

UNIVERSIDADE FEDERAL DE SANTA CATARINA

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

***Comparação de Modelos de Agentes para
Gerenciamento de Redes: Uma Abordagem
via Simulação***

Dissertação submetida à Universidade Federal de Santa Catarina

para obtenção do grau de

Mestre em Ciência da Computação

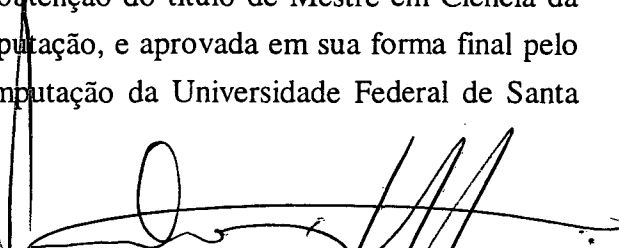
Iwens Gervásio Sene Júnior

Florianópolis, Junho de 1997


Comparação de Modelos de Agentes para Gerenciamento de Redes: Uma Abordagem via Simulação

IWENS GERVÁSIO SENE JÚNIOR

Esta dissertação foi julgada adequada para a obtenção do título de Mestre em Ciência da Computação, Especialidade Sistemas de Computação, e aprovada em sua forma final pelo Curso de Pós-Graduação em Ciência da Computação da Universidade Federal de Santa Catarina

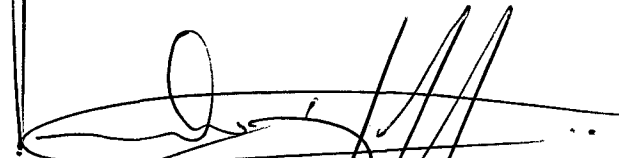


Prof. Paulo José de Freitas Filho, Dr.
Orientador, INE, UFSC



Prof. Murilo Silva de Camargo, Dr.
Coordenador do Curso, INE, UFSC

Banca Examinadora



Prof. Paulo José de Freitas Filho, Dr.
Presidente, INE, UFSC



Prof. Vitório Bruno Mazzola, Dr.
INE, UFSC



Prof. Maria Marta Leite, M.Sc.
INE, UFSC

Agradecimentos

Agradeço aos amigos e colegas e aos professores do CPGCC da UFSC.

À UFSC e à CAPES, pelo suporte material e apoio financeiro.

Ao companheiro do laboratório do CPGCC Eduardo Hoff, pelos momentos de apoio.

Ao “quartel”, que contribui e teve presença marcante.

À Verinha e à Valdete pela dedicação no atendimento aos alunos do curso, mesmo nos múltiplos pedidos para ontem.

Aos professores Vitório Bruno Mazzola e Maria Marta Leite pelas contribuições ao presente trabalho.

Em especial pela professora Elizabeth Sueli Specialski, pelo incentivo e pela co-orientação durante o curso de Mestrado. E principalmente pelo exemplo de vida que ela é.

Ao professor Paulo José de Freitas Filho, pela orientação, companheirismo, apoio e motivação no desenvolvimento deste trabalho.

Aos amigos Braulio Adriano, Celso Kopp e Roberto Scheffel, que com certeza formaram minha família durante este importante período demonstrando amizade em situações, ora de muitas dificuldades e outras de alegrias e comemorações.

À minha namorada que mesmo a distância se empenhou com compreensão e muito amor.

Aos meus pais e meus irmãos, pelo incentivo, dedicação e pela presença em todos os momentos.

Ao restante de meus familiares com o incentivo e apoio.

Agradeço a Deus por esta oportunidade e colocar em minha volta sempre pessoas maravilhosas.

Resumo

Atualmente, a complexidade de novas gerações de sistemas de computação, principalmente no gerenciamento de redes de computadores, tem exigido que plataformas de gerência sejam construídas.

Levando-se em conta os atuais avanços tecnológicos, e em função de grande variedade e capacidade dos elementos disponíveis nestes ambientes.

Este trabalho apresenta uma avaliação de desempenho em tecnologia de computação de alta performance, no desenvolvimento de aplicações de gerenciamento de redes.

Utiliza-se simulação para esta avaliação entre o modelo de objetos passivos e modelo de objetos ativos, sendo ambos baseados no modelo de gerência OSI/ISO.

A avaliação destes modelos é realizada com base nos resultados de simulações com diferentes parâmetros, através de análises, comparação e estatística.

O presente trabalho tem por objetivo comparar os resultados obtidos, de modo a determinar o modelo que apresenta melhor performance no gerenciamento de redes de computadores.

Abstract

Nowadays, mainly in computer networks management, new computer systems generations complexity has required the construction of management platforms.

Considering today's technological advances, and because of the great variety and capacity of the elements available in these environments.

This work presents a performance analysis on high performance computing technology, concerning network management applications development.

Simulation techniques are used for this analysis between the passive objects and active object models, which are both, based on the OSI/ISO management model.

The models evaluation is done based on simulation results using different parameters, through analysis, comparisons and statistics.

This work's goal is to compare the obtained results, so that the network management model with the best performance can be determined.

Sumário

RESUMO	V
ABSTRACT	VI
SUMÁRIO	VII
LISTA DE TABELAS	X
LISTA DE FIGURAS	XI
CAPÍTULO I - INTRODUÇÃO	1
1.1 - APRESENTAÇÃO.....	1
1.2 - MOTIVAÇÃO E JUSTIFICATIVA.....	1
1.3 - DEFINIÇÃO DO PROBLEMA.....	2
1.4 - OBJETIVOS DO TRABALHO.....	3
1.4.1 - <i>Objetivo Geral</i>	3
1.4.2 - <i>Objetivos Específicos</i>	3
1.5 - ESTRUTURA DO TRABALHO.....	4
CAPÍTULO II - GERENCIAMENTO OSI	5
2.1 - INTRODUÇÃO.....	5
2.2 - GERÊNCIA DE REDES.....	5
2.2.1 - <i>Modelo de Gerência OSI/ISO</i>	5
2.2.2 - <i>Plataformas de Gerenciamento</i>	7
2.2.3 - <i>Aplicações de Gerenciamento</i>	8
2.3 - O PROJETO DA PLATAFORMA DE GERENCIAMENTO.....	10
2.3.1 - <i>Uma Visão Geral do Projeto</i>	11
2.3.2 - <i>Modelos de Implementação de Objetos Gerenciados</i>	12
2.3.2.1 - <i>Modelo de Implementação de Objetos Passivos</i>	12
2.3.2.2 - <i>Modelo de Implementação de Objetos Ativos</i>	15
2.4 - SUMÁRIO.....	17
CAPÍTULO III - AVALIAÇÃO DE DESEMPENHO	19
3.1 - INTRODUÇÃO.....	19
3.2 - TÉCNICAS E MEDIDAS DE DESEMPENHO.....	19
3.3 - SIMULAÇÃO.....	22

