

UNIVERSIDADE DO VALE DO RIO DOS SINOS - UNISINOS
UNIDADE ACADÊMICA DE PESQUISA E PÓS-GRADUAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO E SISTEMAS
NÍVEL MESTRADO

Diego Augusto Steffen

DESENVOLVIMENTO DE INTERFACES AUTOMATIZÁVEIS PARA CONSTRUÇÃO E VERIFICAÇÃO DE MODELOS DE SIMULAÇÃO COMPUTACIONAL: UM PROTÓTIPO PARA UM LABORATÓRIO DE ANÁLISES QUÍMICAS

São Leopoldo
2009

Diego Augusto Steffen

DESENVOLVIMENTO DE INTERFACES AUTOMATIZÁVEIS PARA CONSTRUÇÃO E VERIFICAÇÃO DE MODELOS DE SIMULAÇÃO COMPUTACIONAL: UM PROTÓTIPO PARA UM LABORATÓRIO DE ANÁLISES QUÍMICAS

Dissertação apresentada como requisito parcial para a obtenção do título de Mestre, pelo Programa de Pós-Graduação em Engenharia de Produção e Sistemas da Universidade do Vale do Rio dos Sinos.

Orientador: Luis Henrique Rodrigues

São Leopoldo
2009

Diego Augusto Steffen

DESENVOLVIMENTO DE INTERFACES AUTOMATIZÁVEIS PARA CONSTRUÇÃO E VERIFICAÇÃO DE MODELOS DE SIMULAÇÃO COMPUTACIONAL: UM PROTÓTIPO PARA UM LABORATÓRIO DE ANÁLISES QUÍMICAS

Dissertação apresentada como requisito parcial para a obtenção do título de Mestre, pelo Programa de Pós-Graduação em Engenharia de Produção e Sistemas da Universidade do Vale do Rio dos Sinos.

Aprovado em ____/____/____

BANCA EXAMINADORA

Gustavo Severo Borba – Universidade do Vale do Rio dos Sinos

Guilherme Luís Roehe Vaccaro – Universidade do Vale do Rio dos Sinos

Ricardo Augusto Cassel – Universidade do Vale do Rio dos Sinos

S817d

Steffen, Diego Augusto.

Desenvolvimento de interfaces automatizáveis para construção e verificação de modelos de simulação computacional : um protótipo para um laboratório de análises químicas / Diego Augusto Steffen. – 2009.

136 f. : il. ; 30 cm.

Dissertação (mestrado) – Universidade do Vale do Rio dos Sinos, Programa de Pós-Graduação em Engenharia de Produção e Sistemas, 2009.

"Orientador: Luis Henrique Rodrigues."

1. Simulação (Computadores). 2. Construção automática de modelos. 3. Engenharia de produção. I. Título.

CDU 658.5

Dados Internacionais de Catalogação na Publicação (CIP)
(Bibliotecário: Flávio Nunes – CRB 10/1298)

*Dedico este trabalho a todos aqueles
que contribuíram para sua realização*

Agradecimentos

Para a realização desta pesquisa pude contar com o apoio e ajuda de algumas pessoas. Meus sinceros agradecimentos:

À minha família, incentivadora pela busca de conhecimento e cultura, meu porto seguro;

À minha namorada, companheira e amiga, presente nos momentos mais importantes da minha vida;

À família Zorzo pelo apoio necessário para que este momento acontecesse;

Ao professor Luis Henrique Rodrigues, orientador deste trabalho, pelos seus conhecimentos, sua atenção, sua vontade e principalmente, pela sua amizade;

Ao Eduardo, Jonatas e Menezes pela amizade e companhia nos momentos importantes da pesquisa;

À Alexandre Costa, pelo companheirismo nos projetos desenvolvidos;

Aos professores do Programa de Pós-Graduação em Engenharia de Produção e Sistemas pelo conhecimento e amizade;

Aos colegas de mestrado, pelo incentivo, brincadeiras e estudos ao longo desta trajetória;

Ao Guilherme Vaccaro e Ricardo Cassel, pelos ensinamentos em Simulação Computacional;

À Proffittare pela minha formação e pelas oportunidades profissionais;

À Accera Supply Chain Solutions por sempre me incentivar e aos colegas de trabalho pela paixão por vencer;

À UNISINOS pelo auxílio concedido.

*"Uma paixão forte por qualquer objeto
assegurar  o sucesso, porque o desejo
pelo objetivo mostrar  os meios"*

William Hazlitt

Resumo

A presente pesquisa é um estudo em simulação computacional desenvolvido em um laboratório de análises químicas onde a geração do modelo de simulação é construído de forma automática, por meio de interfaces automatizáveis. O objetivo do trabalho foi desenvolver uma abordagem de geração automática de modelos de simulação computacional para projetos repetitivos. Foram desenvolvidos templates e um gerador de modelos de simulação que foram aplicados em um projeto de simulação computacional em um laboratório de análises químicas. Ao término do projeto, puderam-se observar diversos benefícios que o uso dos templates proporciona, entre os quais autonomia para o usuário final, rapidez para a construção de novos modelos, geração automática do código do modelo de simulação, rapidez na atualização de um modelo previamente utilizado.

Palavras-chave: Simulação Computacional; Construção automática de modelos.

Abstract

This research is a study in a computer simulation developed in a laboratory for chemical analysis where the generation of the simulation model is built automatically through parametrizable interfaces. The objective was to develop an approach to automatic generation of computer simulation models for repetitive projects. Templates were developed and a generator of simulation models that were applied in a project of computer simulation in a laboratory for chemical analysis. At the end of the project, could be observed several benefits that provide the use of templates, including autonomy for the end user, faster to build new models, automatic code generation of the simulation model, speed of update of a model previously used.

Keywords: Computational simulation; Construction automatic models.

Lista de Figuras

Figura 1 - Condução de projetos (VACCARO, 1999)	18
Figura 2 - Trade-off e evolução dos programas de simulação (RODRIGUES, 1994)	41
Figura 3 - Malha de Atividades do Micro Saint (Retirado do Software Micro Saint)	43
Figura 4 - Lista de Variáveis do Micro Saint (Retirado do Software Micro Saint).....	43
Figura 5 - Arquitetura de um ambiente de modelagem (WAGNER, et al., 1996).....	47
Figura 6 - Estrutura do Modelo Orientado a Objeto (TRAVASSOS, 2002 p. 33).....	49
Figura 7 - Simulação em Arena (LOVEJOY, 2001)	51
Figura 8 - Exemplo do OMNet++ (OMNET++, 2009)	54
Figura 9 - Diagrama Modular do ASDA (BRUSCHI, 2003 p. 70).....	58
Figura 10 - Fluxo de informação do sistema proposto (PAIVA, 2005 p. 111).....	64
Figura 11 - Esquema do Método Hipotético-Dedutivo. Adaptado (LAKATOS e MARCONI, 1990)	70
Figura 12 - Exemplo do fluxo das amostras no laboratório (Elaborada pelo Autor)	75
Figura 13 - Exemplo de um teste (Elaborada pelo Autor)	76
Figura 14 - Exemplo de uma bancada	77
Figura 15 - Escopo do 1º estudo de simulação (Elaborada pelo Autor).....	79
Figura 16 - Metodologia utilizada no 2º projeto (Elaborada pelo Autor).....	81
Figura 17 - Exemplo de código gerado a partir da demanda (Elaborada pelo Autor).....	83
Figura 18 - Templates desenvolvidos (Elaborado pelo Autor)	88
Figura 19 - Template Eventos (Elaborada pelo Autor).....	89
Figura 20 - Template Atividade (Elaborada pelo Autor)	90
Figura 21 - Template Variáveis (Elaborada pelo Autor).....	91

Figura 22 - Template Produtos (Elaborada pelo Autor)	92
Figura 23 - Template Amostras (Elaborada pelo Autor)	93
Figura 24 - Template Recursos Técnicos (Elaborada pelo Autor).....	94
Figura 25 - Template Bancadas (Elaborada pelo Autor)	95
Figura 26 - Template Finalidade (Elaborada pelo Autor).....	
Figura 27 - Template Configuração dos Turnos (Elaborada pelo Autor).....	97
Figura 28 - Template Configurações Gerais (Elaborada pelo Autor).....	98
Figura 29 - Template Painel de Acompanhamento (Elaborada pelo Autor)	99
Figura 30 - Condução da Criação do Gerador Automático de Modelos (Elaborada pelo Autor)	101
Figura 31 - Modelo de Simulação Vazio (Retirado do Software Micro Saint)	102
Figura 32 - Ambiente de Desenvolvimento NetBeans (Retirado do Software NetBeans)	104
Figura 33 - Gerador Automático de Modelos - Salvar novo modelo (Retirado do Software Gerador Automático de Modelos).....	105
Figura 34 - Gerador Automático de Modelos - Abrir Templates (Retirado do Software Gerador Automático de Modelos).....	105
Figura 35 - Modelo Computacional Construído a partir dos Templates (Retirado do Software Micro Saint)	106
Figura 36 - Curva de desenvolvimento (Elaborada pelo Autor).....	112
Figura 37 - Inserir atividades no modelo (Retirado do Software Micro Saint)	115
Figura 38 - Inserção das regras na atividade (Retirado do Software Micro Saint).....	116
Figura 39 - Conectar teste nas prioridades (Retirado do Software Micro Saint)	117
Figura 40 - Inserção de novo teste com templates (Elaborada pelo Autor).....	117
Figura 41 - Exemplo da inserção de uma nova bancada (Retirado do Software Micro Saint)	118
Figura 42 - Regras na inserção de uma nova bancada (Retirado do Software Micro Saint)..	119
Figura 43 - Inserção de nova bancada utilizando templates (Elaborada pelo Autor).....	120
Figura 44 - Configuração do número de técnicos (Retirado do Software Micro Saint).....	121
Figura 45 - Alteração dos técnicos com uso dos templates (Elaborada pelo Autor).....	121
Figura 46 - Fluxo de construção das regras de modelagem (Elaborada pelo Autor)	124

Lista de Tabelas

Tabela 1 - Termos procurados na EBSCO.....	20
Tabela 2 - Termos procurados na Biblioteca Digital Brasileira de Teses e Dissertações.....	20
Tabela 3 - Termos procurados na Biblioteca Eletrônica Scielo.....	21
Tabela 4 - Layout dos dados de demanda.....	83
Tabela 5 - Avaliação do usuário	112

Sumário

1 INTRODUÇÃO	16
1.1 Justificativa.....	19
1.2 Problema de Pesquisa	24
1.3 Objetivo	25
1.3.1 Objetivo Geral.....	26
1.3.2 Objetivos Específicos.....	26
1.4 Delimitações e Pressupostos	27
1.5 Estrutura do Trabalho	29
2 REFERENCIAL TEÓRICO.....	31
2.1 Modelagem	31
2.2 Simulação	35
2.2.1 Definição de Simulação.....	36
2.2.2 Método de Condução de Projetos de Simulação.....	38
2.2.3 Ferramentas de Modelagem	40
2.2.4 Micro Saint.....	42
2.3 Visual Interactive Simulation e Visual Interactive Modeling.....	44
2.3.1 Ambientes de Simulação Orientados a Objetos	48

	14
2.3.1.1 Arena.....	51
2.3.1.2 OMNet++.....	52
2.3.1.3 ASiA.....	54
2.4 Ambientes de Simulação Automáticos	55
2.4.1 ASDA – Um Ambiente de Simulação Distribuída Automático.....	56
2.4.2 Geração Automática de Modelos de Simulação na Indústria Têxtil.....	62
2.4.3 Comparativo entre as abordagens apresentadas.....	66
3 MÉTODO.....	68
3.1 Método de Pesquisa	69
3.2 Método de Trabalho.....	72
4 ESTUDO DE CASO	74
4.1 O Contexto do Caso.....	74
4.2 As Aplicações Anteriores ao uso dos Templates.....	78
4.2.1 Primeiro Projeto.....	78
4.2.2 Segundo Projeto.....	80
4.3 A Aplicação com o uso dos Templates.....	85
4.3.1 Descrição do Projeto.....	85
4.3.2 Os Templates Desenvolvidos.....	87
4.3.3 O Gerador Automático de Modelos.....	100
5 AVALIAÇÃO DA ABORDAGEM DESENVOLVIDA.....	108
5.1 Avaliação do Uso de Templates	108
5.1.1 Avaliação da Pesquisa Realizada	108
5.1.2 Avaliação do Estudo Realizado no Laboratório	114
5.2 Pontos Críticos e Dificuldades Encontradas	123
6 CONCLUSÕES FINAIS E TRABALHOS FUTUROS.....	126

6.1 Conclusões Finais	126
6.2 Trabalhos Futuros	128
7 REFERÊNCIAS BIBLIOGRÁFICAS.....	130

1 INTRODUÇÃO

Um modelo é a uma criação cultural que se destina a representar uma realidade, ou alguns de seus aspectos, a fim de torná-los descritíveis qualitativa e quantitativamente e está presente no cotidiano das pessoas e no ambiente empresarial (SAYÃO, 2001) (TRIVELATO, 2003).

Os modelos buscam a formalização do universo através de meios de expressões controláveis pelo ser humano, que necessita entender a realidade complexa da realidade que está inserido (SAYÃO, 2001).

Na busca de novos esclarecimentos e conhecimentos, de novos fenômenos e eventos, o ser humano não os identifica somente pelas sensações ou pelas manifestações imediatas, mas recorre à reflexão e ao conhecimento acumulado, através da formulação de hipóteses e da estruturação de modelos(SAYÃO, 2001 p. 82).

Cada modelo expressa e justifica um método de abordagem de uma realidade física, ao mesmo tempo em que cada método subentende um modelo, nem que seja um modelo meramente operacional (SAYÃO, 2001).

Para o mundo dos negócios, a utilização de modelos tem grande valia em sistemas de informação, que auxiliam na tomada de decisões. Neste contexto está inserida a simulação computacional. Modelos de simulação são usados para avaliar eventos do mundo real e os sistemas existentes, sob diferentes condições operacionais (NAGABHUSHANA, 2006).

A simulação computacional é uma ferramenta de modelagem e análise de sistemas que possibilita reproduzir o mundo real em um ambiente virtual. Esta possibilidade de reproduzir de forma dinâmica o comportamento do mundo real pode ser utilizada nos mais diversos ambientes, para um grande número de aplicações, principalmente quando se quer testar hipóteses do tipo "o que acontece se...", ou seja, testar diversas hipóteses que poderão acontecer no sistema real, se forem tomadas certas decisões. Dentre as aplicações pode-se citar:

- Área de manufatura, onde a simulação pode auxiliar na redução do tempo de implantação de projetos, evitando rupturas na cadeia produtiva, identificando gargalos, simulando cenários de quebra de máquinas ou aumento de produção;
- Área de prestação de serviços com aplicação em lojas, bancos, hospitais e clínicas visando melhorar o atendimento aos clientes e pacientes re-dimensionando áreas de espera, leitos, número de salas de cirurgias e minimizar os custos com pessoal e insumos;
- Área de logística readequando locais de estoque, visando diminuir as distâncias percorridas, otimizando rotas de distribuição.

A presente pesquisa é um terceiro estudo em simulação computacional aplicado em um laboratório de análises químicas. A pesquisa será realizada em um ambiente onde o laboratório é um setor de apoio ao processo produtivo, ou seja, é um meio para que a empresa produza alguma coisa.

A gestão de um laboratório de análises químicas possui uma elevada complexidade, devido a suas características estocásticas. Com a simulação, pode-se ter um melhor entendimento das complexas relações existentes no processo como um todo. Porém, construir modelos de simulação computacional é uma tarefa que requer habilidades específicas, deixando a cargo de pessoas que detêm o conhecimento necessário para tanto.

O desenvolvimento de projetos de simulação computacional envolve diversas etapas, que se estendem desde a compreensão do contexto em que o objeto em estudo está inserido e a formulação dos objetivos e das questões a serem respondidas, até a implementação e documentação das soluções resultantes do estudo (VACCARO, 1999).

Assim, a construção de um projeto de simulação computacional segue alguns passos que vão desde o entendimento do ambiente real que se está querendo simular, até os resultados das rodadas da simulação.

Law e Kelton (1991), Gogg e Mott (1996), entre outros autores, apresentam métodos para o desenvolvimento de projetos de simulação que de um modo geral é realizado através de um conjunto de etapas, com a seguinte natureza:

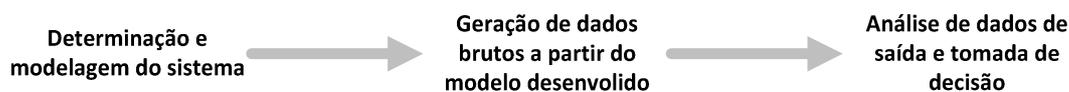


Figura 1 - Condução de projetos (VACCARO, 1999)

O presente estudo apresentará uma abordagem para reduzir o esforço da etapa que o autor descreve como “determinação e modelagem do sistema”.

Gogg e Mott (1996) enfatizam a utilização de uma regra para determinação de esforços em simulação: “40-20-40”. Os 40% do tempo significam o custo nas primeiras etapas de definição do problema e coleta dos dados; 20% é o custo da tradução do problema, validação e geração de dados brutos a partir do modelo desenvolvido; e, finalizando, os 40% do tempo devem ser utilizados na experimentação, análise de resultados e documentação.

Visto isso, este trabalho apresenta uma abordagem para construção e verificação de modelos de simulação computacional na qual é proposta a construção de

interfaces automatizáveis com o objetivo de reduzir a complexidade e tempo de construção e verificação dos modelos computacionais.

As interfaces automatizáveis serão construídas em forma de templates, para que haja um padrão nas informações que constituirão o modelo. “Padrões definem a sintaxe e semântica de um modelo de dados”(SILBERSCHATZ, et al., 1999 p. 646).

Com o uso dos templates, se buscará eliminar inconsistência e redundância dos dados, centralizar o acesso aos dados, evitando o isolamento das informações, como também reduzir problemas de integridade, possibilitando que informações até então independentes no laboratório, possam interagir (SILBERSCHATZ, et al., 1999).

1.1 Justificativa

Nesta seção serão explicados os motivos que tornam esta pesquisa relevante. Será abordada a originalidade do estudo, o tema simulação computacional e o motivo de realizar esta pesquisa.

A justificativa do ponto de vista da originalidade é evidenciada pela pesquisa ampla em bases de dados nacionais e internacionais. Para tanto, foram selecionadas base de pesquisa internacional, base de dados de dissertações e teses nacionais e base de periódicos nacionais.

Nesta pesquisa foi verificado, realizando pesquisa com palavras-chave abrangentes e específicas, que o tema simulação computacional está em discussão no meio acadêmico, por meio de trabalhos científicos e publicações.

As tabelas Tabela 1, Tabela 2 e Tabela 3 apresentam os resultados destas pesquisas em bases de dados:

Tabela 1 - Termos procurados na EBSCO

Termos procurados	Resultados
Computer Simulation	13.320
Computer models building	1
Discrete Event Dynamic Systems	21
Simulation Methods	12.130
Total	25.472

Fonte: EBSCO¹

Tabela 2 - Termos procurados na Biblioteca Digital Brasileira de Teses e Dissertações

Termos procurados	Resultados
Ambiente de Simulação Automático	6
Automatic simulation environment	10
Simulação Computacional	556
Computer Simulation	340
Ambiente de Modelagem Computacional	97
Construção de modelos computacionais	40
Simulação de Filas	16
Simulação discreta	25
Gestão automática de modelos de processos	2
Total	1.092

Fonte: Biblioteca Digital Brasileira de Teses e Dissertações²

¹ A EBSCO é uma base de dados referenciais, com serviço de fornecimento de periódicos e livros eletrônicos e assinatura de impressos, ferramenta de gerenciamento de recursos eletrônicos, base de dados em textos completos e secundários, e serviços relacionados para todos os tipos de bibliotecas, corporações e organizações de pesquisa (EBSCO PUBLISHING, 2009).

Na Biblioteca Digital Brasileira de Teses e Dissertações o termo “Simulação Computacional” retorna 556 trabalhos, entre Artigos, Dissertações e Teses. O termo “Ambiente de Modelagem Computacional” retorna 97 trabalhos e o termo “Gestão automática de modelos de processos” retorna 2 trabalhos.

Tabela 3 - Termos procurados na Biblioteca Eletrônica Scielo

Termos procurados	Resultados
Ambiente de Simulação Automático	0
Automatic simulation environment	0
Simulação Computacional	10
Computer Simulation	22
Construção de modelos computacionais	0
Simulação discreta	0
Total	32

Fonte: SciELO³

Na base de dados Scielo o termo “Simulação Computacional” retorna 10 artigos, sendo que nenhum trabalho é na área deste estudo.

Na base internacional EBSCO o termo “Computer Simulation” retorna 13.320 trabalhos. Já o termo “Computer Models Building” retorna 1 trabalho.

Os resumos e trabalhos que mais se enquadraram no tema proposto foram analisados, a fim de identificar trabalhos correlatos ou que complementam a proposta desta pesquisa.

² A Biblioteca Digital Brasileira de Teses e Dissertações (BDTD) é mantida pelo Ministério da Ciência e Tecnologia do Governo Brasileiro, disponibilizando teses e dissertações existentes no País, além de serviços integrados para a realização de pesquisas (BDTD, 2009).

³ A Scientific Electronic Library Online (SciELO) é uma biblioteca eletrônica que abrange uma coleção selecionada de periódicos científicos brasileiros.

Nota-se que as publicações são voltadas ao uso da simulação computacional como ferramenta e não no estudo de construção de modelos computacionais, ou seja, as pesquisas utilizam a simulação computacional como meio para alcançar os objetivos das pesquisas e não estudam a simulação computacional como objeto principal.

Uma pesquisa correlata foi desenvolvida por Paiva (2005) com o título de “Geração Automática de Modelos de Simulação de uma Linha de Produção na Indústria Têxtil”, onde foi desenvolvido um sistema genérico para o software Arena.

A pesquisa teve como objetivo a construção de um sistema de apoio à decisão para estudar o impacto de diversas alternativas de gestão da produção aplicadas a um sistema produtivo da indústria têxtil (PAIVA, 2005).

Outra pesquisa correlata foi desenvolvida por Bruschi (2003) com o título “ASDA – um ambiente de simulação distribuído automático”, onde propôs um ambiente automático para desenvolvimento de simulação distribuída.

A simulação computacional, graças a sua versatilidade, tomou força nas últimas três décadas. Também pelo acesso simplificado aos computadores de menor porte. A Ciência Estatística tem na simulação computacional uma das principais ferramentas (VACCARO, 1999).

O desenvolvimento e a popularização dos recursos computacionais transformaram a simulação em uma importante ferramenta para planejamento, desenvolvimento e controle de sistemas complexos. Diversos avanços foram conseguidos no sentido de criar ambientes apropriados para a modelagem e execução de experimentos de simulação (VACCARO e RODRIGUES, 2005 p. 4666).

Entretanto, construir modelos computacionais não é tarefa fácil (BRUSCHI, 2003).

Segundo Bruschi (2003 p. 2),

Desenvolver uma simulação computacional não é uma tarefa muito fácil, principalmente para usuários com pouca experiência. Necessitam-se conhecimentos em modelagem, programação, linguagem para simulação e análise dos resultados.

Assim, a justificativa de realizar esta pesquisa é de facilitar para o usuário o uso da simulação computacional, pois o uso de templates tem o propósito de substituir habilidades do usuário que são necessárias para desenvolver um modelo.

A justificativa de se realizar a pesquisa em um laboratório de análises químicas se deve ao fato de o autor já ter realizado trabalhos de simulação neste ambiente, contribuindo para uma melhor condução desta pesquisa.

A escolha por laboratórios também se deve ao fato do dinamismo do gerenciamento do mesmo. Como o laboratório é um órgão de apoio dentro do processo produtivo, ele está subordinado ao gerenciamento do processo mais amplo, tendo que se adequar rapidamente às decisões tomadas.

Dentre os benefícios que a simulação computacional pode proporcionar ao laboratório, pode-se citar uma maior acuracidade no atendimento de demandas não programadas, agilidade no atendimento das necessidades (por exemplo, vendas), melhor balanceamento dos recursos no laboratório e disponibilidade dos recursos do sistema produtivo externos ao laboratório (uma vez que estes recursos externos dependem dos testes do laboratório para serem liberados).

Assim, tornar a simulação computacional uma ferramenta de apoio à tomada de decisão, dinâmica e flexível é uma motivação para a realização desta pesquisa. Fatores como a extensão do modelo, em outras palavras, o número de atividades, variáveis, eventos, dados de entrada e de saída que devem ser modelados, tendem a tornar a simulação engessada e com vida útil limitada, pois é da natureza deste sistema.

Além da complexidade de codificar para uma linguagem de simulação, está o trabalho operacional de edição e codificação de cada uma das atividades. Erros na especificação do projeto podem acarretar alterações e inserções de novas características (variáveis, demandas, regras, etc.) durante a programação do modelo, surgindo a necessidade de editar cada uma das atividades do modelo, tornando esta etapa demorada e trabalhosa.

O conceito das interfaces configuráveis está sendo proposto com o objetivo de auxiliar o usuário no desenvolvimento computacional e manutenção da simulação, dando maior autonomia ao usuário, assim como realizar a verificação dos dados.

1.2 Problema de Pesquisa

Conhecer a realidade, entender a dinâmica e lógica do sistema, definir uma estratégia de modelagem são os primeiros passos em um projeto de simulação computacional. Para cada ambiente a ser simulado tem características que nem sempre são reproduzíveis em outro ambiente. Por um lado, simular ambientes diferentes requer estratégias diferentes, mas, por outro lado, simulações em ambientes semelhantes podem facilitar o entendimento e construção do modelo de simulação.

Este estudo será realizado em um ambiente onde o autor já realizou outros dois projetos de simulação. Os projetos de simulação, em sua primeira construção, são custosos (no sentido de esforço) porque, como comentado, deve-se conhecer a realidade, entender a dinâmica e lógica do sistema, definir uma estratégia de modelagem, etc. Entretanto, quando o projeto de simulação é realizado em um ambiente onde já foram desenvolvidos outros projetos, pode-se aproveitar não só o conheci-

mento adquirido das modelagens anteriores, como também modelos computacionais desenvolvidos.

Neste contexto, a vida útil de um modelo de simulação está relacionada às mudanças do ambiente real. Se o ambiente representado na simulação muda constantemente, o modelo deve suportar estas mudanças. Caso contrário, sua serventia se limita a representar o ambiente naquele espaço de tempo em que foi estudado. Esta idéia de o modelo suportar mudanças vale também quando uma reedição do projeto é realizada no mesmo ambiente (projetos de simulação computacional repetitivos).

Assim, esta pesquisa tem como propósito responder à seguinte questão:

Como desenvolver um gerador automático de modelos de simulação, visando dar agilidade na etapa de modelagem de projetos de simulação computacional repetitivos?

Para responder a esta pergunta, uma hipótese irá nortear o estudo. A hipótese é que a automatização da interface reduz o tempo de construção de modelos e verificação dos dados, na simulação computacional para futuras utilizações do mesmo.

Para responder a esta questão, é necessário realizar o objetivo, que será apresentado no capítulo 1.3.

1.3 Objetivo

Para esta pesquisa, os seguintes objetivos foram traçados:

1.3.1 Objetivo Geral

O objetivo deste trabalho é o desenvolvimento de uma abordagem de geração automática de modelos de simulação computacional para projetos repetitivos.

Além disso, foram traçados três objetivos específicos, apresentados na sequência.

1.3.2 Objetivos Específicos

Para a realização deste objetivo, alguns objetivos específicos serão abordados:

- Desenvolver uma simulação computacional em um laboratório de análises químicas, utilizando as interfaces previamente construídas;
- Identificar, por meio de uma aplicação prática, os benefícios da aplicação de interfaces automatizáveis na construção de um modelo de simulação computacional;
- Avaliar qualitativamente a abordagem proposta, aplicando um questionário com os envolvidos no projeto, buscando identificar os potenciais benefícios da abordagem proposta.

Para tanto, a pesquisa possui delimitações e pressupostos, apresentadas na seção 1.4.

1.4 Delimitações e Pressupostos

Diferentemente da proposta apresentada por Paiva (2005), a pesquisa será para projetos de simulação repetitivos, (melhor detalhado no capítulo 1.2), para laboratórios de análises químicas, ou seja, recomenda-se a abordagem construída para projetos de simulação computacional que terão mais de uma versão. A linguagem de programação a ser utilizada será do Micro Saint e a ferramenta, além de construir o modelo computacional, irá realizar a verificação do padrão dos dados que alimentarão a simulação computacional.

Em relação à proposta apresentada por Bruschi (2003), a proposta deste estudo não é de criar um software de simulação computacional, mas uma interface para criação de modelos computacionais no software Micro Saint.

