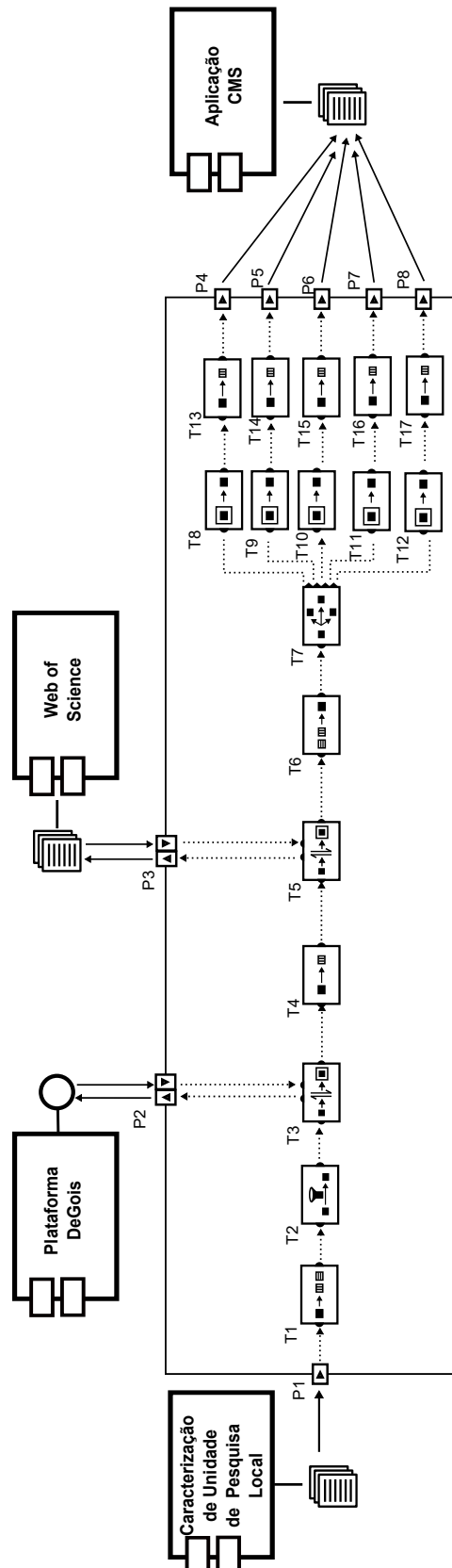


Figura 3.3: Modelo Conceitual. [42].

Finalmente, as tarefas T13, T14, T15, T16 e T17 realizam transformação de mensagem, mais precisamente, transformação de dados representados em XML em documentos HTML, correspondente as cinco categorias diferentes de indicadores de pesquisa. A produção dessas tarefas (cinco documentos HTML) é encaminhada através de escalas P4-P8 para "Aplicação CMS" na forma de artigos CMS, e imediatamente acessível pela instância CMS ("Joomla®").

3.2 Equivalência entre redes de Petri e Guaraná

A proposta de tradução do modelo conceitual da solução de integração em redes de Petri é possível pela relação análoga entre seus componentes, conforme a Tabela §3.1 e pela semelhança do modelo de execução. Os *tokens* das redes de Petri são análogos às mensagens da solução de integração, e os lugares são equivalentes aos *slots*, que atuam como *buffers*. Mesmo havendo uma equivalência entre componentes, cada tarefa do Guaraná tem uma semântica implementada que pode ser traduzida, que não foi feita nesta dissertação. Foi criado um modelo de simulação que abstrai detalhes da semântica das tarefas. No modelo, as tarefas são eventos que quando ocorrem, mudam o estado do sistema.



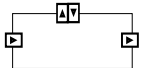

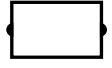



Notação	Conceito	Notação	Conceito
	Aplicação		Porta de Entrada
	Processo		Porta de Saída
	Tarefa		Porta de Resposta
	Slot		Porta de Solicitação

Tabela 3.1: Comparação entre os Elementos do Guaraná e redes de Petri [38].

Roos-Frantz [38], propõe a analogia entre a troca de estados em uma solução modelada com Guaraná e uma troca de estados em uma rede de Petri como é apresentado na Figura §3.4.

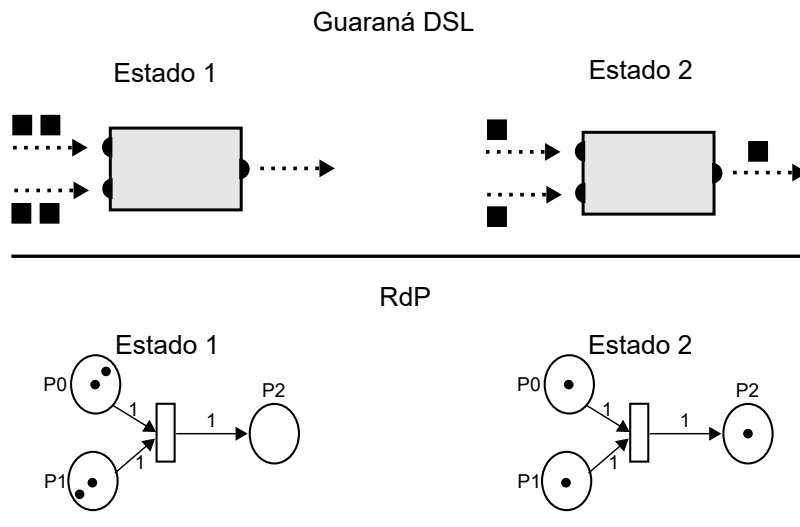


Figura 3.4: Semelhança entre troca de estados [38].

Para tradução de uma solução Guaraná específica em uma rede de Petri é levada por meio da conexão dos componentes da rede de Petri no mesmo estilo dos componentes na solução Guaraná. Dentre vários tipos de tarefas Roos-Frantz [38] apresenta alguns tipos: comunicador, mapeador, modificador e roteador.

A Tabela §3.2 ilustra as tarefas do comunicador, que permitem enviar ou receber mensagens através de portas para aplicações sendo integradas.

Guaraná	RdP	Nome da Tarefa
		In Communicator
		Out Communicator
		In-Out Communicator
		Out-In Communicator

Tabela 3.2: Tarefas Comunicadores

A Tabela §3.3 ilustra as tarefas do mapeador, que permitem mapear o corpo de uma mensagem de XML para fluxo ou vice-versa.

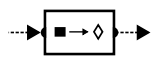
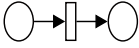
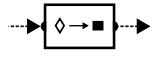
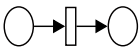
Guaraná	RdP	Nome da tarefa
		XML2Stream
		Stream2XML

Tabela 3.3: Tarefas Mapeadores

A Tabela §3.4 ilustra as tarefas do roteador, que obtém uma ou mais mensagens de seus slots de entrada e transferem ou copiam para um ou mais slots de saída.


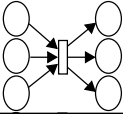
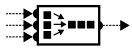
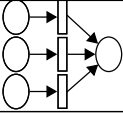

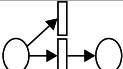

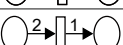

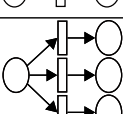

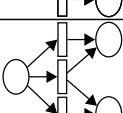

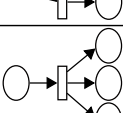
Guaraná	RdP	Nome da tarefa
		Correlator
		Merger
		Filter
		Idempotent Transfer
		Dispatcher
		Distributor
		Replicator

Tabela 3.4: Tarefas Roteadoras

A Tabela §3.5 ilustra as tarefas de modificação, que adicionam ou removem dados do corpo e do cabeçalho das mensagens de entrada.

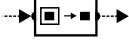


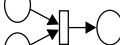
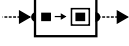
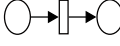

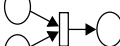
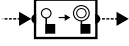

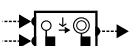
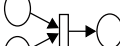

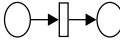

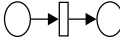
Guaraná	RdP	Nome da Tarefa
		Slimmer
		Context-based Slimmer
		Content Enricher
		Context-based Content Enricher
		Header Enricher
		Context-based Header Enricher
		Header Promoter
		Header Demoter

Tabela 3.5: *Tarefas Modificadoras*

A tradução do Guaraná para a rede de Petri é um processo que embora as informações necessárias para a simulação possam ser traduzidas, os componentes da tarefa da linguagem de domínio específico tem informações semânticas que não podem ser traduzidas. O resultado é que vários componentes em Guaraná semanticamente diferentes são mapeados para uma rede de Petri idêntica.

Quando traduzimos uma solução, todas as mensagens em Guaraná torna-se um símbolo, e cada unidade de trabalho em Guaraná torna-se um disparo de transição na rede de Petri, de modo que a execução da rede de Petri é equivalente ao sistema de execução de uma sequência de instruções únicas em Guaraná.

3.3 Modelo de Simulação Proposto

A Figura §3.5 representa o modelo de simulação da solução de integração do estudo de caso, apresentado na Figura §3.3. A rede foi construída

seguindo a tradução das tarefas em Guaraná DSL para as respectivas redes de Petri. Estas redes estavam ligadas para cada estado na mesma forma das interligações das tarefas representadas na Tabela §3.6. Na rede de Petri resultante, as transições foram traduzidas de acordo com suas tarefas correspondentes e os lugares de acordo com seus respectivos *slots*, observando a organização e ligação das tarefas, a disposição das portas e a direção do fluxo de mensagens.