3.3.1 - <i>Sistemas</i>	23
3.3.2 - <i>Modelos</i>	23
3.4 - TERMINOLOGIA DE MODELAGEM E SIMULAÇÃO DE SISTEMAS	23
3.4.1 - <i>Variáveis de Estado</i>	24
3.4.2 - <i>Eventos</i>	24
3.4.3 - <i>Modelos Discretos e Modelos Contínuos</i>	24
3.4.4 - <i>Entidades e Atributos</i>	25
3.4.5 - <i>Recursos</i>	25
3.5 - JUSTIFICATIVAS PARA O USO DE SIMULAÇÃO	25
3.6 - VANTAGENS E DESVANTAGENS DA SIMULAÇÃO.....	27
3.7 - PASSOS EM UM ESTUDO DE SIMULAÇÃO.....	29
3.8 - LINGUAGEM DE SIMULAÇÃO	30
3.8.1 - <i>Características de um Programa para Simulação</i>	32
3.8.2 - <i>O ARENA</i>	32
3.9 - VARIÁVEIS E MEDIDAS DE DESEMPENHO	33
3.10 - PROJETO EXPERIMENTAL.....	34
3.10.1 - <i>Tipos de Projetos Experimentais</i>	35
3.11 - ANÁLISE DOS RESULTADOS DA SIMULAÇÃO.....	36
3.11.1 - <i>Análise Estatística</i>	37
3.11.2 - <i>Verificação e Validação dos Modelos de Simulação</i>	38
3.11.3 - <i>Tipos de Sistemas</i>	39
3.11.3.1 - <i>Sistema Terminais</i>	39
3.11.3.2 - <i>Sistemas Não-Terminais</i>	39
3.12 - AVALIAÇÃO DE DESEMPENHO EM REDES DE COMPUTADORES	40
3.13 - SUMÁRIO	41
CAPÍTULO IV - MODELAGEM DOS AMBIENTES SIMULADOS	43
4.1 - INTRODUÇÃO.....	43
4.2 - VISÃO GERAL DOS MODELOS DESENVOLVIDOS	43
4.2.1 - <i>Modelagem dos Gerentes</i>	44
4.2.1.1 - <i>Gerente Síncrono</i>	44
4.2.1.2 - <i>Gerente Assíncrono</i>	45
4.2.2 - <i>Modelagem dos Agentes</i>	45
4.2.2.1 - <i>Agente com Objetos Passivos</i>	46
4.2.2.2 - <i>Agente com Objetos Ativos</i>	48
4.3 - PRINCIPAIS DIFERENÇAS ENTRE OS MODELOS.....	50
4.4 - <i>Sumário</i>	50

CAPÍTULO V - EXPERIMENTAÇÃO E RESULTADOS DA SIMULAÇÃO	52
5.1 - INTRODUÇÃO.....	52
5.2 - TÉCNICAS E FERRAMENTAS DE AVALIAÇÃO DE DESEMPENHO UTILIZADAS.....	52
5.2 - EXPERIMENTOS.....	54
5.2.1 - <i>Fatores e Níveis</i>	54
5.2.2 - <i>Variáveis de Resposta</i>	57
5.3 - ANÁLISE DOS RESULTADOS DA SIMULAÇÃO.....	58
5.3.1 - <i>Caracterização das Replicações</i>	59
5.4 - RESULTADOS DO MODELO DE AGENTE COM OBJETOS PASSIVOS.....	60
5.4.1 - <i>Tempo médio de uma requisição no sistema (TTR)</i>	61
5.4.2 - <i>Número de Requisições Atendidas (NRA)</i>	62
5.4.3 - <i>Taxa de Atendimento das Requisições (NRA/TTR)</i>	63
5.5 - RESULTADOS DO MODELO DE AGENTE COM OBJETOS ATIVOS.....	63
5.5.1 - <i>Tempo médio de uma requisição no sistema (TTR)</i>	65
5.5.2 - <i>Número de Requisições Atendidas (NRA)</i>	67
5.5.3 - <i>Taxa de Atendimento das Requisições (NRA/TTR)</i>	68
5.6 - COMPARAÇÃO DOS MODELOS.....	68
5.7 - ANÁLISE ESTATÍSTICA	71
5.7.1 - <i>Cálculo dos Efeitos</i>	71
5.7.2 - <i>Interpretação dos Resultados</i>	73
5.8 - SUMÁRIO.....	74
CONCLUSÃO	76
REFERÊNCIAS BIBLIOGRÁFICAS.....	79

Lista de Tabelas

TABELA 3.1- CONSIDERAÇÕES QUE PODEM LEVAR À ESCOLHA DA TÉCNICA UTILIZADA	21
TABELA 3.2 - MEDIDAS RELACIONADAS COM A REALIZAÇÃO SATISFATÓRIA DO SERVIÇO	33
TABELA 4.1 - PRINCIPAIS DIFERENÇAS ENTRE AGENTES COM OBJETOS ATIVOS E COM OBJETOS PASSIVOS	50
TABELA 5.1 - FATORES UTILIZADOS NA SIMULAÇÃO DE OBJETOS PASSIVOS.	60
TABELA 5.2 – RESULTADOS OBTIDOS DO MODELO OBJETOS PASSIVOS	61
TABELA 5.3 - FATORES UTILIZADOS NA SIMULAÇÃO DE OBJETOS ATIVOS	64
TABELA 5.4 – RESULTADOS OBTIDOS DO MODELO OBJETOS ATIVOS	64
TABELA 5.5 - RESULTADOS DO EXPERIMENTO 1 COM INTERVALO DE REQUISIÇÕES DE 367 MS	65
TABELA 5.6 - PARÂMETROS IGUAIS UTILIZADOS NA SIMULAÇÃO TANTO EM AGENTES COM OBJETOS PASSIVOS QUANTO EM AGENTES COM OBJETOS ATIVOS.....	69
TABELA 5.7 - RESULTADOS DA SIMULAÇÃO DOS MODELOS COM COMPARAÇÃO.....	70
TABELA 5.8 - COEFICIENTES DE CONTRASTE PARA UM FATORIAL 2 ⁴	72

Lista de Figuras

FIGURA 2.1 - INTERAÇÕES DO GERENCIAMENTO DE SISTEMAS [BRISA93].....	6
FIGURA 2.2 - GERENCIAMENTO E A CAMADA DE APLICAÇÃO [BRISA93]	9
FIGURA 2.3 - SISTEMA DE OBJETO PASSIVOS.....	13
FIGURA 2.4 - OBJETOS GERENCIADOS COMO OBJETOS PASSIVOS	15
FIGURA 2.5 - SISTEMA OBJETO ATIVO	16
FIGURA 3.1 – FLUXOGRAMA DE UM ESTUDO DE SIMULAÇÃO.....	29
FIGURA 2.1 - MODELAGEM DO GERENTE SÍNCRONO.....	44
FIGURA 2.2 - MODELAGEM DO GERENTE ASSÍNCRONO.....	45
FIGURA 2.3 - MODELAGEM DO AGENTE COM OBJETOS PASSIVOS	46
FIGURA 2.4 - TRATAMENTO DAS NOTIFICAÇÃO NA MODELAGEM DO AGENTE COM OBJETOS PASSIVOS	47
FIGURA 2.5 - MODELAGEM DO AGENTE COM OBJETOS ATIVOS.....	48
FIGURA 2.6 - TRATAMENTO DAS NOTIFICAÇÃO NA MODELAGEM DO AGENTE COM OBJETOS ATIVOS.....	49
FIGURA 5.1 - GRÁFICO DE TEMPOS MÉDIOS DE REQUISIÇÃO - OBJETOS PASSIVOS.....	62
FIGURA 5.2 - GRÁFICO DOS NÚMEROS DE REQUISIÇÕES ATENDIDAS - OBJETOS PASSIVOS.....	62
FIGURA 5.3 - GRÁFICO DAS TAXAS DE ATENDIMENTO DAS REQUISIÇÕES - OBJETOS PASSIVOS	63
FIGURA 5.4 - GRÁFICO DOS TEMPOS MÉDIOS DE REQUISIÇÃO - OBJETOS ATIVOS	65
FIGURA 5.5 - GRÁFICO DOS TEMPOS MÉDIOS DE REQUISIÇÃO PARA O EXPERIMENTO 1 MODIFICADO	66
FIGURA 5.6 - GRÁFICO DOS NÚMEROS DE REQUISIÇÕES ATENDIDAS - OBJETOS PASSIVOS.....	67
FIGURA 5.7 - GRÁFICO DAS TAXAS DE ATENDIMENTO DAS REQUISIÇÕES (NRA/TTR)	68

Capítulo I

INTRODUÇÃO

1.1 - Apresentação

A complexidade das novas gerações de sistemas de computação, heterogêneos, distribuídos e concorrentes, exige o uso de ferramentas de administração e gerência, que permitam a monitoração e controle destes ambientes. O número destas ferramentas disponíveis tem aumentado significativamente e, com isto, surge a necessidade de comparar alternativas, avaliando seu custo/benefício. Uma das alternativas para a necessidade em questão seria o uso de ferramentas de modelagem e avaliação que é uma possível utilização para esta análise de desempenho.