A pesquisa foi desenvolvida utilizando o software de simulação Micro Saint, porque é um software de baixo custo comercial, com linguagem de programação de domínio do pesquisador e disponível na Universidade na versão acadêmica.

Outro fator para o desenvolvimento desta pesquisa foi a oportunidade da realização do terceiro projeto de simulação computacional em um laboratório de análises químicas. O laboratório, onde foram desenvolvidos os projetos anteriores de simulação, requisitou um novo projeto de simulação computacional, permitindo assim a execução desse trabalho.

São os seguintes as principais delimitações do trabalho:

- As interfaces foram desenvolvidas para laboratórios de análises químicas, pelo fato de o pesquisador ter acesso às informações.

- A avaliação da abordagem proposta será qualitativa, por não haver documentação sobre os tempos de execução de cada uma das etapas do método de condução dos projetos de simulação anteriores
- É intenção da pesquisa verificar a utilização das interfaces automatizáveis na construção e verificação do modelo de simulação computacional, não contemplando uma análise nas demais etapas da construção da simulação como também análises dos resultados obtidos com a mesma;
- O trabalho não pretende exaurir as questões de modelagem e simulação em laboratórios de análises químicas, incentivando outros pesquisadores a ampliarem os conceitos discutidos durante a pesquisa, introduzindo questões novas e ampliando a aderência à realidade.
- Vale ressaltar que em nenhum momento está se colocando as interfaces como única alternativa para construção e verificação de modelos de simulação computacional. O intuito da pesquisa é mostrar que as interfaces se configuram como uma importante ferramenta de construção de modelos, porém não se pode relegar outros fatores, como a expertise do profissional adquirida ao longo de sua carreira, por exemplo;
- A ferramenta que codificará os templates em linguagem de simulação será desenvolvida em linguagem Java, que é uma linguagem orientada a objetos, desenvolvida pela Sun Microsystems;
- As planilhas eletrônicas serão criadas utilizando Microsoft Excel, com possibilidade de serem editadas por qualquer editor de texto;
- O software de simulação utilizado nesta pesquisa é o Micro Saint. O Micro Saint é um simulador genérico, o qual roda nos ambientes Windows, Mac e Sun-workstations. A versão utilizada foi a Micro Saint Release 3.2 Build J whit ActionView and OptQuest, Standart Version de março de 2001.

1.5 Estrutura do Trabalho

No capítulo 1 procurou-se comentar as razões pelas quais se decidiu fazer um estudo sobre geração automática de modelos de simulação computacional. Também foram explicados os motivos do uso da abordagem proposta, bem como o objetivo desta pesquisa. Na introdução foram expostas as delimitações do trabalho bem como a sua estrutura.

No capítulo 2 são abordados os temas relevantes para a pesquisa. Estão descritos o conceito e definições da modelagem, conceito e definições da simulação computacional, trabalhos correlatos e softwares de simulação que utilizam abordagem da pesquisa.

No capítulo 3 é apresentado o método de pesquisa, a metodologia utilizada na construção do trabalho, assim como o detalhamento do método de trabalho.

No capítulo 4 é detalhada a abordagem proposta para construção de modelos de simulação computacional em laboratórios de análises químicas, é descrito o ambiente que será tratado na simulação computacional assim como as regras existentes no sistema real.

Neste capítulo será apresentado também o protótipo desenvolvido para este trabalho. Serão detalhadas todas as funções, relacionamentos e interações do usuário com o software, bem como apresentação da sua usabilidade.

No capítulo 5 é realizada a avaliação da abordagem proposta, levantando os benefícios que a abordagem trouxe para a construção de modelos de simulação computacional. Esta avaliação será realizada por meio de entrevistas com os envolvidos no processo de construção e validação da simulação computacional.

O capítulo 6 é um resumo das conclusões obtidas durante as etapas anteriores do trabalho. Neste capítulo é respondida a questão norteadora do estudo, bem como o objetivo proposto.

Ainda neste capítulo serão apresentadas sugestões de trabalhos futuros a serem realizados.

2 REFERENCIAL TEÓRICO

Este capítulo contém o referencial teórico construído ao longo da pesquisa.

Nas ciências sociais a teoria não é um luxo para o pesquisador, é muito mais uma necessidade; afirmar poder abster-se dela é uma impostura que deve ser recusada sob pena de privação do próprio fundamento de toda ciência (BRUYNE, et al., 1991 p. 101).

Em última instância, a razão de ser das teorias científicas,

é o papel que elas desempenham diante dos fatos: elas dão um quadro coerente dos fatos conhecidos, indicam como são organizados e estruturados, explicam-nos, prevêm-nos e fornecem, assim, pontos de referência para a observação dos fatos novos. Sem fatos novos, as teorias não teriam objeto (O'NEIL, 1972 *apud* BRUYNE et al, 1991 p. 102).

O seguinte referencial teórico será utilizado durante o trabalho: Modelagem, Simulação Computacional (simuladores comerciais e pesquisas que utilizam o conceito de geração automática de modelos) e Interface Homem-Máquina.

2.1 Modelagem

Para um melhor entendimento do significado dos processos de modelagem e simulação, pode-se iniciar por algumas definições de modelo:

- “É uma representação dos aspectos essenciais de um sistema, que apresenta conhecimento desse sistema em uma forma utilizável” (EYKHOFF, 1974 *apud* (GARCIA, 2005 p. 19).
- “Um modelo é a representação simplificada da realidade ou das principais características de um sistema. Ele é composto por um conjunto de relações que podem ser expressas sob a forma de palavras, diagramas, tabelas de dados, gráficos, equações matemáticas ou qualquer combinação desses elementos e que possibilite a simulação de fenômenos observados empiricamente ou não.” (MOREIRA, 2001, p.17 *apud* (PEDRO, 2006 p. 6)).
- “Um modelo pode ser visto como um novo mundo construído para representar fatos/eventos/objetos/processos que acontecem no nosso mundo ou num mundo imaginário.” (SAMPAIO, 1998).

De forma geral, pode-se entender um modelo como uma representação, por meio de símbolos, de algum problema ou fenômeno que se queira estudar. Os símbolos utilizados variam conforme o recurso ou ferramenta disponível, podendo ser mais simples ou mais sofisticados, de acordo com o grau de fidelidade que se pretende dar ao modelo criado e com o nível de abstração desejado (PEDRO, 2006).

Ou seja, por modelo entende-se a representação de um sistema real ou imaginário usando uma linguagem, um meio e segundo um ponto de vista. Um fato importante na construção de um modelo é manter a fidelidade do ambiente, mas de uma forma simplificada. Modelos são usados para representar o conhecimento e comportamento de sistemas complexos (TRIVELATO, 2003).

A criação e utilização de modelos estão presentes no cotidiano das pessoas. Normalmente, quando se deseja representar alguma coisa de forma simplificada, utilizam-se modelos.

O processo de criação e construção de um modelo é denominado modelagem. Existem diversas formas de se modelar uma determinada realidade ou objeto.

O fato de haver diversas formas de se modelar determinado objeto se dá por fatores como a percepção do modelador, pelas informações consideradas para construir o modelo, finalidade do modelo, por exemplo.

A criação e a simulação de modelos podem ser realizadas em qualquer área do conhecimento, necessitando, no entanto, que se escolha um recurso adequado para a criação de modelos que represente satisfatoriamente, por meio de símbolos, os aspectos que se deseja explorar.

No caso de estudos de fenômenos (sejam eles físicos, ambientais, sociais, econômicos, dentre outros), devido às suas complexidades, é necessário o uso de ferramentas de modelagem que possam representar diferentes contextos, que sejam fáceis de manusear e que permitam a contínua alteração e teste dos modelos (PEDRO, 2006 p. 8).

Para Trivelato (2003 p. 6),

Modelar é o primeiro passo para a análise de um sistema de qualquer natureza e sob qualquer aspecto. Quando o modelo é uma representação válida de um sistema, informações significativas podem ser retiradas sobre sua dinâmica ou seu desempenho.

Modelagem é um processo complexo e, em vários campos, envolve a capacidade de dedução e inferência. A modelagem de um sistema será mais fácil se conhecermos as leis pertinentes ao sistema, se for fácil obter uma representação gráfica ou simbólica do sistema e se as incertezas entre as entradas, parâmetros e saídas puderem ser quantificadas (TRIVELATO, 2003).

Portanto, pode-se definir um modelo como: "Uma representação física, matemática, lógica ou computacional qualquer de um sistema, processo, fenômeno ou entidade" (TRIVELATO, 2003 p. 6).

Não faz sentido se valer de ferramentas de modelagem que não trazem facilidades para a construção e testes de modelos. Segundo Pedro (2006 p. 8), "A finalidade do modelo é poder ser utilizado para a análise e entendimento do fenômeno original, tornando mais simples esse processo de estudo e aprendizagem".

A utilização de modelos é muito ampla e se dá em todos os ramos da ciência, sendo múltiplas as suas funções. Isto está relacionado à maneira com a qual o modelo é usado para mostrar os comportamentos dos componentes e suas interações nas entidades modeladas.

Segundo Shannon (1978) *apud* Strack (1984), os modelos têm cinco funções básicas.

A primeira é o auxílio na estruturação do pensamento e raciocínio, pois o modelo força uma organização, avaliação e validação das idéias para que a representação destes pensamentos seja clara e objetiva. Assim os conceitos são ordenados e inconsistências eliminadas.

A segunda função é de facilitar a comunicação, pois

em muitas situações a linguagem verbal é ambígua, enquanto que os modelos tornam possível uma descrição única da estrutura e revelam relações importantes (STRACK, 1984 p. 4).

A terceira função é em relação aos riscos e custos em treinamentos e instruções, onde a prática pode ser exercida sem a pressão de não poderem acontecer erros.

A quarta função, que, segundo o autor, é a mais importante, é a realização de previsões. Os modelos servem para antecipar comportamentos e situações, antevendo-os antes de sua ocorrência.

A quinta função é a realização de experimentos controlados que tornam viáveis experimentos que seriam impraticáveis ou com custo muito alto.

Visto isso, a próxima etapa é entender como os modelos fazem parte da simulação computacional, uma vez que a simulação computacional utiliza os conceitos de modelagem em sua concepção. Estes conceitos estão expostos na seção 2.2:

2.2 Simulação

Nesta seção serão apresentados os conceitos de simulação computacional, o seu surgimento, definições de simulação, método de condução de um projeto de simulação, além de ferramentas de simulação.

A simulação computacional surgiu a partir da evolução das técnicas de Pesquisa Operacional (NETO, 2006). Então, para melhor compreender o surgimento da simulação, faz-se necessário entender o conceito de pesquisa operacional. A pesquisa operacional, segundo Ackoff e Sasieni (1971 p. 55),

pode ser considerada como uma aplicação do método científico por equipes interdisciplinares sobre problemas que dizem respeito aos controles de sistemas organizados (homem-máquina) com a finalidade de obter as soluções que melhor satisfaçam aos objetivos da organização como um todo.

Ainda, conforme Ackoff e Sasieni (1971 p. 11),

(...) a PO procura dar tratamento sistemático e racional aos problemas fundamentais relacionados com o controle de sistemas, conduzindo as decisões que, de acordo com determinado critério, produzam melhores resultados à luz de toda informação que possa ser usada com proveito.

O uso de modelos é uma característica na pesquisa operacional como forma de compreender e estruturar o problema para encontrar a sua resposta (OENNING, 2004).

Sobre o surgimento da simulação computacional Neto (2006 p. 79) descreve,

Precisamente, ela teve sua origem como uma extensão do Método de Monte Carlo, que foi proposto por Von Neumann e Ulam, em 1940, durante a Segunda Guerra Mundial, em pesquisas realizadas no laboratório de Los Alamos, para um projeto chamado Manhattan, cujo objetivo era a invenção da bomba atômica. Por razões de sigilo, o código "Monte Carlo" foi adotado.

Modelos de simulação podem ser de diversos tipos: desde simples maquetes até sistemas virtuais, desenvolvidos em computadores digitais (VACCARO, 1999).

O processo de simulação computacional é amplo e irrestrito na sua percepção. Desta forma diferentes pessoas entendem e percebem o sistema a ser modelado de forma diferente. Simulação é uma forma economicamente viável de pré-testar propostas de sistemas ou políticas sem a necessidade do desenvolvimento de protótipos caros, campos de teste ou sistemas reais (VACCARO, 1999).

2.2.1 Definição de Simulação

A simulação computacional é uma ferramenta que possibilita expressar uma realidade em um ambiente virtual. Por sua flexibilidade, a simulação contempla a variabilidade do sistema, gerando resultados dinâmicos, de acordo com os parâmetros estipulados. A sua utilização possibilita uma melhor visualização e entendimento do sistema real (CASSEL, 1996) (STEFFEN, 2006).

Segundo Vaccaro (1999), simulação é o processo de representação de determinadas características – estáticas ou dinâmicas – de um ente, seja ele tangível ou intangível. Estas características podem ser entendidas como estruturas que podem interagir com o meio, como uma mudança de comportamento frente a estímulos externos.

Para Hollocks (1992) *apud* Carvalho (2006),

a simulação computacional é uma técnica de pesquisa operacional que envolve a criação de um programa computacional para representar alguma parte do mundo real, de tal forma que os experimentos no modelo são como a antevisão do que acontecerá na realidade.

Já para Szymankiewci, McDonald e Turner (1988) *apud* Neto (2006),

simulação é uma das mais poderosas técnicas disponíveis para a solução de problemas. Ela consiste na construção de um modelo matemático, correspondente ao sistema real, que pode ser experimentado e avaliado, quando submetido a diversos cenários de ação.

Um modelo de simulação computacional nada mais é que uma função, em geral complicada, que transforma dados de entrada em dados de saída (VACCARO, 1999). Desta forma, a simulação computacional tem nos dados de entrada uma das etapas primordiais para a sua correta concepção, pois se o modelo não for abastecido com dados corretos, dados incorretos resultarão da simulação.

Law e Kelton (1991) apresentam as possibilidades de realização de estudos sobre um sistema de forma bastante clara e concisa.

- Experimentos no mundo real: quando lidamos com sistemas que representam risco seu uso pode ser recomendado no caso de treinamentos;
- Desenvolvimento de modelos matemáticos para representação do sistema: Esta abordagem pode resultar em modelos complexos e, em certos casos, insolúveis, dado o nível de exatidão requerido.

A simulação computacional está sendo usada em diversas situações, como por exemplo: aplicações militares (simulações de guerra e ambientes de treinamento), projetos aeroespaciais, sistemas urbanos, meteorologia, projetos de engenharia, sistemas de manufatura, sistemas ecológicos e ambientais e sistemas de computação (hardware, redes de comunicação, arquiteturas, etc.) (BANKS, 1998).

Para desenvolver um projeto de simulação computacional, é necessário um método de condução. Na seção 2.2.2 será apresentado o método de condução definido por Law e Kelton (1991).

2.2.2 Método de Condução de Projetos de Simulação

O desenvolvimento de projetos de simulação envolve diversas etapas e nenhum projeto de simulação pode ser considerado bem sucedido se não resultar na tomada de uma decisão. O objetivo fundamental de um estudo de simulação é fornecer ajuda na compreensão do comportamento de um sistema (VACCARO, 1999) (LAW e KELTON, 1991).

A seguir, um detalhamento das etapas de condução de projetos de simulação computacional, segundo Law e Kelton.

Law e Kelton (1991) apresentam um método para o desenvolvimento de projetos de simulação. As etapas do método são apresentadas abaixo:

1. Formulação do problema e planejamento do estudo: Esta etapa consiste na visualização clara e precisa de seus objetivos principais e dos pontos específicos, bem como as alternativas a serem seguidas.
2. Coleta de dados e formulação do modelo: Os dados do sistema real devem ser coletados para especificar as distribuições de probabilidade a fim de manter a variabilidade no modelo a ser construído.
3. Validade: Etapa em que as pessoas que conhecem o sistema real participam de modo a verificar se o modelo é válido.

4. Construção de um programa computacional e verificação: Codificação do modelo teórico em um programa computacional. Após a conclusão do programa, deve-se verificá-lo usando técnicas de verificação, sugeridas por Law e Kelton (1991), as quais podem ser encontradas no trabalho de Cassel (1996).
5. Fazer rodadas-piloto: Após verificar o modelo computacional, devem-se fazer rodadas-piloto para que este sistema possa ser validado.
6. Validação: Comparação dos dados de saída com os dados do sistema real.
7. Planejamento de experimento: Law e Kelton (1991) dizem que alguns parâmetros devem ser avaliados e definidos, como por exemplo, condições iniciais da Simulação, a duração de cada rodada de Simulação e o número de replicações associadas ao modelo, devido à natureza estocástica da Simulação.
8. Fazer rodadas produtivas: Considerando os parâmetros estabelecidos na etapa anterior, fazem-se as rodadas para a coleta dos dados do sistema de interesse.
9. Análise dos dados de saída: Técnicas estatísticas são utilizadas para analisar os dados de saída. Law e Kelton (1991) sugerem que sejam utilizados intervalos de confiança para determinar a precisão estatística dos resultados, bem como saídas gráficas como histogramas e gráficos de barra.
10. Documentação, apresentação e implementação dos resultados: Nesta etapa os resultados do estudo devem ser apresentados e documentados, para uma possível replicação futura do projeto. Além disso, Law e Kelton (1991) afirmam que o sucesso de um estudo de Simulação está associado à apresentação e implementação dos resultados obtidos.

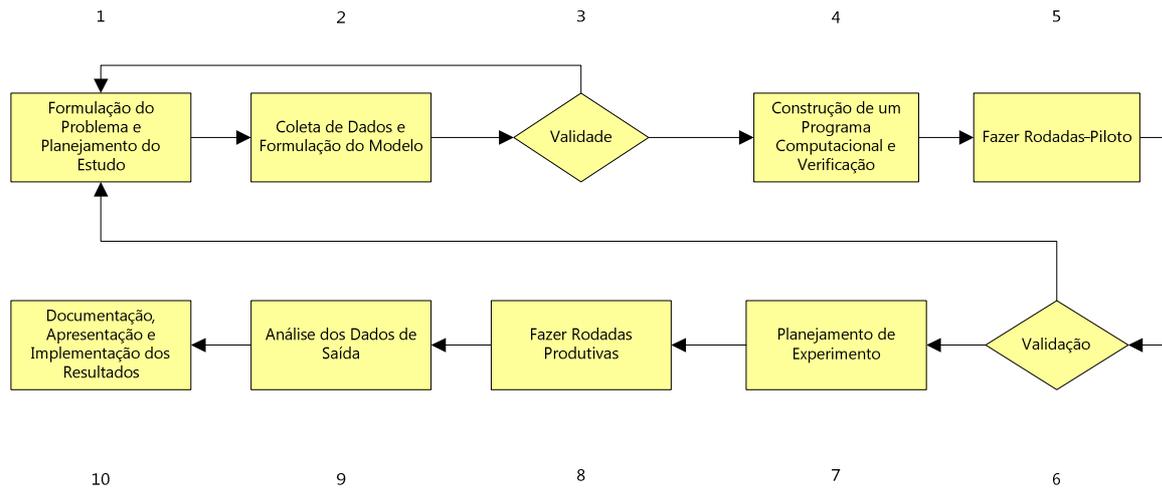


Figura 1 - Passos em um Estudo de Simulação (LAW e KELTON, 1991)

O estudo realizado nesta pesquisa seguirá o método de Law e Kelton para conduzir o projeto de simulação. Na seção 2.2.3 serão apresentadas algumas ferramentas de modelagem, assim como trabalhos correlatos que utilizam a abordagem desta pesquisa.

2.2.3 Ferramentas de Modelagem

Para construir um modelo de simulação computacional, é necessária uma ferramenta que interprete o ambiente e transcreva para o computador. Esta transcrição e codificação da lógica do ambiente real para o computador pode ser realizada através de linguagens genéricas, como Java e C++, ou através de linguagens de programação desenvolvidas para criação de modelos de simulação, como Arena e Micro Saint (BORBA, 1998).

Segundo Rodrigues(1994), existe um *trade-off* entre flexibilidade e especialização do usuário (*expertise*) na escolha de um programa de simulação, conforme a Figura 2:

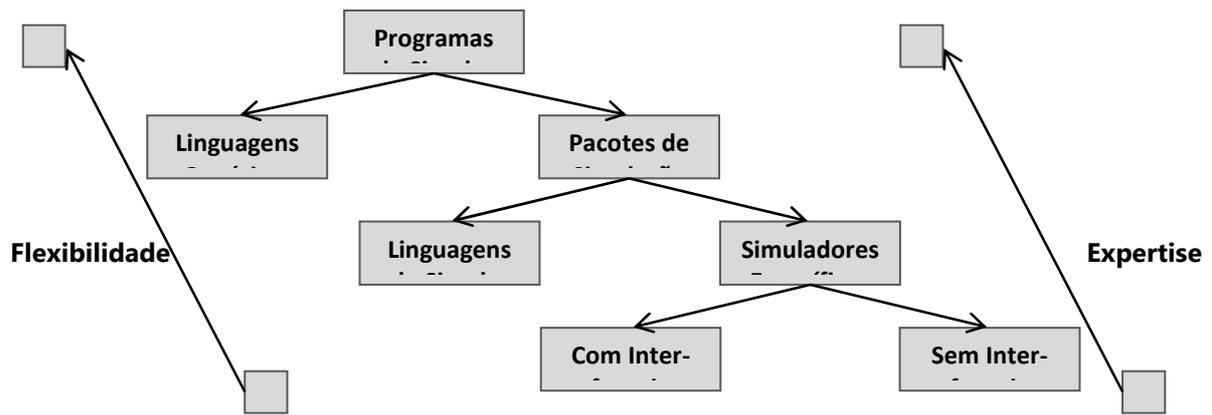


Figura 2 - Trade-off e evolução dos programas de simulação (RODRIGUES, 1994)

Segundo Cassel (1996),

quanto mais flexível o software, maior a dificuldade de programá-lo, pois em softwares mais específicos, várias funções já estão previamente programadas, facilitando a modelagem de sistemas indicados para aquele software. Este *trade-off* vem sendo eliminado pelos novos programas, que utilizam bibliotecas de funções já prontas, mas que permitem também a criação de novas funções por parte do usuário.

Complementando a idéia de Cassel, Pidd (1997) diz que softwares de simulação que tem interface gráfica possuem algumas vantagens, como por exemplo:

- Quando um software com interface gráfica é utilizado, não existe a necessidade de desenvolvimento de um modelo completo antes de rodá-lo para que seu comportamento seja observado. O modelo pode ser construído por etapas, facilitando a sua verificação e construção;
- Alguns destes sistemas, como o Micro Saint, possibilitam a interação entre o usuário e o sistema durante a rodagem do modelo. Desta forma, experimentações podem ser realizadas durante a execução do modelo.

A seguir será apresentado o software de simulação Micro Saint que objetiva atender bem ao *trade-off* entre flexibilidade e especialização, uma vez que permite criar modelos voltados ao problema de estudo, o que torna possível sua aplicação a vários tipos de sistema.

2.2.4 *Micro Saint*

O Micro Saint é um pacote de Simulação de propósito genérico com interface com o usuário. Por sua flexibilidade e devido ao seu método de modelagem por rede de atividades, tem sido utilizado em diversas áreas de aplicação, como indústrias de manufatura e serviços (MENEZES, 1998).

Cassel (1996) define que

O Micro Saint é um programa de simulação que usa fluxograma de tarefas, ou seja, as tarefas são definidas juntamente com suas variáveis, sendo então associadas na seqüência em que serão realizadas. Ao se rodar programa, este gera uma animação simbólica das tarefas realizadas, facilitando a verificação dos inter-relacionamentos dos elementos do sistema modelado de uma forma dinâmica, o que contribui para a detecção de erros lógicos. Além desta animação simbólica, o Micro Saint possui um módulo de animação gráfica chamado ActionView, que gera ícones que se mexem sobre uma figura de fundo e que estão associados com itens que se movimentam através da rede de tarefas.

A Figura 3 exemplifica um fluxograma de tarefas, onde cada figura oval representa as atividades, o retângulo na frente da atividade é a fila e o losango representa nodos de decisão.

Não só o tempo de cada atividade precedente é condição suficiente para a realização de uma tarefa.

Podem impedir a realização da tarefa acontecimentos como falta de material, operadores ocupados, máquina quebrada, etc. Para contemplar isto, o Micro Saint permite a especificação de certas condições necessárias, que são avaliadas testando as variáveis de estado do sistema. São exemplos disto as seguintes expressões: "operadores ≥ 1 ;", "estoque > 0 ;", "maquina $= 1$;" e "clock < 19 ;" (MENEZES, 1998 p. 37).

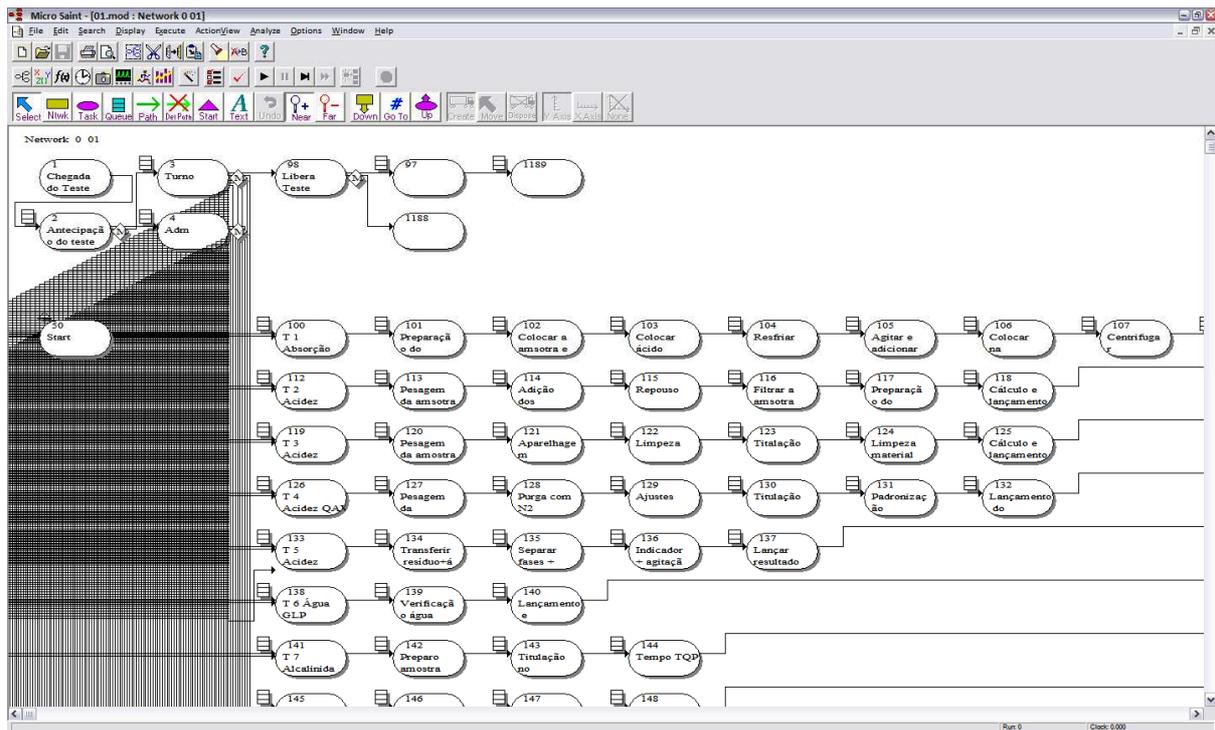


Figura 3 - Malha de Atividades do Micro Saint (Retirado do Software Micro Saint)

Assim, cada variável utilizada para construir condições e regras no modelo deve ser cadastrada na lista de variáveis, conforme a Figura 4:

Name	Value
{variaveis}	0
ano	7
atrasado	0.000000
atrasoFin[]	...
bancada[]	...
chegadaTeste[]	...
clock	0.000000
codigoAmos[]	...
codigoProdi[]	...
contTurnoBan[]	...
cont	0
contaSaiTest	0.000000
contaTeste	0
contAtrasoFi[]	...
contPrazoFin[]	...
cTest[]	...
dia	1
diaM	1
diaS	6
dobraAdm[]	...
dobralnteira[]	...
dobraTurno[]	...
duration	0.000000
eficTecAdm	0.000000
eficTecAdmTu	0.000000
eficTecTurno	0.000000
EQP001	10
EQP002	10
EQP003	10
EQP004	10
EQP005	10
EQP006	10
EQP007	10
EQP008	10
EQP009	10
EQP010	10
EQP011	10
EQP012	10
EQP013	10
EQP014	10
EQP015	10
EQP016	10
EQP017	10

Figura 4 - Lista de Variáveis do Micro Saint (Retirado do Software Micro Saint)

Em interfaces semelhantes, o Micro Saint permite que sejam cadastradas listas de atividades, exportações de resultados, acompanhamento de indicadores.