A representação da porta de entrada no modelo de simulação é equivalente a transição P1, que quando disparada insere um *token* no lugar S1, que representa o *slot* S1, a porta de saída envia mensagens para as aplicações e indica o fluxo de *tokens* para fora do processo. Cabe destacar a tarefa correlacionador, que é representada pela transição T3d e T5d, pois são as únicas tarefas que para tornarem-se aptas à disparar dependem de *tokens* em dois *slots*, no caso o T3d, depende dos *slots* S7 e S8 e o T5d dos *slots* S15 e S16.

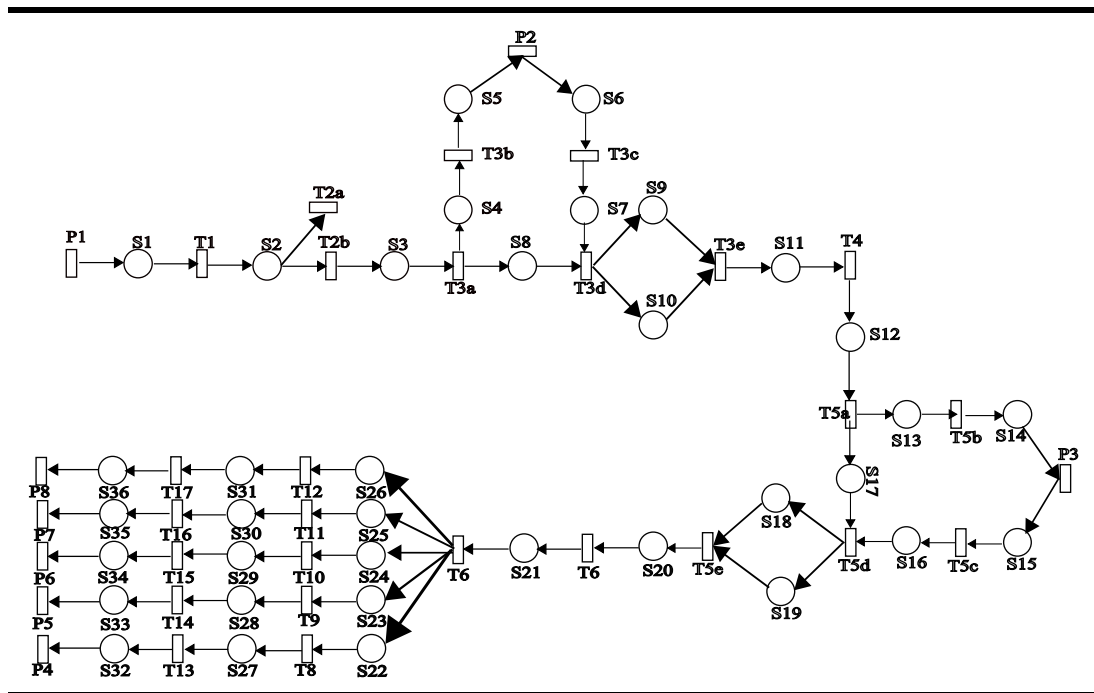


Figura 3.5: Modelo de Simulação das redes de Petri Estocásticas.

A Figura §3.6 apresenta o modelo de simulação da solução de integração, na qual são inseridas transições imediatas no lugar de algumas transições temporizadas para a análise do comportamento do sistema.






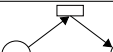

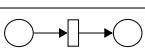



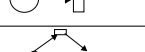

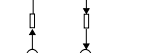

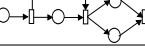

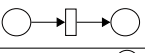

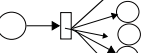

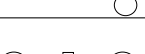


Guaraná	Grupo/Nome	Funcionalidade	RdP
	Porta de Entrada	Insere mensagens na solução de integração	
	Porta de Saída	Envia mensagens a uma aplicação integrada	
	Porta de Solicitação	Solicita informações à uma aplicação integrada	
	Transformador/Splitter	Adequa formato da mensagem	
	Roteador/Filter	Remove mensagens do fluxo ou permite a sua passagem	
	Enquiper	Tarefa Composta por um correlacionador, enriquecedor de conteúdo, replicador e transformador	
	Transformador/Aggregator	Adequa formato da mensagem	
	Roteador/Replicator	Faz cópias idênticas de uma mensagem	
	Modificador/Slimmer	Adiciona conteúdo a uma mensagem	
	Transformador/Traslator	Adequa formato da mensagem	
	Correlacionador	Processa juntamente mensagens com alguma relação	
	Enriquecedor de Conteúdo	Adiciona conteúdo a uma mensagem	

Tabela 3.6: Tabela de Equivalência entre Guaraná DSL e redes de Petri.

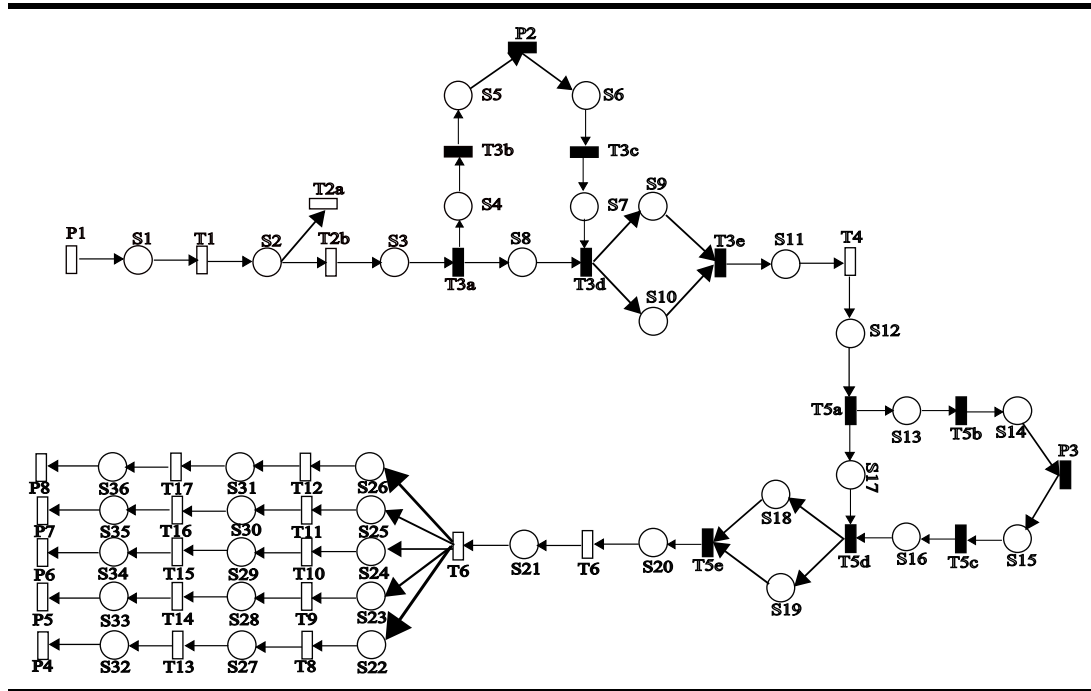


Figura 3.6: Modelo de Simulação das redes de Petri Estocásticas Generalizadas.

A primeira etapa na construção do modelo de simulação baseado em redes de Petri Estocásticas Generalizadas é a definição de quais as transições devem ser temporizadas e quais devem ser imediatas. Para isso, foi necessário identificar quais transições no modelo de simulação baseado em RdP Estocásticas geram gargalos de desempenho. Após a identificação dessas transições decidiu-se inicialmente inserir tarefas imediatas nas transições T3d e T5d (transições com gargalo de desempenho) e observar o comportamento. Após a análise foi concluído que não houve uma redução ou eliminação dos gargalos como o esperado, foi então que foram incluídos tarefas imediatas nas transições que fazem todo o processo de mensagem até chegarem nas transições que indicam gargalos, mais especificamente nas tarefas pertencentes a tarefa Enquirer do modelo conceitual, que é chamada de tarefa composta, sua composição é representada por outras quatro tarefas que são replicador, correlacionador, enriquecedor de conteúdo e transformador, conforme Figura §3.7.

Foram feitas simulações para a seleção de quais as transições temporizadas deixariam o sistema mais dinâmico com a inserção das transições imediatas, selecionou-se 12 transições temporizadas que foram transforma-

das em imediatas, como ficaram mais de uma transição imediata foram adotadas prioridades para a execução das transições, a ordem foi definida pelo sentido do fluxo inicial de mensagens no sistema.

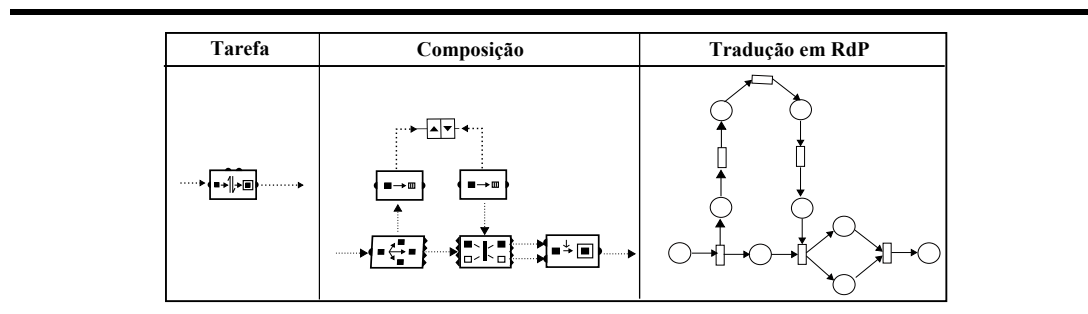


Figura 3.7: Tarefa Enquirer

3.4 Resumo do Capítulo

Neste capítulo, foram apresentados o problema de integração, o ecossistema de software e o modelo conceitual do caso de estudo utilizado. Além disso, o processo de construção do modelo de simulação foi descrito e o modelo de simulação foi detalhado.