A modelagem e sua posterior análise em determinados sistemas de computação, têm recebido com ênfase por parte dos pesquisadores e projetistas que necessitam entender bem o comportamento destes sistemas.

Este documento descreve um projeto de pesquisa abordando a modelagem, experimentação e simulação de sistemas de gerência de redes de computadores, o que possibilita a avaliação de seu desempenho em sistemas distribuídos [SLC95].

Uma análise mais apurada é feita, com o auxílio da simulação, e são aplicadas técnicas de análise estatística, para determinar quais são os melhores e piores pontos relativos ao desempenho destes modelos.

1.2 - Motivação e Justificativa

Atualmente, estamos convivendo com um crescente processo de transformação dos ambientes computacionais de centralizados para distribuídos. Esta migração é motivada por um conjunto de vantagens que incluem menor custo, maior poder computacional global e maior confiabilidade.

Tal migração tem exigido maior evolução na área, o que resulta no desenvolvimento de sistemas de alta performance e baixo custo. Atualmente, computadores pessoais, estações de trabalho e redes de computadores possuem desempenho maior que os sistemas anteriores, com custos cada vez menores.

Na medida da evolução, amadurecimento e aumento da competitividade na indústria da informática, as decisões entre as alternativas envolvendo projetos de sistemas computacionais, redes de computadores e, em especial, os sistemas de gerência de redes, tornam-se cada vez mais importantes. Estas decisões devem ser baseadas em procedimentos criteriosos os quais permitam chegar a soluções com a necessidade real obtendo o máximo desempenho com o mínimo custo.

Dentro desta perspectiva, o grupo de redes do Laboratório de Integração de Software e Hardware (LISHA) da Universidade Federal de Santa Catarina (UFSC), está desenvolvendo um projeto que visa a implementação de uma plataforma de suporte para aplicações de gerência de redes, que segue o modelo OSI/ISO [MMS96]. Como todo projeto, este apresenta diferentes alternativas para a sua implementação. Aplicando a técnica de avaliação de desempenho é que se deseja saber qual a melhor solução antes da implementação.

É neste sentido que a presente dissertação se posiciona. Buscando basicamente a comparação entre dois modelos de agentes de gerência OSI. Um implementado com objetos ativos e outro com objetos passivos, ambos descritos em [Matias96] para a implementação de agente na plataforma, com o objetivo de medir o desempenho de ambos para determinar a melhor alternativa.

1.3 - Definição do Problema

Pode-se conceituar sistemas computacionais como dispositivos ou conjuntos de equipamentos que realizam o tratamento de informações. Isto envolve a aquisição, processamento e armazenamento, recuperação, transmissão e apresentação de estruturas de dados. Este ambiente é tanto mais complexo quanto mais cresce o número de máquinas de diferentes fabricantes, com suas características diversificadas (hardware e software), interligadas em redes e necessitando de uma boa interoperabilidade. Estes complexos fatores combinados, justificam a necessidade de seu gerenciamento.

Vários projetos de pesquisa [NoMe96], [SiNo96] visam a especificação e implementação de plataformas para gerenciamento de redes com o objetivo de fornecer um gerenciamento integrado em ambientes heterogêneos.

Este trabalho, como parte componente do projeto de uma plataforma de gerenciamento OSI, aborda fundamentalmente a avaliação de desempenho da implementação de agentes de duas maneiras diferentes. Esta avaliação de desempenho é fundamental pois servirá de comparação entre os dois modelos, um que segue a maioria das plataformas de gerência de redes atualmente disponíveis utilizando, objetos passivos e outro, proposto com objetos ativos.

A avaliação de desempenho não é necessariamente uma ferramenta de previsão do funcionamento de sistemas, que já está definido, mas sim uma ferramenta quantitativa que vai indicar o desempenho do futuro sistema. Sua utilização possibilita apresentar através de números, as características de uma determinada realidade. A análise de desempenho é, portanto, realizada tendo como base a comparação entre valores numéricos [Jain91].

1.4 - Objetivos do Trabalho

1.4.1 - Objetivo Geral

Este trabalho tem como principal objetivo apresentar uma comparação entre dois modelos de servidores de informação de gerenciamento. Apesar de ambos terem execuções diferentes pretende-se mostrar em quais pontos um é melhor e pior que o outro. Um modelo utilizado na maioria das plataformas (enfoque de objetos passivos), e outro proposto em [Matias96] (enfoque de objetos ativos). Ambos são baseados no modelo de gerência OSI/ISO. Desta avaliação de desempenho serão obtidos indicadores que orientarão qual a melhor proposta a ser adotada e em quais situações.

1.4.2 - Objetivos Específicos

- modelar os agentes, com objetos ativos e com objetos passivos;

- modelar o gerente que vai interagir com estes agentes, realizando interações de gerência;
- efetuar a avaliação de desempenho, com o auxílio da simulação, levantando métricas de acordo com as características mais relevantes a serem analisadas, através de experimentos e estudos estatísticos.

1.5 - Estrutura do Trabalho

O presente trabalho está organizado em cinco capítulos. O capítulo 2 apresenta uma visão do modelo de gerência OSI, seus componentes, seu modelo de informação, a aplicação de gerenciamento, interações de objetos gerenciados e recursos reais. Este capítulo também tratará das plataformas existentes bem como o modelo da plataforma de gerência à qual este trabalho está relacionado.

A avaliação de desempenho de sistemas computadorizados é abordada no capítulo 3. As ferramentas para a avaliação de desempenho são abordadas, com ênfase especial na simulação, que é a ferramenta utilizada neste trabalho. Também é abordada a construção de projetos de experimentos para a simulação.

No capítulo 4 é apresentada a modelagem dos dois tipos de agentes, com objetos ativos e passivos, além dos gerentes, para fins de simulação. São descritos os mecanismos que implementam, nos modelos, o funcionamento e a interação entre os componentes nos sistemas reais.

O capítulo 5 apresenta as experimentações e os resultados provenientes das simulações. São feitas as análises e comparações sobre estes resultados, através de tabelas, gráficos e análises estatísticas. Em seguida são apresentadas as conclusões sobre os resultados obtidos, com base na proposta de comparação deste trabalho.

Capítulo II

GERENCIAMENTO OSI

2.1 - Introdução

Hoje em dia, quando se fala de computadores interligados através de uma rede, a principal preocupação é reduzir custos enquanto, ao mesmo tempo, proporciona-se novos níveis de serviços. Em geral, o problema encontrado é que poucos sistemas de informação podem dar-se ao luxo de possuírem recursos provenientes de um único fornecedor. Normalmente a rede possui uma grande variedade de equipamentos e sistemas de computação de diferentes tipos e arquiteturas, adquiridos de vários fornecedores e com pouca ou praticamente nenhuma interoperabilidade.

Interoperabilidade significa que os elementos da rede tais como modems, comutadores de pacotes, PABX e outros dispositivos podem comunicar-se uns com os outros. Para maximizar os benefícios que as redes heterogêneas oferecem, os usuários e fornecedores devem investir em algum sistema de gerenciamento para coordenar, controlar e monitorar o comportamento dos elementos desta rede de forma integrada.