2.3 Visual Interactive Simulation e Visual Interactive Modeling

Com o passar dos anos, a simulação se difundiu para as mais diferentes áreas de aplicação, causando um crescimento na complexidade das simulações, bem como no volume de dados gerados e analisados pelo usuário (WAGNER, et al., 1996).

O uso de representações gráficas e animações no objeto de estudo foram sendo utilizadas para que o usuário tivesse uma melhor compreensão dos dados, através de representações mais eficientes dos mesmos.

Além da necessidade de uma melhor visualização dos dados, é muito importante que os ambientes de simulação ofereçam recursos para a interação do usuário com os experimentos, tornando desta forma os termos interativo e visual cada vez mais presentes no processo de simulação. Enquanto que o propósito da simulação é obter um nível mais aprofundado no entendimento dos dados, o propósito da interação é oferecer ao usuário a facilidade de poder trabalhar diretamente com o modelo e o processo de simulação, influenciando no seu comportamento e realizando experimentações visando compreender suas propriedades e relacionamentos (WAGNER, et al., 1996 p. 87).

Wagner et al. (1996) diz que estudos na área de simulação estão voltados para o desenvolvimento de sistemas que ofereçam ao usuário maiores facilidades para a simulação e modelagem.

Os conceitos de interatividade apresentados em VIS (Visual Interactive Simulation) começaram a ser propagados com naturalidade para todas as fases do processo de simulação, sendo que o processo de modelagem é o que mais faz uso destas características. O conceito de VIM (Visual Interactive Modeling) se refere à construção do modelo de simulação com o auxílio de um ambiente gráfico interativo. Enquanto que VIS se concentra nas necessi-

dades do usuário do modelo, VIM concentra-se no construtor do modelo de simulação (WAGNER, et al., 1996 p. 88).

O conceito VIS surgiu com o propósito de facilitar a condução dos experimentos de simulação pelo usuário, possibilitando a visualização de dados intermediários e controle completo do experimento e de parâmetros do modelo, ou seja, tornar a simulação uma ferramenta mais amigável ao usuário (WAGNER, et al., 1996)(SCHRAMM e FORMOSO, 2007).

A Simulação Interativa Visual é uma técnica que consiste no uso de representações dinâmicas nas quais os usuários podem alterar os parâmetros do modelo durante a execução da rotina a fim de analisar seus impactos (AU e PAUL, 1996 *apud* SCHRAMM e FORMOSO, 2007 p. 3).

Um modelo VIS pode ser composto por uma mistura de diagramas de blocos, ícones, gráficos e textos utilizados para representar o comportamento do sistema durante a simulação. As características dinâmicas e as mudanças discretas de um processo podem ser vistas na tela do computador. Através da visualização destas mudanças, o usuário pode testar o modelo de simulação e validar seus resultados, obter insights sobre o comportamento do sistema real, comparar vários cenários alternativos e prever o comportamento futuro do sistema (SCHRAMM e FORMOSO, 2007).

O autor continua dizendo que a motivação principal de VIS é a integração com o processo de simulação, caracterizando-se pelo controle da simulação, ou seja, há a visualização de dados intermediários e capacidade de interação com o modelo durante o transcorrer da simulação. Podem ser modificados parâmetros e definições de variáveis, forma de apresentação dos resultados, etc., fazendo com que as consequências no processo de simulação sejam imediatas. Deste modo, VIS engloba a realização interativa das várias etapas do processo de simulação, utilizando representações gráficas e, em algumas, animação (WAGNER, et al., 1996).

Schramm e Formoso (2007) citam como vantagens de utilizar VIS:

- Visualização gráfica: o usuário acompanhar os eventos enquanto estes ocorrem e identificar erros potenciais, isto é, é mais fácil verificar e validar o modelo.
- Entendimento do modelo: ambientes VIS são adequados para promover um melhor entendimento do modelo, bem como a participação do usuário nos processos de modelagem e simulação. A VIS melhora a comunicação do modelo e de seus resultados para todos os participantes do processo, quer estes sejam especialistas ou não, fornecendo assim uma boa solução para promover a discussão conjunta e interação entre os integrantes do estudo de simulação.

O ponto principal de VIS é prover ferramentas ao usuário para que ele possa melhorar o entendimento do comportamento do sistema que está sendo modelado. E para isto Rooks (1991) *apud* Wagner et al (1996) apresenta 4 requisitos para ambientes VIS:

- Intervenção: visa prover o usuário de uma forma de interação com o modelo;
- Inspeção: deve permitir ao usuário o acesso a todos os dados relevantes do modelo para realizar os experimentos;
- Especificação: o usuário deve ser capaz de especificar os parâmetros do modelo, de acordo com os objetivos para a análise do modelo;
- Visualização: deve fornecer ao usuário a capacidade de visualizar os dados do modelo de forma que ele possa compreender o seu comportamento.

Já os ambientes de modelagem, que utilizam conceito de VIM, começaram a ser desenvolvidos a partir da segunda metade da década de 90, sendo constituídos por inúmeras ferramentas que auxiliam o desenvolvimento de modelos de simulação. O objetivo principal destes ambientes é aumentar a produtividade do modela-

dor, com ênfase na construção do modelo através de recursos gráficos(WAGNER, et al., 1996).

A Figura 5 apresenta uma arquitetura típica de um ambiente VIM:

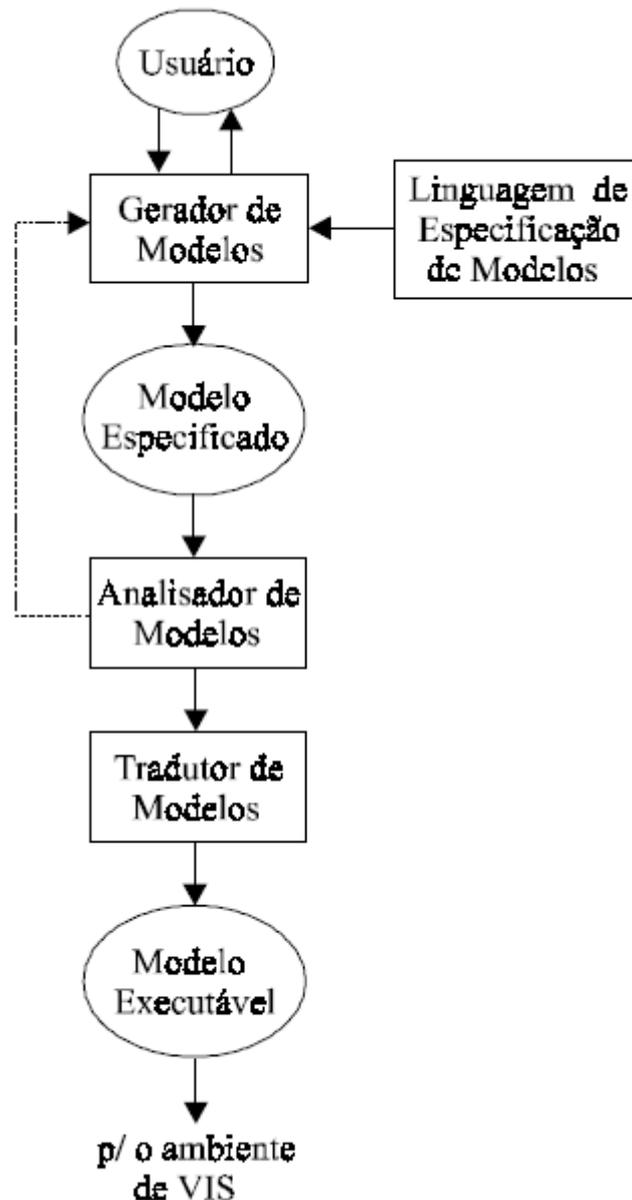


Figura 5 - Arquitetura de um ambiente de modelagem (WAGNER, et al., 1996)

O módulo gerador de modelos permite que o modelador construa graficamente o modelo de simulação e a sua visualização para uma posterior análise por parte do usuário. Por intermédio de uma linguagem de especificação de modelos

(LEM), o modelador especifica a lógica do modelo. Seu propósito principal é descrever o comportamento dos elementos do sistema modelado.

Após a construção do modelo, o módulo analisador de modelos verifica a correção da especificação do modelo gerada.

Caso esta especificação seja assegurada, ela é passada ao tradutor de modelos que a transforma em uma representação executável. Normalmente o modelo executável é representado através de uma linguagem de programação. A partir deste momento, o ambiente de VIS pode ser ativado para realizar os experimentos sobre o modelo construído(WAGNER, et al., 1996).

Por fim, o autor diz,

Independentemente de como o modelo de simulação é construído (forma textual ou através de recursos gráficos), o sistema deve prover uma ferramenta altamente amigável e composta de diversos recursos gráficos que permita ao usuário projetar o cenário que será utilizado para a apresentação de resultados(WAGNER, et al., 1996 p. 91).

Assim, ambientes de VIM permitem a construção do modelo de simulação através de recursos visuais e interativos. Tais ambientes estão interessados em oferecer recursos para o modelador, não possuindo recursos para a fase de experimentação do modelo.

2.3.1 Ambientes de Simulação Orientados a Objetos

O conceito de orientação a objetos oferece vantagens sobre as linguagens convencionais de simulação, pois a abordagem busca facilitar a manutenção, a reusabilidade e a extensibilidade do código/sistema (PENTEADO, 1996).

Programas desenvolvidos segundo a orientação a objetos são mais fáceis de serem lidos e entendidos do que os desenvolvidos segundo a abordagem tradicional (PENTEADO, 1996 p. 9).

A primeira linguagem de programação orientada a objetos foi a SIMULA, que era originalmente designada como uma linguagem de simulação de finalidade geral. A linguagem já fornecia objetos, classes e herança, algumas das características fundamentais da programação orientada a objetos (TRAVASSOS, 2002).

Segundo Travassos (2002 p. 32),

Na simulação orientada a objetos o sistema em estudo é descrito como uma coleção de objetos que interagem. Os comportamentos dos objetos são descritos por meio de métodos, e as suas características estão contidas nos atributos dos objetos que podem ser representadas por variáveis. Tudo que o objeto conhece está em suas variáveis e tudo que ele pode fazer são os seus métodos. Essa estrutura mapeia claramente os objetos significativos do mundo real que constituem o sistema sendo modelado.

A Figura 6 apresenta uma visão simplificada da estrutura de um modelo orientado a objetos. As elipses representam classes de objetos, as setas cheias indicam as heranças implementadas e as setas tracejadas a colaboração existentes entre os objetos instanciados. Na orientação a objetos, os objetos que pertencem às classes, interagem ou se comunicam através de troca de mensagens.

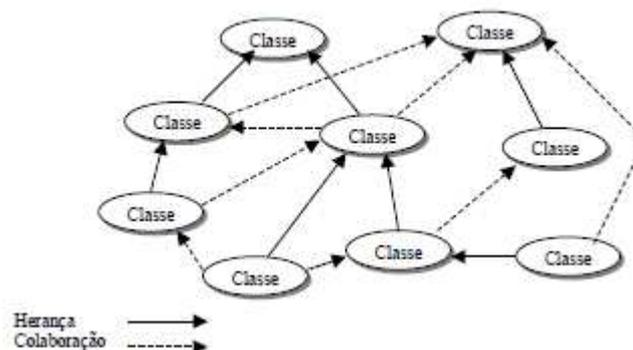


Figura 6 - Estrutura do Modelo Orientado a Objeto (TRAVASSOS, 2002 p. 33)

Os conceitos básicos da abordagem orientada a objetos são (ROBERTS e HEIM, 1988 *apud* TRAVASSOS, 2002):

- Classe: é usada para definir um tipo de dado abstrato particular. A definição de classe irá especificar a estrutura de dados implementada e as operações que podem ser executadas sobre os dados. Essa propriedade de juntar dados e código em um único objeto (ou classe) é conhecida como encapsulamento. Uma classe é descrita pelos atributos (variáveis) que seus objetos possuem e métodos (funções) que podem atuar sobre esses objetos;
- Objeto: é uma instância de uma classe específica. Ela contém todas as informações desta classe específica e as ações que podem ser executados nos dados, invocando-se os métodos da classe. O mecanismo usado para se efetuar a chamada dos métodos de um objeto é pelo meio de mensagens. Uma mensagem para um objeto requisita a execução de um procedimento. Uma mensagem invoca um método (procedimento) no objeto receptor para produzir um comportamento requerido;
- Herança possibilita que as classes sejam organizadas em hierarquias de modo que uma classe que é subordinada a outra classe automaticamente herde todas as variáveis e métodos da classe superior;
- Polimorfismo é a possibilidade de se enviar uma mesma mensagem a um objeto, cuja execução vai depender do tipo de objeto instanciado. Uma mesma operação pode se comportar diferentemente quando definida em classes diferentes.

A seguir, serão apresentadas algumas ferramentas que utilizam a abordagem orientada a objetos:

2.3.1.1 Arena

O Arena é um software com um ambiente gráfico integrado de simulação, que contém todos os recursos para modelagem de processos, desenho e animação, análise estatística e análise de resultados.

Sakurada e Miyake(2003) explicam que

o software de simulação Arena foi desenvolvido inicialmente pela *System Modeling Corporation* e baseia-se na linguagem de simulação SIMAN. A construção do modelo é feita através da seleção do módulo que contém as características do processo a ser modelado. O modelador não precisa necessariamente conhecer a linguagem SIMAN para construir um modelo.

Para criar um modelo, não é necessário escrever nenhuma linha de código, pois todo o processo de criação do modelo de simulação é gráfico e visual, e de maneira integrada, conforme Figura 7.

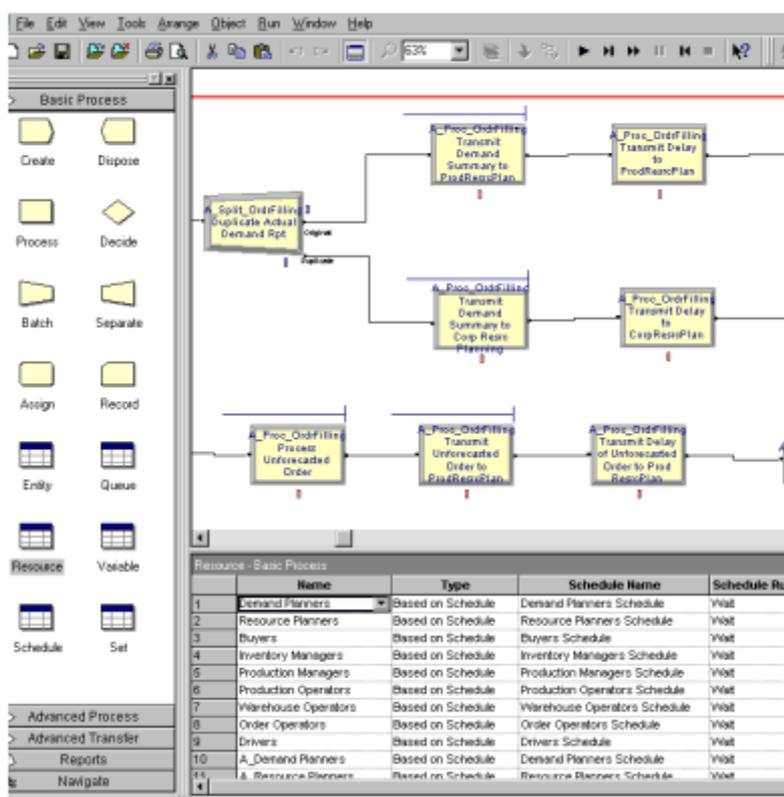


Figura 7 - Simulação em Arena (LOVEJOY, 2001)

Este software se utiliza de modelos de atividades específicas (coleção de objetos e ferramentas). Estes modelos pré-configurados também são chamados de templates pela ferramenta. Este conceito é diferente do conceito dos templates utilizados na pesquisa. O uso dos templates no Arena permite ao usuário descrever o comportamento do processo em análise, através de respostas às perguntas pré-elaboradas, sem programação, de maneira visual e interativa.

O Arena também oferece a modelagem denominada de baixo nível para usuários que preferem ter o total controle da modelagem. O modelo é construído diretamente na linguagem SIMAN (SAKURADA e MIYAKE, 2003).

2.3.1.2 OMNet++

OMNeT ++ é um ambiente de simulação de código aberto e modular, que roda em ambiente gráfico. Sua principal aplicação é a área simulação de redes de comunicação e por causa de sua arquitetura flexível e genérica. Vários modelos de simulação de código aberto têm sido publicados, no domínio das simulações.

O OMNet++ é gratuito para as universidades e aplicações sem propósito comercial.

Segundo Varga (2001), o OMNet++ é um simulador de eventos discretos, baseado em C++, para modelar relacionamentos de atividades, multitarefa e outros sistemas distribuídos ou paralelos. O software tenta preencher a lacuna existente entre os softwares livres e os softwares comerciais.

Dentre os benefícios da utilização do OMNet++, pode-se citar:

- Apto a simulações de larga escala, modelos de simulação que necessitam ser hierarquizados e pode reaproveitar componentes sempre que possível;

- Os programas de simulação são infames pelos longos períodos de depuração. Assim, o software de simulação deve ter ênfase na rastreabilidade e na facilidade de depuração. (O mesmo recurso é útil para uso educacional do software);
- O software de simulação deve ser modular, personalizável e permitir maior incorporação de modelos de simulação;
- As interfaces de informações devem ser abertas: deve ser possível gerar e processar entradas e saídas de dados com comodidade.

O OMNet possui os seguintes componentes (VARGA, 2001):

- Biblioteca de simulação;
- Compilador para a linguagem de descrição de topologia – NED (*Network Description*);
- Editor gráfico de redes para os arquivos NED – GNED (*Graphical Network Editor*);
- Interface gráfica para execução da simulação;
- Interface em linha de comando para a execução da simulação;
- Ferramenta para impressão dos gráficos de saída;
- Utilitários (gerador de números aleatórios);
- Documentação, exemplos de simulação.

A Figura 8 apresenta o software OMNet++.

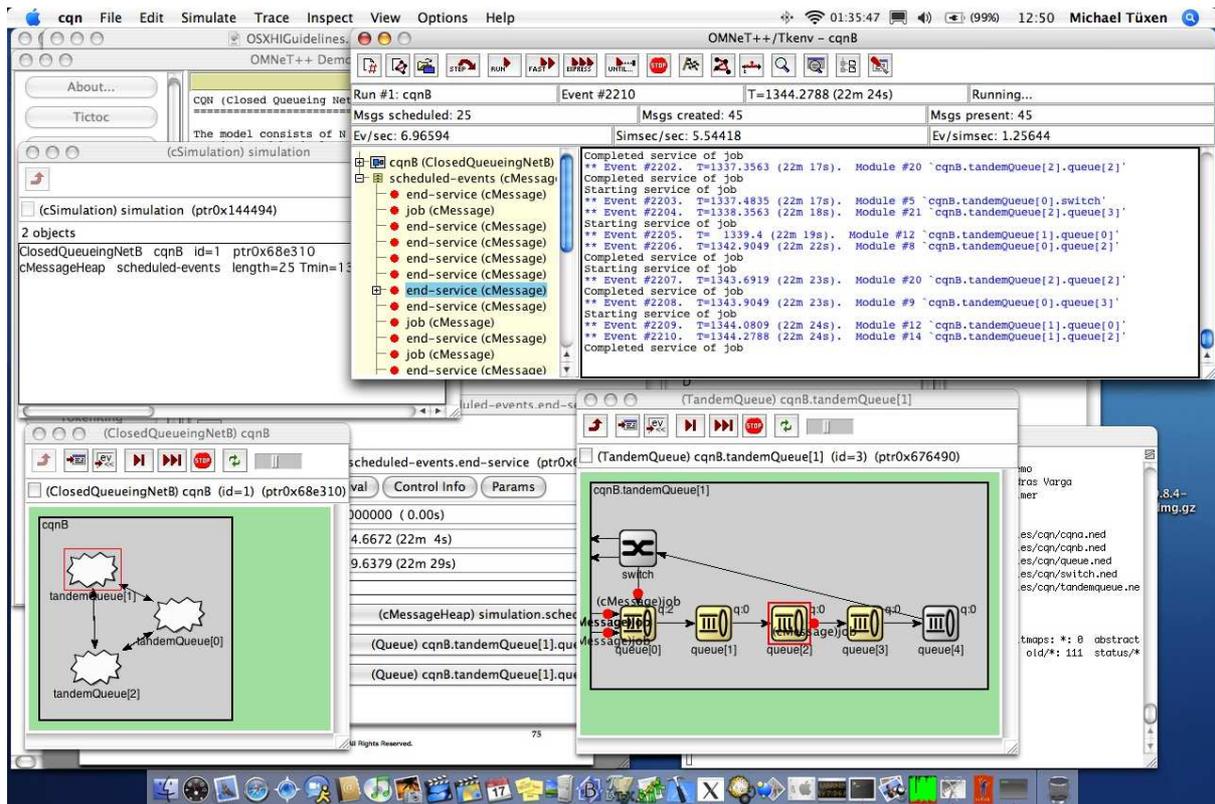


Figura 8 - Exemplo do OMNeT++ (OMNET++, 2009)

2.3.1.3 ASiA

O ASiA (Ambiente de Simulação Automático) é um ambiente que gera automaticamente o programa, desenvolvido na área acadêmica para simulação seqüencial, baseado em uma especificação gráfica do modelo fornecida pelo usuário (Santana, 1996 *apud* (BRUSCHI, 2003).

Ainda, segundo o autor, o ambiente tem por objetivo afastar o usuário da tarefa de transcrição do modelo em um programa de simulação. A idéia é que o usuário forneça, através de um editor gráfico, os parâmetros da simulação. As demais tarefas serão executadas automaticamente pelo ASiA.

Quatro estágios básicos são utilizados para a obtenção dos resultados no sistema de simulação automático:

- Editor gráfico: fornece ao usuário uma interface amigável, organizando e testando a consistência dos dados de entrada;
- Gerador de aplicações: responsável pela geração do programa de simulação;
- Estágio de saída e análise dos dados: responsável por analisar os resultados fornecidos pela simulação;
- Saída gráfica: responsável por apresentar os resultados de uma maneira gráfica.

O ambiente constantemente é atualizado, com o intuito de facilitar cada vez mais a construção de modelos (BRUSCHI, 2003).

2.4 Ambientes de Simulação Automáticos

Em um projeto de simulação, a transcrição do modelo definido pelo usuário pode se tornar um obstáculo em um modelo de simulação. Neste sentido, foi desenvolvido o conceito de ambiente de simulação que, por uma série de recursos, auxilia o usuário no desenvolvimento de uma simulação (BRUSCHI, 2003).

Para isso, segundo Tanir e Sevinc (1994) *apud* Bruschi (2003), "(..) é necessário a inclusão de editores gráficos, interpretadores, compiladores e otimizadores". O ambiente de simulação é definido então como um ambiente de simulação automático.

Nos ambientes de simulação automáticos uma característica importante é a programação visual. Utilizando a programação visual, o modelo de simulação é criado graficamente na tela e os resultados (comportamento do sistema) são observados através de gráficos ou animações. A programação visual e a saída visual são características da simulação visual interativa (VIS –

Visual Interactive Simulation) (BELL e O'KEEFE, 1987) *apud* (BRUSCHI, 2003 p. 60).

A autora discorre ainda que o usuário pode manipular as informações visuais de um modo mais eficiente, tornando a programação visual particularmente desejável e agradável, ou seja, os ambientes de simulação automáticos utilizam os conceitos de VIS e VIM em sua concepção.

Existem vários ambientes de simulação, sendo que a maioria é comercial. A seguir serão apresentados alguns destes ambientes de simulação.

A seguir serão apresentadas ferramentas e aplicações que utilizam os conceitos de geração automática de modelos.

2.4.1 ASDA – Um Ambiente de Simulação Distribuída Automático

O trabalho de Bruschi (2003) vem ao encontro do propósito desta pesquisa, pois tem propósito e objetivos semelhantes, onde propõe um ambiente automático para o desenvolvimento de simulação com o intuito de facilitar a utilização e desenvolvimento da mesma. O ambiente proposto, denominado ASDA (Ambiente de Simulação Distribuída Automático), possui as características apresentadas pelos ambientes de simulação automáticos, isto é, uma interface amigável, geração e compilação automática do código de simulação, execução e análise dos resultados. Adicionalmente, o ASDA permite a utilização de simulação distribuída por usuários que não possuem profundos conhecimentos de computação paralela, dos métodos de sincronização para simulação distribuída ou dos métodos para análise dos resultados (BRUSCHI, 2003).

O objetivo do ASDA é oferecer ao usuário um ambiente flexível e amigável. O ASDA foi especificado em sete módulos:

- Interface com o usuário: é onde o usuário interage com a ferramenta. É através deste módulo que o usuário pode definir o modelo, utilizando as técnicas de modelagem. Uma vez definido o modelo, o programa de simulação é gerado com base nas linguagens/bibliotecas de simulação disponíveis no ambiente. Tem a função de identificar e gerenciar os usuários do sistema, receber os dados armazenados para os usuários e para os módulos do sistema e permitir a especificação gráfica do modelo a ser simulado;
- Avaliador: é o módulo que em que se escolhe a abordagem a ser adotada na geração do programa de simulação. Esta escolha pode ser definida pelo usuário ou pelo próprio ambiente. Este módulo tem a função de auxiliar o usuário nas decisões durante a definição do modelo e geração do código de simulação. O módulo avaliador foi dividido em três níveis, sendo: I) verifica se o modelo do usuário está sendo especificado corretamente; II) ajuda o usuário a escolher entre simulação seqüencial ou distribuída SRIP ou MRIP; III) ajuda o usuário a escolher a abordagem de simulação;
- Gerador: baseado na especificação gráfica e na decisão do módulo avaliador, este módulo gera o código do programa de simulação;
- Executor: responsável por controlar a execução dos programas de simulação;
- Interface de Software: responsável por fornecer uma interface entre os programas de simulação já desenvolvidos e o módulo replicador;
- Replicador: gerencia as replicações da simulação e garante a obtenção de resultados confiáveis através da abordagem MRIP;

- Escalonamento: responsável pelo escalonamento dos processos gerados pelo módulo executor.

A Figura 9 apresenta o diagrama da estrutura modular proposto para o ASDA:

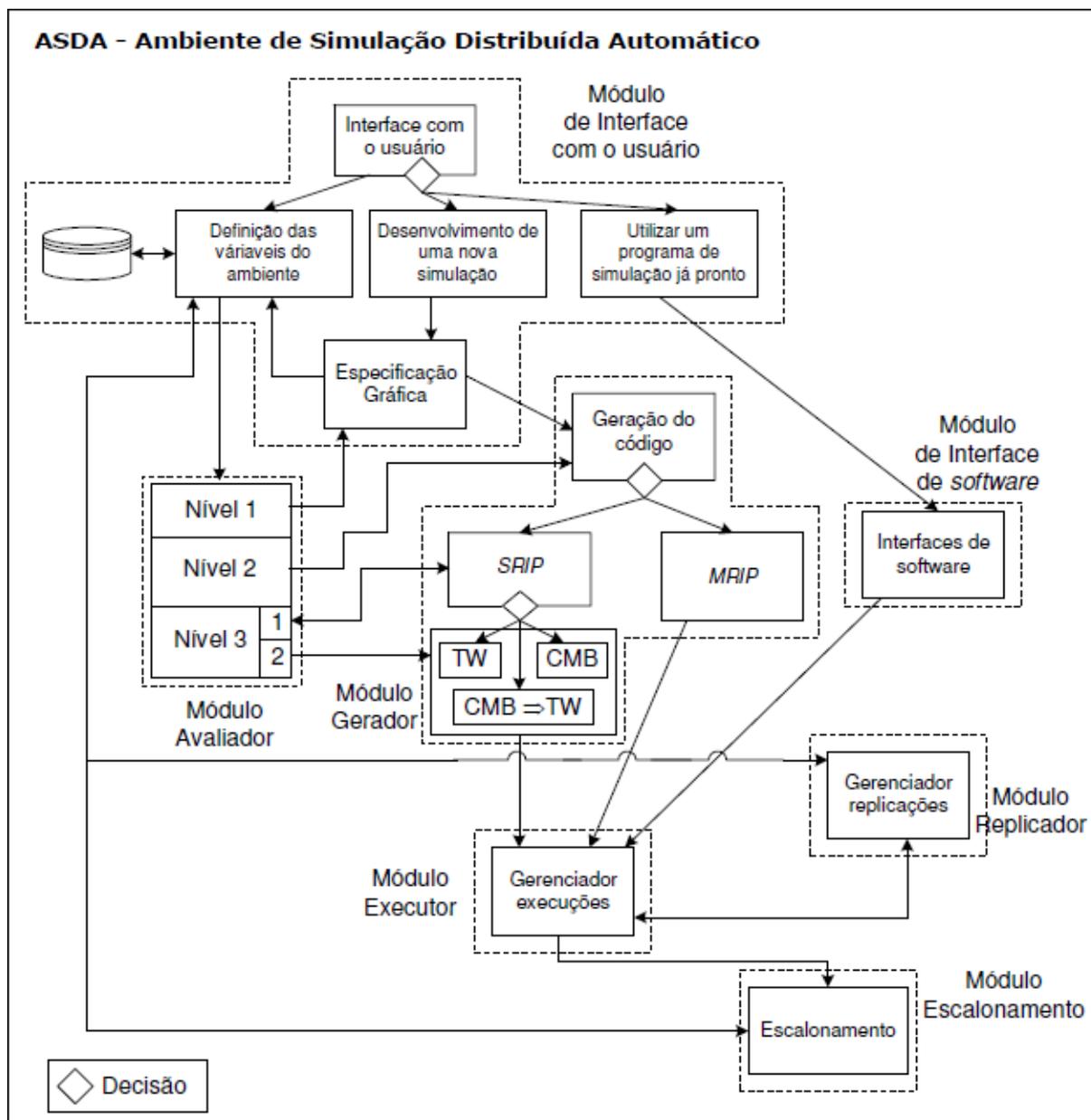


Figura 9 - Diagrama Modular do ASDA (BRUSCHI, 2003 p. 70)

Segundo Bruschi (2003 p. 4), três características se destacam no ambiente ASDA:

- Possibilidade de desenvolvimento de programas de simulação distribuída, utilizando várias abordagens (SRIP⁴, MRIP⁵ ou SRIP + MRIP);
- Possibilidade de utilizar um programa de simulação já desenvolvido e replicá-lo;
- Auxílio fornecido para decisões referentes à escolha da abordagem e escolha do protocolo de sincronização.