Capítulo 4

Experimentação

Nenhum homem realmente produtivo
pença como se estivesse escrevendo uma dissertação.

Albert Einstein

Neste capítulo é apresentado o experimento de simulação do modelo proposto. A Seção §4.1 descreve o cenário utilizado, a ferramenta de simulação, as variáveis observadas e os resultados experimentais. Na Seção §4.2 é realizada a análise e interpretação dos resultados obtidos a partir da simulação da solução de integração modelada. A Seção §4.3 apresenta a verificação do modelo de simulação proposto. A Seção §4.4 encerra o capítulo apresentando um resumo dos temas abordados.

4.1 Experimento

Para executar as simulações utilizou-se a ferramenta computacional PIPE2, para a exclusão de qualquer dúvidas em relação a veracidade do experimento, uma vez que experimentos desta característica deve-se utilizar em torno de 25 repetições, conforme especifica Grinstead e Snell [16]. O objetivo das repetições é eliminar qualquer anormalidade nos resultados, por se tratar de um processo estocástico.

A ferramenta de simulação PIPE2 apresenta nos seus resultados a média de *tokens* acumulados em cada lugar e o intervalo de confiança de cada média. O mecanismo de disparos leva em conta modelos probabilísticos.

Para a determinação da carga de entrada foi feito um levantamento da quantidade de pesquisadores que atuam nas universidades e centros politécnicos de Portugal, tendo em vista que o estudo de caso se refere a uma unidade local de pesquisa, e então tomado por parâmetro uma média de pesquisadores de cada unidade. Dessa forma, definiu-se como carga de entrada 300 *tokens*, correspondente a média de pesquisadores.

Para definir o número de disparos a serem utilizados na simulação, utilizou-se a seguinte metodologia. Para cada mensagem que entra na solução, 1 (uma) mensagem deve ser gerada em cada porta de saída. Para representar esse comportamento na rede de Petri, para cada *token* no lugar P1, são necessários 33 disparos para que se gere um *token* nos lugares P4, P5, P6, P7 e P8. Dessa forma, tomando como entrada 300 *tokens*, são necessários no mínimo 9900 disparos para simular o cenário proposto.

Portanto, para este experimento, cada repetição teve como parâmetro 10000 disparos.

4.1.1 Variáveis Observadas

Para a identificação de gargalos de desempenho e uma possível redução destes gargalos em uma solução de integração, deve-se observar algumas variáveis que são essenciais neste processo. Identificou-se as seguintes variáveis para a realização do experimento: acúmulo de *tokens* nos lugares, número de execuções realizadas por cada transição, frequência de disparos das transições e tempo médio de espera de uma transição apta a executar.

O acúmulo de *tokens* nos lugares representa a formação de filas nos *slots* da solução de integração. As filas acontecem quando uma ou mais mensagens se acumulam nos *slots* para serem processadas pelas tarefas.

O número de disparos das transições na simulação representa o número de execuções das tarefas do modelo conceitual.

A frequência de disparos representa o percentual de disparos de cada tarefa em relação ao total de disparos.

O tempo médio de espera de uma transição apta a disparar na simulação representa o tempo médio de espera de uma determinada tarefa para executar.

4.1.2 Apresentação da Ferramenta

A ferramenta utilizada para a simulação do modelo foi o *Platform Independent Petri net Editor 2 (PIPE2)* [1]. PIPE2 é uma plataforma de código aberto para edição e análise de redes de Petri, com interface de fácil utilização, conta com muitos módulos de análise, animação dos projetos e possibilita exportação de redes para arquivos de imagem.

PIPE2 foi inicialmente desenvolvida por um grupo de alunos de pós graduação do Departamento de Computação do *Imperial College de Londres*, por meio de um projeto de pesquisa nomeado "*The Platform Independent Petri net Editor PIPE*". O projeto teve início entre 2002 e 2003 e mantém-se em funcionamento e aperfeiçoamento com lançamento de versões atualizadas, disponibilizadas para download no endereço <http://sourceforge.net/projects/pipe2/> [9].

Na Figura §4.1 é ilustrada a interface gráfica do PIPE2 para a criação e edição de redes de Petri. Além da funcionalidade básica de design, a ferramenta oferece recursos de animação e análise das redes.

As transições das Redes de Petri Estocásticas Generalizadas são representadas pelo PIPE2 com retângulos com preenchimento na cor preta, também chamadas de transições imediatas e as redes de Petri Estocásticas com retângulos sem preenchimento, denominadas transições temporizadas. A taxa de disparo referente a cada transição estocástica são editadas através da aba *Transition Editor*, como apresenta a Figura §4.2, na aba *Name*, identificamos a transição e na aba *Rate* é determinada a taxa de disparo referente a transição, os marcadores *Single* e *Timed* representam, respectivamente, um único servidor para o sistema e transição temporizada e o marcador *Immediate* para transição imediata.

As transições estocásticas do PIPE2 são sensibilizadas com *Rate* igual 1 que representa a taxa porcentual, o que significa que as transições são definidas com probabilidade de disparo de 100%. Desta forma, só é necessário alterar transições que tenham taxa de disparo inferior e nas transições das redes de Petri Estocásticas Generalizadas ao selecionar a transição como imediata, ela terá prioridade sobre as outras transições temporizadas e se tiver outra transição imediata define-se uma prioridade.

Para realizar a simulação, utiliza-se o módulo *Simulation*, conforme a Figura §4.3 no qual calcula o número médio de *tokens* por lugar, com intervalo

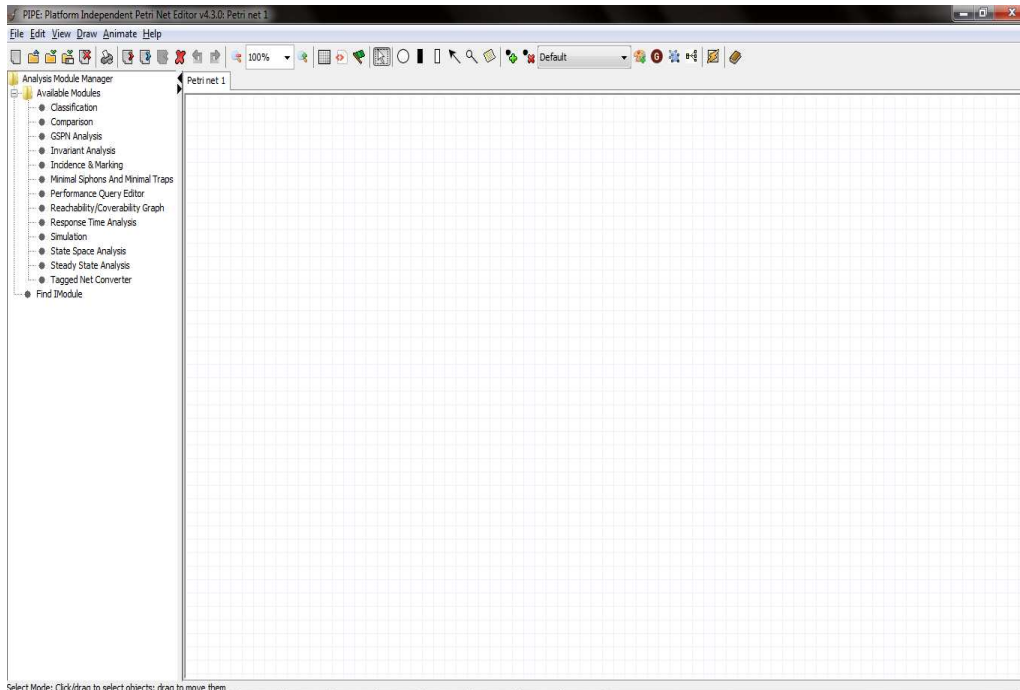


Figura 4.1: Interface Gráfica PIPE2

de confiança de 95% para cada lugar da rede de Petri, este módulo também realiza a evolução da rede de Petri de acordo com as regras de disparo das transições, permitindo a visualização do fluxo de *tokens* e a identificação dos estados alcançáveis do sistema modelado. O módulo de simulação de redes apresenta duas abas de edição dos parâmetros de simulação, *Firings* (disparos) e *Replications* (repetições), onde será implementado o cenário proposto.

Após a realização da simulação, é apresentada conforme a Tabela §4.1, uma tabela com três colunas, uma identificando o Lugar, outra o número médio de *tokens* e por fim uma com 95% do intervalo de confiança, cabe ressaltar que por se tratar experimento estocástico, em cada repetição de simulação os valores dos resultados são diferentes, porém todos com valores aceitáveis. A ferramenta disponibiliza a tabela no formato HTML e pode ser obtida uma cópia.