Visto a necessidade de ferramentas que auxiliem este gerenciamento, a seguir será apresentado o modelo de gerenciamento OSI, afim de introduzir conceitos básicos que auxiliarão no decorrer dos outros capítulos.

2.2 - Gerência de Redes

2.2.1 - Modelo de Gerência OSI/ISO

O sistema de gerência de redes é definido como um conjunto de processos que provêem mecanismos para monitoração, controle e coordenação de recursos em um ambiente OSI e define protocolos padrões para a troca de informações entre estes processos [BRISA93].

Um objeto gerenciado representa um recurso real sujeito ao gerenciamento. Ele é definido em termos de seus atributos, das operações a que pode ser submetido, das notificações que pode emitir e de seus relacionamentos com outros objetos gerenciados. O conjunto de objetos gerenciados, juntamente com seus atributos, operações e notificações, constitui a MIB (*Management Information Base*) [ISO/TEC91b].

O modelo de gerenciamento mais eficaz e genérico é o modelo de gerência OSI/ISO. O fato de ser eficaz e completo torna este modelo bastante complexo, pela quantidade de detalhes que deve manipular.

A Figura 2.1 apresenta, de maneira conceitual, a interação entre os componentes do modelo. As interações entre entidades de aplicação de gerenciamento são abstraídas em termos de operações de gerenciamento (CREATE, DELETE, GET, CANCEL-GET, SET e ACTION) e relatórios de eventos (EVENT-REPORT), sendo estas interações efetuadas pelo protocolo de gerenciamento CMIP [ISO/TEC91a]].

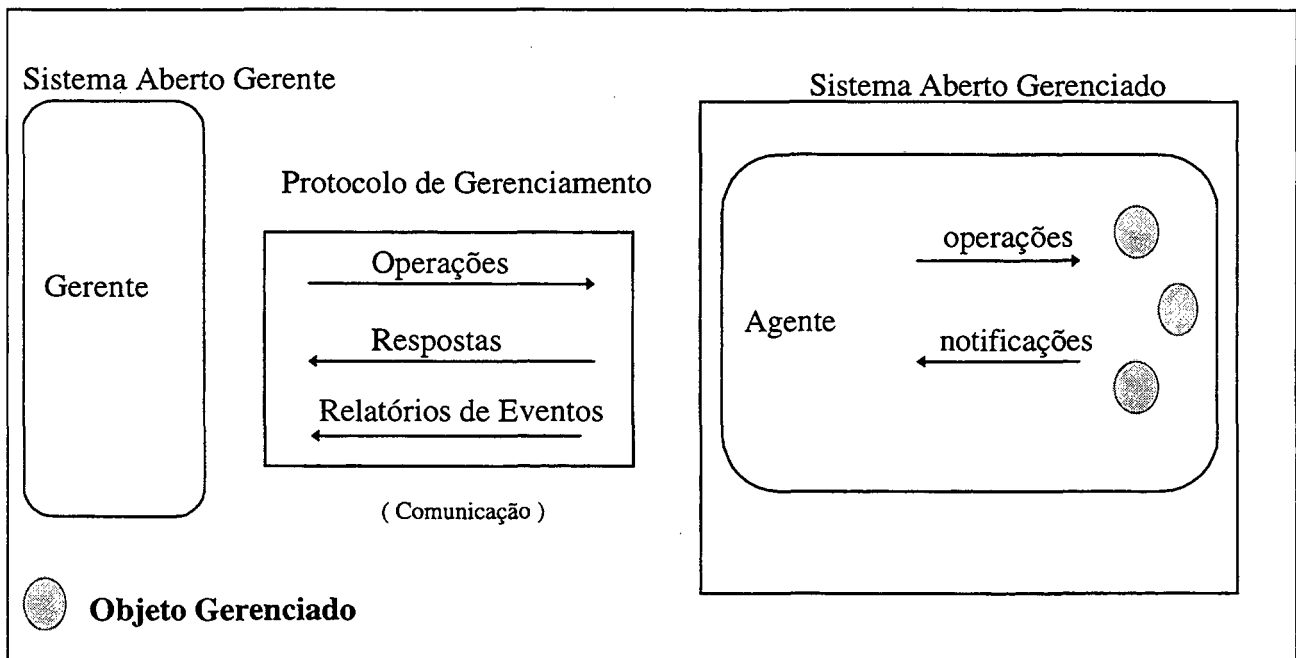


Figura 2.1 - Interações do gerenciamento de sistemas [BRISA93]

O gerenciamento de redes é feito através de associações estabelecidas entre processos da camada de aplicação (chamados processos de gerenciamento) [BRISA93]. Um processo de gerenciamento pode assumir um dos dois papéis possíveis: gerente ou

agente. Uma vez que um processo assume o papel de gerente ou agente, este não pode mais ser alterado enquanto a associação for mantida.

Aplicações de gerência no papel de agente têm como objetivo monitorar e controlar os objetos gerenciados, dentro de seu ambiente de gerenciamento, executando operações de gerenciamento requisitadas pelo gerente, e emitindo relatórios de eventos em função das notificações geradas pelos objetos gerenciados.

Nas aplicações distribuídas, a aplicação gerente é responsável por uma ou mais atividades de gerenciamento, envolvendo a emissão de operações e/ou a recepção de relatórios de eventos.

Cada agente OSI possui uma MIB (*Management Information Base*) [ISO/TEC91a]], que é uma base de dados guardando informações a respeito dos objetos gerenciados. A estrutura da MIB no modelo OSI segue uma abordagem orientada a objetos. Um objeto gerenciado OSI é definido em termos de seus atributos, comportamentos, notificações e operações. Atributos são características específicas de um objeto, operações são ações que podem ser executadas no objeto, notificações são emitidas pelo objeto para indicar algum evento, e o comportamento dita as mudanças no objeto devido às operações executadas nele.

O ambiente de comunicação a ser gerenciado é distribuído, implicando no fato de que as atividades de gerenciamento sejam também distribuídas. Duas ou mais entidades de aplicação de gerenciamento de sistemas podem associar-se para prover uma instância de um gerenciamento, bem como estabelecer um ambiente distribuído.

2.2.2 - Plataformas de Gerenciamento

Dentre alguns trabalhos desenvolvidos na área de gerenciamento de redes, discutidos na atualidade, tem-se diversos relacionados à implementação de plataformas de gerenciamento de redes. Um exemplo é a análise comparativa entre as diferentes metodologias utilizadas no desenvolvimento de aplicações de gerenciamento de redes pelas plataformas AIDE (*Administration, Intelligence, Diagnostic and Expertise*) [SOS95] e OSIMIS (*OSI Management Information Service*) [GoNo95], no contexto de sistemas TMNs (*Telecommunication Management Networks*). Onde a descrição de um protótipo, em fase de implementação, contendo a migração do AIDE para o OSIMIS. São investigados aspectos de suporte oferecidos por estas plataformas, com ênfase na interface

Q3 (intra TMN) e na interface X (inter TMN) da arquitetura funcional TMN, proposta pelo ITU-T (ex-CCITT) [M.3010], [SOS95].

Outro importante trabalho discute a utilização do modelo de gerenciamento TMN (*Telecommunication Management Networks*) para o gerenciamento de redes baseadas em nós de comutação ATM (*Asynchronous Transfer Mode*). As características principais do ATM e os seus requisitos de gerenciamento, OSI e TMN são analisados para subsidiar a escolha do modelo mais adequado para o gerenciamento das redes heterogêneas [GoNo95].

Tem-se ainda a arquitetura de gerência integrada da plataforma SIS [SiNo96]. Neste trabalho foram apresentadas as funcionalidades e os seus componentes, bem como o modelo gerente/agente utilizado para a integração de gerenciamento de redes de computadores e de telecomunicações. A interface de programação de aplicações de gerência oferecida também é abordada. Este esquema de gerenciamento visa otimizar o tempo e o esforço gastos na construção de aplicações de gerência distribuídas.