O autor salienta que um ponto forte do ASDA é a sua flexibilidade. “Uma opção oferecida é a geração de programas de simulação seqüenciais e a utilização destes programas para execução paralela através da abordagem MRIP” (BRUSCHI, 2003 p. 67).

O público-alvo para o qual esta abordagem foi desenvolvida são alunos, professores, pesquisadores e profissionais interessados em desenvolver projetos em simulação computacional, que podem apresentar diferentes tipos de conhecimentos (BRUSCHI, 2003):

- Usuários que apresentam um grande conhecimento em simulação;
- Usuários que possuem conhecimento superficial em simulação;
- Usuários com conhecimento em simulação, mas que dispunham de pouco tempo para o desenvolvimento da simulação;
- O uso das interfaces deve oferecer ao usuário um ambiente de fácil aprendizagem e utilização;
- Possibilitar a geração completa de um problema de simulação para um usuário menos experiente;

⁴ Single Replication in Parallel (SRIP) significa simulação distribuída. O sistema é particionado em processos lógicos e cada processo lógico é mapeado em um processador.

⁵ Multiple Replication in Parallel (MRIP) significa múltiplas replicações em paralelo. O sistema não é particionado. Replicações independentes de um mesmo programa de simulação seqüencial são executadas em paralelo. O resultado de cada replicação é analisado, chegando a uma média final.

- Oferecer flexibilidade necessária para que um usuário mais experiente possa modificar os programas gerados;
- Facilitar a obtenção de dados confiáveis através da verificação das informações no momento da construção computacional;
- Minimizar o tempo de execução da simulação, oferecendo um escalonamento de processos eficientes.

O ambiente foi desenvolvido para usuários com interesse na avaliação de desempenho de sistemas computacionais, principalmente os baseados na teoria de filas.

A autora lista como benefício da utilização do ASDA:

- Oferece ambiente de fácil aprendizagem e utilização ao usuário;
- Possibilidade de uma geração completa de um problema de simulação para um usuário menos experiente;
- Oferece flexibilidade necessária para que um usuário menos experiente possa modificar os programas gerados;
- Possibilita a utilização de simulações já desenvolvidas;
- Oferece diretrizes para que o usuário possa escolher abordagens para a simulação distribuída. Se o usuário preferir, o sistema escolhe a melhor abordagem;
- Facilita a obtenção de dados confiáveis através da avaliação estatística dos resultados gerados pela simulação;
- Minimiza o tempo de execução da simulação, oferecendo um escalonamento de processos eficientes.

Bruschi (2003 p. 181) conclui sua pesquisa, informando que

a necessidade de desenvolvimento de ambientes que tenham como objetivo facilitar a utilização de simulação distribuída deve-se principalmente à di-

ficuldade encontrada pelo usuário em todo o processo de desenvolvimento da simulação. São necessários conhecimentos referentes a modelagem, simulação, análise estatística, computação paralela e finalmente simulação distribuída.

A autora complementa, dizendo que o ASDA foi especificado para que possa ser utilizado tanto por usuários sem muita experiência, quanto por usuários com experiência, mas que queiram minimizar o tempo necessário para o desenvolvimento de programas de simulação.

Além de desenvolver uma simulação distribuída desde o início (especificando o modelo, gerando o código, escolhendo a abordagem a ser utilizada na geração do programa de simulação, etc.), o usuário pode utilizar outra característica do ambiente: a possibilidade de diminuir o tempo necessário para se obter resultados confiáveis de um programa de simulação já desenvolvido, replicando com a abordagem MRIP (BRUSCHI, 2003 p. 182).

Usando a abordagem, também se eliminam os problemas de desenvolvimento de um novo programa para um modelo que já possuía um programa desenvolvido. Um programa de simulação seqüencial existente pode ser replicado pelo ambiente, que ainda oferece todas as ferramentas necessárias para a obtenção de resultados confiáveis (BRUSCHI, 2003).

A autora salienta como característica importante a capacidade que o ambiente tem de fornecer ao usuário um auxílio no desenvolvimento da simulação. "Na especificação tem-se a verificação e validação do modelo e dos parâmetros, sendo que o ambiente avisa quando o modelo apresenta inconsistências"(BRUSCHI, 2003 p. 182).

Por fim, segundo Bruschi (2003), as contribuições da pesquisa são:

- Realização de uma análise crítica da bibliografia existente, fornecendo a contextualização do conceito de simulação distribuída utilizado neste trabalho (ferramenta para avaliação de desempenho);

- Levantamento na literatura dos ambientes de simulação que tem como objetivo facilitar o desenvolvimento de simulação, tanto seqüencial como distribuída;
- Identificação das principais dificuldades encontradas no desenvolvimento de simulações seqüenciais e distribuídas;
- Especificação de um ambiente para desenvolvimento de simulação distribuída, o qual apresenta uma série de características que não são encontradas em ambientes que possuem o mesmo objetivo proposto;
- Desenvolvimento de um protótipo para o ambiente proposto.

2.4.2 Geração Automática de Modelos de Simulação na Indústria Têxtil

Paiva (2005) também realizou uma pesquisa com a abordagem da geração automática de modelos de simulação. A pesquisa teve como objetivo a construção de um sistema de apoio à decisão para estudar o impacto de diversas alternativas de gestão da produção, aplicadas a um sistema produtivo da indústria têxtil. Para alcançar o objetivo, desenvolveu-se um modelo-base, em linguagem de simulação, com as características da indústria têxtil. As características específicas dos recursos utilizados foram desenvolvidas de forma a permitir uma utilização interativa e suscetível de ser parametrizável pelo usuário (PAIVA, 2005).

Para a implementação dos modelos, o autor utilizou o simulador Arena, onde foi desenhado o modelo característico do sistema produtivo. Este modelo contém um módulo para aplicar os critérios e estratégias definidas pelo usuário.

O módulo que gera automaticamente modelos de simulação com especificações e parametrizações personalizadas foi construído na linguagem Visual Basic.

Segundo o autor, a ferramenta foi implementada no sentido de permitir flexibilidade ao nível de materiais, do número de linhas de produção ou do número de máquinas.

Paiva (2005) justifica a utilização do ambiente de simulação Arena, pois

O Arena foi considerado por especialistas de renome em simulação como "O mais inovador software de simulação", por unir os recursos de uma linguagem de simulação à facilidade de uso de um simulador, num ambiente integrado, que contém todos os recursos para modelação de processos, desenho e animação, análise estatística e análise de resultados.

Entre as vantagens já citadas, Paiva (2005) complementa:

- Compatibilidade com Microsoft Office;
- Inclui a tecnologia ActiveX, o que garante a sua conectividade com outras aplicações tipo Excel, Word. Etc. Tendo sido desenvolvido na linguagem Visual C++, utilizando técnicas de orientação a objetos e recursos da Microsoft Foundation Classes (MFC), o Arena faz uso das facilidades oferecidas pela interface gráfica do ambiente;
- O Arena também gera código de simulação na linguagem SIMAN, linguagem específica para simulação, o que permite a depuração de erros da lógica do modelo com facilidade;
- Inclui VBA (*Visual Basic for Applications*), permitindo ao usuário desenvolver rotinas em linguagens de propósito geral, tais como Visual Basic, FORTRAN, C e C++, visando automatizar tarefas, inserir multimídia, desenvolver ferramentas de formação, etc.

Assim como na proposta ASDA, o trabalho desenvolvido por Paiva (2005) aponta como pressuposto facilitar o uso por parte dos usuários sem conhecimentos profundos de informática ou simulação.

A ferramenta denominada GAMSTAF (Gerador Automático de Modelos de Simulação para a Têxtil António Falcão) é constituída pela ferramenta informática, o Arena e uma base de dados (PAIVA, 2005).

A ferramenta informática aqui denominada GAMSTAF permite a definição das características dos elementos que vão compor o nosso sistema produtivo. São também selecionados os critérios e estratégias que serão aplicados no sistema (PAIVA, 2005 p. 90).

O autor complementa que a ferramenta faz a interligação entre o usuário e o simulador. Para tal, apóia-se na parametrização de todos os tipos de definições e características necessárias ao modelo de simulação, na escolha das estratégias a aplicar, na criação de novos cenários de teste e na obtenção e análise de resultados. Toda a parametrização definida pelo usuário fica registrada na base de dados.

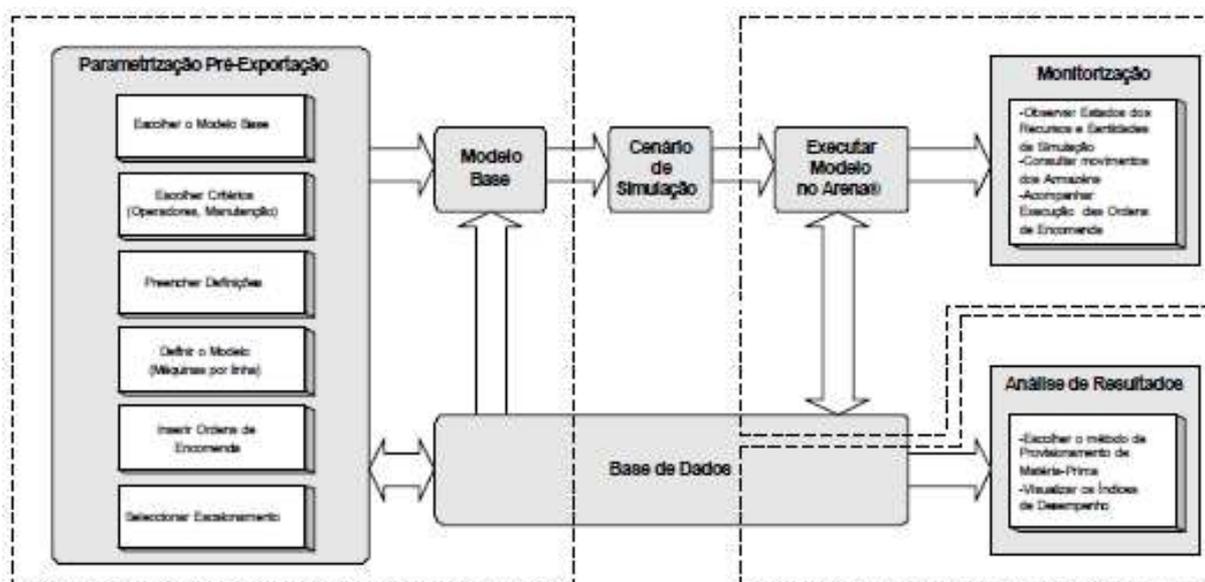


Figura 10 - Fluxo de informação do sistema proposto (PAIVA, 2005 p. 111)

As três fases que integram o sistema são: Parametrização, Monitoração e Análise dos resultados.

O processo inicia-se com a introdução das definições para o novo projeto, que interage com a base de dados, para consulta dos valores existentes e arquivo das alterações inseridas pelo usuário.

Depois que a definição e o novo cenário estiver no GAMSTAF, incluindo a seleção do modelo-base, ativa-se o procedimento de exportação.

Ao término da exportação, a simulação está apta para ser executada. No próprio GAMSTAF pode-se acompanhar a simulação, observando as alterações geradas e as mudanças provocadas no sistema.

Ao final da simulação, a análise de resultados possibilita efetuar um estudo sobre o comportamento do sistema.

Ao final da pesquisa, o autor conclui que o sistema tem capacidade de gerar modelos automaticamente possuindo forte componente visual, possibilitando ao usuário uma melhor compreensão e validação do modelo. As estratégias de base foram elaboradas para terem um caráter genérico e flexível do ponto de vista da utilização.

Entre as contribuições da pesquisa em relação ao GAMSTAF, o autor cita:

- A importância da técnica de simulação computacional para a resolução de problemas em que o controle, a parametrização do sistema, flexibilidade das políticas e definições de recursos são pressupostos básicos para uma representação fidedigna do mundo real;
- Entendimentos de como a construção de modelos de simulação podem contribuir para análise de fenômenos, ou seja, evidenciar o papel da simulação nos processos de tomada de decisão.

2.4.3 Comparativo entre as abordagens apresentadas

Na revisão bibliográfica foram apresentados os conceitos de VIS e VIM, além das abordagens de orientação a objetos e ambiente de simulação automático.

Wagner et al. (1996) já dizia que o uso dos termos VIS e VIM causam confusão quanto ao ambiente de simulação, e propôs um novo conceito VISM (Visual Interactive Simulation and Modeling), que é a unificação de VIS e VIM.

Algumas ferramentas, como Arena e ASDA, trazem esta abordagem, pois oferecem recursos visuais de interação no modelo (VIS), como também auxiliam na construção do modelo, através de coleções de objetos e ferramentas (VIM).

Outras ferramentas, como o Micro Saint, utilizam somente a abordagem VIS, transferindo para o usuário a tarefa de construir o modelo, através de uma linguagem de programação.

Sendo assim, a diferença entre VIS e VIM é a função e o modo de ação das teorias. Enquanto a VIS está voltada a interação gráfica e manipulação do modelo no momento da simulação, a VIM está para a construção do modelo, utilizando uma ferramenta amigável e composta por recursos gráficos.

Já as abordagens simulação orientada a objetos e ambiente de simulação automático se complementam, pois a primeira é uma forma de construção de modelos onde ganhos com a utilização dos objetos, classes e heranças que pode ser utilizada para construir ambientes de simulação automáticos. O exemplo desta união é o trabalho de Bruschi (2003 p. 182) que cita na especificação do ASDA "como a especificação do modelo foi realizada utilizando conceitos de orientação a objetos, esta

apresenta uma maneira mais fácil de acrescentar novas classes que representem outras formas de modelagem”.

A abordagem desenvolvida nesta pesquisa se difere das demais abordagens de ambiente de simulação automático, pois trabalha em outro nível de automatização. Essa abordagem foi construída para laboratórios de análises químicas, onde o usuário final somente manipula os templates que contém todos os parâmetros, não interagindo na construção dos modelos. As demais abordagens são genéricas, onde a codificação do modelo é realizada de forma automática, mas com interação do usuário nesse processo.

Por fim, para que a abordagem desenvolvida nesta pesquisa seja aplicada em um ambiente diferente de um laboratório de análises químicas, é necessária a presença de um especialista para alterar as regras de modelagem dentro do Gerador Automático de Modelos.

3 MÉTODO

Neste capítulo é apresentado o método utilizado durante a pesquisa. Antes, porém, é necessário entender melhor a tarefa da ciência. Segundo VIRIEUX-REYMOND (1972) *apud* BRUYNE et al (1991), “a tarefa de uma ciência é procurar os invariantes e as leis que regem fenômenos e fatos”. A operacionalização da ciência se dá pela metodologia.

O objetivo da metodologia é o de esclarecer a unidade subjacente a uma multiplicidade de procedimentos científicos particulares, ela ajuda a desimpedir os caminhos da prática concreta da pesquisa dos obstáculos que esta encontra. Ela não pretende refletir a progressão concreta de cada pesquisa particular, pois esta é eminentemente variável, mas quer se pensar em sua própria progressão e em suas relações com os procedimentos concretos da prática científica (BRUYNE, et al., 1991 p. 27).

Antes de determinar a metodologia de uma pesquisa, é preciso situar o lugar do campo de pesquisa.

O campo de pesquisa, concebido como lugar efetivo do trabalho dos pesquisadores, é essencialmente o lugar dinâmico e dialético no qual elabora uma prática científica que constrói objetos de conhecimentos específicos, os quais impõem, por sua vez, sua matriz particular de apreensão e de interpretação dos fenômenos (BRUYNE, et al., 1991 p. 28).

Portanto, o campo de pesquisa é o lugar prático de construção do conhecimento científico, sistêmico e da análise empírica dos fatos resultantes da pesquisa.

Segundo BRUYNE et al (1991), a metodologia deve não só explicar os produtos resultantes da pesquisa como também todo o processo que levou a estes produtos.

“A pesquisa científica se constrói por referência com o mundo dos acontecimentos, que são apenas os efeitos de estados de coisas presumidos, de uma realidade suposta” (BRUYNE, et al., 1991 p. 201).

A pesquisa coletará dados em função dos quais elaborará seus fatos. As informações são os acontecimentos, fenômenos significantes antes mesmo da pesquisa. A informação torna-se dado pela própria aplicação das técnicas de coleta. Desta maneira, os fatos científicos são construídos pelas técnicas que os coletaram, contextualizada pelo sistema teórico (BRUYNE, et al., 1991).

A coleta de dados se baseia em um conjunto de técnicas, cada técnica adequada a determinado tipo de pesquisa.

3.1 Método de Pesquisa

Para Silva e Menezes (2001), a investigação científica depende de um conjunto de procedimentos intelectuais e técnicos para que seus objetivos sejam atingidos: os métodos científicos. Método científico é o conjunto de processos ou operações mentais que se devem empregar na investigação. É a linha de raciocínio adotada no processo de pesquisa. Para Cervo e Bervian (2002), entende-se por método o conjunto de processos empregados na investigação e na demonstração da verdade.

A pesquisa segue o método hipotético-dedutivo. Para Popper *apud* Lakatos e Marconi (1990 p. 93),

o método científico parte de um *problema* (P_1) ao qual se oferece uma espécie de solução provisória, uma *teoria-tentativa* (TT), passando-se depois a criticar a solução, com vista à *eliminação do erro* (EE) e, tal como no caso da dialética, esse processo se renovaria a si mesmo, dando origem a *novos problemas* (P_2).

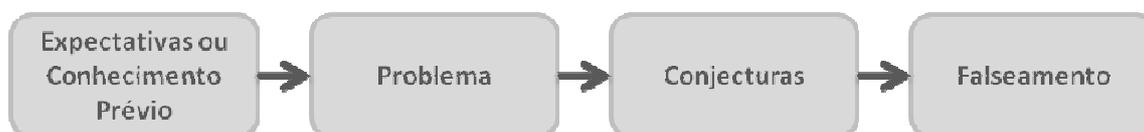


Figura 11 - Esquema do Método Hipotético-Dedutivo. Adaptado (LAKATOS e MARCONI, 1990)

A Figura 11 apresenta as etapas do processo investigatório de Popper. O processo inicia com o problema. O problema surge de conflitos frente a expectativas e teorias existentes. A conjuntura é justamente a proposta (nova teoria) em forma de proposições passíveis de teste. O teste de falseamento é a tentativa de refutar a nova proposta, por meio de observação e experimentação, entre outros (LAKATOS e MARCONI, 1990).

Segundo Silva e Menezes(2001), em relação à finalidade, pode-se concluir que, do ponto de vista da sua natureza, este trabalho é uma Pesquisa Aplicada, que objetiva gerar conhecimentos para aplicação prática dirigidos à solução de problemas específicos, envolvendo verdades e interesses locais.

Quanto à natureza, a pesquisa pode constituir-se em um trabalho científico original ou em resumo de assunto. Por trabalho científico original entende-se uma pesquisa realizada pela primeira vez, que venha a contribuir com novas conquistas e descobertas para a evolução do conhecimento científico. Naturalmente, esse tipo de pesquisa é desenvolvido por cientistas e especialistas em determinada área de assunto (ANDRADE, 2006 p. 123).

A forma de abordagem do problema será através de uma Pesquisa Qualitativa, que considera que há uma relação dinâmica entre o mundo real e o sujeito, isto é, um vínculo indissociável entre o mundo objetivo e a subjetividade do sujeito que não pode ser traduzido em números. A interpretação dos fenômenos e a atribuição de significados são básicas no processo de pesquisa qualitativa. Não requer o uso de métodos e técnicas estatísticas. O ambiente natural é a fonte direta para coleta de

dados, e o pesquisador é o instrumento-chave. É descritiva. Os pesquisadores tendem a analisar seus dados indutivamente. O processo e seu significado são os focos principais de abordagem.

A classificação, baseada nos objetivos gerais da pesquisa, pode ser apontada como pesquisa exploratória. Esta classificação se dá pelo fato de a pesquisa aprimorar idéias e descobrir intuições (GIL, 1996).

Na maioria dos casos essas pesquisas envolvem: a) levantamento bibliográfico; b) entrevistas com pessoas que tiveram experiências práticas com o problema pesquisado; e c) análise de exemplos que estimulem a compreensão (GIL, 1996 p. 45).

Para ANDRADE (2006 p. 124), a pesquisa exploratória proporciona “maiores informações sobre determinado assunto; facilitar a delimitação de um tema de trabalho”.

Segundo Gil (2007), classificar a pesquisa é útil para estabelecer o marco teórico, conceitual do estudo. Entretanto para fazer o cruzamento dos dados empíricos com a teoria, torna-se necessário traçar um modelo conceitual e operativo da pesquisa. Este delineamento expressa a idéia do modelo, sinopse e plano.

O delineamento refere-se ao planejamento da pesquisa em sua dimensão mais ampla, que envolve tanto a diagramação quanto a previsão de análise e interpretação de coleta de dados. Entre outros aspectos, o delineamento considera o ambiente em que são coletados os dados e as formas de controle das variáveis envolvidas (GIL, 2007 p. 43).

Desta forma, torna-se possível classificar as pesquisas segundo o seu delineamento. Gil(2007) frisa que o procedimento da coleta de dados é o elemento mais importante para identificação do delineamento.

A pesquisa bibliográfica será constituída de livros e artigos científicos. Nesta pesquisa serão utilizadas várias fontes bibliográficas, entre elas, os livros de referên-

cia informativa⁶. Outra fonte bibliográfica utilizada são as publicações periódicas. As principais publicações periódicas são os jornais e revistas, que representam nos tempos atuais uma das mais importantes fontes bibliográficas (GIL, 2007).

A vantagem da pesquisa bibliográfica é o acesso às informações mais amplas do que se poderia pesquisar diretamente. Entretanto, se as fontes secundárias apresentarem dados coletados ou processados de forma equivocada, pode comprometer a qualidade da pesquisa.

3.2 Método de Trabalho

A seguir serão detalhadas as etapas do método de trabalho:

- Identificação do tópico de pesquisa: Consiste em identificar e delimitar o escopo da pesquisa, justificando o motivo para desenvolver o trabalho;
- Revisão Bibliográfica: Após identificar a pesquisa será construído o referencial teórico, focado no tema da pesquisa;
- Estudo de campo: Será realizado um estudo de campo onde serão levantados os requisitos para a elaboração dos templates, assim como o gerador automático de modelos.
- Construção dos templates: Nesta etapa serão construídos os templates que serão utilizados durante o projeto de simulação computacional;

⁶ Livros de referência informativa são aqueles que contêm a informação que se busca, diferentemente dos livros de referência remissiva, que remetem a outras fontes.

- Construção do Gerador Automático de Modelos: Esta etapa do trabalho contempla a construção da ferramenta, que também será utilizada na construção da simulação computacional;
- Desenvolvimento de um projeto de simulação computacional: Nesta etapa será construído um modelo utilizando as interfaces e o Gerador Automático de Modelos, buscando responder às questões de pesquisa;
- Validação da abordagem proposta: Após a construção da simulação computacional, a abordagem proposta será validada junto aos participantes da simulação computacional.

4 ESTUDO DE CASO

Neste capítulo é apresentado o estudo que aborda o uso de templates para construção de um modelo de simulação. Inicialmente é retratado o contexto onde o caso foi aplicado, detalhando a realidade de um laboratório de análises químicas. Será apresentado o contexto dos dois projetos anteriores realizados no laboratório de análises químicas, detalhando a condução do projeto, a construção dos modelos e evolução da modelagem em cada um dos projetos.

Após será descrita a aplicação do uso dos templates, com a descrição do projeto, o desenvolvimento dos templates e a construção e aplicação do gerador automático de modelos.

4.1 O Contexto do Caso

Nesta seção será apresentado o contexto em que a pesquisa está envolvida. Esta pesquisa foi desenvolvida em um laboratório de análises químicas, que é um órgão de apoio dentro do processo produtivo, e está subordinado ao gerenciamento do processo mais amplo, tendo que se adequar às decisões tomadas. Neste laboratório já houve outros projetos de simulação. O fato de já ter realizado outros proje-

tos de simulação computacional fez com que o Laboratório procurasse novamente auxílio para desenvolver este projeto.

Além disso, a escolha por um laboratório de análises químicas se deve à complexidade, características e regras presentes no dia-a-dia de um laboratório.

As amostras para análises chegam ao Laboratório para serem testadas. Estas amostras são de rotina, ou seja, elas têm periodicidade para serem testadas e fazem parte do trabalho normal do Laboratório e amostras de outras finalidades, sendo que as amostras chegam sem uma regra definida de horário.

Estas amostras não rotineiras acabam por estragar o gerenciamento e seqüenciamento dos trabalhos dentro do Laboratório, pois na maioria das vezes este só tem conhecimento da necessidade de testar estas amostras no momento em que elas chegam. Como não é o Laboratório quem define as prioridades dos testes, um novo seqüenciamento e alocação de recursos devem ser realizados praticamente a cada novo turno.

Para cada amostra o Laboratório deve fazer um ou mais testes. Dependendo da sua finalidade (vendas, rotinas, por exemplo), o teste tem mais prioridade para ser realizado. A Figura 12 exemplifica o fluxo das amostras no Laboratório.

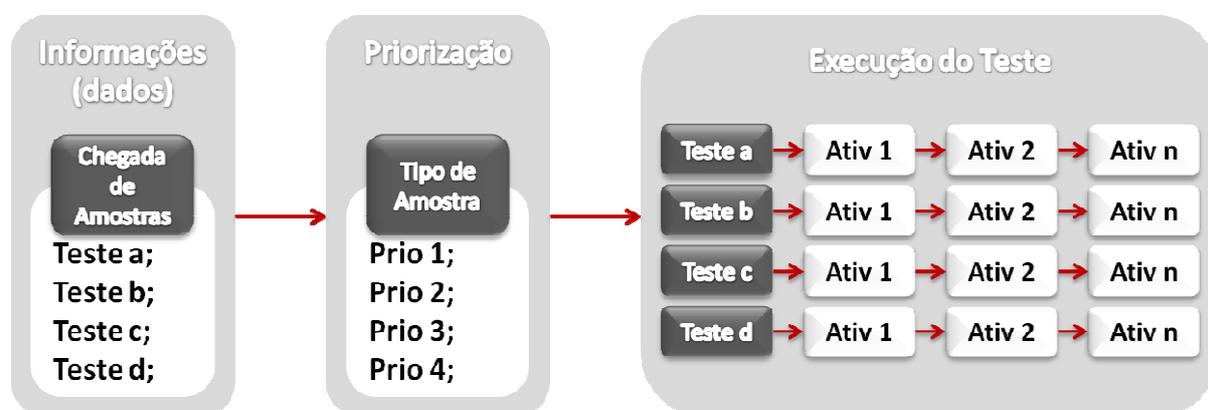


Figura 12 - Exemplo do fluxo das amostras no laboratório (Elaborada pelo Autor)

Cada teste também tem um tempo máximo para a sua realização. Quanto mais próximo do prazo máximo, maior prioridade o teste terá para ser realizado.