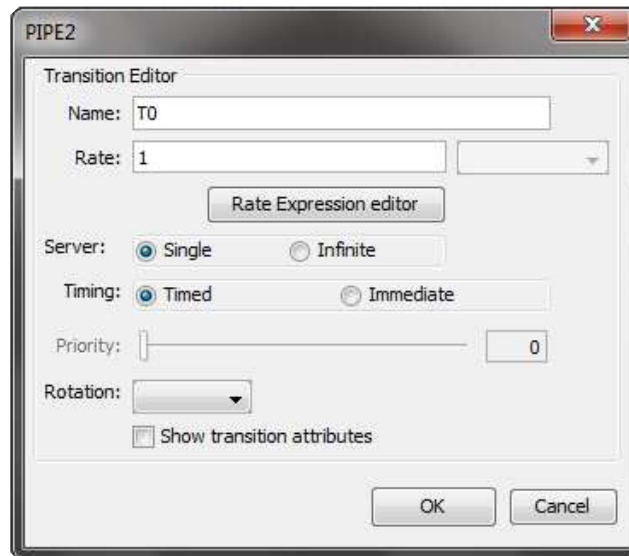


Figura 4.2: *Aba de Sensibilização das Transições Estocásticas e Estocásticas Generalizadas do PIPE2*

4.2 Resultados e Discussão

Inicialmente o modelo de simulação foi submetido ao cenário previamente destacado para que fosse feita a avaliação do seu comportamento perante a carga de entrada e a análise do modelo diante das redes de Petri Estocásticas e redes de Petri Estocásticas Generalizadas. Os resultados são representados por meio de tabelas e gráficos e suas respectivas interpretações.

As Tabelas §4.2 e §4.3 representam as informações do modelo de simulação e dados do experimento divididas em colunas que apresentam o comportamento do modelo perante as condições impostas nas simulações nas duas situações levantadas neste trabalho.

Na coluna 1, são mostradas todas as transições que compõem o modelo. As linhas de 1 a 8 representam as portas do modelo conceitual, e as demais as tarefas. A última linha é utilizada para apresentar a soma total dos valores de cada coluna.

Na coluna 2, estão descritas as taxas de disparos das respectivas transições. Na coluna 3 estão as probabilidades globais de disparo para cada

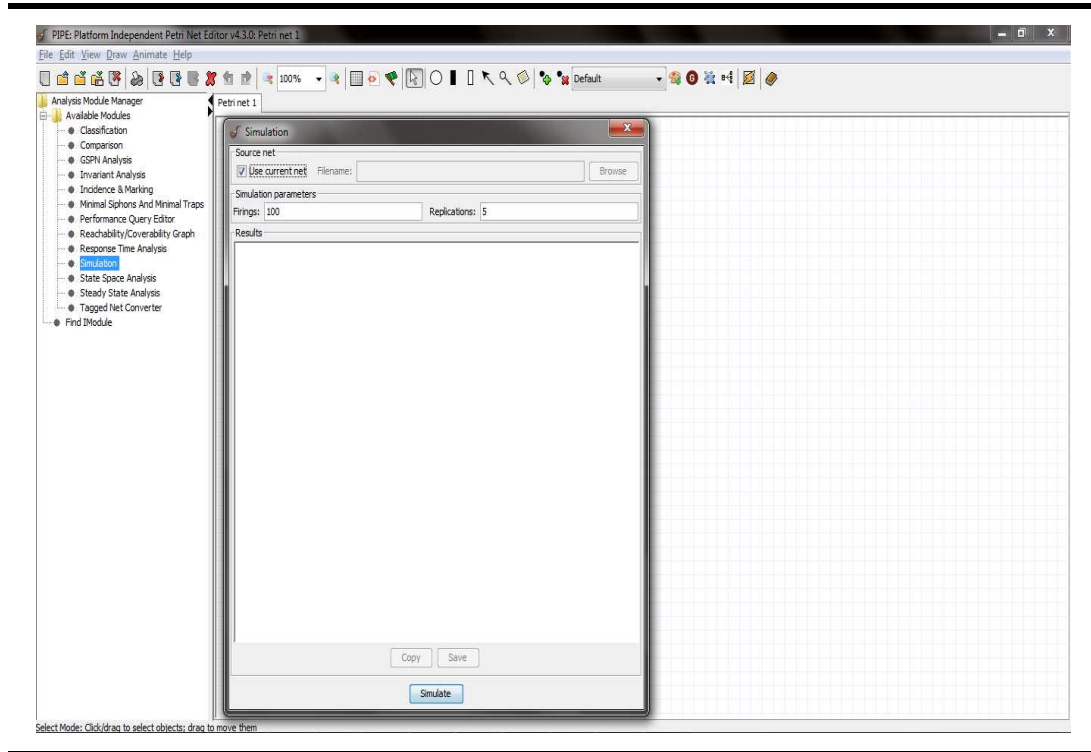


Figura 4.3: Interface Simulation PIPE2

transição, que é obtida pela razão da taxa de disparo individual com a soma das taxas de disparo. A probabilidade global possibilita calcular o valor esperado de disparos para cada transição pela multiplicação da probabilidade global pelo número total de disparos do experimento, como descrito na coluna 4. A soma dos disparos esperados é igual ao total de disparos implementados no sistema como mostra a última linha da coluna 4, com soma sendo igual a 10.000 disparos.

A coluna 5, apresenta a média do número de disparos realizados de cada transição no experimento realizado. A coluna 6, apresenta o total de disparos realizados, obtido levando em consideração a variação de zero até dobro de *tokens*. A coluna 7, apresenta a frequência de disparos das transições no experimento, obtida pela razão do número de disparos da transição pelo total de disparos implementado na simulação. A frequência representa o percentual de disparos da referida tarefa em relação ao número total de disparos.

Na coluna 8, determina-se o atraso médio de espera de uma tarefa apta a disparar, quantificado em disparos, obtido pela razão do número total de disparos pelos disparos realizados.

Petri net simulation results

Place	Average number of tokens	95% confidence interval (+/-)
S1	21,86541	12,43398
S2	7,63774	3,85383
S3	0,0305	0,001
S4	0,12179	0,00389
S5	0,0305	0,00097
S8	0,0304	0,00097
S6	0,0305	0,00097
S7	0,0304	0,00097
S10	0,0304	0,00097
S9	0,0304	0,00097
S11	4,67763	1,68815
S12	0,0304	0,0005
S13	0,12169	0,00199
S14	0,0304	0,0005
S17	0,0304	0,0005
S16	0,0304	0,0005
S18	0,0304	0,0005
S19	0,0304	0,0005
S20	2,79802	1,50098
S21	3,91661	1,66407
S24	7,58544	2,27811
S23	5,21388	2,24372
S25	4,51675	2,28982
S26	5,45875	3,92481

Tabela 4.1: Tabela de Resultados das Simulações

Ao analisar os resultados experimentais das tabelas, observa-se poucas diferenças entre os valores de disparos esperados quanto aos resultados realizados, uma vez que o mecanismo de seleção de disparos é estocástica, aleatória. Porém ao analisarmos a quantidade de disparos realizados na porta de entrada P1 ela é muito maior que as portas de saída P4, P5, P6, P7, P8, o que demonstra o acúmulo de *tokens* em algum lugar dentro do sistema.

Outro fator que deve ser considerado é a estimativa da frequência de disparo das transições que serve para estimar o atraso médio de uma transição apta a disparar. A espera representa o atraso das redes de Petri Estocásticas e redes de Petri Estocásticas Generalizadas. Define-se que cada transição tem um tempo médio de espera entre seus disparos, usando como unidade de tempo disparos. A análise do tempo de disparo mostra que a transição T1

Transição	Taxa de Disparo	Probabilidade Global	Expectativa de Disparos	Média de disparos	Disparos Realizados	Frequência de Disparos	Tempo Médio de Espera para Disparo
P1	1	0,030120482	301,20	240,43	480,86	4,81%	20,80
P2	1	0,030120482	301,20	164,31	328,61	3,29%	30,43
P3	1	0,030120482	301,20	147,53	295,05	2,95%	33,89
P4	1	0,030120482	301,20	132,68	265,36	2,65%	37,68
P5	1	0,030120482	301,20	134,03	268,06	2,68%	37,30
P6	1	0,030120482	301,20	134,06	268,12	2,68%	37,30
P7	1	0,030120482	301,20	133,84	267,67	2,68%	37,36
P8	1	0,030120482	301,20	134,40	268,80	2,69%	37,20
T1	1	0,030120482	301,20	216,01	432,02	4,32%	23,15
T2a	0,2	0,006024096	60,24	40,10	80,19	0,80%	124,70
T2b	1	0,030120482	301,20	173,34	346,68	3,47%	28,84
T3a	1	0,030120482	301,20	170,62	341,24	3,41%	29,30
T3b	1	0,030120482	301,20	167,10	334,20	3,34%	29,92
T3c	1	0,030120482	301,20	162,23	324,47	3,24%	30,82
T3d	1	0,030120482	301,20	159,13	318,25	3,18%	31,42
T3e	1	0,030120482	301,20	156,94	313,87	3,14%	31,86
T4	1	0,030120482	301,20	154,90	309,80	3,10%	32,28
T5a	1	0,030120482	301,20	152,46	304,92	3,05%	32,80
T5b	1	0,030120482	301,20	149,73	299,46	2,99%	33,39
T5c	1	0,030120482	301,20	145,77	291,55	2,92%	34,30
T5d	1	0,030120482	301,20	144,14	288,27	2,88%	34,69
T5e	1	0,030120482	301,20	142,56	285,13	2,85%	35,07
T6	1	0,030120482	301,20	140,70	281,39	2,81%	35,54
T7	1	0,030120482	301,20	139,18	278,35	2,78%	35,93
T8	1	0,030120482	301,20	136,83	273,67	2,74%	36,54
T9	1	0,030120482	301,20	137,21	274,42	2,74%	36,44
T10	1	0,030120482	301,20	137,46	274,92	2,75%	36,37
T11	1	0,030120482	301,20	137,03	274,07	2,74%	36,49
T12	1	0,030120482	301,20	137,64	275,29	2,75%	36,33
T13	1	0,030120482	301,20	134,71	269,42	2,69%	37,12
T14	1	0,030120482	301,20	135,74	271,49	2,71%	36,83
T15	1	0,030120482	301,20	135,70	271,40	2,71%	36,85
T16	1	0,030120482	301,20	135,25	270,50	2,70%	36,97
T17	1	0,030120482	301,20	136,25	272,50	2,73%	36,70
TOTAL	33,2	1	10000	5000	10000	1	*****