2.2.3 - Aplicações de Gerenciamento

Os documentos de padronização não definem a organização interna dos processos de gerenciamento, sendo esta uma questão local de cada implementação [ISO/TEC92b]. A fim de implementar tais processos, será necessário identificar seus requisitos e características, propondo uma estrutura na qual os elementos de serviços e protocolos disponíveis para as aplicações de gerenciamento, podem ser utilizadas nestes processos.

Processos de aplicação de gerenciamento são aquelas aplicações que se utilizam dos serviços providos pelo elemento de serviço de aplicação de gerenciamento de sistemas (SMAE). Este conceito definido em [ISO/TEC92a]], está ilustrado na figura 2.2.

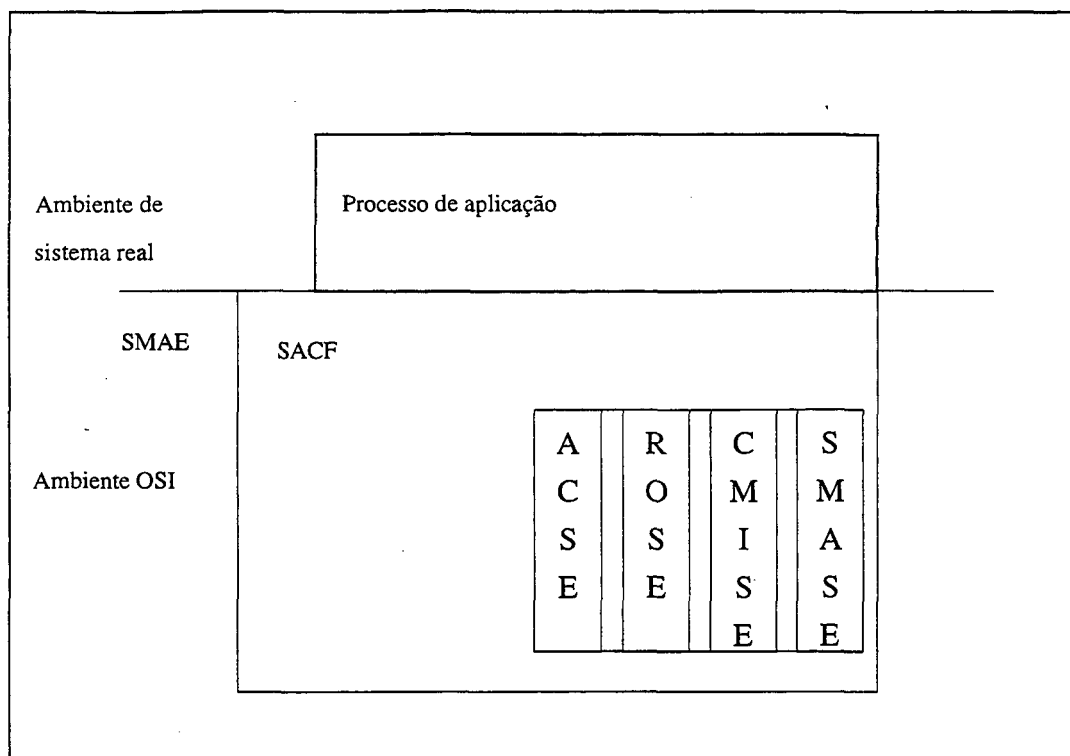


Figura 2.2 - Gerenciamento e a camada de aplicação [BRISA93]

De forma genérica, foram identificados três requisitos básicos que a estrutura de um processo de aplicação de gerenciamento deve conter [BRISA93]:

- 1) suportar uma abordagem orientada a eventos;
- 2) suportar a execução de *polling* em tempo real, tanto para recursos reais do sistema quanto para comunicação entre aplicações de gerenciamento;
- 3) estruturar seus serviços, protocolos e informações (ACSE, SMASE, CMIP, LPP, SMI, etc.) de maneira a atender os requisitos (1) e (2).

Em função das aplicações de gerenciamento necessitarem de um suporte para comunicações externas (ambientes distribuídos), estas são organizadas seguindo um enfoque orientado a eventos.

Os tipos de comunicação entre as aplicações podem ser identificadas assumindo as possíveis interações em que uma aplicação de gerenciamento pode estar envolvida:

- 1) gerentes \Rightarrow agentes
- 2) agentes \Rightarrow gerentes
- 3) agentes \Rightarrow recursos gerenciados
- 4) gerentes \Rightarrow gerentes

A primeira interação é feita para a execução de operações de gerenciamento requisitadas pelo gerente.

A segunda interação, pode ser tanto para a emissão de respostas às operações requisitadas, quanto para a emissão de relatórios de eventos.

A terceira interação é realizada quando se tem recursos gerenciados os quais estão fracamente acoplados ao sistema onde se localiza o agente. Um exemplo deste tipo de recurso, pode ser uma impressora que está conectada diretamente à rede e que deve ser gerenciada por um agente localizado em uma estação de trabalho. Este tipo de comunicação pode ser visto no exemplo seguinte:

A comunicação agente-impressora não é padronizada e deve ser feita de maneira transparente para o processo gerente. Esta comunicação é totalmente dependente da interface do recurso (neste caso a impressora). Recursos fortemente acoplados são aqueles que podem ser acessados no mesmo sistema local. Processos sendo gerenciados em sistemas que implementam proteção de memória, por exemplo UNIX, são considerados recursos fortemente acoplados.

A quarta interação se dá para propósitos de comunicações entre gerentes, quando se tem gerentes integrados ao nível de interações inter-domínios, visto em [Tanenbaum88].

2.3 - O Projeto da Plataforma de Gerenciamento

Dois modelos são largamente adotados para a construção de plataformas de gerenciamento de redes. Um baseado no protocolo CMIP [BRISA93], e outro baseado no protocolo SNMP. Em redes baseadas em TCP/IP, a utilização do protocolo SNMP é predominante, em função de sua implantação simples, podendo ser embutida em vários dispositivos de redes (roteadores, hubs, switches, etc.).

O CMIP, por ser mais robusto em função de sua maior complexidade e funcionalidade, exige um maior esforço em sua implementação, consumindo maiores recursos para seu processamento, não sendo tão atrativo implementá-lo em dispositivos simples tais como modems, hubs, etc.

Contudo, existem atualmente vários fornecedores de produtos (HP, SUN, etc.) para gerenciamento de redes, que implementam o protocolo CMIP, visto ser uma padronização internacional e oferecer várias funcionalidades não apresentadas pelo SNMP.

Com o crescimento e aumento da complexidade dos ambientes de redes de computadores e de telecomunicações, as necessidades de utilização de protocolos de gerenciamento mais robustos será um fator encorajador para que o CMIP seja largamente utilizado [ISO/TEC91a]].

Várias plataformas no mercado já estão implementando *gateways* para ambos os protocolos, o que permite uma interação das operações de gerenciamento dos dois ambientes. O objetivo de se utilizar protocolos abertos para gerenciar redes de comunicação, é reduzir o custo do gerenciamento destas redes, fornecendo interfaces padronizadas dentre as diferentes tecnologias e serviços, aumentando a interoperabilidade em ambientes multifornecedores [VaBB96].

2.3.1 - Uma Visão Geral do Projeto

O grupo de redes do Laboratório de Integração de Software e Hardware (LISHA) da Universidade Federal de Santa Catarina (UFSC), está desenvolvendo um projeto que visa a implementação de uma plataforma para gerência de redes, a qual segue o modelo OSI/ISO [MMS96].