Cada teste tem uma ou mais atividades, conforme exemplificado na Figura 13:

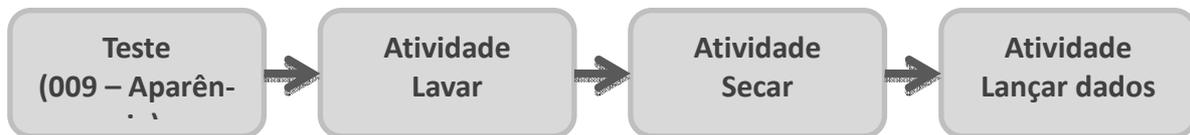


Figura 13 - Exemplo de um teste (Elaborada pelo Autor)

Cada atividade pode usar recursos e técnicos com tempo distinto para cada atividade. O número de técnicos é definido pelo número de amostras que deverão ser testadas no turno. Pela volatilidade do seqüenciamento, a definição do número de técnicos é realizada durante o turno de trabalho, conforme o acúmulo de testes que estão atrasados.

Em alguns casos é solicitada ao Laboratório a previsão para a realização de um teste específico. Como ele não dispõe de uma ferramenta de seqüenciamento, não consegue prever com garantia o prazo da solicitação, tampouco quais outros testes serão prejudicados.

Para melhor distribuir o serviço no Laboratório, os testes são divididos por bancadas. A Figura 14 exemplifica uma bancada em um laboratório.

Para cada bancada há um conjunto de recursos e técnicos, que trabalham em turnos predefinidos. O balanceamento de trabalho também é prejudicado pela volatilidade do gerenciamento do laboratório, onde uma bancada acaba por ter mais trabalho que outra, por causa dos testes não previstos durante o turno.

Os técnicos são multifuncionais, ou seja, o técnico pode testar mais de um teste simultaneamente, respeitando um número máximo de testes sem que haja queda de produtividade. Em algumas atividades é obrigatória a presença do técnico

no final, como por exemplo, aquecer (se a amostra ficar no forno ela muda de propriedade).



Figura 14 - Exemplo de uma bancada

Existem dois tipos de técnicos no laboratório. O primeiro tipo é aquele que trabalha no turno administrativo. O turno administrativo trabalha somente no horário comercial. O segundo tipo de técnico é o que trabalha no turno denominado igualmente de Turno. Este técnico realiza jornadas que podem estar dentro do horário comercial como em jornadas fora do horário comercial. Ele segue uma legislação própria que, por exemplo, diz que o técnico que realiza hora-extra tenha no mínimo x horas de descanso até o próximo turno de trabalho.

Visto isso, o Laboratório tem carência em gerenciar os técnicos em função da demanda volátil, não conseguindo manter um nível de atendimento esperado.

4.2 As Aplicações Anteriores ao uso dos Templates

Neste capítulo serão resgatados os dois projetos de simulação realizados no Laboratório que antecederam esta pesquisa, que subsidiaram o desenvolvimento da abordagem para projetos de simulação repetitivos.

4.2.1 Primeiro Projeto

Em 2003, foi realizado o primeiro estudo com o uso da simulação computacional no Laboratório de análises químicas. O objetivo do estudo foi a construção de um modelo sistêmico, integrado e agregado do fluxo de processos do Laboratório, visando determinar o dimensionamento da capacidade produtiva necessária frente às novas demandas da ampliação das novas unidades de produção. Ou seja, utilizou-se a simulação computacional como ferramenta para responder de forma objetiva a algumas questões relacionadas à capacidade produtiva, uma vez que a demanda por testes no Laboratório iria aumentar em decorrência da expansão da capacidade no processo produtivo.

A questão da agregação do modelo estava relacionada ao tempo disponível para a construção do mesmo. Assim sendo, buscou-se o desenvolvimento de modelos simplificados da realidade, mas que não afetem sobremaneira a representatividade da realidade.

Nesta simulação procurou-se desenvolver um modelo de simulação computacional que retratasse a realidade atual do Laboratório, o qual deveria possibilitar o entendimento sistêmico das relações dos atuais processos do Laboratório, a parame-

trização dos recursos existentes, em termos da sua capacidade disponível em relação à quantidade, turnos de trabalho, etc., a ampliação e eliminação de análises e seus impactos na capacidade produtiva, avaliar as atuais e novas políticas de gerenciamento da capacidade do Laboratório, avaliar a viabilidade econômica da utilização de uma ferramenta de Planejamento e Programação das atividades desenvolvidas pelo Laboratório, objetivando a melhoria do seqüenciamento dos recursos envolvidos e capacitação básica para um grupo de funcionários na operacionalização dos modelos de simulação computacional desenvolvidos. A Figura 15 exemplifica o escopo do primeiro projeto de simulação computacional desenvolvido no Laboratório.



Figura 15 - Escopo do 1º estudo de simulação (Elaborada pelo Autor)

Durante o projeto, percebeu-se que algumas características básicas do laboratório haviam sido compreendidas de forma errada, refletindo no resultado da simu-

lação. Assim, o resultado do projeto teve que relevar estas questões, para que tivesse aceitação final.

4.2.2 Segundo Projeto

No ano de 2005, dois anos após a conclusão do primeiro projeto de simulação, surgiu a necessidade de uma atualização do modelo frente à concretização da ampliação da capacidade simulada em 2003.

Assim, o Laboratório chamou novamente a equipe do primeiro projeto para executar este segundo projeto. Alguns do primeiro projeto não participaram do segundo, como também outros que não haviam participado da primeira edição colaboraram neste projeto.

O segundo projeto de simulação computacional no Laboratório teve como objetivo a reanálise do modelo sistêmico, integrado e agregado do fluxo de processos do Laboratório, visando agregar ao modelo as modificações dos processos, determinando o dimensionamento da capacidade produtiva necessária frente às novas demandas da ampliação das novas unidades de produção. Ou seja, uma simulação semelhante à do primeiro projeto, atualizando regras, adicionando e removendo testes, redistribuindo testes por bancadas e atualizando os recursos humanos e materiais.

Nota-se que ainda na segunda versão os processos foram analisados de forma integrada e agregada com algumas simplificações da realidade.

O escopo do trabalho era idêntico ao realizado dois anos antes. A atualização do modelo deveria também trazer melhorias, entre elas, apresentar novos indicado-

res de desempenho, como utilização dos recursos do Laboratório, lead time dos testes realizados, atendimento dos prazos estipulados, número de testes na fila e tempo médio de espera do teste na fila.

Para executar este projeto, foi utilizada metodologia apresentada na Figura 16:

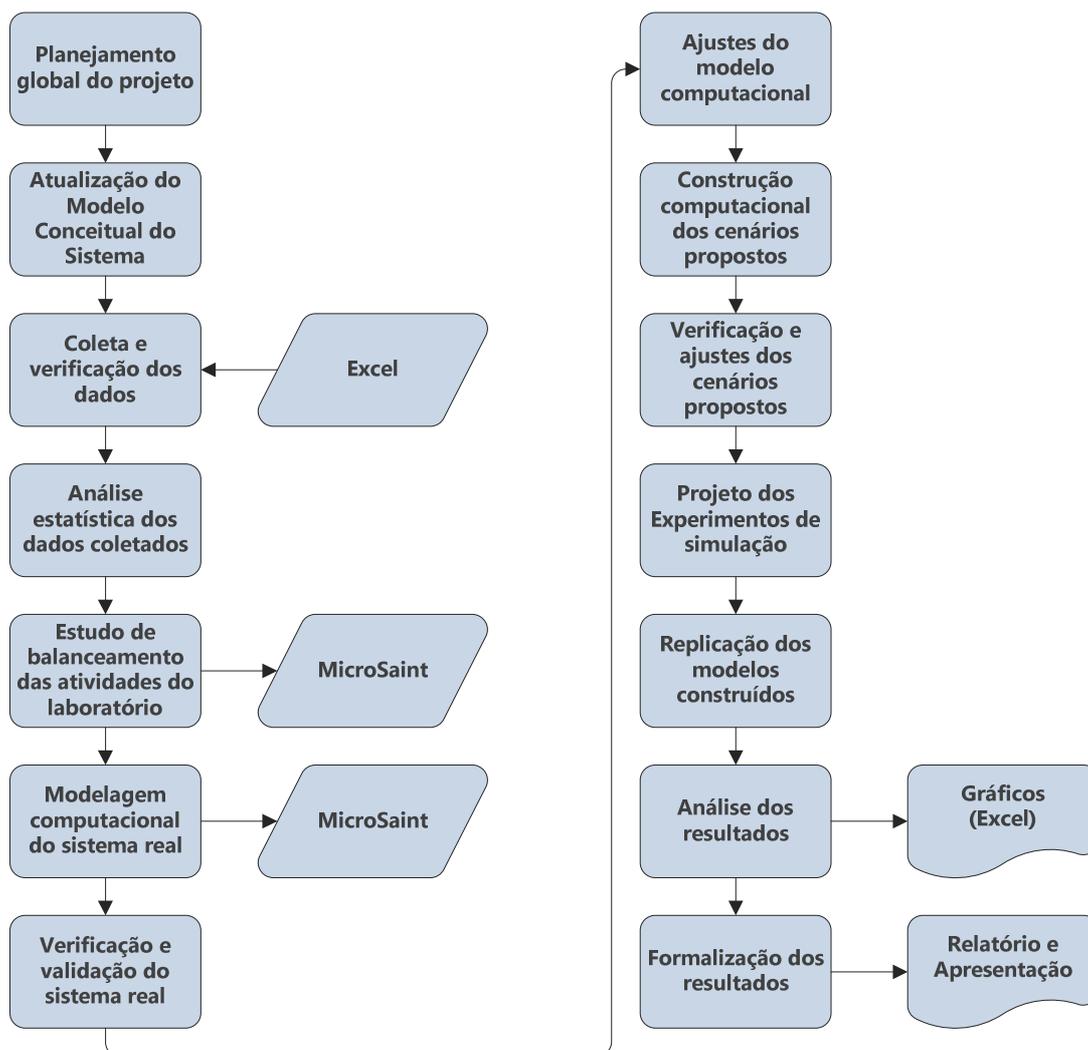


Figura 16 - Metodologia utilizada no 2º projeto (Elaborada pelo Autor)

No planejamento global do projeto, foi decidido que o modelo computacional seria refeito a partir do zero, sendo possível aproveitar somente os conceitos e regras gerais da simulação realizada em 2003.

Assim, a estrutura-base do modelo não sofreu alteração em relação ao modelo passado, onde cada bancada foi representada isoladamente, conforme o layout físico do Laboratório. As atualizações contemplaram a exclusão e adição de testes por bancada, atualização dos recursos por teste/bancada, mudança nas demandas dos testes, alteração das configurações dos técnicos, etc.

Os dados coletados já concretizavam o aumento da capacidade que havia sido do foco do primeiro projeto de simulação. Estes dados foram disponibilizados por meio de formulários impressos.

Após a digitação manual dos dados de entrada, foi realizado o tratamento estatístico dos mesmos.

Nesta simulação averiguou-se o balanceamento da capacidade do Laboratório. Para tanto, foi adicionada uma melhoria frente ao modelo anterior em relação à configuração dos técnicos. Nesta modelagem eles ganharam uma característica de multifuncionalidade, ou seja, alguns técnicos predefinidos poderiam “flutuar” entre as bancadas, ajudando assim os técnicos fixos por bancada.

Os dados de demanda seguiram um padrão de layout, onde foi possível automatizar parcialmente a inserção das informações no modelo, através de macros construídas na linguagem Microsoft Visual Basic nas planilhas em Microsoft Excel. Esta automação parcial trouxe ao modelo uma flexibilidade, pois com poucos passos era possível atualizar a tabela de demandas. A Tabela 4 exemplifica o layout de dados utilizados na automatização parcial dos dados de demanda.

Tabela 4 - Layout dos dados de demanda

Código Produto	Produto	Chegada	Data Fim análise	Bancada	TQP	Teste	Número Teste
R9	Produto A	125	350	4	2	Teste 1A	10
R9	Produto A	125	350	3	0	Teste 2A	50
R9	Produto A	125	350	3	0	Teste 3A	20
R9	Produto A	125	350	1	1	Teste 3A	20
R9	Produto A	125	350	3	2	Teste 4A	60
R9	Produto A	125	350	1	1	Teste 4A	60
R9	Produto A	125	350	2	0	Teste 5A	14
R9	Produto A	125	350	2	2	Teste 6A	43
R9	Produto A	125	350	1	1	Teste 6A	43
T6	Produto B	220	600	2	1	Teste 1B	98
T6	Produto B	220	600	4	1	Teste 2B	45
T6	Produto B	220	600	2	2	Teste 3B	76
T6	Produto B	220	600	1	0	Teste 4B	22
T6	Produto B	220	600	2	1	Teste 5B	18
T6	Produto B	220	600	2	0	Teste 6B	82

Fonte: Autor

A macro desenvolvida codificava as informações da planilha para a linguagem do Micro Saint. O código gerado a partir de cada linha está exemplificado na Figura 17. Este código é inserido manualmente no arquivo principal da modelagem.

```

510
3
0
0
1.000000
1
1000.000000
start(4,cont);tag:=cont;cont+=1;
produto[tag] := 220;
ensaio[tag] := 41;
bancada[tag] := 4;
tipo[tag] := 2;
---
++++

```

Figura 17 - Exemplo de código gerado a partir da demanda (Elaborada pelo Autor)

Apesar de a geração da demanda ser era codificada automaticamente, ela não estava integrada ao modelo.

As informações exemplificadas na Figura 17 representam códigos da linguagem interna do Micro Saint e as regras de modelagem. Um exemplo das regras de

modelagem pode ser visto no trecho que inicia em "start(4,cont); tag:=cont+=1;" e vai até "tipo[tag] := 2;". Para cada linha da Tabela 4 é gerado um conjunto de informações semelhantes ao apresentado acima.

Salienta-se que no planejamento deste segundo projeto de simulação ficou definido que seria aproveitado somente o conceito da primeira modelagem, sendo construído um novo modelo para esta segunda versão. Foi definido assim, pois o trabalho manual de atualização do modelo era tão complexo quanto construir novamente o modelo.

A complexidade do modelo se devia ao número de interações manuais necessárias para atualização ou construção do modelo. Pelas características do Laboratório que na época realiza 150 testes e para cada teste uma média de 10 operações, isso representava editar 1500 atividades manualmente no modelo. Além deste trabalho manual de edição das atividades, há a complicação das regras específicas para cada uma das atividades, que requerem não só trabalho manual, mas também inteligência de programação das regras de negócio do Laboratório.

Essa particularidade do modelo engessava qualquer intenção de flexibilizar a modelagem se aplicado nesta concepção de modelagem.

Da mesma forma que a primeira versão construída em 2003, este projeto teve alguns conceitos mal compreendidos, refletindo no resultado final da simulação.

Graças à realização desses dois estudos, foi possível avançar na concepção do terceiro modelo que será construído, utilizando novas características que buscarão facilitar a usabilidade do modelo, prolongando sua vida útil.

4.3 A Aplicação com o uso dos Templates

Nesta seção será detalhada a condução do projeto de simulação computacional que utilizou a abordagem proposta na pesquisa.

4.3.1 Descrição do Projeto

Em vista dos projetos passados, alguns pressupostos foram elencados para este projeto. O modelo deveria ter uma estrutura física mais limpa e clara que as versões anteriores, com mecanismos de entrada de novos dados de demanda, assim como possibilidade de edição do modelo no que tange a:

- Adicionar, editar e remover testes;
- Adicionar e remover bancadas;
- Alterar o número de técnicos e recursos;
- Alterar as bancadas de teste.

Entre as diferenças das regras de modelagem desta abordagem para a anterior desenvolvida no Laboratório, pode-se citar:

	Versão 2005	Versão 2008
Input do Modelo	Chegada de testes	Chegada de amostras, onde cada amostra é um conjunto de testes
Leadtime máximo por amostra/teste	Não considera	A simulação deverá respeitar o lead time máximo cadastrado por amostra/teste. Serão informados os testes "mandatórios" da amostra.
Horário Input	No momento do início	Os testes entrarão no modelo x ho-

	do teste	ras antes do horário de início
Gerenciamento dos Técnicos	Técnicos por bancada + Técnico Flutuante	Técnico Totalmente Flutuante
Horário de Trabalho	Técnico trabalha (se necessário) nos intervalos	Técnico não trabalha nos intervalos
Estrutura "física" da modelagem	Testes separados por bancada	Lista com todos os testes

Quadro 1 - Diferenças nas versões de modelos desenvolvidos (Elaborado pelo autor)

A diferença na concepção da modelagem inicia-se pela entrada dos dados. Nesta pesquisa as informações que alimentarão a simulação são as amostras e não mais os testes individuais. Uma vez no modelo, as amostras dispararão o conjunto de testes que deverão ser executados. O benefício desta característica é um melhor controle sobre o tempo máximo que cada amostra pode permanecer dentro da simulação, por questões de prioridade e cumprimento do prazo. Nas versões anteriores não era observado o tempo de permanência máximo de cada amostra na simulação.

Outra diferença é em relação ao momento em que a amostra é inserida na simulação. No modelo desta pesquisa foram consideradas várias regras de permanência dos técnicos no final do turno de trabalho (hora-extra). Para tanto, foi necessário inserir a demanda de amostras um turno antes de realmente os testes serem processados. Isto serviu para calcular o número de técnicos necessários para processar todos os testes, dentro do limite máximo de permanência de cada amostra na simulação.

Em relação aos técnicos, optou-se por não fixá-los nas bancadas, diferentemente das versões anteriores, onde os técnicos pertenciam a uma bancada específica. Isto possibilitou avaliar em quais momentos as bancadas necessitam mais técnicos.

Ainda em relação aos técnicos, nesta versão o tempo de intervalo deles impacta diretamente no atendimento das amostras.

Outra característica que difere dos projetos passados é em relação à disposição das bancadas na simulação. Nas duas primeiras versões levou-se em conta o local físico das bancadas. A alteração para uma lista de testes, onde a configuração da bancada está inserida nas regras de todos os testes, serviu para simplificar a sua construção e facilitar a manutenção dos mesmos.

Partindo dessas informações, buscaram-se alternativas para contemplar os requisitos do modelo de simulação.

A alternativa escolhida foi desenvolver templates, que serão apresentados na seção 4.3.2.

4.3.2 Os Templates Desenvolvidos

Para tanto, foram definidos doze templates (arquivos com os modelos padrão de dados) que detêm as informações necessárias para a construção computacional do modelo de simulação, exemplificados na Figura 18.

Dos estudos anteriores pouca coisa foi aproveitada. Do segundo estudo realizado aproveitou-se a idéia da alimentação automática da demanda, que consiste em cadastrar o histórico da demanda dos testes para alimentar a simulação.

O Laboratório mantém um histórico dos testes realizados em uma base de dados. Este histórico serve como dados de entrada para a simulação. Portanto, foi construída uma interface para que esta informação fosse lida no simulador de forma automática.



Figura 18 - Templates desenvolvidos (Elaborado pelo Autor)

O arquivo de tem esta informação é o "eventos.csv". Este arquivo tem a demanda de testes que o modelo irá executar. As informações do arquivo "eventos.csv", exemplificadas na Figura 19, são:

- Código Amostra: código da amostra que será testada;
- Código Produto: código do produto que será testado;
- Chegada: informação do tempo de início do teste na simulação;
- Saída Max do teste: tempo máximo de permanência do teste na simulação;
- Bancada: informação do local físico em que será realizado o teste;
- Teste Mandatório: informação se o teste é mandatório dentre os testes da amostra;

- Finalidade: finalidade para o qual o teste está sendo executado. Serve para determinar a prioridade do teste, se houver concorrência por falta de capacidade;
- Teste (cod): número (código) do teste a ser executado.

	A	B	C	D	E	F	G	H
1	Código Amostra	Código Produto	Chegada	Saída max do teste	Bancada	Teste Mandatário	Finalidade	Teste (cod)
2	610	88601I	100	200	AGUAS	N	1	2
3	610	88601I	100	201	AGUAS	N	1	2
4	610	88601I	100	202	AGUAS	N	1	2
5	610	88601I	100	203	AGUAS	N	1	2
6	610	88601I	100	204	AGUAS	N	1	2
7	610	88601I	100	205	AGUAS	N	1	2
8	610	88601I	100	206	AGUAS	N	1	2
9	610	88601I	100	207	AGUAS	N	1	2
10	610	88601I	100	208	AGUAS	N	1	2
11	610	88616C	100	209	AGUAS	N	1	2
12	610	88616C	100	210	AGUAS	N	1	2
13	610	88616C	100	211	AGUAS	N	1	2
14	610	88616C	100	212	AGUAS	N	1	2
15	610	88616C	100	213	AGUAS	N	1	2
16	610	88616C	100	214	AGUAS	N	1	2
17								

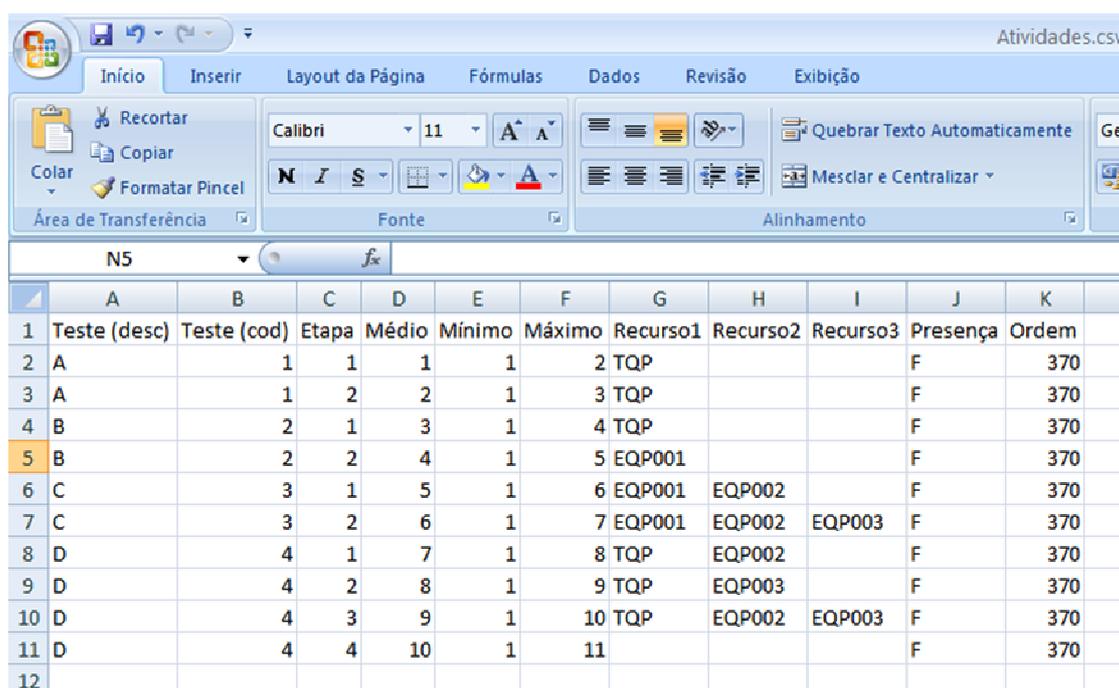
Figura 19 - Template Eventos (Elaborada pelo Autor)

O Laboratório dispõe de uma lista de testes, atividades, recursos e tempos de execução das atividades, que pode ser exportada do sistema de gestão. Assim foi definida a segunda interface, com toda a malha de atividades da simulação.

O arquivo "atividades.csv" contém as informações do fluxo e relacionamento das atividades no modelo. As informações do arquivo "atividades.csv", exemplificadas na Figura 20, são:

- Teste (desc): nome do teste;
- Teste (cod): número do teste;

- Etapa: nome das etapas que contêm o teste;
- Médio: tempo médio de execução de cada etapa⁷;
- Mínimo: tempo mínimo de execução de cada etapa;
- Máximo: tempo máximo de execução de cada etapa;
- Recurso1: nome do recurso necessário para executar a etapa;
- Recurso2: nome do recurso necessário para executar a etapa;
- Recurso3: nome do recurso necessário para executar a etapa;
- Presença: algumas atividades não necessitam a presença do técnico durante a sua execução, como por exemplo, atividade que utiliza o forno. Entretanto esta atividade necessita que o técnico esteja presente no seu término, para que a amostra não se danifique. Neste caso, a atividade necessita de um técnico ao final da atividade.



	A	B	C	D	E	F	G	H	I	J	K
1	Teste (desc)	Teste (cod)	Etapa	Médio	Mínimo	Máximo	Recurso1	Recurso2	Recurso3	Presença	Ordem
2	A		1	1	1	2	TQP			F	370
3	A		1	2	2	3	TQP			F	370
4	B		2	1	3	4	TQP			F	370
5	B		2	2	4	5	EQP001			F	370
6	C		3	1	5	6	EQP001	EQP002		F	370
7	C		3	2	6	7	EQP001	EQP002	EQP003	F	370
8	D		4	1	7	8	TQP	EQP002		F	370
9	D		4	2	8	9	TQP	EQP003		F	370
10	D		4	3	9	10	TQP	EQP002	EQP003	F	370
11	D		4	4	10	11				F	370
12											

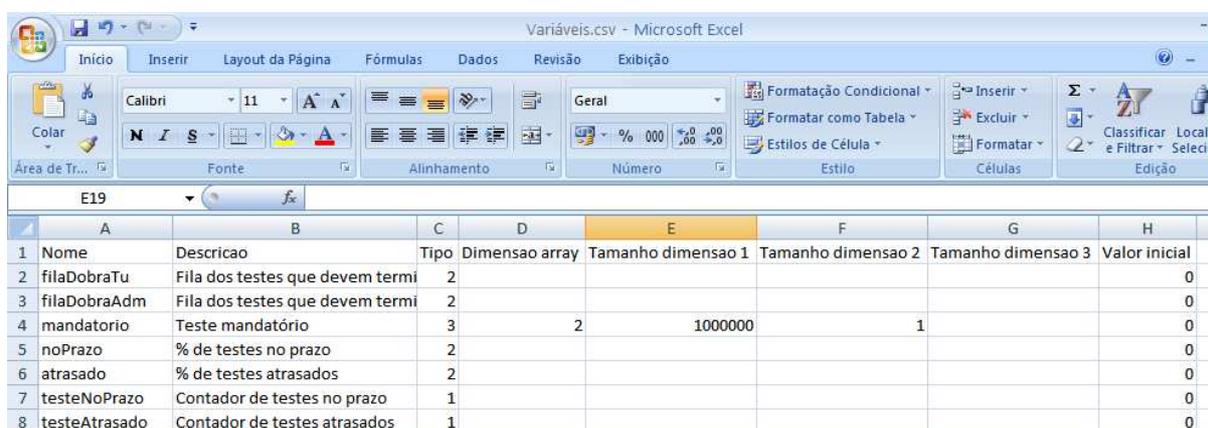
Figura 20 - Template Atividade (Elaborada pelo Autor)

⁷ Para os tempos de cada atividade, foi utilizada a distribuição triangular por não haver dados suficientes para um tratamento de dados.

Visando dar maior flexibilidade, as variáveis do modelo computacional são publicadas em um arquivo denominado "variáveis.csv". As informações deste arquivo, exemplificadas na Figura 21, são:

- Nome: nome da variável no formato que será importado para o modelo, com tamanho máximo de 12 caracteres;
- Descrição: descrição da variável;
- Tipo: existem 4 tipos de variáveis, sendo eles:
 1. Inteiro: números inteiros;
 2. Real: números reais;
 3. Array de inteiro: array de números inteiros;
 4. Array de real: array de números reais;
- Dimensão array: quando o tipo da variável for 3 ou 4, é necessário informar a dimensão da variável (pode variar de 1 a 3);
- Tamanho dimensão 1: tamanho da primeira dimensão;
- Tamanho dimensão 2: tamanho da segunda dimensão;
- Tamanho dimensão 3: tamanho da terceira dimensão;
- Valor inicial: o valor inicial que a variável assume no sistema.

Estas informações são alteradas manualmente, sempre que necessário.