Tabela 4.2: Resultados Experimentais das SPN.

tem um tempo de espera menor que as demais, o que reflete no seu número de execuções maior que as demais, da mesma forma que as transições que possuem baixa taxa de disparo têm um atraso bem elevado. Em tese um *token* depende de 33 disparos para percorrer todo o modelo se não for filtrado, o que determina que cada tarefa teria atraso de 33 disparos. A média geral do atraso médio das transições do experimento, desconsiderando as transições com função de filtragem (T2a), nas redes de Petri Estocásticas é de 33,88 dis-

Transição	Taxa de Disparo	Probabilidade Global	Expectativa de Disparos	Média de disparos	Disparos Realizados	Frequência de Disparos	Tempo Médio de Espera para Disparo
P1	1	0,030120482	301,20	224,22	448,43	4,48%	22,30
P2	1	0,030120482	301,20	158,36	316,72	3,17%	31,57
P3	1	0,030120482	301,20	153,50	307,00	3,07%	32,57
P4	1	0,030120482	301,20	135,45	270,90	2,71%	36,91
P5	1	0,030120482	301,20	132,40	264,81	2,65%	37,76
P6	1	0,030120482	301,20	130,30	260,59	2,61%	38,37
P7	1	0,030120482	301,20	134,27	268,55	2,69%	37,24
P8	1	0,030120482	301,20	133,57	267,13	2,67%	37,43
T1	1	0,030120482	301,20	202,35	404,70	4,05%	24,71
T2a	0,2	0,006024096	60,24	36,26	72,52	0,73%	137,90
T2b	1	0,030120482	301,20	158,45	316,91	3,17%	31,55
T3a	1	0,030120482	301,20	158,42	316,85	3,17%	31,56
T3b	1	0,030120482	301,20	158,39	316,79	3,17%	31,57
T3c	1	0,030120482	301,20	158,33	316,66	3,17%	31,58
T3d	1	0,030120482	301,20	158,30	316,60	3,17%	31,59
T3e	1	0,030120482	301,20	158,27	316,54	3,17%	31,59
T4	1	0,030120482	301,20	153,59	307,19	3,07%	32,55
T5a	1	0,030120482	301,20	153,56	307,13	3,07%	32,56
T5b	1	0,030120482	301,20	153,53	307,07	3,07%	32,57
T5c	1	0,030120482	301,20	153,47	306,94	3,07%	32,58
T5d	1	0,030120482	301,20	153,44	306,88	3,07%	32,59
T5e	1	0,030120482	301,20	153,41	306,82	3,07%	32,59
T6	1	0,030120482	301,20	150,61	301,23	3,01%	33,20
T7	1	0,030120482	301,20	146,70	293,39	2,93%	34,08
T8	1	0,030120482	301,20	143,11	286,21	2,86%	34,94
T9	1	0,030120482	301,20	141,48	282,97	2,83%	35,34
T10	1	0,030120482	301,20	139,11	278,22	2,78%	35,94
T11	1	0,030120482	301,20	142,18	284,36	2,84%	35,17
T12	1	0,030120482	301,20	141,24	282,48	2,82%	35,40
T13	1	0,030120482	301,20	138,60	277,20	2,77%	36,08
T14	1	0,030120482	301,20	136,72	273,44	2,73%	36,57
T15	1	0,030120482	301,20	134,30	268,61	2,69%	37,23
T16	1	0,030120482	301,20	137,22	274,44	2,74%	36,44
T17	1	0,030120482	301,20	136,86	273,72	2,74%	36,53
TOTAL	33,2	1	10000	5000,00	10000,00	100%	*****

Tabela 4.3: Resultados Experimentais das GSPN.

paros e nas redes de Petri Estocásticas Generalizadas é de 34,61 disparos. As transições com a função de filtragem são desconsideradas no cálculo da média tendo em vista a taxa de disparo ser muito baixa e o atraso muito elevado e pelo fato de que é estimado o atraso médio considerando que a mensagem não é filtrada. A análise do tempo de espera das transições revela que em média as transições tem um atraso muito semelhante.

Assim o tempo de espera de uma tarefa, nas duas redes, de acordo com esse experimento, não interfere na formação de filas, pois o tempo médio de espera e o tempo real de uma tarefa são muito próximos e não chegam

a interferir na formação de gargalos.

Para uma melhor visualização, foram construídos gráficos para a representação e comparação do comportamento dos *tokens* nos lugares. A Figura §4.4, remete-se a simulação de uma rede de Petri Estocástica, percebe-se um acúmulo elevado desproporcional de *tokens* nos *slots* S4 e S13, em relação aos outros *slots* do sistema, sendo que o *slot* S4 tem uma média de armazenamento de 5,5 vezes maior de *tokens* que os demais *slots* e no *slot* S13 a média é de 4 vezes maior, os *slots* S4 e S13 alimentam as transições T3d e T5d, que por equivalência com o Guaraná DSL pode ser denominado como tarefa correlacionador, visto que a formação de filas é uma característica dos sistemas orientados a mensagens, mas a fila pode se tornar um gargalo quando apresenta um acúmulo desproporcional ao estado geral do sistema, com base nisto as tarefas acima podem ser possíveis gargalos de desempenho.

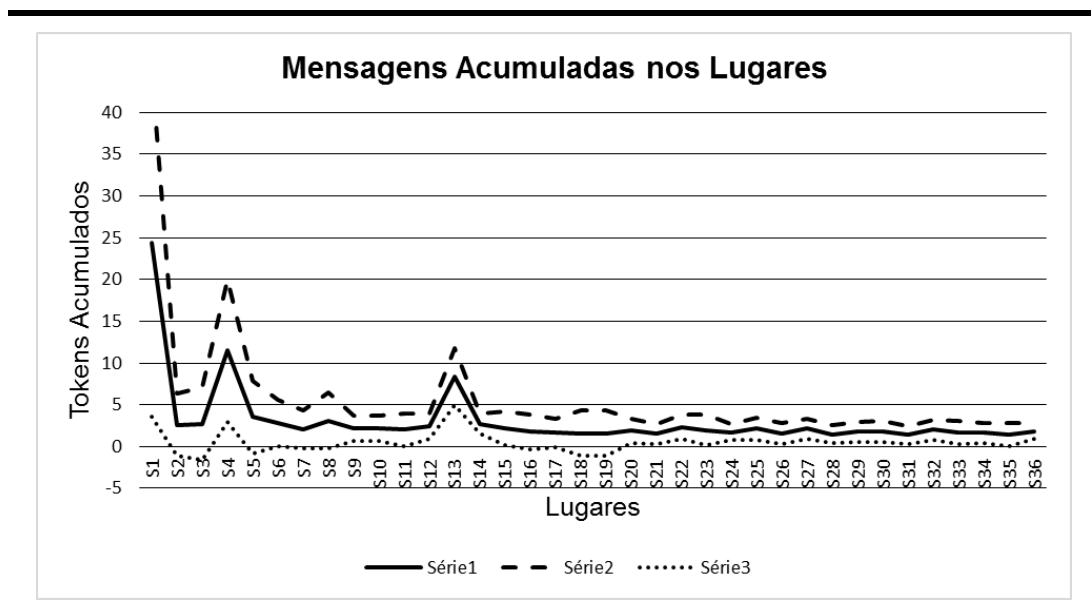


Figura 4.4: Gráfico de Formação de Filas nos Slots

Na Figura §4.5, o qual refere-se a simulação de uma rede de Petri Estocástica Generalizada, onde são atribuídas transições imediatas nas transições que são possíveis gargalos de desempenho nas redes de Petri Estocásticas, observa-se a diminuição de *tokens* tendendo a zero, ou seja, não tem acúmulo *tokens* nesses *slots*. Porém observa-se um crescimento de *tokens* no *slot* S11 e do *slot* 20 em diante, vale ressaltar que o aumento de *tokens* acontecem de maneira proporcional nos *slots* que são alimentados por transições

temporizadas.