Neste projeto participam alunos de graduação e pós-graduação em Ciência da Computação da Universidade Federal de Santa Catarina, sendo objeto de vários trabalhos de conclusão de curso e dissertações de mestrado.

Como a plataforma tem por objetivo inicial atuar em um ambiente de rede TCP/IP, ambiente da rede UFSC, foi utilizada a implementação da pilha TCP/IP fornecida pelo próprio sistema operacional, o qual implementa ao nível de *Kernel*.

O sistema operacional utilizado é o SunOs 5.4, que segue a linha UNIX SVR4, o qual acompanha o pacote Solaris 2.4. O ambiente de hardware é composto por estações de trabalho Sun SPARCstation do Departamento de Informática, as quais estão ligadas em uma rede ethernet [JoOI96], utilizando como protocolos para interconexão em rede a pilha TCP/IP.

Juntamente com as estações de trabalho, estão também conectados à rede inf., vários PC's com sistemas DOS/Windows e Linux. A linguagem de programação utilizada na implementação da plataforma é a linguagem C++, utilizando-se o compilador GNU C++ 2.7.2.

2.3.2 - Modelos de Implementação de Objetos Gerenciados

Existem duas possibilidades de implementação de entidades agentes, as quais seguem duas abordagens diferentes no tratamento de requisições de operações de gerenciamento.

A primeira possibilidade corresponde à implementação dos objetos gerenciados na maioria das plataformas de gerenciamento atualmente disponíveis, conhecida como objetos passivos. A segunda abordagem, descrita em [Matias96], faz uma implementação diferenciada dos agentes e dos objetos gerenciados, com uma implementação *multithreaded* [JoOI96], conhecida como objetos ativos.

A seguir, serão descritos os detalhes funcionais de ambas as abordagens.

2.3.2.1 - Modelo de Implementação de Objetos Passivos

O modelo de implementação de objetos passivos pode ser representado como na Figura 2.3. Neste modelo, uma estação gerente está ligada a um agente através de um canal de comunicação. O sistema aberto gerenciado é composto de um serviço genérico - CMIS, que contém a MIB.

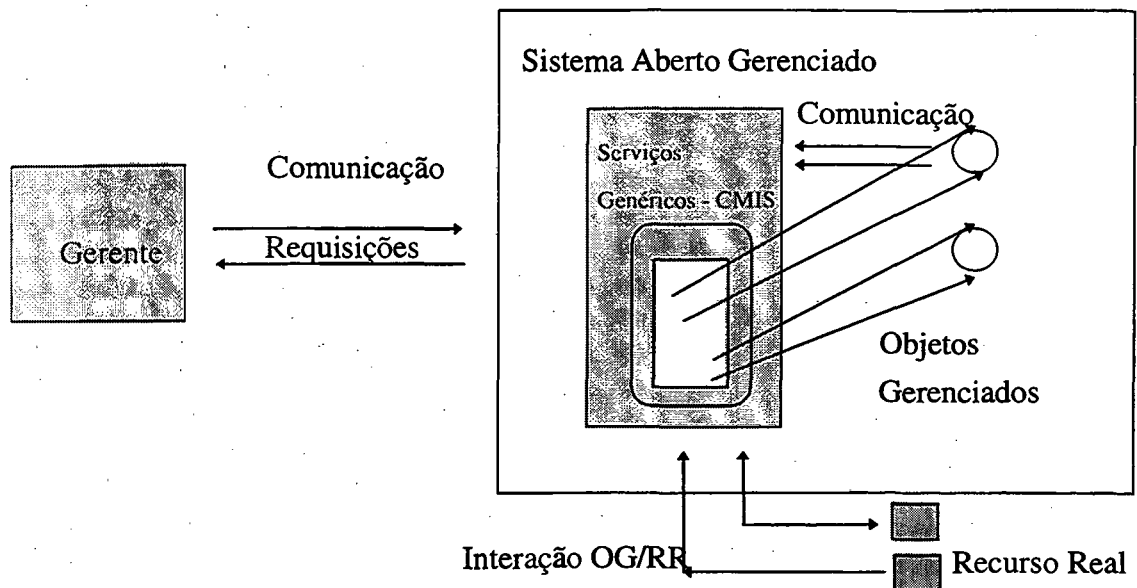


Figura 2.3 - Sistema de objeto passivos

A interação entre gerente-agente acontece através de um canal de comunicação, onde o gerente emite operações de gerenciamento (requisições) para o agente. O gerente pode enviar as seguintes requisições para um agente:

- CREATE: cria um objeto gerenciado na MIB;
- GET: busca valores do objeto gerenciado na MIB;
- SET: modifica os valores de um objeto gerenciado na MIB;
- CANCEL-GET: cancela uma operação GET;
- DELETE: apaga um objeto gerenciado na MIB;
- ACTION: executa uma ação em objetos gerenciados.

Já o agente emite as respostas das requisições, bem como relatórios de eventos (notificações), que são alarmes oriundos do recurso real, informando ao gerente que algum evento ocorreu. Estas notificações são denominadas M-EVENT-REPORT.

Os canais de comunicação descrevem a comunicação gerente-agente baseado no protocolo CMIP. No sistema aberto gerenciado, apresentado na figura 2.3, estão incluídas as funções que cuidam da recepção e envio de mensagens ao gerente.

Dentro dos serviços genéricos estão incluídas as funções que cuidam da recepção e envio de mensagens CMIP ao gerente, a manutenção da árvore de informação de gerenciamento (MIT) que contém as instâncias dos objetos gerenciados em uso e os

mecanismos de acesso a essas instâncias para a execução dos comandos do CMIP (GET, SET, CREATE, DELETE e ACTION).

Os mecanismos de acesso, citados anteriormente, ativam a execução de cada comando para cada instância específica. Por exemplo, quando o gerente solicita um GET para um certo atributo de uma determinada instância, os mecanismos genéricos se encarregarão de encontrar a instância desejada dentro da árvore. É então extraído o valor daquele atributo de acordo com sua sintaxe e a mensagem que retornará este valor ao gerente é montada.

O módulo conhecido como *callback* apresentado na Figura 2.4, deve ser escrito manualmente, para possibilitar que o agente genérico CMIP comunique-se com o recurso gerenciado. Uma vez que cada recurso real tem sua própria linguagem de acesso proprietária, é impossível uma automação total do processo de implementação de um agente destinado a controlar qualquer tipo de equipamento. Além disso, o tempo gasto para realizar as comunicações do agente com o recurso real dependerá exclusivamente do recurso que está sendo gerenciado, uma vez que cada um apresenta características diferentes de comunicação.

Nos modelos de gerenciamento não há uma padronização na interação entre objeto gerenciado e recurso real.

A cada requisição do gerente é necessário que a operação seja processada em todas as fases da implementação do agente com objeto passivo, até chegar ao gerente esta resposta, para que o gerente possa enviar outra requisição, ou seja, apenas uma requisição em trânsito neste tipo de agente.

A implementação do agente com objetos passivos pode ser ilustrada como mostra a Figura 2.4.