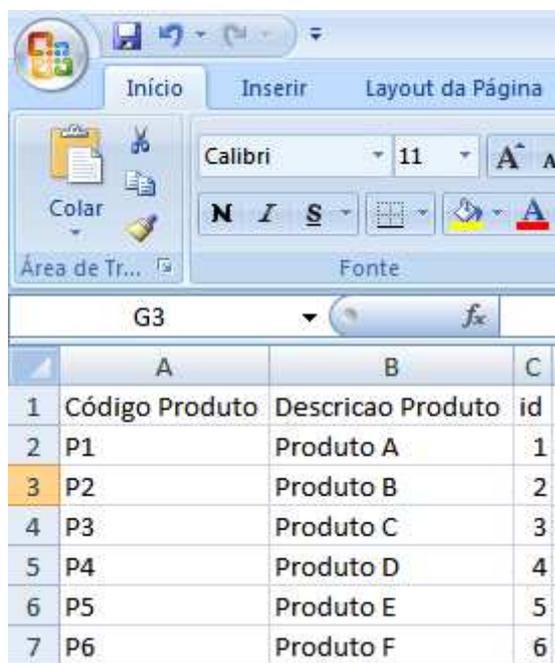


	A	B	C	D	E	F	G	H
	Nome	Descricao	Tipo	Dimensao array	Tamanho dimensao 1	Tamanho dimensao 2	Tamanho dimensao 3	Valor inicial
2	filaDobraTu	Fila dos testes que devem termi	2					0
3	filaDobraAdm	Fila dos testes que devem termi	2					0
4	mandatorio	Teste mandatório	3	2	1000000	1		0
5	noPrazo	% de testes no prazo	2					0
6	atrasado	% de testes atrasados	2					0
7	testeNoPrazo	Contador de testes no prazo	1					0
8	testeAtrasado	Contador de testes atrasados	1					0

Figura 21 - Template Variáveis (Elaborada pelo Autor)

Do sistema de gestão sai também a lista dos produtos. Para a simulação, os produtos ganham nova chave, pois os registros originais contêm números e letras que não são reconhecidos pelo Micro Saint. As informações deste arquivo, exemplificadas na Figura 22, são:

- Código Produto: código do produto no sistema corporativo da empresa;
- Descrição Produto: descrição do produto;
- Id: identificação para a simulação computacional.



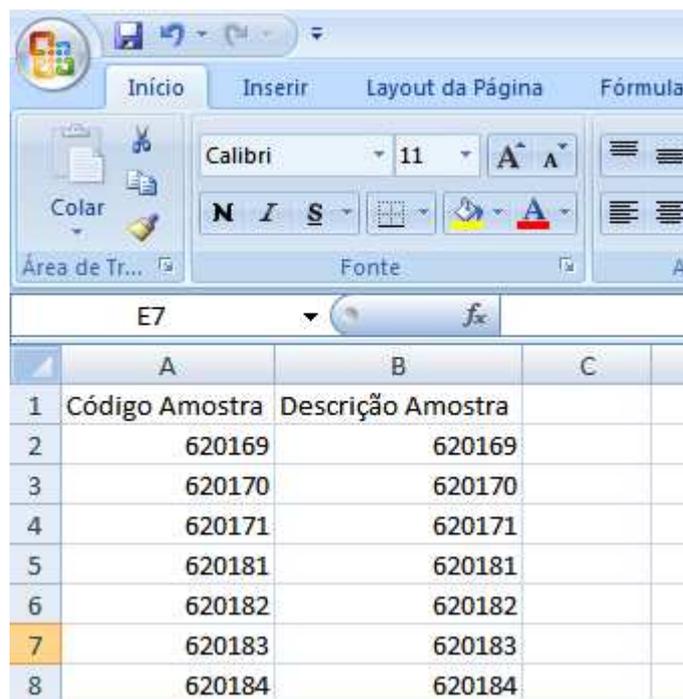
	A	B	C
1	Código Produto	Descricao Produto	id
2	P1	Produto A	1
3	P2	Produto B	2
4	P3	Produto C	3
5	P4	Produto D	4
6	P5	Produto E	5
7	P6	Produto F	6

Figura 22 - Template Produtos (Elaborada pelo Autor)

As informações sobre produtos são utilizadas principalmente nos relatórios.

Outra interface é a das amostras. As amostras são tratadas pela simulação pelo código. Para melhor identificá-las, esse arquivo traz a descrição das amostras. As informações deste arquivo, exemplificadas na Figura 23, são:

- Código Amostra: código da amostra;
- Descrição Amostra: descrição da amostra.



	A	B	C
1	Código Amostra	Descrição Amostra	
2	620169	620169	
3	620170	620170	
4	620171	620171	
5	620181	620181	
6	620182	620182	
7	620183	620183	
8	620184	620184	

Figura 23 - Template Amostras (Elaborada pelo Autor)

Os técnicos são configurados como recursos no modelo, com algumas peculiaridades. Existe diferenciação de calendário (número de turnos, intervalos, regras para hora-extra) e número de técnicos. Os templates são flexíveis ao ponto de aceitar tantas configurações de turno quantas forem necessárias. O arquivo recursotqp.csv indica o número de técnicos que estão entrando a cada novo turno. As informações deste arquivo, exemplificadas na Figura 24, são:

- Data: data e hora de início de cada turno;
- Dia Semana: dia da semana;
- Tempo: tempo em que os técnicos entrarão na simulação;
- TQP Turno: número de técnicos do turno "turno";
- TQP Turno-ADM: número de técnicos do turno "turno-administrativo";
- TQP ADM: número de técnicos do turno "administrativo".

	A	B	C	D	E	F
1	Data	Dia Semana	Tempo	TQP Turno	TQP Turno-ADM	TQP ADM
2	1/6/2007 00:00	sex	0	4	0	0
3	1/6/2007 08:00	sex	480	4	1	7
4	1/6/2007 16:00	sex	960	4	0	0
5	2/6/2007 00:00	sáb	1440	4	0	0
6	2/6/2007 08:00	sáb	1920	3	0	6
7	2/6/2007 16:00	sáb	2400	4	0	0
8	3/6/2007 00:00	dom	2880	4	0	0
9	3/6/2007 08:00	dom	3360	3	0	0
10	3/6/2007 16:00	dom	3840	4	0	0

Figura 24 - Template Recursos Técnicos (Elaborada pelo Autor)

O template dos recursos é semelhante ao recurso técnico. Entretanto preferiu-se construir dois templates, pois o template dos técnicos sofre mais atualizações que os recursos fixos, e as informações dos técnicos têm maior importância para os gestores do Laboratório.

As bancadas foram cadastradas em templates para facilitar a adição ou remoção das mesmas. As bancadas são configuradas no arquivo bancadas.csv. Cada bancada tem uma descrição, uma identificação numérica e um turno associado (turno previamente cadastrado no template dos turnos). As informações deste arquivo, exemplificadas na Figura 25, são:

- Bancada: nome da bancada;
- Número: número da bancada;
- Turno: turno em que a bancada trabalha.

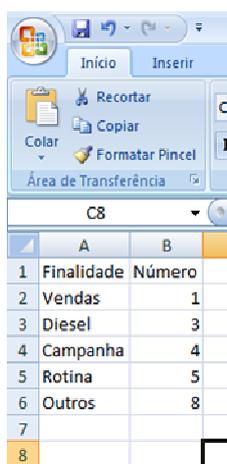


	A	B	C
1	Bancada	Número	Turno
2	BANCADA 1	1	ADM
3	BANCADA 2	2	ADM
4	BANCADA 3	3	ADM
5	BANCADA 4	4	TURNO
6	BANCADA 5	5	TURNO
7	BANCADA 6	6	TURNO
8	BANCADA 7	7	ADM
9	BANCADA 8	8	ADM
10	BANCADA 9	9	ADM
11	BANCADA 10	10	ADM
12	BANCADA 11	11	ADM
13	BANCADA 12	12	ADM
14	BANCADA 13	13	ADM
15	BANCADA 14	14	TURNO
16	BANCADA 15	15	TURNO
17	BANCADA 16	16	TURNO

Figura 25 - Template Bancadas (Elaborada pelo Autor)

O cadastro das finalidades dos testes segue a mesma idéia do cadastro das bancadas. Os testes do modelo terão prioridades diferentes. Estas prioridades são configuradas pela finalidade do teste. As informações deste arquivo, exemplificadas na Figura 26, são:

- Finalidade: descrição da finalidade do teste;
- Número: nível de importância da finalidade, sendo 1 mais importante.



	A	B
1	Finalidade	Número
2	Vendas	1
3	Diesel	3
4	Campanha	4
5	Rotina	5
6	Outros	8
7		
8		

Figura 26 - Template Finalidade (Elaborada pelo Autor)

Os turnos de trabalho dos técnicos são configurados no arquivo Configuração dos turnos.csv. O modelo suporta até 3 turnos. Cada turno pode ter uma parada, ou seja, além das paradas entre turnos, há também uma dentro do turno. Nesta simulação há duas configurações de turno: uma chamada de TURNO, que tem 3 turnos e outra chamada ADM, com dois turnos somente. As informações deste arquivo, exemplificadas na Figura 27, são:

- Descrição: nome do turno;
- Início: hora de início do turno 1;
- Parada: tempo da parada dentro do turno;
- Fim: hora do término do turno 1;
- Início: hora de início do turno 1;
- Parada: tempo da parada do turno;
- Fim: hora do término do turno 1;
- Início: hora de início do turno 2;
- Parada: tempo da parada dentro do turno;
- Fim: hora do término do turno 2;
- Início: hora de início do turno 2;
- Parada: tempo da parada do turno;
- Fim: hora do término do turno 2;
- Início: hora de início do turno 3;
- Parada: tempo da parada dentro do turno;
- Fim: hora do término do turno 3;
- Início: hora de início do turno 3;
- Parada: tempo da parada do turno;
- Fim: hora do término do turno 3.

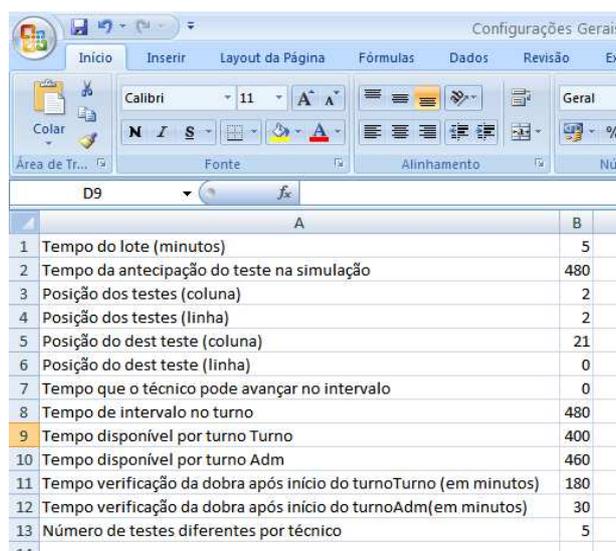
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	Descrição	Início	Parada	Fim															
2	Turno	00:00		04:00	04:00		07:20	08:00		12:00	12:40		15:20	16:00		20:00	20:40		23:20

Figura 27 - Template Configuração dos Turnos (Elaborada pelo Autor)

Além da configuração dos técnicos, bancadas, turnos, etc., há configurações gerais. Estas informações são características e regras que podem ser modificadas para gerar novos cenários. As informações deste arquivo, exemplificadas na Figura 28, são:

- Tempo do lote (minutos): minutos que o teste espera para acumular um lote de testes;
- Tempo da antecipação do teste na simulação: O número de técnicos necessários a cada turno é definido pelos testes que estão atrasados do turno anterior e por aqueles que irão chegar. Assim, esta informação serve para configurar o tempo de antecipação dos testes (que permanecerão em uma fila) para poder calcular a real necessidade de técnicos;
- Posição dos testes (coluna): cada atividade presente no Micro Saint possui uma localização (posição visual) da atividade. Esta informação servirá para posicionar os testes dentro do Micro Saint;
- Posição dos testes (linha): mesma funcionalidade explicada acima, mas para a configuração da linha, que será base para posicionar os testes dentro do Micro Saint;
- Posição do destino do teste (coluna): posição da atividade final da simulação, onde todos os testes deverão apontar;
- Posição do destino do teste (linha): posição da atividade final da simulação, onde todos os testes deverão apontar;

- Tempo que o técnico pode avançar no intervalo: Na simulação existe a possibilidade de liberar o técnico a trabalhar nos intervalos de parada. Este tempo é configurado em minutos;
- Tempo de intervalo no turno: periodicidade de início de cada turno;
- Tempo disponível por turno Turno: tempo disponível no turno "turno";
- Tempo disponível por turno Adm: tempo disponível no turno "administrativo";
- Tempo verificação da dobra após início do turnoTurno (em minutos): tempo em que a simulação deve recalculer a necessidade de técnicos adicionais (técnicos do turno anterior que permaneceram para auxiliar no turno seguinte);
- Tempo verificação da dobra⁸ após início do turnoAdm (em minutos): tempo em que a simulação deve recalculer a necessidade de técnicos adicionais (técnicos do turno anterior que permaneceram para auxiliar no turno seguinte);
- Número de testes diferentes por técnico: número de testes que cada técnico pode trabalhar simultaneamente.



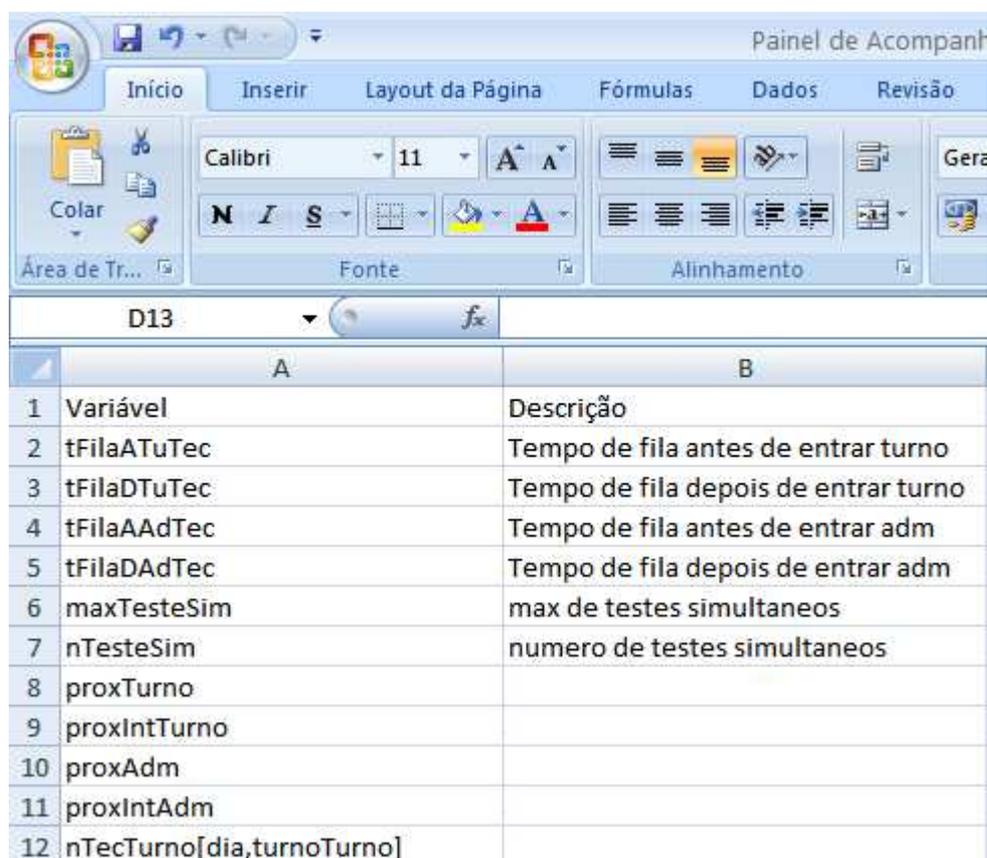
	A	B
1	Tempo do lote (minutos)	5
2	Tempo da antecipação do teste na simulação	480
3	Posição dos testes (coluna)	2
4	Posição dos testes (linha)	2
5	Posição do dest teste (coluna)	21
6	Posição do dest teste (linha)	0
7	Tempo que o técnico pode avançar no intervalo	0
8	Tempo de intervalo no turno	480
9	Tempo disponível por turno Turno	400
10	Tempo disponível por turno Adm	460
11	Tempo verificação da dobra após início do turnoTurno (em minutos)	180
12	Tempo verificação da dobra após início do turnoAdm(em minutos)	30
13	Número de testes diferentes por técnico	5

Figura 28 - Template Configurações Gerais (Elaborada pelo Autor)

⁸ Para o Laboratório, a "dobra" significa a hora-extra realizada pelo técnico. O técnico "dobra" o turno de trabalho.

O Micro Saint tem um painel de visualização de indicadores que podem ser acompanhados durante a simulação. As informações que podem ser vistas são variáveis do modelo. Estas informações podem ser cadastradas previamente no arquivo Painel de Acompanhamento.csv. As informações deste arquivo, exemplificadas na Figura 29, são:

- Variável: variável que será acompanhada no painel de visualização do Micro Saint;
- Descrição: descrição da variável que será acompanhada no painel de visualização do Micro Saint.



	A	B
1	Variável	Descrição
2	tFilaATuTec	Tempo de fila antes de entrar turno
3	tFilaDTuTec	Tempo de fila depois de entrar turno
4	tFilaAAdTec	Tempo de fila antes de entrar adm
5	tFilaDAdTec	Tempo de fila depois de entrar adm
6	maxTesteSim	max de testes simultaneos
7	nTesteSim	numero de testes simultaneos
8	proxTurno	
9	proxIntTurno	
10	proxAdm	
11	proxIntAdm	
12	nTecTurno[dia,turnoTurno]	

Figura 29 - Template Painel de Acompanhamento (Elaborada pelo Autor)

Estes templates foram suficientes para flexibilizar a construção e atualização do modelo.

O passo seguinte no projeto foi a construção do software Gerador Automático de Modelos que tem a função de ler as informações dos templates e codificá-los no formato que o Micro Saint reconhece.

4.3.3 O Gerador Automático de Modelos

A construção do Gerador Automático de Modelos aconteceu durante o projeto de simulação computacional. A metodologia de condução do projeto de simulação utilizada foi a proposta por Law e Kelton (1991).

Na primeira etapa do projeto foi realizado o planejamento do estudo e a formulação do problema. O objetivo geral deste estudo foi analisar de forma sistêmica, integrada e agregada o fluxo de processos do Laboratório, visando ao dimensionamento da capacidade produtiva necessária frente às demandas das unidades de produção da refinaria, recursos e layout para melhor ajustar os prazos de entrega dos testes realizados. Foi definido que o desenvolvimento do trabalho seria realizado em conjunto com uma equipe interna de analistas.

Na segunda etapa do estudo foi definido que seria construído um novo modelo computacional que retratasse a realidade atual do Laboratório, possibilitando entender os seus processos atuais. Com essas definições, foi possível parametrizar os recursos existentes, em termos da sua capacidade disponível (quantidade, turnos de trabalho, etc.).

Como premissa, o modelo deveria ser flexível ao ponto de se poder ampliar as análises para avaliar os impactos que os mesmos causam na capacidade produtiva assim como avaliar as atuais e novas políticas de gerenciamento da capacidade do

laboratório para definir o nível de utilização dos diversos recursos. O modelo deveria ainda determinar os prazos de entrega dos testes realizados.

Uma vez formulado o problema e definido o planejamento global do projeto, foi possível elaborar a especificação da simulação e liberar a equipe do Laboratório a coletar os dados.

O próximo passo do projeto consistiu em validar a especificação, ajustando as informações necessárias.

Na especificação foram definidos os cenários. Os cenários seriam construídos a partir de alterações nos templates. Por exemplo, no cenário 1, o número de técnicos administrativos seria igual a oito. No cenário 2, o número de técnicos seria igual a 5.

Após a validação da especificação, iniciou-se a construção dos templates e do Gerador Automático de Modelos. Para tanto, a condução da construção do gerador automático e dos templates seguiu os passos apresentados na Figura 30.

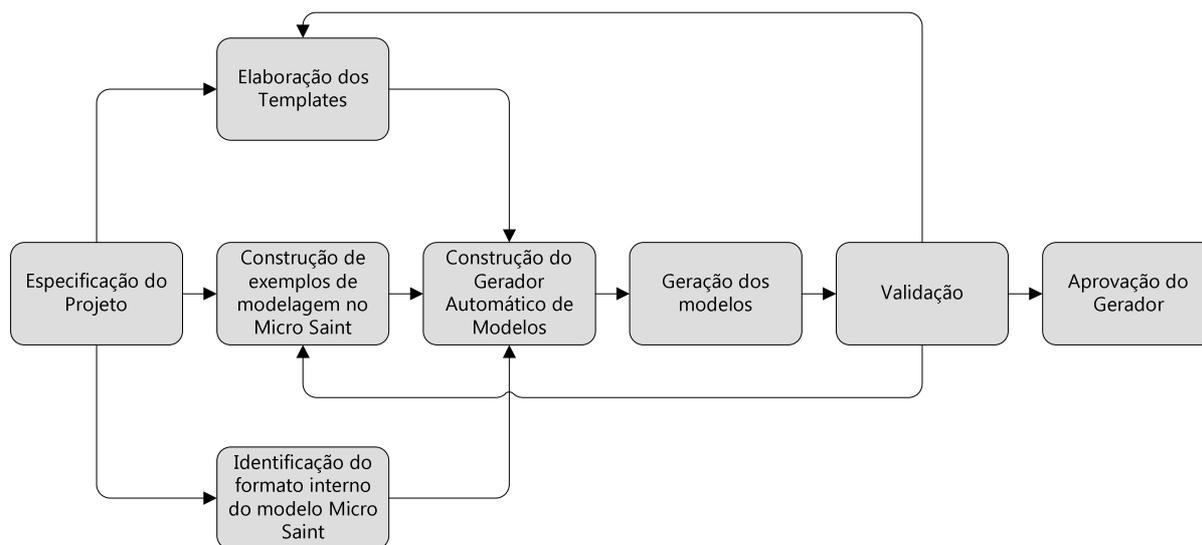


Figura 30 - Condução da Criação do Gerador Automático de Modelos (Elaborada pelo Autor)

Identificação do formato interno do modelo Micro Saint: Para identificar o padrão de armazenamento das informações, foi necessário construir o modelo computacional diretamente no Micro Saint.

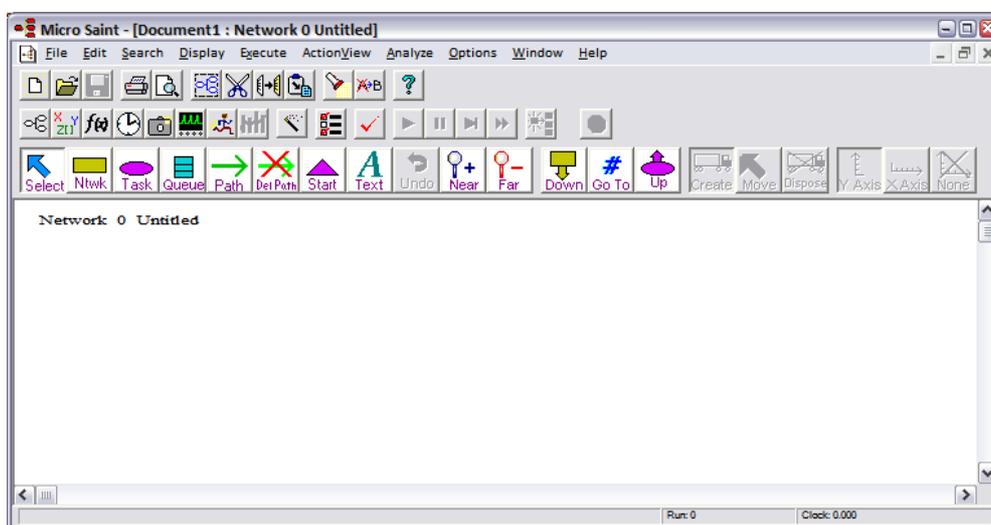


Figura 31 - Modelo de Simulação Vazio (Retirado do Software Micro Saint)

Desta forma, para identificar as informações, foi necessário um trabalho paciente. A cada nova inserção no modelo no Micro Saint, era necessário editar o arquivo e identificar o código que havia gerado.

Construção de exemplos de modelagem no Micro Saint: Em paralelo à identificação da linguagem interna do Micro Saint foram construídos exemplos de modelagem com as regras de funcionamento do laboratório. Essas regras estão inseridas no código do Gerador Automático de Modelos, não sofrendo atualizações a cada construção de um novo modelo.

Assim, as regras estão configuradas no código do Gerador Automático de Modelos e, para atualizá-las, é necessário atualizar o código interno, na linguagem Java.

Optou-se por esta modelagem justamente porque as regras do Laboratório não alteram constantemente, não sendo necessário construir um template exclusivo para isso, durante este projeto.

Vale ressaltar que este fato não engessa a simulação, como também não exclui a atualização do Gerador Automático de Modelos, quando uma regra necessita alteração.

Elaboração dos Templates: Ainda em paralelo à construção de exemplos de modelagem, foram construídos os Templates. Para a construção dos templates, manteve-se um padrão dos campos, para melhor identificar as informações. Na importação dos templates é realizada uma verificação na integridade dos dados, analisando se os arquivos estavam no padrão correto para a simulação. Ao longo da construção do Gerador Automático de Modelos e dos templates, foram sendo realizadas rodadas-piloto.

A idéia de utilizar templates para a construção de modelo surgiu com a evolução dos projetos de simulação no Laboratório. Na especificação do terceiro projeto, os templates das atividades, recursos físicos, recursos humanos e eventos haviam sido definidos. Os demais templates (bancada, variáveis, amostras, configurações gerais, produtos, configuração dos turnos, finalidade e painel de acompanhamento) foram desenvolvidos ao longo do projeto, conforme ia sendo necessário.

Uma vez que o formato interno do Micro Saint e as regras de modelagem estavam conhecidos e os templates preenchidos, pôde-se construir o Gerador Automático de Modelos.

Construção do Gerador Automático de Modelos: O Gerador Automático de Modelos foi desenvolvido utilizando o ambiente de desenvolvimento NetBeans, por ser um software Open Source, com uma comunidade de parceiros ao redor do mun-

do. A Sun Microsystem é o principal patrocinador deste projeto. O NetBeans é um produto gratuito (NETBEANS, 2009).

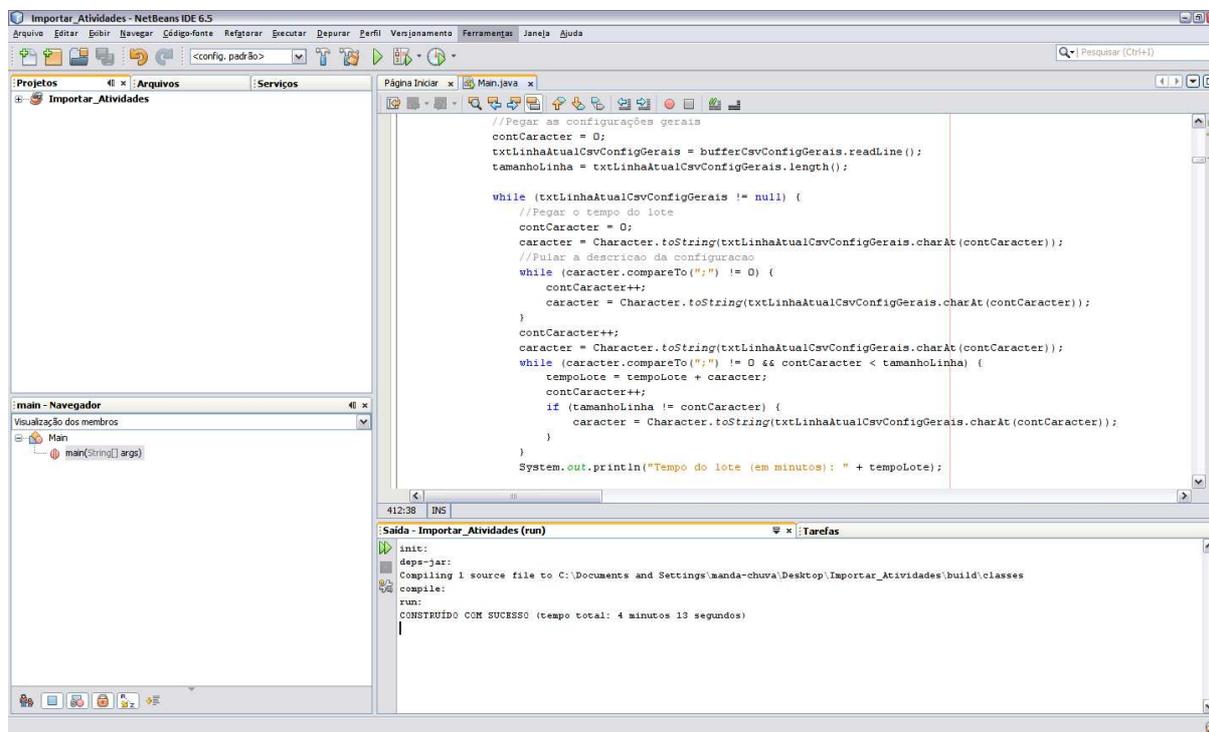


Figura 32 - Ambiente de Desenvolvimento NetBeans (Retirado do Software NetBeans)

A linguagem de programação utilizada para desenvolver o Gerador Automático de Modelos foi a Java. A tecnologia Java é uma plataforma lançada pela Sun Microsystems em 1995, que inicialmente era conhecida como OAK.