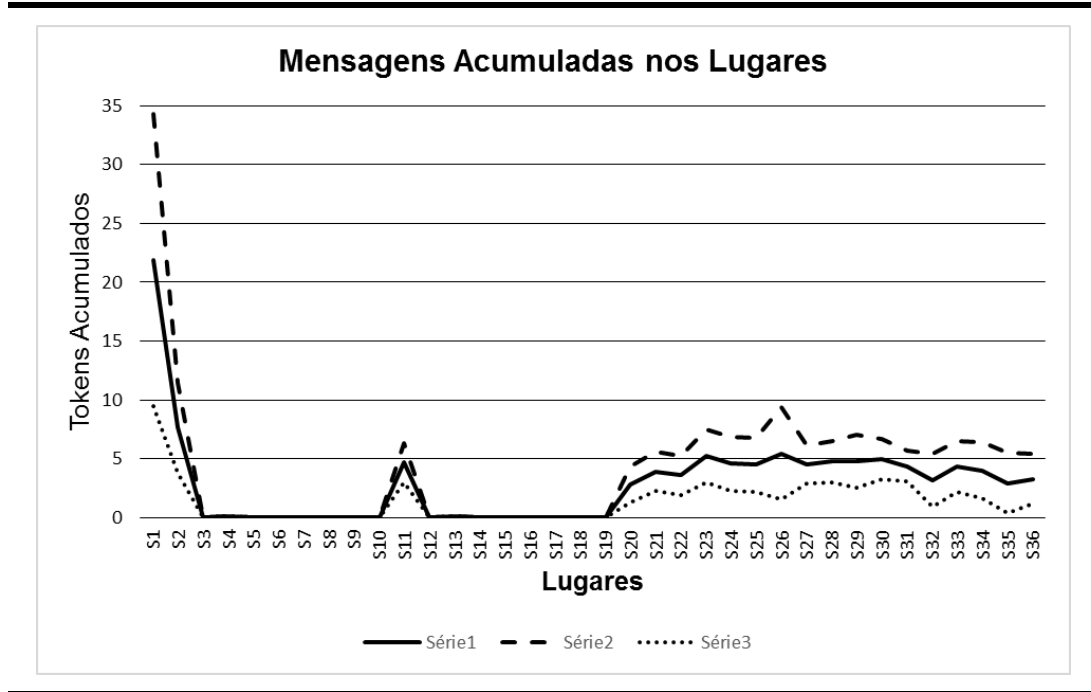


Figura 4.5: Gráfico de Formação de Filas nos Slots

Comparado os gráficos das Figuras §4.4 e §4.5, conclui-se que o acúmulo de tokens nos *slots* que alimentam as tarefas temporizadas são distribuídas de maneira uniforme/linear, como observa-se no gráfico da simulação de redes de Petri Estocásticas, os *slots* onde não há um acúmulo de *tokens* permanece com uma média de 2 *tokens* por *slot* e no gráfico das redes de Petri Estocásticas Generalizadas o acúmulo desses *tokens* nos *slots* tem uma média de 4 *tokens* por *slot*. Portanto, os *tokens* distribuem-se de forma uniforme/linear nos *slots*, não caracterizando acúmulos excessivos e desproporcionais nas redes de Petri Estocásticas Generalizadas.

4.3 Verificação do Modelo de Simulação

Os modelos de simulação estão cada vez sendo mais explorados a fim de auxiliar nas tomadas de decisões. Porém, a questão é que quando esses modelos são submetidos a um ambiente de simulação, os resultados ou dados obtidos refletem ou não a realidade. Para que isso seja feito é utilizado a

verificação do modelo, assegurando assim que o programa, modelo e sua implementação estejam corretos. Para o problema desta dissertação foi utilizada algumas técnicas de verificação disponíveis na literatura para os dois experimentos propostos, tanto para as redes de Petri Estocásticas quanto para redes de Petri Estocásticas Generalizadas. Inicialmente foi utilizada a técnica de Kleijnen [26], onde a expectativa do comportamento do sistema é determinada por um especialista da área. Para o especialista, uma solução de integração é um sistema de mensagem composto por unidades de armazenamento temporário denominado *slot*, no qual espera-se que tenha um armazenamento de mensagens. A tarefa correlacionador (T3d) é alimentada pelos *slot* S4 e S8, onde o *slot* S4 é alimentado pela tarefa replicador (T3a) e o *slot* S8 depende do processamento das tarefas T3b, T3c e da porta P2. Da mesma forma a tarefa correlacionador T5d é alimentada pelos *slot* S13 e S17, onde o *slot* S13 é alimentado pela tarefa replicador (T5a) e o *slot* S17 depende do processamento das tarefas (T5b, T5c) e da porta (P3). Assim espera-se que tenha um maior acúmulo de mensagens nos *slots* S4 e S13, uma vez que armazenam mensagens para serem correlacionadas com as mensagens dos *slots* S8 e S17 pelas tarefas T3d e T5d. A característica das tarefas T3d e T5d é correlacionar as mensagens, tornando assim os *slots* S4 e S8, prováveis gargalos de desempenho. O experimento inicialmente foi realizado com o modelo de simulação baseadas nas redes de Petri Estocásticas e após o final da simulação, observou-se um acúmulo de *tokens* nestes *slots*. O *slot* S4 acumulou cerca de 3,5 vezes mais que os demais, e o *slot* S13 cerca de 3 vezes mais.

A comparação das características esperadas com as encontradas nos experimentos, seguindo a proposta de Kleijnen [26], demonstra que o acúmulo de mensagens esperado nos *slots* se repetiu nos Lugares, o que demonstra que o modelo de simulação se comportou muito próximo de um sistema orientado a mensagem. A formação de filas mais expressiva era esperada nos *slots* que alimentam as tarefas T3d e T5d, neste caso os *slots* S4 e S13, o que se constatou quando houve acúmulo nos Lugares adjacentes das transições T3d e T5d. De fato, o acúmulo nos Lugares S4 e S13, foi em média maior que os demais, comportamento que era esperado para estes *slots*. A comparação entre o comportamento esperado pelos especialistas da área e o experimento desenvolvido deixa evidente que há semelhança entre comportamentos.

A técnica de verificação proposta por Sargent [40] visa descrever o fluxo de mensagens na perspectiva de verificar se o modelo está implementado corretamente, utilizou-se os dados iniciais sendo uma entrada de 10000 disparos. O modelo de simulação foi avaliado em setores e colocados contadores de disparos em determinados m determinados pontos do modelo representados com linha pontilhada *slots*, para dois experimentos, a Figura §4.6,

representa as redes de Petri Estocásticas (SPN) e a Figura 4.7, representa as redes de Petri Estocásticas Generalizadas (GSPN). Este experimento foi executado uma vez, e sua intenção é descrever o fluxo de mensagem, com intuito de verificar se o modelo está implementado corretamente na ferramenta.

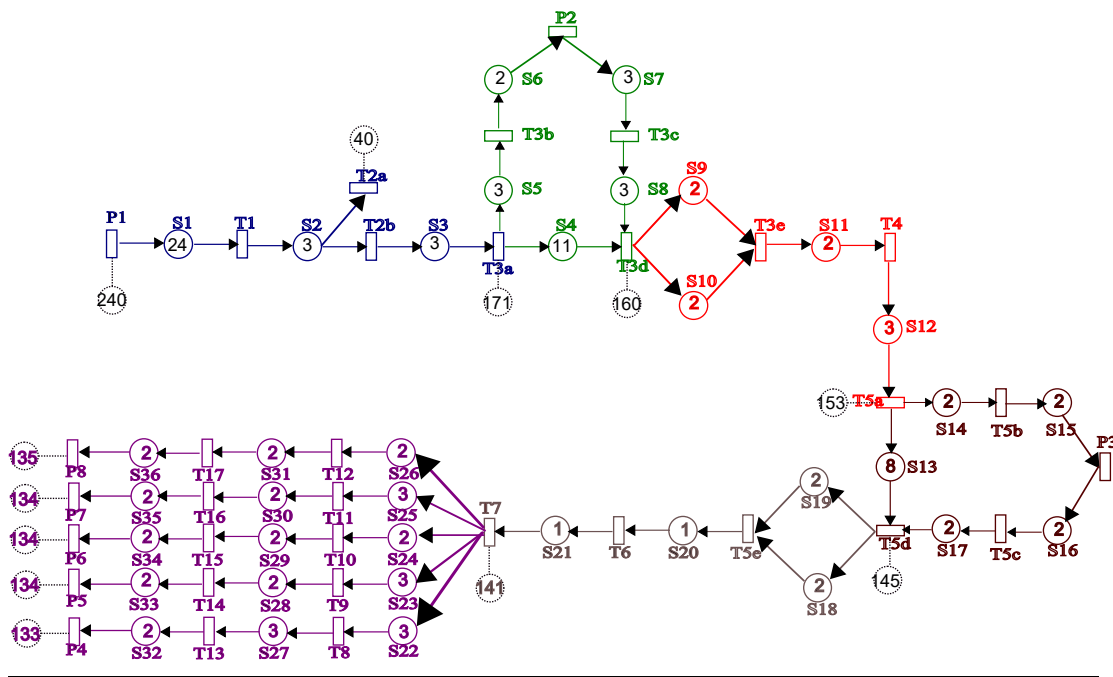


Figura 4.6: Diagrama de Aplicação da Técnica de Sargent na SPN

O modelo foi dividido por setores coloridos e a contagem de mensagens partirá das portas de saídas em direção a porta P1. Os contadores de disparos estão representados com linhas pontilhadas na cor preta e no interior o número de disparos que realizou-se, como mostras as Figuras 4.6 e 4.7. Os lugares apresentam em seu interior o número de mensagens que ficaram acumuladas no final do experimento.

A Tabela 4.4 e a Tabela 4.5, foram organizadas para uma melhor compreensão da contagem das mensagens, a representação do número de disparos de uma Transição (T_n) é determinada por $D(T_n)$. A formação de filas nos Lugares (S_n) é determinada por $F(T_n)$.