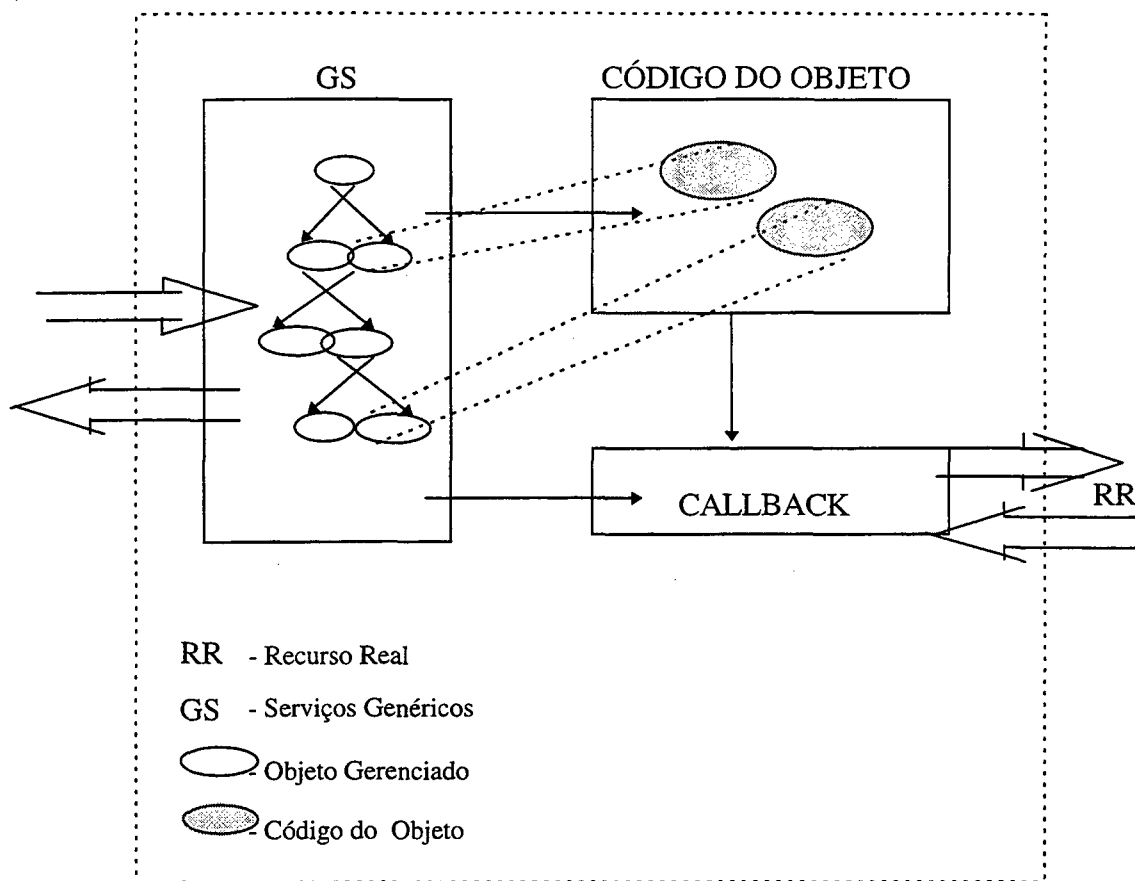


Figura 2.4 - Objetos gerenciados como objetos passivos

2.3.2.2 - Modelo de Implementação de Objetos Ativos

A Figura 2.5 mostra um sistema de gerência, que basicamente tem as mesmas condições que o sistema com objetos passivos apresentado anteriormente, com algumas diferenças que serão vistas a seguir.

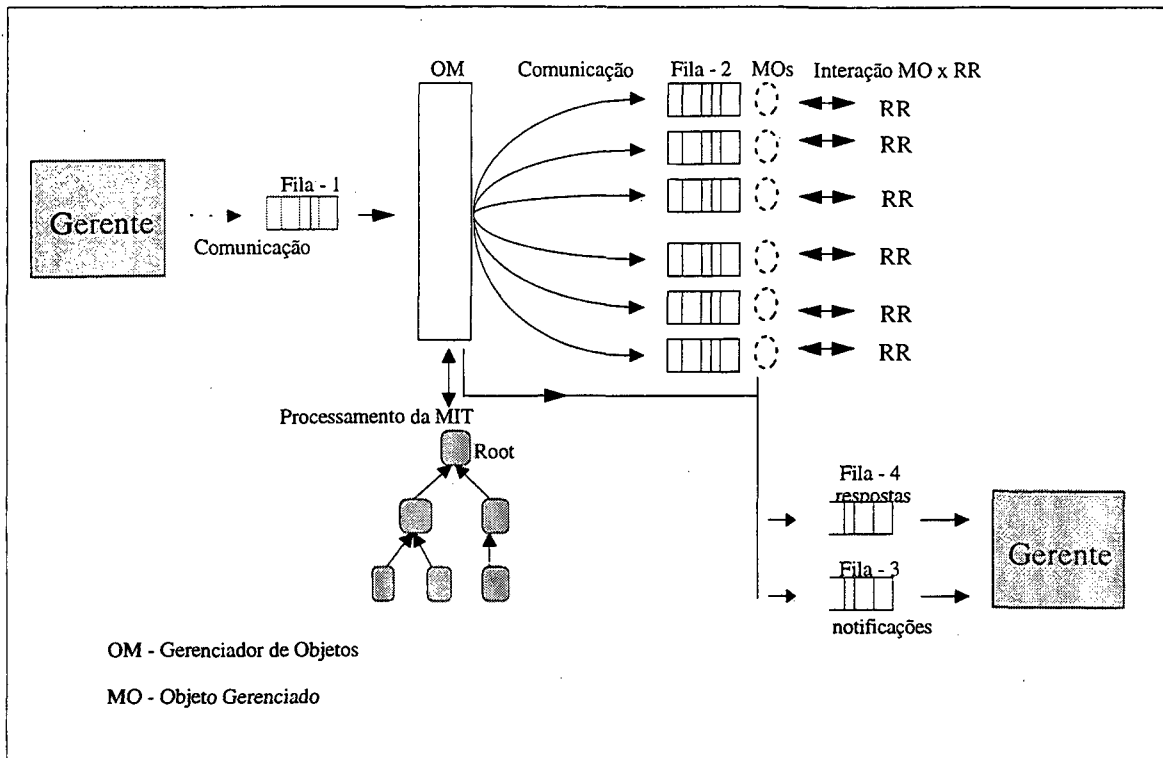


Figura 2.5 - Sistema objeto ativo

Uma estação gerente está ligada a um agente através de um canal de comunicação. Tem-se o OM (*object manager*) que é um gerente de objetos. Ligado a este OM está a MIT. A MIT está a identificação única de cada objeto, onde este objeto comunica-se com o seu recurso real. Os objetos comunicam-se com os recursos de maneira independente, podendo as comunicações dos objetos ocorrerem em paralelo. Como no sistema passivo, a comunicação entre o objeto e o recurso real não é padronizada, dependendo das características de cada um [ISO/TEC91b].

No sistema com objetos ativos os procedimentos acontecem analogamente ao sistema com objeto passivo, com a diferença de que os processos acontecem concorrentemente. Exemplo: pode-se ter uma solicitação de um GET pelo gerente, logo após uma solicitação de um SET em um outro objeto, e ao mesmo tempo os objetos sem requisições pendentes podem estar fazendo *polling* nos recursos reais, e ainda um recurso ter causado alguma falha e estar emitindo uma notificação para o gerente. Isto acontece concorrentemente no sistema de agentes com objetos ativos, pois cada objeto é implementado como uma *thread*, a qual executa concorrentemente com as demais. No sistema com objetos passivos, todas as operações ocorrem de maneira seqüencial. Desta forma, todas as operações de gerenciamento acima ocorrem seqüencialmente, já que o agente como um todo é implementado como uma *thread* apenas.

No sistema de objetos ativos as respostas das requisições são enviadas diretamente para uma fila de respostas. Estas respostas são enfileiradas, onde serão enviadas diretamente para o gerente, sem ter que passar novamente pelo OM. As notificações são enviadas diretamente para uma fila de notificações.

As filas internas do agente são responsáveis pela capacidade de atendimento concorrente de diversas operações de gerenciamento. O OM apresenta uma fila para o recebimento das requisições do gerente. Desta forma, mesmo processando uma requisição anterior, uma nova requisição pode chegar, sendo armazenada na fila para posterior atendimento. Isto implica que o gerente pode enviar quantas requisições forem necessária.