O Java permite que você jogue on-line, converse com pessoas do mundo inteiro, calcule a taxa de juros de um financiamento e veja imagens em 3D, apenas para citar alguns exemplos. Ele também faz parte dos aplicativos de intranet e de outras soluções de e-business que formam a base da computação corporativa (SUN MICROSYSTEM, 2009).

Esta linguagem tem como benefícios:

- Criar um software em uma plataforma e executá-lo em outra;
- Criar programas para execução em navegadores e serviços da Web;
- Combinar aplicativos ou serviços, usando a linguagem Java para criar aplicativos ou serviços personalizados;

- Criar aplicativos para telefones celulares, processadores remotos, produtos de consumo de baixo custo.

A única interface do Gerador Automático de Modelos é a solicitação do local onde o modelo será salvo e do local de cada um dos templates, como mostram as Figura 33 e Figura 34.

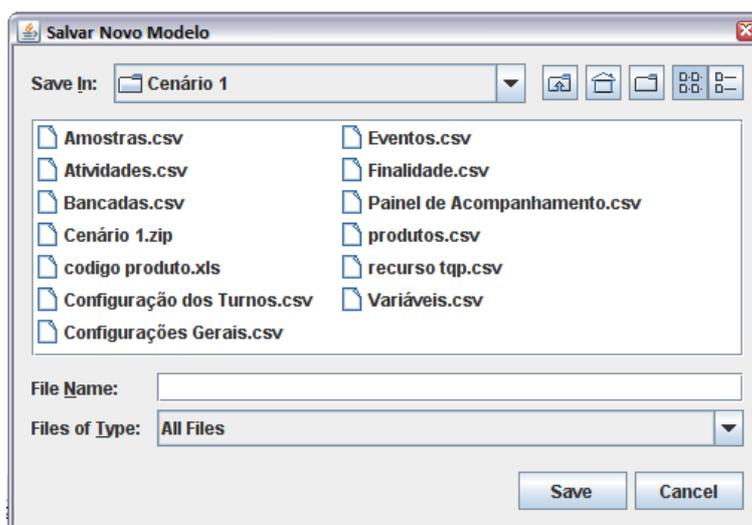


Figura 33 - Gerador Automático de Modelos - Salvar novo modelo (Retirado do Software Gerador Automático de Modelos)



Figura 34 - Gerador Automático de Modelos - Abrir Templates (Retirado do Software Gerador Automático de Modelos)

Desta forma, o usuário construirá um modelo de simulação computacional somente manipulando os templates e indicando onde eles estão salvos.

A Figura 35 exemplifica um modelo computacional construído a partir dos templates:

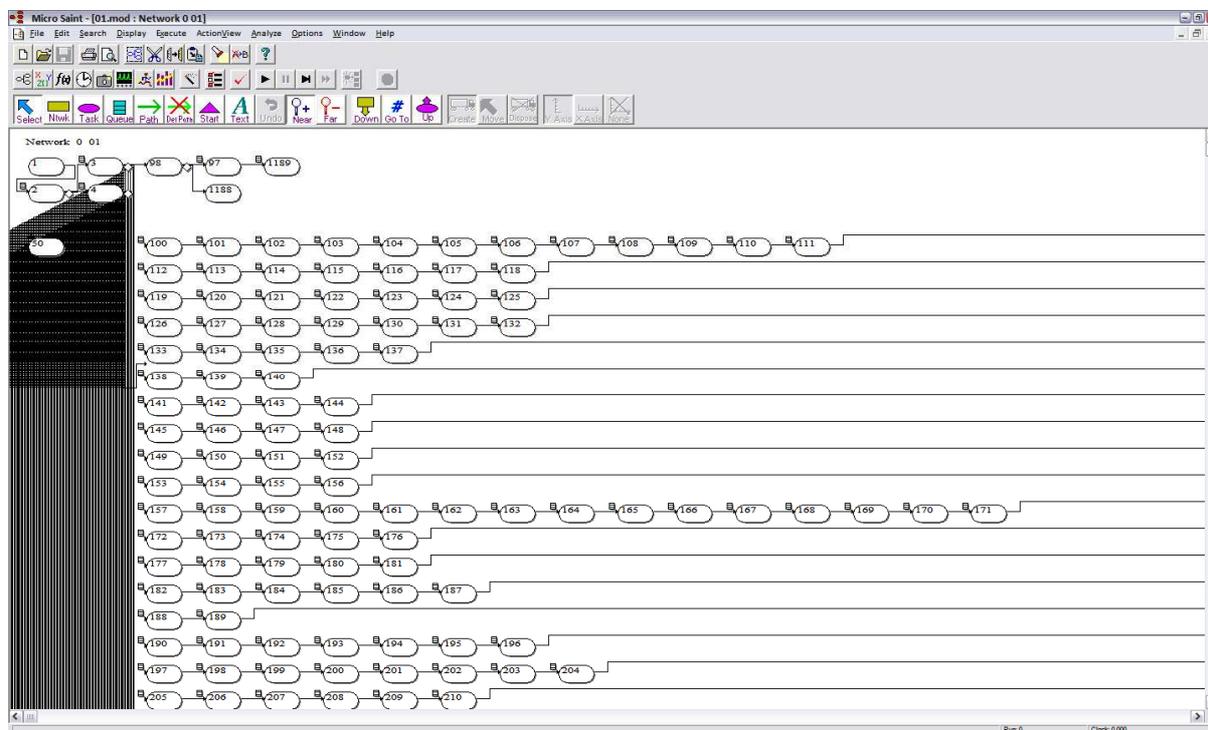


Figura 35 - Modelo Computacional Construído a partir dos Templates (Retirado do Software Micro Saint)

Geração dos Modelos: Com o Gerador Automático de Modelos finalizado, pôde-se gerar modelos para início dos testes. O fato de construir um modelo completo dificultou a execução dos testes do modelo. Assim, para facilitar esta etapa foram utilizados dados reduzidos até que houvesse confiabilidade no Gerador.

Validação: A validação ocorreu durante a geração dos modelos. Houve necessidades de ajustes, tanto das regras de modelagem, quanto dos templates. Nas regras de modelagens foram ajustados erros de entendimento e de código e nos templates foram ajustados campos e dados. Assim que os templates e o modelo computacional foram validados, pôde-se realizar as rodadas produtivas.

Foi elaborada uma tabela-resumo para analisar os resultados, com os principais indicadores da simulação, além do detalhamento de cada indicador em forma de gráficos.

Aprovação do Gerador: Quando os resultados do modelo gerado pelo Gerador Automático de Modelos foram validados após as rodadas produtivas, foi formalizada a conclusão do projeto, com a elaboração do documento final, apresentação dos resultados que contiveram todos os passos que envolveram a simulação, bem como os resultados atingidos.

5 AVALIAÇÃO DA ABORDAGEM DESENVOLVIDA

Este capítulo irá avaliar a abordagem desenvolvida, a fim de identificar os benefícios do uso de templates para geração de modelos de simulação.

5.1 Avaliação do Uso de Templates

Realizou-se uma pesquisa com o usuário do Laboratório onde foi desenvolvido o projeto de simulação, assim como a avaliação do autor.

5.1.1 Avaliação da Pesquisa Realizada

Para avaliar a pesquisa apresentada, foi realizada uma entrevista com o usuário do laboratório que participou dos três projetos de simulação. Ele foi definido, pois nos três projetos, foi o único usuário do laboratório a interagir na construção dos modelos, foco desta pesquisa.

O usuário inicia a entrevista, elencando uma série de características do Laboratório, destacando a função de apoio ao processo produtivo que tem o laboratório. Ele destaca que, em um laboratório onde a realização dos testes é o negócio-fim da empresa, o gerenciamento é realizado sem interferências. O que determina o seqüenciamento são as regras e prioridades que o laboratório define, diferentemente do caso do laboratório onde foi realizada a pesquisa, que fica subordinada ao gerenciamento do processo produtivo. Qualquer alteração do planejamento no nível mais amplo tem impacto direto no gerenciamento.

Neste sentido, o Laboratório escolheu realizar um projeto de simulação em detrimento de um projeto de seqüenciamento, pois com a simulação poderia contemplar a dinamicidade e variabilidade dos processos.

Os projetos de simulação foram focados em construir cenários levando em consideração a demanda futura em relação à quantidade de recursos humanos e recursos materiais para averiguar a necessidade de alocação dos mesmos.

Entre as dificuldades dos primeiros projetos de simulação listadas pelo usuário, estão:

- Entendimento da realidade: para o usuário, houve falhas de entendimento do ambiente real a ser simulado nos dois primeiros projetos. A modelagem conceitual do sistema não foi desenhada de forma satisfatória, gerando retrabalhos futuros. Segundo o usuário, uma das causas desta situação foi justamente pensar no modelo de forma estática, ou seja, sem atualizações futuras;
- Levantamento dos dados: dificuldade na obtenção dos dados, que não estavam disponíveis ou em formato que não poderia ser utilizado na simulação;

- Regras do negócio: as regras do negócio não estavam bem definidas, ou não havia consenso. Isto se deve à não-formalização das regras, ficando estas subjetivas;
- Defasagem do modelo: a defasagem do modelo se devia a dois motivos. O primeiro era o erro em alguma modelagem ou regra basilar do modelo. Por ser basilar, não havia como alterar esta modelagem no decorrer da simulação. O segundo motivo era a alteração natural do sistema real, onde também não era possível representar esta alteração no modelo no tempo de construção. A consequência para esta defasagem era o descrédito da simulação.

Ainda nos primeiros projetos o usuário lembra que não tinha autonomia sobre a simulação por ser muito complexa onde a atualização era item a item. Por não ter esta autonomia e pela complexidade de atualização do modelo, a simulação tinha vida útil somente no tempo do projeto.

Pelo fato de a simulação se utilizar de linguagens e padrões próprios, os demais usuários do Laboratório tiveram uma participação como espectadores, não conseguindo auxiliar diretamente no modelo.

Outro detalhe lembrado pelo usuário foi em relação aos primeiros resultados da simulação. Ele recorda que os primeiros resultados não tardaram a aparecer, entretanto com respostas erradas, ou seja, apesar de ter resultados em um tempo curto, houve retrabalhos até que os resultados pudessem ser considerados válidos.

Apesar destas dificuldades, os projetos chegaram ao fim com resultados satisfatórios.

Para o terceiro projeto, com a experiência dos anteriores, buscou-se incorporar melhorias já na sua concepção e planejamento.

Para este projeto, foram utilizados os conceitos de geração automática de modelos. O usuário discorre que houve uma maturidade nos conceitos, e que isto se deve ao amadurecimento obtido com os projetos anteriores. O usuário percebe que nesta versão os conceitos estavam compreendidos de forma correta.

Entre os benefícios que esta abordagem proporcionou para a simulação, o usuário enumera:

- Regras: possibilidade de alterar regras basilares de uma forma fácil, utilizando os templates;
- Cenários: com o uso de templates, se consegue criar um número vasto de cenários para analisar a capacidade produtiva no Laboratório de uma maneira mais adequada. Os ajustes de capacidade não são estáticos e desta forma consegue-se colocar um dinamismo no modelo, pois se fazem diversas simulações de uma maneira rápida e fácil;
- Dados: os dados que alimentam a simulação estão padronizados, com fácil manipulação;
- Usuários: possibilidade de os demais usuários do Laboratório atualizarem e utilizarem a simulação, sem a necessidade de um usuário especialista. Usuários que nos projetos anteriores não se envolviam na modelagem agora podem ajudar de forma concreta;
- Atualização: o fim do projeto de simulação não representa o fim das atualizações e utilizações do modelo, como eram nas versões anteriores. A vida útil do modelo é estendida.

A dificuldade apontada pelo usuário na utilização desta abordagem é em relação aos primeiros resultados. Diferentemente dos projetos anteriores, nesta versão os primeiros resultados tardaram a ser disponibilizados, entretanto não houve retrabalhos. O usuário chamou esta demora como “curva lenta de desenvolvimento”, exemplificado na Figura 36.

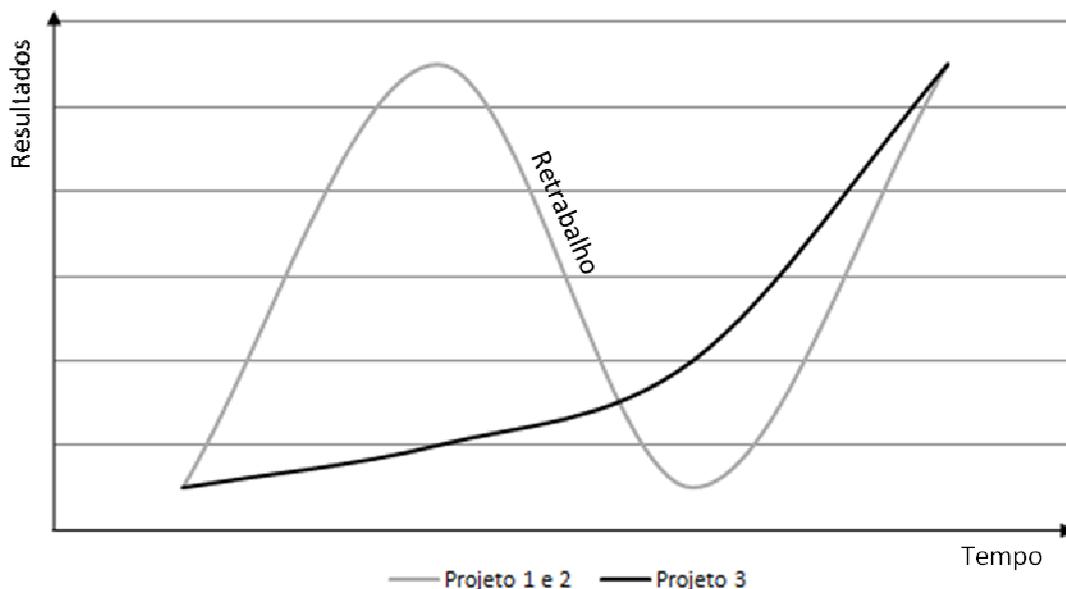


Figura 36 - Curva de desenvolvimento (Elaborada pelo Autor)

Para melhor visualizar e compreender a percepção do usuário em relação à autonomia, atualização, tempo de projeto e vida útil do modelo, foi requisitado que ele ponderasse numa escala de 1 a 5, sendo 1 Baixo, 3 Médio e 5 Alto, conforme a Tabela 5:

Tabela 5 - Avaliação do usuário

	Primeiro Projeto	Segundo Projeto	Terceiro Projeto
Grau de Autonomia dos usuários	1	1	4
Facilidade de Atualização do modelo	2	3	4
Vida útil do modelo	2	2	4

Fonte: Autor

Em relação à autonomia, o usuário considerou que nos dois primeiros projetos não houve autonomia, sendo que no terceiro projeto esta autonomia é completa, pois as regras do Laboratório estão inseridas dentro do código do Gerador Automático de Modelos.

No aspecto atualização do modelo, segundo o autor, foi em uma crescente: no primeiro projeto, era mais difícil de atualizar o modelo, ganhando certa facilidade no segundo projeto em relação à atualização dos dados de demanda e uma maior flexibilidade no terceiro projeto com o uso dos templates.

Anteriormente se disse que o usuário teve expectativa em ter resultados no terceiro projeto, em relação aos projetos passados, mas o tempo total de projeto foi semelhante nas três edições.

Já em relação à vida útil do modelo, o usuário respondeu que nos dois primeiros projetos o modelo teve menor vida útil, comparando com o terceiro projeto.

Como melhorias futuras para a abordagem utilizada nesta pesquisa, o usuário cita:

- Retirar as regras de modelagem que estão dentro do programa para templates;
- Melhorar o método de condução do projeto no que diz respeito à validação do modelo. O usuário sugere que no início do projeto já se tenham dados e indicadores de desempenho (parâmetros de controle) reais em relação a estes dados, para que a validação do modelo possa ser conduzida de forma autônoma, sem a participação direta do usuário;
- Construção de casos de teste para validar as regras de modelagem;
- Transformar o gerador de modelos em um gerador genérico para que possa construir modelos para outros simuladores;
- Elaboração de um questionário para validar a modelagem conceitual e evitar retrabalhos futuros;
- Integrar os templates da simulação com o sistema de gestão do Laboratório;

Essas foram as considerações do usuário do Laboratório em relação ao projeto de simulação computacional que utilizou a abordagem proposta por esta pesquisa. Nota-se que em diversos momentos o usuário comenta sobre a condução da simulação e não da abordagem proposta, mas que também serão consideradas como sugestões de trabalhos futuros.

5.1.2 Avaliação do Estudo Realizado no Laboratório

O estudo proposto possui características, como interface amigável, geração e compilação automática do código de simulação. Esta proposta permite a utilização da simulação por usuários que não possuem profundos conhecimentos de computação, dos métodos de construção e de execução de uma simulação computacional.

O público-alvo para o qual esta abordagem foi desenvolvida são alunos, professores, pesquisadores e profissionais interessados em desenvolver projetos em simulação computacional que não apresentem conhecimentos em modelagem computacional. Para que o usuário construa modelos computacionais, basta saber manipular os templates desenvolvidos durante a pesquisa.

Diferentemente de outros projetos de simulação, este estudo aconteceu em um ambiente onde tanto o autor quanto os responsáveis do laboratório já tinham experiência na construção de modelos de simulação.

A experiência adquirida nos primeiros projetos facilitou o entendimento do sistema, permitindo que o modelo computacional fosse mais bem modelado a cada versão de projeto.

Para este estudo, se teve um cuidado para que a modelagem fosse flexível para ser atualizada. O engessamento das versões anteriores fez com que a simulação fosse abandonada tão logo os projetos se encerrassem.

O prolongamento da vida útil da simulação era requisito para que este estudo fosse desenvolvido. Assim, desde os primeiros planejamentos, esta era a idéia que balizava as modelagens conceituais: flexibilidade, usabilidade, prolongamento do uso da simulação, entre outras, assim como a preocupação de que a manipulação da simulação fosse através de uma interface amigável.

Dentre os benefícios identificados pelo autor, pode-se citar:

1 Atualização do modelo: este benefício se desdobra em outros quatro, que são: atualização dos testes, das bancadas, dos cenários e das prioridades.

Um primeiro benefício é a rapidez e facilidade de inserir ou atualizar os testes no modelo. Nos projetos anteriores esta ação era feita conforme a Figura 37:

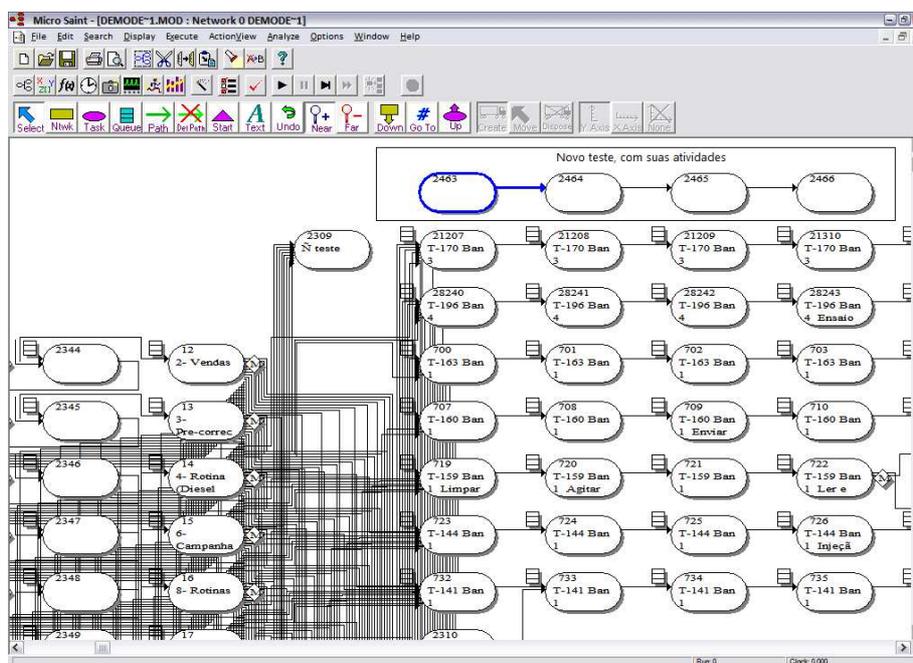


Figura 37 - Inserir atividades no modelo (Retirado do Software Micro Saint)

Recorda-se que nas versões anteriores havia um conjunto de testes para cada bancada separadamente. Para cada atividade (no exemplo, são quatro) era necessário inserir as regras manualmente, conforme a Figura 38.

Task Description
Edit

Looking at Task < >

Task Number Name Appearance

Task Timing Information Time Distribution

Mean Time: Standard Deviation:

Release Condition: Beginning Effect:

Launch Effect: Ending Effect:

Data Collection

Figura 38 - Inserção das regras na atividade (Retirado do Software Micro Saint)

Após inserir as regras em cada uma das atividades, era necessário conectar o teste e inserir a regra nas prioridades, conforme a Figura 39.

Vale ressaltar que, em média, são 170 testes por bancada, 8 prioridades por bancada, sendo que foram consideradas 7 bancadas. Ou seja, para cadastrar todas as atividades presentes na simulação anterior, foi necessário conectar e inserir a regra 9520 vezes.

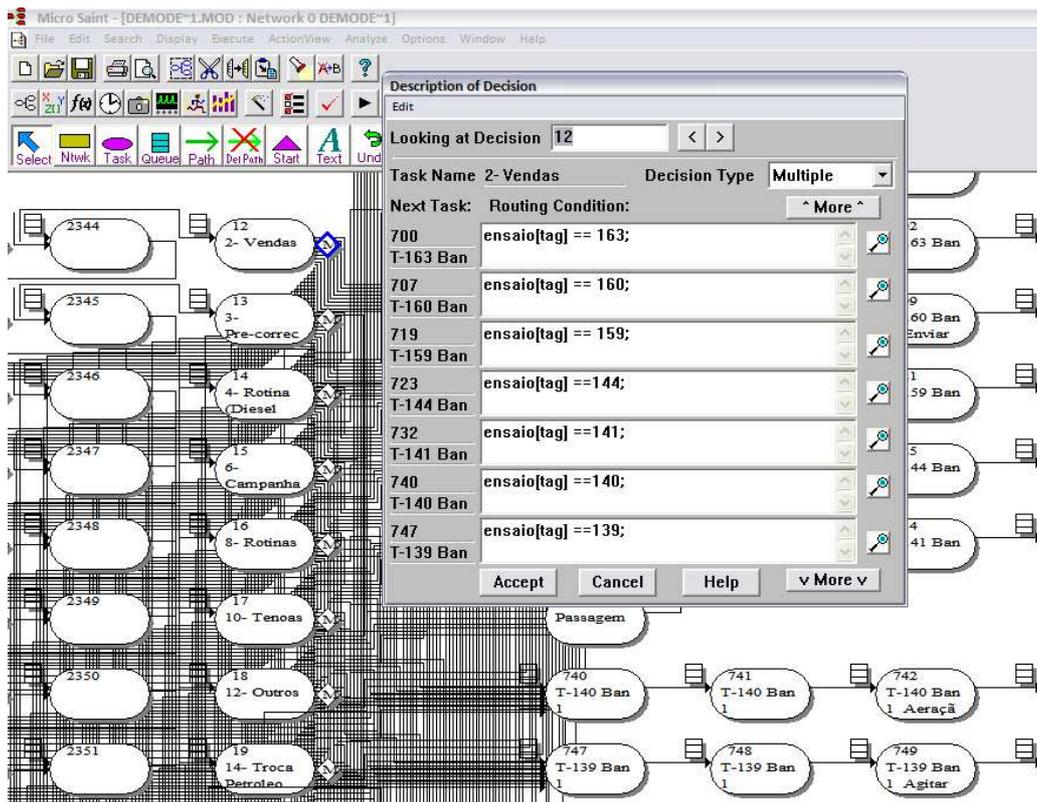


Figura 39 - Conectar teste nas prioridades (Retirado do Software Micro Saint)

Na abordagem proposta nesta pesquisa, a mesma atividade é realizada cadastrando a atividade no template, conforme a Figura 40:

	A	B	C	D	E	F	G	H	I	J	K
1	Teste (desc)	Teste (cod)	Etapa	Médio	Mínimo	Máximo	Recurso1	Recurso2	Recurso3	Presença	Ordem
2	A		1	1	1	2	TQP			F	370
3	A		1	2	2	3	TQP			F	370
4	B		2	1	3	4	TQP			F	370
5	B		2	2	4	5	EQP001			F	370
6	C		3	1	5	6	EQP001	EQP002		F	370
7	C		3	2	6	7	EQP001	EQP002	EQP003	F	370
8	D		4	1	7	8	TQP	EQP002		F	370
9	D		4	2	8	9	TQP	EQP003		F	370
10	D		4	3	9	10	TQP	EQP002	EQP003	F	370
11	D		4	4	10	11				F	370
12											

Figura 40 - Inserção de novo teste com templates (Elaborada pelo Autor)

Nota-se uma economia de esforço no que tange à operacionalização das atividades. Este trabalho manual, suscetível ao erro pelo número de interações que o usuário deve fazer para cadastrar todas as atividades no modelo, é executado pelo Gerador Automático de Modelos, que interpreta o template e o insere no modelo de simulação automaticamente, em segundos.

Outro benefício é na inserção ou atualização das bancadas. Nas modelagens anteriores, o trabalho de inserção de uma nova bancada era realizado manualmente. Cada bancada tem uma série de regras que definem as prioridades e seqüenciamento dos testes, que levam em consideração a ocupação da bancada, disponibilidade de técnicos e recursos.

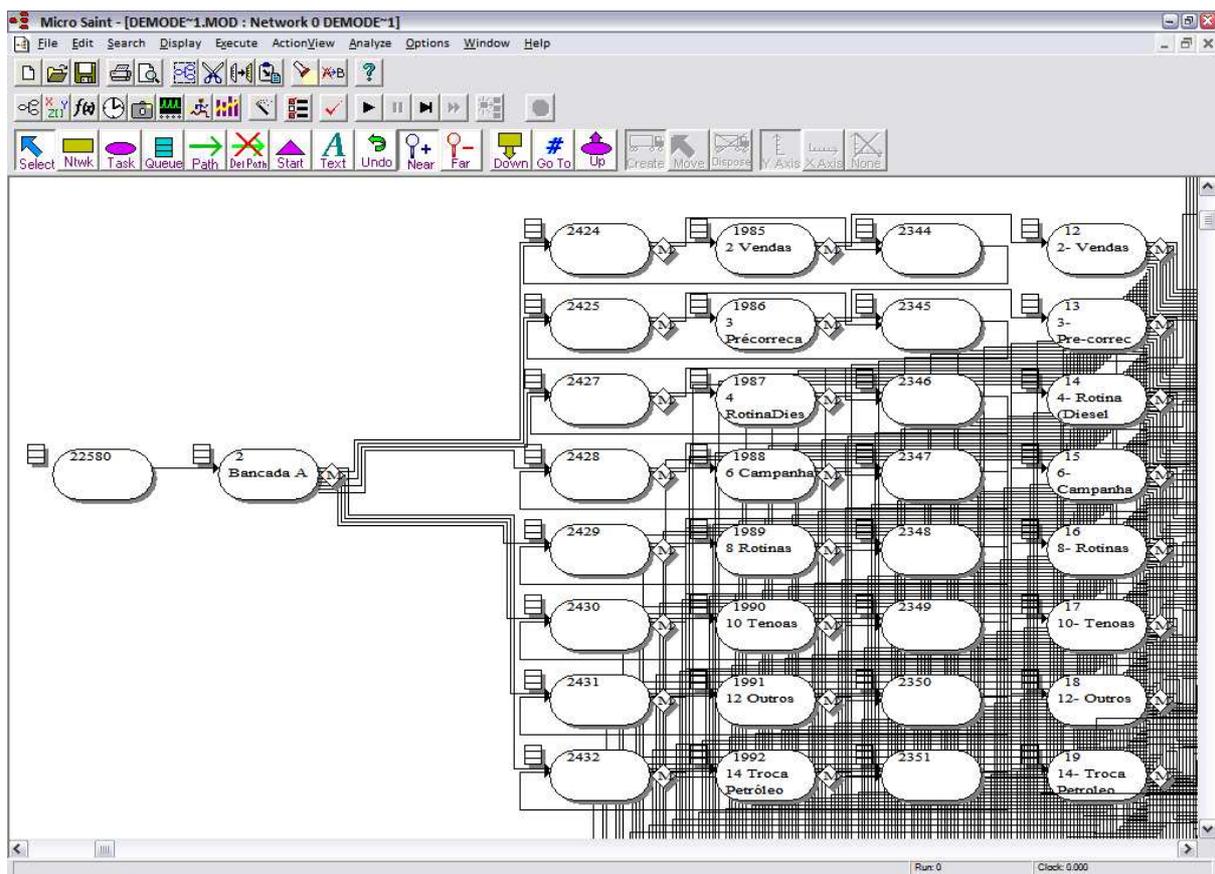


Figura 41 - Exemplo da inserção de uma nova bancada (Retirado do Software Micro Saint)

A Figura 41 exemplifica uma bancada na modelagem das versões anteriores. Para cada atividade de configuração da bancada, era necessário cadastrar uma série de regras, exemplificado na Figura 42.