As referidas Tabelas determinam o setor de aplicação da técnica de acordo com sua cor indicado na coluna 1, a coluna 2 compara o valor do número de disparos da transição e a soma das filas e disparos das transições dos setores e

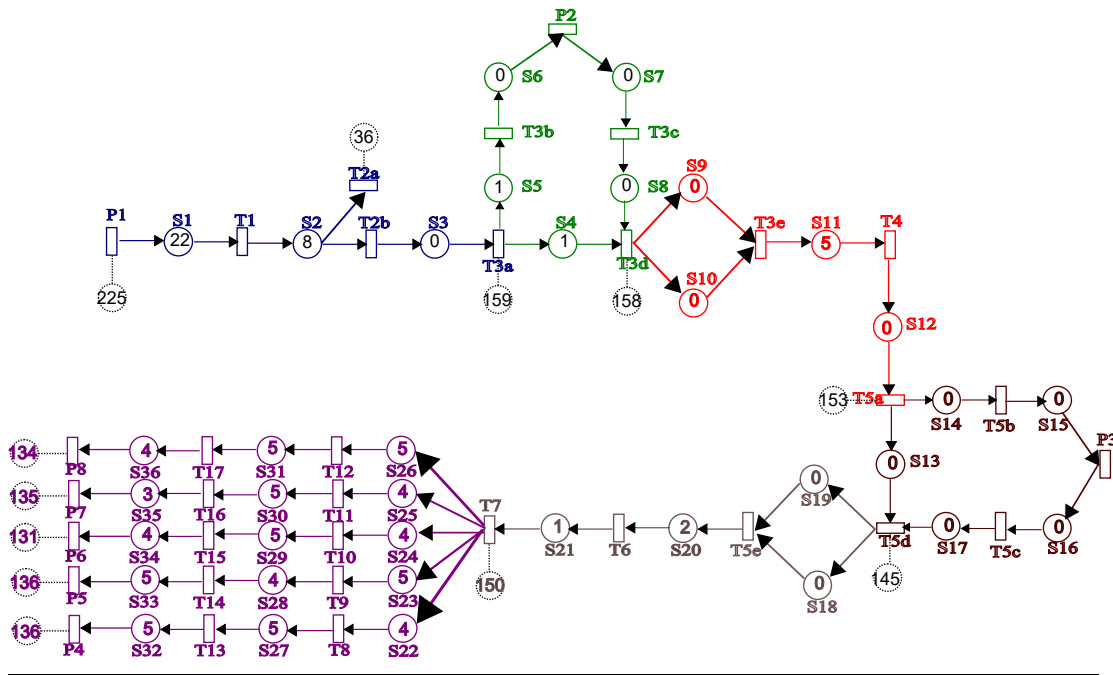


Figura 4.7: Diagrama de Aplicação da Técnica de Sargent na GSPN

a coluna 3 mostra o *checklist* que indica se o modelo foi implementado corretamente naquele setor, de acordo com a aplicação da técnica. Analisando os resultados, verifica-se que todos os setores satisfazem a premissa de partida da técnica $Entrada = Filas + Saída$. A verificação parte do número de disparos da transição que antecipa (entrada) o setor com valor de disparos dispostos nos marcadores de disparos e é igualada com o valor da transição de saída do setor (ou do modelo) somada as filas formadas no setor, havendo igualdade de valores o setor está verificado e não apresenta erros e é adicionado à coluna 3 um *checklist* positivo. Assim conclui-se que de acordo com a técnica de Sargent [40] que os modelos de simulação foram implementados de forma correta e não apresentaram erros em nenhum setor.

A aplicação da técnica de validade de eventos proposta por Sargent [40] implica na ocorrência de disparos das transições que representam a tarefa filtro (T2) no modelo de simulação perante os experimentos apresentados. A técnica avalia se a transição disparou conforme a sua taxa de disparo. A calibragem das transições que representam a funcionalidade de remover mensagens do fluxo foi feita de acordo com a estimativa de especialistas. A análise está descrita na Tabela §4.6 que demonstra a perspectiva de funcio-

Setor	Contagem	Resultado
Azul	$D(P1) = F(S1) + F(S2) + D(T2a) + F(S3) + D(T3a)$ $240 = 24 + 3 + 40 + 3 + 171 = 240$	ok
Verde	$D(T3a) = F(S4) + D(T3d) = F(S5) + F(S6) + F(S7) + F(S8) + D(T3d)$ $171 = 11 + 160 = 3 + 2 + 3 + 3 + 160 = 171$	ok
Vermelho	$D(T3d) = F(S9) + F(S10) + F(S11) + F(S12) + D(T5a)$ $160 = 2 + 2 + 2 + 3 + 153 = 160$	ok
Marron	$D(T5a) = F(S13) + D(T5d) = F(S14) + F(S15) + F(S16) + F(S17) + D(T5d)$ $153 = 8 + 145 = 2 + 2 + 2 + 2 + 145 = 153$	ok
Cinza	$D(T5d) = F(S18) + F(S19) + F(S20) + F(S21) + D(T6)$ $145 = 2 + 2 + 1 + 1 + 141 = 145$	ok
Roxo	$D(T6) = F(S22) + F(S27) + F(S32) + D(P4)$ $141 = 3 + 3 + 2 + 133 = 141$ $D(T6) = F(S23) + F(S28) + F(S33) + D(P5)$ $141 = 3 + 2 + 2 + 134 = 141$ $D(T6) = F(S24) + F(S29) + F(S34) + D(P6)$ $141 = 2 + 2 + 2 + 134 = 141$ $D(T6) = F(S25) + F(S30) + F(S35) + D(P7)$ $141 = 3 + 2 + 2 + 134 = 141$ $D(T6) = F(S26) + F(S31) + F(S36) + D(P8)$ $141 = 2 + 2 + 2 + 135 = 141$	ok

Tabela 4.4: Aplicação da Técnica de Verificação na SPN.[40]

namento da tarefa filtro do ponto de vista da Tecnologia Guaraná e do modelo de simulação em redes de Petri para os dois experimentos.

A coluna 1 identifica a tarefa filtro, a coluna 2 a taxa de filtragem estimada para cada uma delas, a coluna 3 a representação da tarefa filtro em redes de Petri, sendo as transições T2a e T2b representantes da função filtrar mensagem, a coluna 4 mostra a taxa de disparo estipulada para a transição, na coluna 5 está expresso o valor total médio de disparos ocorridos durante o experimento, o número total médio de mensagens processadas pelo Filtro é conhecido somando-se o número de disparos da tarefa com função filtrar com os disparos da tarefa não filtrar, deste total obtêm-se por uma regra de três simples a taxa de disparo em porcentagem de mensagens filtradas no experimento, descritas na coluna 6.

Portanto ao comparar o número de mensagens filtradas esperado com o obtido no experimento, pode-se constatar que seus valores são muito próximos, tratando-se de processos estocásticos, o que leva a concluir que há uma

Setor	Contagem	Resultado
Azul	$D(P1) = F(S1) + F(S2) + D(T2a) + F(S3) + D(T3a)$ $225 = 22 + 8 + 36 + 0 + 159 = 225$	ok
Verde	$D(T3a) = F(S4) + D(T3d) = F(S5) + F(S6) + F(S7) + F(S8) + D(T3d)$ $159 = 1 + 158 = 1 + 0 + 0 + 0 + 0 = 159$	ok
Vermelho	$D(T3d) = F(S9) + F(S10) + F(S11) + F(S12) + D(T5a)$ $158 = 0 + 0 + 5 + 0 + 153 = 158$	ok
Marron	$D(T5a) = F(S13) + D(T5d) = F(S14) + F(S15) + F(S16) + F(S17) + D(T5d)$ $153 = 0 + 153 = 0 + 0 + 0 + 0 + 153 = 153$	ok
Cinza	$D(T5d) = F(S18) + F(S19) + F(S20) + F(S21) + D(T6)$ $153 = 0 + 0 + 2 + 1 + 150 = 153$	ok
Roxo	$D(T6) = F(S22) + F(S27) + F(S32) + D(P4)$ $150 = 4 + 5 + 5 + 136 = 150$ $D(T6) = F(S23) + F(S28) + F(S33) + D(P5)$ $150 = 5 + 4 + 5 + 136 = 150$ $D(T6) = F(S24) + F(S29) + F(S34) + D(P6)$ $150 = 5 + 6 + 6 + 133 = 150$ $D(T6) = F(S25) + F(S30) + F(S35) + D(P7)$ $150 = 5 + 5 + 5 + 135 = 150$ $D(T6) = F(S26) + F(S31) + F(S36) + D(P8)$ $150 = 5 + 5 + 6 + 134 = 150$	ok

Tabela 4.5: Aplicação da Técnica de Verificação na GSPN.[40]

equivalência significativa nos eventos esperados no modelo conceitual e no modelo de simulação.

Modelo Conceitual Tecnologia Guaraná		Modelo de Simulação Redes de Petri Estocásticas			
Tarefa	Taxa de Filtragem	Transição	Taxa de Disparo	Média de Disparos	Porcentagem Média Efetiva de Mensagens Filtradas
T2	0,2	T2a	0,2	40,097	0,187863
		T2b	1	173,340	

Tabela 4.6: Aplicação da Técnica de Validade de Eventos.[40]

4.4 Resumo do Capítulo

Nesse capítulo, foi realizada a simulação do modelo conceitual, através da plataforma de simulação PIPE2, utilizando uma carga de entrada de 10.000 disparos, sendo feitas 25 repetições para as redes de Petri Estocásticas e Estocásticas Generalizadas.

Foram identificadas e observadas algumas variáveis para verificar o comportamento dos *tokens* no sistema, com a finalidade de encontrar ou justificar os possíveis gargalos de desempenho.

Com o auxílio dos dados disponibilizados pela plataforma de simulação, foi possível fazer uma comparação entre as médias de *tokens* e o intervalo de confiança nos lugares, e qual o impacto que uma transição temporizada ou estocástica realiza no sistema quando substituído por uma transição imediata.

Por fim, o capítulo apresenta as técnicas de verificação do modelo de simulação, a fim de dar veracidade no que está sendo pesquisado e proposto.

Capítulo 5

Conclusões e Trabalhos Futuros

*O cientista não é o homem que fornece as verdadeiras respostas;
é quem faz as verdadeiras perguntas.*

Claude Lévi-Strauss



aquisição de novas aplicações por parte das empresas visa apoiar as tomadas de decisões e o aperfeiçoamento dos processos de negócios. O conjunto dessas aplicações compõe um ecossistema de software da empresa e geralmente são heterogêneas. A solução de integração faz um processo de sincronização dos dados e funcionalidades entre as aplicações, ou produz novas funcionalidades a partir das existentes.

A modelagem matemática e simulação das soluções de integração auxiliam na análise e interpretação do comportamento do sistema, quando submetidos a determinados cenários.

Esta pesquisa partiu de um modelo conceitual de solução de integração, baseadas na tecnologia Guaraná, o qual oferece um conjunto de ferramentas e tecnologias para modelagem, implementação e execução das soluções de integração. Este modelo foi traduzido para um modelo de simulação em redes de Petri para possibilitar a análise do seu comportamento no que refere-se a gargalos de desempenho.

Com a simulação identificou-se os possíveis gargalos de desempenho, ou seja, os locais onde ocorrem o maior acúmulo de mensagens no sistema, tudo isso na fase de projeto da solução de integração. Então, buscou-se com as

redes de Petri Estocásticas Generalizadas reduzir os possíveis gargalos. As simulações foram submetidas a técnicas de verificação, a qual demonstrou que os resultados experimentais se aproximam dos esperados por especialistas.

Portanto, um modelo conceitual de solução de integração pode ser representado por um modelo matemático, e é possível identificar, reduzir ou eliminar os possíveis gargalos de desempenho de um sistema, ainda em sua fase de projeto.

Após a revisão da literatura, esta pesquisa centrou-se na aplicação das redes de Petri Estocásticas e redes de Petri Estocásticas Generalizadas. A sugestão para trabalhos futuros é a proposta de outros tipos de redes de simulação, para as soluções de integrações propostas pela tecnologia Guaraná, assim os resultados encontrados poderão acrescentar, até mesmo, corroborar o que foi proposto nesta pesquisa. Sugere-se também a utilização de outras técnicas de verificação a fim de consolidar os resultados encontrados.

Bibliografia

- [1] M. Ajmone Marsan, G. Conte e G. Balbo. *A class of generalized stochastic petri nets for the performance evaluation of multiprocessor systems*. *ACM Transactions on Computer Systems (TOCS)*, 2(2):93–122, 1984.
- [2] P. Bonet, C. M. Lladó, R. Puijaner e W. J. Knottenbelt. *Pipe v2. 5: A petri net tool for performance modelling*. Em *Proc. 23rd Latin American Conference on Informatics (CLEI 2007)*, 2007.
- [3] R. S. Cargnin. *Modelagem e simulação de uma solução de integração para identificação de gargalos de desempenho baseadas em formalismo matemático. uma abordagem orientada a redes de petri*. Dissertação de Mestrado, Universidade Regional do Noroeste do Estado do rio Grande do Sul, 2015.
- [4] D. Chappell. *Enterprise service bus*. "O'Reilly Media, Inc.", 2004.
- [5] G. Chiola, M. A. Marsan, G. Balbo e G. Conte. *Generalized stochastic petri nets: A definition at the net level and its implications*. *Software Engineering, IEEE Transactions on*, 19(2):89–107, 1993.
- [6] L. Chwif. *Redução de modelos de simulação de eventos discretos na sua concepção: uma abordagem causal*. Tese Doutoral, Universidade de São Paulo, 1999.
- [7] L. F. de Medeiros, A. Moser e N. dos Santos. *A simulação computacional como técnica de pesquisa na administração*. *REVISTA INTERSABERES*, 9(Espec):463–485, 2015.
- [8] J. Desel e J. Esparza. *Free choice petri nets*, volume 40. Cambridge university press, 2005.
- [9] N. J. Dingle, W. J. Knottenbelt e T. Suto. *Pipe2: a tool for the performance evaluation of generalised stochastic petri nets*. *ACM SIGMETRICS Performance Evaluation Review*, 36(4):34–39, 2009.

- [10] D. Dossot, J. D’Emic e V. Romero. *Mule in action*. Manning, 2014.
- [11] M. Fisher, J. Partner, M. Bogoevici e I. Fuld. *Spring integration in action*. Manning Publications Co., 2012.
- [12] 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.
- [13] R. Z. Frantz, R. Corchuelo, F. Roos-Frantz e S. Sawicki. *A technology proposal to realise enterprise application integration*. *International Journal of Computer Research*, 22(3):307, 2015.
- [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. *Enterprise application integration: an easy-to-maintain model-driven engineering approach*. Tese Doutoral, Universidad de Sevilla, 2012.
- [16] C. M. Grinstead e J. L. Snell. *Introduction to probability*. American Mathematical Soc., 2012.
- [17] P. J. Haas. *Stochastic petri nets: Modelling, stability, simulation*. Springer Science & Business Media, 2006.
- [18] F. S. Hillier e G. J. Lieberman. *Introdução à pesquisa operacional*. McGraw Hill, 2010.
- [19] G. Hohpe. *Your coffee shop doesn’t use two-phase commit [asynchronous messaging architecture]*. *IEEE software*, 22(2):64–66, 2005.
- [20] G. Hohpe e B. Woolf. *Enterprise integration patterns*. Em *9th Conference on Pattern Language of Programs*, páginas 1–9, 2002.
- [21] G. Hohpe e B. Woolf. *Enterprise integration patterns: Designing, building, and deploying messaging solutions*. Addison-Wesley Professional, 2004.
- [22] P. Huber, K. Jensen e R. M. Shapiro. *Hierarchies in coloured petri nets*. Em *Advances in Petri Nets 1990*, páginas 313–341. Springer, 1989.

- [23] C. Ibsen e J. Anstey. *Camel in action*. Manning Publications Co., 2010.
- [24] K. Jensen. *Coloured petri nets: basic concepts, analysis methods and practical use*, volume 1. Springer Science & Business Media, 2013.
- [25] W. D. Kelton e A. M. Law. *Simulation modeling and analysis*. McGraw Hill Boston, 2000.
- [26] J. P. Kleijnen. *Validation of models: statistical techniques and data availability*. Em *Proceedings of the 31st conference on Winter simulation: Simulation—a bridge to the future-Volume 1*, páginas 647–654. ACM, 1999.
- [27] M. J. Klein. *Formalização da linguagem guaraná DSL - uma abordagem matemática para especificação formal da sintaxe abstrata utilizando notação Z*. Dissertação de Mestrado, Universidade Regional do Noroeste do Estado do rio Grande do Sul, 2015.
- [28] A. Q. Lencina e F. Roos-Frantz. *As redes de petri como instrumento para modelagem matemática de um problema de integração do sistema de currículos da agência nacional e tecnologia de portugal*, 2016. URL: <https://www.publicacoeseventos.unijui.edu.br/index.php/salaoconhecimento/article/view/6782/5550>.
- [29] A. Q. Lencina e F. Roos-Frantz. *Modelagem e simulação de uma solução de integração baseada no formalismo matemático redes de petri*, 2016. URL: http://www.univates.br/editora-univates/media/publicacoes/203/pdf_203.pdf.
- [30] D. S. Linthicum. *Enterprise application integration*. Addison-Wesley Professional, 2000.
- [31] P. R. Maciel, R. D. Lins e P. R. Cunha. *Introdução às redes de petri e aplicações*. UNICAMP-Instituto de Computacao, 1996.
- [32] M. A. Marsan, G. Balbo, G. Conte, S. Donatelli e G. Franceschinis. *Modelling with generalized stochastic petri nets*. John Wiley & Sons, Inc., 1994.
- [33] F. Menge. *Enterprise service bus*. Em *Free and open source software conference*, volume 2, páginas 1–6, 2007.
- [34] M. K. Molloy. *Performance analysis using stochastic petri nets*. *Computers, IEEE Transactions on*, 100(9):913–917, 1982.

- [35] C. Petri. *Kommunikation mit automaten. schriften des iim nr. 2, institut fur instrumentelle mathematic*. Relatório técnico, English translation: Technical Report RADCTR-65-377, Griffiths Air Base, New York, 1966.
- [36] A. A. B. Pritsker. *Introduction to stimulation and slam ii*. 1986.
- [37] C. A. L. Reis. *Uma abordagem flexível para execução de processos de software evolutivos*. 2003.
- [38] F. Roos-Frantz, M. Binelo, R. Z. Frantz, S. Sawicki e V. B. Fernandes. *Using petri nets to enable the simulation of application integration solutions conceptual models*. Em *ICEIS (1)*, páginas 87–96, 2015.
- [39] N. Sakurada e D. I. Miyake. *Aplicação de simuladores de eventos discretos no processo de modelagem de sistemas de operações de serviços*. *Gestão & Produção*, 16(1):25–43, 2009.
- [40] R. G. Sargent. *Verification and validation of simulation models*. Em *Proceedings of the 37th conference on Winter simulation*, páginas 130–143. winter simulation conference, 2005.
- [41] 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*. IGI Global, 2015.
- [42] F. R. Sequeira, R. Z. Frantz, I. Yevseyeva, M. T. Emmerich e V. Basto-Fernandes. *An eai based integration solution for science and research outcomes information management*. *Procedia Computer Science*, 64: 894–901, 2015.
- [43] L. C. Silva. *Simulação de processos*. Universidade Federal do Espírito Santo <http://www.agais.com/simula.htm>, 2, 2008.
- [44] J. Wang. *Timed petri nets: Theory and application*, volume 9. Springer Science & Business Media, 2012.

This document was typeset on June 1, 2017 using class `RGBOK` $\alpha 2.14$ for `LATEX2 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.