Figura 42 - Regras na inserção de uma nova bancada (Retirado do Software Micro Saint)

Da mesma forma que inserir um novo teste na simulação, a inserção das regras e atividades na nova bancada era um trabalho manual e oneroso.

Na proposta desta pesquisa, a mesma atividade de inserção de uma nova bancada é demonstrada na Figura 43.

Basta adicionar uma nova linha, informando o número da nova bancada e informar qual turno de trabalho ela irá seguir. As configurações e regras do modelo são geradas automaticamente, sem a interferência do usuário.

	A	B	C
1	Bancada	Número	Turno
2	BANCADA 1	1	ADM
3	BANCADA 2	2	ADM
4	BANCADA 3	3	ADM
5	BANCADA 4	4	TURNO
6	BANCADA 5	5	TURNO
7	BANCADA 6	6	TURNO
8	BANCADA 7	7	ADM
9	BANCADA 8	8	ADM
10	BANCADA 9	9	ADM
11	BANCADA 10	10	ADM
12	BANCADA 11	11	ADM
13	BANCADA 12	12	ADM
14	BANCADA 13	13	ADM
15	BANCADA 14	14	TURNO
16	BANCADA 15	15	TURNO
17	BANCADA 16	16	TURNO

Figura 43 - Inserção de nova bancada utilizando templates (Elaborada pelo Autor)

O terceiro benefício de atualização é na construção de cenários. Uma característica existente no Laboratório é o gerenciamento constante dos técnicos, para que não haja ociosidade, assim como horas-extras durante o mês. Assim, um dos propósitos da execução da simulação foi avaliar a distribuição destes técnicos ao longo do tempo. Assim, foi estabelecida uma série de cenários onde a única modificação foi a configuração dos técnicos.

Nas versões anteriores, esta modificação do número de técnicos se dava dentro da "Lista de Eventos" do Micro Saint. No tempo da virada de turno, havia a configuração do número de técnicos presentes naquele turno.

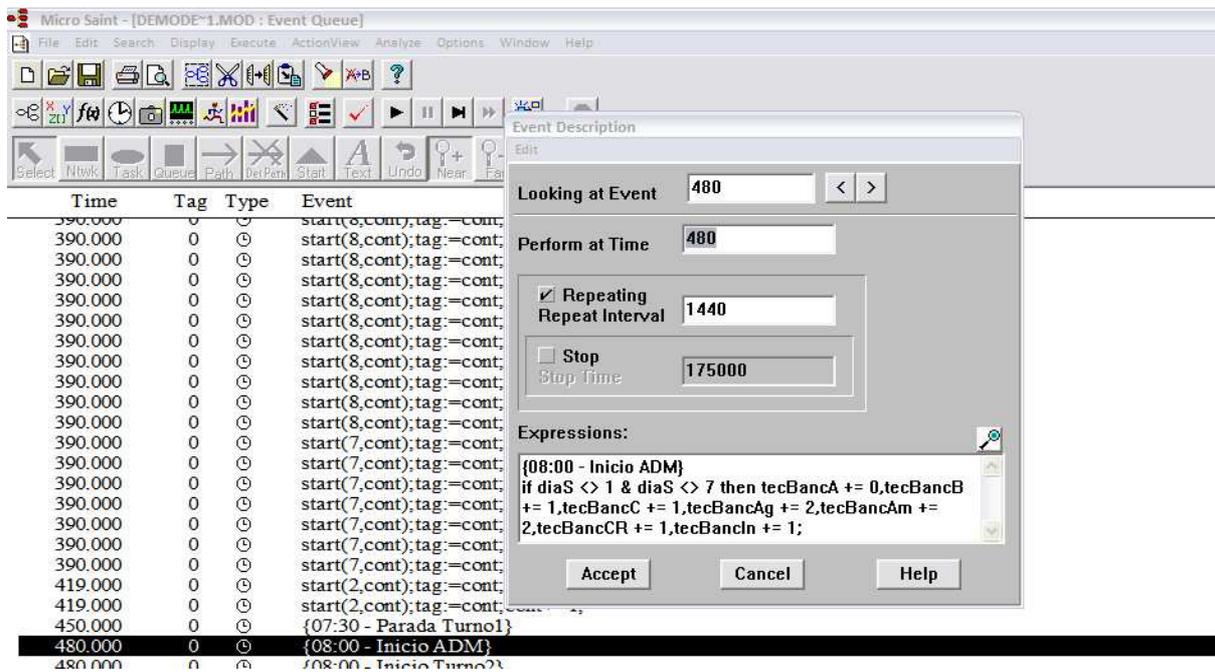


Figura 44 - Configuração do número de técnicos (Retirado do Software Micro Saint)

Ou seja, a configuração dos técnicos ficava na lista de atividades, que continha entre outras coisas, toda a demanda da simulação. Assim, para criar um novo cenário modificando o número de técnicos, era necessário percorrer a lista de atividades, identificando o momento da virada de turno.

Na proposta atual, esta mesma configuração pode vir do sistema de informação do próprio laboratório, conforme a Figura 45.

	A	B	C	D	E	F	G
1	Data	Dia Semar	Tempo	TQP Turnc	TQP Turnc	TQP ADM	
2	1/6/2007 00:00	sex	0	4	0	0	
3	1/6/2007 08:00	sex	480	4	1	7	
4	1/6/2007 16:00	sex	960	4	0	0	
5	2/6/2007 00:00	sáb	1440	4	0	0	
6	2/6/2007 08:00	sáb	1920	3	0	6	
7	2/6/2007 16:00	sáb	2400	4	0	0	
8	3/6/2007 00:00	dom	2880	4	0	0	
9	3/6/2007 08:00	dom	3360	3	0	0	
10	3/6/2007 16:00	dom	3840	4	0	0	
11	4/6/2007 00:00	seg	4320	4	0	0	
12	4/6/2007 08:00	seg	4800	3	1	6	
13	4/6/2007 16:00	seg	5280	4	0	0	

Figura 45 - Alteração dos técnicos com uso dos templates (Elaborada pelo Autor)

Desta forma, a alteração do número de técnicos é realizada de um modo simples e de fácil acesso, via templates.

Esses exemplos mostram que a atualização dos modelos dos projetos passados era inviável e que, para desenvolver uma nova versão, não era possível aproveitar os modelos anteriores, pois o número de interações manuais seria tão grande quanto refazer o modelo do zero.

Ainda nos benefícios de atualização do modelo, pode-se citar a inserção ou atualização de prioridades. Muito semelhante à dificuldade de se inserir um novo teste nas modelagens dos projetos anteriores é a inserção de uma nova prioridade. Cada nova prioridade deveria ser ligada a uma bancada e cada teste deveria estar ligado a uma prioridade. O número de interações do usuário seria 170 testes por bancada, para cada uma das 7 bancadas. Isso daria um total de 1190 interações manuais do usuário com o modelo. Via template o usuário somente insere uma linha (semelhante à adição de uma bancada), e o Gerador Automático de Modelos se encarrega de inserir a nova prioridade de forma automática no modelo de simulação.

2 Facilidade para usuários leigos: Uma diferença importante é que o usuário não precisa modificar o modelo dentro do Micro Saint. Toda informação que ele pode alterar foi colocada nos templates. Assim como os benefícios levantados por Brushi (2003) em sua pesquisa, esta proposta possibilita a geração completa de um modelo de simulação para usuários que não detêm conhecimentos necessários para gerar modelos sem o auxílio dos templates, permitindo também que os usuários modifiquem os modelos gerados de uma forma mais fácil e flexível.

3 Verificação dos dados: Outro ponto importante observado como um benefício que esta abordagem apresenta é uma maior confiabilidade na obtenção dos dados, uma vez que na importação dos mesmos acontece uma verificação do padrão das informações.

4 Agilidade para projetos futuros em laboratórios: Ainda observa-se que, a partir deste projeto, o tempo de geração, execução e manutenção do modelo será minimizado, fazendo com que o tempo total de execução da simulação seja inferior ao tempo de simulação de um projeto onde não são utilizados os templates, ou seja, esta abordagem é útil para aqueles usuários que conhecem simulação e que desejam desenvolver modelos mais rapidamente.

5 Autonomia para o usuário final: Tão importante quanto os benefícios já citados é a autonomia para o usuário final. Usando os templates, o usuário não depende do auxílio constante de uma pessoa especialista em simulação. Isso ocorreu nos projetos anteriores, onde o usuário do Laboratório dependia do conhecimento de um especialista para rodar cenários e criar modelos.

6 Preparação dos dados para o tratamento estatístico: O uso de templates também auxilia o tratamento estatístico dos dados, pois seu layout foi construído de modo que as informações sejam facilmente lidas por softwares estatísticos.

Visto isso, com a nova proposta, cada alteração no modelo gera uma reconstrução total da modelagem. Para o usuário, basta atualizar os templates, que o modelo será totalmente reconstruído.

Durante a pesquisa, surgiram alguns pontos críticos e dificuldades, que serão detalhadas na seção 5.2.

5.2 Pontos Críticos e Dificuldades Encontradas

Durante a execução da pesquisa, foram levantadas algumas dificuldades e pontos críticos, que serão detalhadas na seqüência:

- Escopo amplo do modelo: assim como nos dois primeiros projetos de simulação apresentados, o projeto de simulação desenvolvido nesta pesquisa teve um escopo de trabalho amplo, o que tornou a modelagem complexa e trabalhosa. Esta característica, em certos momentos, dificultou a definição das regras de modelagem, pois pela característica do Laboratório as informações se relacionam, dependendo uma da outra.
- Regras de modelagem inseridas no Gerador Automático de Modelos: pelo fato de as regras estarem dentro do código do Gerador Automático de Modelos, as alterações sempre eram realizadas no ambiente de programação NetBeans e posteriormente testadas no Micro Saint. Esta lógica de construção custou um trabalho duplo, uma vez que era necessário estar com o Micro Saint aberto para gerar as regras de modelagens, para depois transcrever dentro do Gerador Automático. Após a transcrição para o Gerador Automático, era necessário executá-lo para certificar que a modelagem estivesse fidedigna à modelagem construída diretamente no Micro Saint, conforme representação na Figura 46:

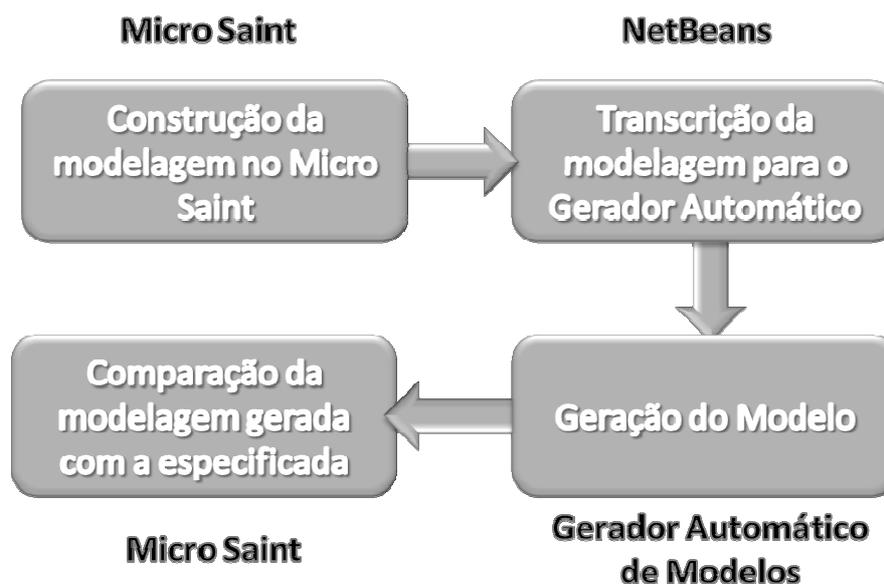


Figura 46 - Fluxo de construção das regras de modelagem (Elaborada pelo Autor)

- Necessidade de modelar 100% das regras para obter as primeiras respostas da simulação, ou seja, diferentemente dos outros onde as primeiras respostas da simulação eram apresentadas à medida que o modelo era construído, nesta versão os primeiros resultados só foram gerados assim que um modelo completo já estivesse rodando. Isto se deve à forma de construção do modelo, que, para poder rodar, necessita-va conter todas as informações;
- Não é possível construir o modelo por etapas, uma vez que ele reconstrói todo o modelo: isto se deve à natureza do Gerador Automático de Modelos. Assim, não é possível validar a construção de um novo modelo por etapas, com vistas a facilitar este processo de validação. Uma sugestão é criar um modelo com as mesmas características, mas com menos informações. Desta forma, mesmo que o Gerador crie um modelo completo com todas as características, ele será simplificado, facilitando a validação do modelo;

Com o término do estudo onde foram detalhados a simulação, os benefícios e dificuldades encontradas durante o projeto, pôde-se chegar a algumas conclusões, que serão detalhadas no capítulo 6.

6 CONCLUSÕES FINAIS E TRABALHOS FUTUROS

Este capítulo apresenta as conclusões da pesquisa e recomendações para trabalhos futuros que possam dar continuidade ao estudo aqui apresentado.

6.1 Conclusões Finais

A presente pesquisa propôs uma abordagem para construção de modelos de simulação computacional em projetos repetitivos.

Realizar uma pesquisa sobre construção automática de modelos com o uso de templates foi motivado por observações do próprio autor e de autores, como Vaccaro (1999), Bruschi (2003), Cassel (1996), entre outros, que citam a construção do modelo de simulação como uma tarefa de custo e que requer habilidades e treinamento apropriado.

O ambiente complexo de um laboratório de análises químicas foi um desafio de modelagem e também foi um dos motivadores da criação de um mecanismo que facilitasse a construção de modelos de simulação.

Durante toda a pesquisa, teve-se o cuidado de identificar pontos importantes relatados por pesquisas correlatas, adaptadas à realidade de um laboratório de análises químicas e ao software Micro Saint.

Ainda durante a construção dos templates sempre se buscava identificar os benefícios proporcionados pela utilização das interfaces configuráveis na construção de modelos de simulação, os quais foram listados no capítulo 5, com destaque à rapidez na construção e atualização de modelos, prolongando sua vida útil, autonomia para o usuário final, geração automática do código do modelo.

Dessa forma, o objetivo geral da pesquisa foi alcançado, assim como os objetivos específicos. Foram construídas as interfaces automatizáveis em formato de planilha eletrônica para facilitar a entrada e organização dos dados no modelo, realizando um projeto de simulação computacional em um laboratório de análises químicas, utilizando os templates desenvolvidos, identificados os benefícios da aplicação de interfaces automatizáveis na construção de um modelo de simulação computacional e, por fim, realizada a avaliação da abordagem proposta, por meio de entrevistas e avaliação empírica do autor.

Foi detalhado o desenvolvimento do gerador automático de modelos, respondendo a questão do problema da pesquisa. Contudo uma hipótese levantada no início da pesquisa não pôde ser respondida. Como não havia registros exatos do tempo de cada uma das etapas dos projetos anteriores, não se averiguou se o tempo de construção dos modelos reduziu em relação aos projetos passados.

Por fim, ressalta-se que a abordagem construída é recomendada para projetos de simulação repetitivos, ou seja, para projetos que terão mais que uma versão. Isto se deve ao fato que, no primeiro projeto desenvolvido, há necessidade de um especialista para definir e incorporar as regras do modelo dentro do Gerador Automático de Modelos.

6.2 Trabalhos Futuros

Durante a pesquisa, foram sendo observados pontos que poderiam ser melhorados em trabalhos futuros. Assim, pode-se citar:

- Para dar uma maior autonomia ao usuário final e para que o gerador automático de modelos possa ser utilizado não só em laboratórios de análises químicas, as regras do ambiente em estudo podem ser também colocadas em templates;
- Alterar o padrão das informações do template das atividades. No template desenvolvido, as atividades seguiram uma lista seqüencial; a sugestão é que as atividades sejam cadastradas de modo que uma atividade aponte a posterior. Ainda neste template, inserir o campo da posição em que a atividade estará visualmente no Micro Saint;
- Identificar as variáveis automaticamente. Durante a construção do gerador automático de modelos, percebeu-se que a identificação das variáveis que compõem o modelo poderia ser realizada automaticamente, sem a necessidade de listá-las em um template;
- Construção de casos-teste para facilitar a validação do modelo (testar o modelo de simulação), ou seja, construir um documento onde se possa estabelecer a resposta esperada para determinada ação, regra ou modelagem dentro do sistema. Este documento deve conter as especificações de entrada e saída, interdependências e exigências especiais (descrição dos casos especiais de inicialização, configuração, etc. que seja necessário para aplicar o teste);
- Transformar o modelo gerado pelo gerador de modelos em um modelo genérico, para ser utilizado em outros simuladores;

- Integrar os templates da simulação com sistemas de gestão dos laboratórios. Durante a pesquisa, foram construídos e utilizados templates preenchidos manualmente pelos usuários, ou importados dos sistemas presentes no laboratório. Como sugestão futura, pode-se pensar na possibilidade de integrar os templates automaticamente com os sistemas de informação do laboratório, tornando esta rotina automática e livre de erros humanos, agilizando ainda mais a atualização dos templates.

7 REFERÊNCIAS BIBLIOGRÁFICAS

ACKOFF, Russel L. e SASIENI, Maurice W. 1971. *Pesquisa Operacional*. Rio de Janeiro : Livros Técnicos e Científicos Editora Ltda, 1971.

ANDRADE, Maria Margarida de. 2006. *Introdução à metodologia do trabalho científico*. 7 ed. São Paulo : Atlas, 2006.

BANKS, Jerry. 1998. *Handbook of Simulation: Principles, Methodology, Advances, Applications, and Practice*. Atlanta : John Wiley & Sons, INC., 1998.

BDTD. 2009. Biblioteca Digital Brasileira de Teses e Dissertações. *Biblioteca Digital Brasileira de Teses e Dissertações*. [Online] Ministério da Ciência e Tecnologia, 15 de Janeiro de 2009. [Citado em: 15 de Janeiro de 2009.] <http://bdtd.ibict.br/>.

BORBA, Gustavo Severo de. 1998. *Desenvolvimento de uma Abordagem a Inserção da Simulação no Setor Hospitalar de Porto Alegre*. Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal do Rio Grande do Sul. Porto Alegre : s.n., 1998. Dissertação de Mestrado.

BRUSCHI, Sarita Mazzini. 2003. *ASDA - um ambiente de simulação distribuída automático*. Instituto de Ciências Matemáticas e de Computação, USP. São Carlos : s.n., 2003. Tese de Doutorado.

BRUYNE, Paul de, HERMAN, Jacques e SCHOUTHEETE, Marc de. 1991. *Dinâmica da Pesquisa em Ciências Sociais*. 5 ed. Rio de Janeiro : Francisco Alves Editora, 1991.

CARVALHO, Leonardo Sanches de. 2006. *Análise das Potencialidades e Vantagens do Uso da Simulação Computacional em Operações Logísticas Complexas, como Ferramenta de Auxílio à Tomada de Decisões: Estudo de Caso em uma Organização Industrial*. Núcleo de Pós-Graduação em Administração, Universidade Federal da Bahia. Salvador : s.n., 2006. Dissertação de Mestrado.

CASSEL, Ricardo Augusto. 1996. *Desenvolvimento de uma abordagem para a divulgação da simulação no setor calçadista gaúcho.* Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal do Rio Grande do Sul. Porto Alegre : s.n., 1996. Dissertação de Mestrado.

CERVO, Amado L. e BERVIAN, Pedro A. 2002. *Metodologia Científica.* 5 ed. São Paulo : Prentice Hall, 2002.

EBSCO PUBLISHING. 2009. EBSCOhost Online Research Databases. *EBSCOhost Online Research Databases.* [Online] EBSCO Industries, Inc., 15 de Janeiro de 2009. [Citado em: 15 de Janeiro de 2009.] <http://ebscobase.com/>.

GARCIA, Claudio. 2005. *Modelagem e simulação de processos industriais e de sistemas eletromecânicos.* São Paulo : EDUSP, 2005.

GIL, Antonio Carlos. 2007. *Como Elaborar Projetos de Pesquisa.* 4 ed. São Paulo : Atlas, 2007.

— **1996.** *Como Elaborar Projetos de Pesquisa.* 3 ed. São Paulo : Atlas, 1996.

GOGG, Thomas e MOTT, Jack. 1996. *Improve Quality & Productivity With Simulation.* 3. ed. s.l. : JMI Consulting Group, 1996. p. 300p.

LAKATOS, Eva Maria e MARCONI, Marina de Andrade. 1990. *Fundamentos de Metodologia Científica.* São Paulo : Atlas, 1990.

LAW, Averill M. e KELTON, W. David. 1991. *Simulation Modeling & Analysis.* Second Edition. New York : McGraw-Hill Books Co., 1991.

LOVEJOY, Tim Curran and Jim. 2001. Supply Chain Simulation. *Techexchange.* [Online] Setembro de 2001. [Citado em: 14 de Fevereiro de 2009.] http://www.techexchange.com/thelibrary/Dama/Supply_Chain_Sim/Fig4.gif.

MENEZES, Alex Fabiane Silveira. 1998. *Desenvolvimento de uma Abordagem para a Divulgação da Simulação Computacional em Sistemas Postais: Um Caso Prático na Empresa Brasileira de Correios e Telégrafos.* Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal do Rio Grande do Sul. Porto Alegre : s.n., 1998. Dissertação de Mestrado.

NAGABHUSHANA, Keshav Prasad. 2006. *Analyze Transition State of Move Rate Changes With Simulation.* Department of Industrial and Manufacturing Engineering, Wichita State University. 2006. Dissertação de Mestrado.

NETBEANS. 2009. Bem vindo ao NetBeans. *NetBeans.* [Online] Sun Microsystem, 22 de Janeiro de 2009. [Citado em: 22 de Janeiro de 2009.] http://www.netbeans.org/index_pt_BR.html.

NETO, Laherce Ribeiro de Castro. 2006. *Modelagem e Simulação da Cadeia Produtiva do Minério de Ferro*. Escola Politécnica. São Paulo : s.n., 2006. Dissertação de Mestrado.

OENNING, Vilmar. 2004. *Decisões sobre Mix de Produção em Indústrias de Produção Conjunta: Uma Abordagem a Partir da Teoria das Restrições e da Pesquisa Operacional*. Programa de Pós-Graduação em Ciências Contábeis, Universidade do Vale do Rio dos Sinos. São Leopoldo : s.n., 2004. Dissertação de Mestrado.

OLIVEIRA, Mário J. de, et al. 2006. Uma plataforma de simulação visual em 3D. *Anais do III SEGeT – Simpósio de Excelência em Gestão e Tecnologia*. 2006.

OMNET++. 2009. Home. *OMNeT++ Community Site*. [Online] OMNeT++ Community Site, 03 de Fevereiro de 2009. [Citado em: 03 de Fevereiro de 2009.] <http://www.omnetpp.org/index.php>.

PAIVA, António Filipe de Oliveira. 2005. *Geração automática de modelos de simulação de uma linha de produção na indústria têxtil*. Departamento de Produção e Sistemas, Universidade do Minho. Guimarães : s.n., 2005. Dissertação de Mestrado.

PEDRO, Márcia Valpassos. 2006. *JLinkt: Desenho e Implementação de um Ambiente de Modelagem Computacional para o Ensino*. Instituto de Matemática, Universidade Federal do Rio de Janeiro. Rio de Janeiro : s.n., 2006. p. 252, Dissertação de Mestrado.

PENTEADO, Rosângela Aparecida Delloso. 1996. *Um método para engenharia reversa orientada a objetos*. Departamento de Física e Informática, Universidade de São Paulo. São Carlos : s.n., 1996. Tese de Doutorado.

RODRIGUES, L. H. 1994. *Developing an approach to help companies synchronise manufacturing*. Lancaster University. 1994. Tese de Doutorado.

SAKURADA, Nelson e MIYAKE, Dario Ikuo. 2003. Estudo Comparativo de Softwares de Simulação de Eventos Discretos Aplicados na Modelagem de um Exemplo de Loja de Serviços. *Anais do XXIII Encontro Nacional de Engenharia de Produção - ENEGEP*. 21 a 24 de Outubro de 2003.

SAMPAIO, Fábio Ferrentini. 1998. Taller Internacional de Software Educativo. [Online] Universidad de Chile, Dezembro de 1998. [Citado em: 22 de Novembro de 2008.] <http://www.c5.cl/ieinvestiga/actas/tise98/html/trabajos/modelag/index.htm>.

SAYÃO, Luís Fernando. 2001. Modelos teóricos em ciência da informação - abstração e método científico. *Ciência da Informação*. jan./abr. de 2001, Vol. 30, pp. 82-91.

SCHRAMM, Fabio K. e FORMOSO, Carlos T. 2007. Uso de Simulação Interativa Visual no Projeto de Sistemas de Produção de Empreendimentos da Construção Civil.

III Encontro de Tecnologia de Informação e Comunicação na Construção Civil. 11 e 12 de Julho de 2007.

SHANNON, Robert Elmaghraby. 1978. *Computer networks and simulation*. Amsterdam : North-Holland, 1978.

SILBERSCHATZ, Abraham, KORTH, Henry F. e SUDARSHAN, S. 1999. *Sistema de Banco de Dados*. São Paulo : MAKRON Books, 1999.

SILVA, Edna Lúcia da e MENEZES, Estera Muszkat. 2001. *Metodologia da pesquisa e elaboração de dissertação*. Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal de Santa Catarina. Florianópolis : Laboratório de Ensino a Distância da UFSC, 2001.

STEFFEN, Diego Augusto. 2006. *Proposição de um método de implantação da técnica de simulação computacional: Estudo de caso em uma empresa de mineração*. UNISINOS. São Leopoldo : s.n., 2006. p. 100, Trabalho de Conclusão.

STRACK, Jair. 1984. *GPSS: Modelagem e Simulação de Sistemas*. Rio de Janeiro : Livros Técnicos e Científicos Editora S.A., 1984.

SUN MICROSYSTEM. 2009. Perguntas Gerais. *Java*. [Online] Sun Microsystem, 22 de Janeiro de 2009. [Citado em: 22 de Janeiro de 2009.] http://java.com/pt_BR/download/faq/whatis_java.xml.

TRAVASSOS, Paulo Roberto Nascimento. 2002. *Um estudo de simulação do sistema brasileiro de coleta de dados baseado em satélites*. Instituto Nacional de Pesquisas Espaciais. São José dos Campos : s.n., 2002. Dissertação de Mestrado.

TRIVELATO, Gilberto da Cunha. 2003. *Técnicas de Modelagem e Simulação de Sistemas Dinâmicos*. Ministério da Ciência e Tecnologia, Instituto Nacional de Pesquisas Espaciais. São José dos Campos : INPE, 2003. Nota Técnica.

VACCARO, Guilherme L. R. 1999. *Modelagem e Análise de Dados em Simulação*. Universidade Federal do Rio Grande do Sul. Porto Alegre : s.n., 1999. p. 129, Exame de Qualificação PPGC.

VACCARO, Guilherme Luís Roeh e RODRIGUES, Luis Henrique. 2005. Características Desejadas em Softwares de Análise de Resultados de Simulação. *Anais do XXV Encontro Nac. de Eng. de Produção*. 29 a 01 de Outubro e Novembro de 2005, pp. 4665-4672.

VARGA, Andrés. 2001. *The OMNeT++ Discrete Event Simulation System*. Prague : s.n., 2001. pp. 6-9.

WAGNER, Paulo Rech, FREITAS, Carla Maria Dal Sasso e WAGNER, Flávio Rech. 1996. Um novo paradigma para modelagem e simulação interativa visual. *Anais do IX SIBGRAPI (1996)*. 1996, pp. 87-94.