Ao chegar uma requisição na fila do OM, ele realizará o processamento da MIT, identificando o objeto. Uma vez identificado o objeto alvo, o OM colocará a requisição na fila do mesmo, ficando liberado para novos processos. Ambas as filas terão uma política FIFO, onde a primeira requisição a entrar será a primeira a sair.

De forma resumida, o funcionamento de um agente no modelo de objetos ativos pode ser descrito como sendo o seguinte:

- o OM verifica se há requisições pendentes, retirando-as da fila de chegada, identificando o objeto através da MIB, e colocando a requisição na fila do objeto;
- cada objeto verifica sua fila, atendendo requisições pendentes, e enviando as respostas para o gerente. No caso de não haver requisições pendentes, o objeto utiliza o período ocioso para executar operações de polling sobre o recurso real, se especificado;
- uma notificação do objeto é enviada diretamente para a fila de notificações, caso exista.

O paralelismo é conseguido pela implementação *multithreaded* dos componentes do agente, o que ainda elimina o bloqueio que um objeto pode causar no processamento dos outros, como no modelo com objetos passivos.

2.4 - Sumário.

Este capítulo tratou dos conceitos necessários para a continuidade deste trabalho, em relação ao modelo de gerenciamento OSI. Foi descrito o modelo de gerenciamento

OSI, plataformas de gerenciamento, aplicações de gerenciamento, e as formas de implementação de agentes.

Em relação a implementação de agentes, foram descritos aspectos relevantes sobre as principais características de ambas as formas de implementação de agentes: com objetos passivos e com objetos ativos. Também foi mostrada a forma de comunicação entre os componentes de uma aplicação de gerenciamento de redes de computadores, com os dois tipos de agentes. Estes aspectos são de importância para a continuidade deste trabalho, já que são a base da construção dos modelos para a simulação.

No próximo capítulo serão abordados os aspectos de avaliação de sistemas computadorizados, em especial a simulação, que será a técnica utilizada neste trabalho.

Capítulo III

AVALIAÇÃO DE DESEMPENHO

3.1 - Introdução

A avaliação é um processo que é utilizado para responder perguntas sobre o comportamento de um sistema em estudo. Este trabalho é voltado para a área de avaliação de desempenho de sistemas computacionais.

A avaliação de desempenho é um passo importante a ser dado quando do projeto e aquisição de novos sistemas computacionais para este projeto, a avaliação da gerência de redes de computadores. Em todos os casos é necessário comparar performance entre sistemas similares e variadas situações.

Tipicamente, a avaliação de desempenho de modelos de gerência de redes necessitará tanto de conhecimentos nesta área quanto de várias técnicas de análise, as quais incluem alguns conhecimentos de estatística, probabilidade, projetos de experimentos e simulação [Tocher63].

Neste capítulo são descritos conceitos utilizados no trabalho tais como modelagem, simulação e avaliação de desempenho, utilizando-se de exemplos relacionados com gerência de redes, visto ser este o domínio do problema.

3.2 - Técnicas e Medidas de Desempenho

Para o desenvolvimento de um projeto de avaliação de desempenho, é necessário selecionar técnicas de avaliação e unidades de medidas que melhor se adaptam ao modelo

em estudo, bem como definir o projeto de experimentos, seus fatores, níveis e variáveis de respostas.

As técnicas de avaliação de desempenho determinam como será medido o desempenho do trabalho em estudo. Alguns pontos auxiliam a determinar uma escolha correta das variáveis, como tempo necessário, ferramentas, precisão e custos. Ferramentas computacionais com o uso de uma linguagem de simulação de propósito específico são de extrema importância para a execução de análises de desempenho em sistemas computacionais.

As técnicas de avaliação de desempenho permitem que sejam escolhidos os métodos que melhor exprimem as características do sistema real que se deseja avaliar. As três principais técnicas são as medições, a modelagem analítica e a simulação.

Quando se utiliza medições, deve medir direto do sistema real em estudo. É um método bastante simples, mas requer a construção propriamente dita do modelo real ou de um protótipo a ser avaliado.

Para a simulação e a modelagem analítica, faz-se necessário a criação de modelos de representação da realidade e, a partir destes, podem ser realizadas as experimentações e coletas de resultados para uma posterior avaliação. As diferentes técnicas empregadas na simulação e modelos analíticos estão relacionados com o tipo de modelo criado para representar a realidade.

Os modelos analíticos permitem a construção de uma representação da realidade através do uso de formalismo matemático, com ou sem auxílio de estruturas gráficas de representação. As três principais ferramentas de modelagem de sistemas por métodos matemáticos são as Cadeias de Markov [Jain91], as Redes de Petri [Jain91] e a Teoria das Filas [MSG96].

A técnica de simulação, é implementada através de linguagens algorítmicas que podem ser adaptadas a uma linguagem específica de simulação ou a uma linguagem de propósito geral. Modelos analíticos são de construção mais simples, entretanto, geralmente consomem muito tempo de desenvolvimento.

A Tabela 3.1 compara alguns critérios na escolha da técnica de avaliação apropriada conforme [Ferrari78]:

Critério	Modelo analítico	Simulação	Medições
Cenário	qualquer	qualquer	pós-protótipo
Tempo necessário para realização	pequeno	médio	alto
Ferramentas	lápiz e papel	linguagem de computador	instrumentação
Precisão	baixa	moderada	alta
Avaliação - escolha	fácil	moderada	difícil
Custo	pequeno	médio	alto
Comerciability	baixa	média	alta

Tabela 3.1- Considerações que podem levar à escolha da técnica utilizada

Considerando o tempo disponível para a avaliação, tem-se na simulação um tempo favorável para realizar os experimentos. Na modelagem analítica o tempo necessário para realizar os experimentos é considerado pequeno, enquanto no caso das medições se tem um alto tempo para realizar toda avaliação [Lucas79].

As ferramentas tem-se modelagem, linguagens de simulação e instrumentos de medidas. A simulação é uma das mais poderosas ferramentas de análise disponíveis para o projeto e operação de sistemas ou processos complexos. Em um mundo altamente competitivo, a simulação torna-se uma ferramenta poderosa para o planejamento, projeto e controle de sistemas [Pedgen95].

O nível de precisão desejado é uma consideração importante. Para modelos analíticos são necessárias muitas tentativas, suposições e simplificações, onde o resultado pode não apresentar uma precisão desejada. Já com as medições, tem-se uma variabilidade muito grande nos parâmetros envolvidos, tais como: configuração do sistema, tipo de carga de trabalho e tempo da medição. Isto pode caracterizar muito tempo desperdiçado e a necessidade de realizar mais de uma experimentação. Para a simulação, muitos detalhes podem ser agregados com menos suposições, o que facilita a obtenção de precisão.

Em termos de comerciabilidade, a melhor escolha recai, certamente na construção de um protótipo em funcionamento, mostrando resultados. Por sua vez, um modelo de simulação, com animação, convence muito mais do que apenas cálculos no papel.

3.3 - Simulação

A simulação probabilística, aqui denominada simplesmente simulação, teve sua origem como extensão do Método de Monte Carlo [Marshall56]. Este método foi proposto por Von Neumann e Ulam para a soluções de problemas matemáticos onde o tratamento analítico não se mostrava variável. Isto se deu durante a Segunda Guerra Mundial, ao longo das pesquisas no Laboratório de Los Alamos, que resultaram na construção da primeira bomba atômica. Ao que tudo indica, por razões de sigilo e também pelo típico de abordagem utilizada.

A aplicação original do Método de Monte Carlo voltava-se à avaliação de integrais múltiplas, usando-se a amostragem aleatória para o estudo do problema da difusão de neutrons. Logo, no entanto, verificou-se que ele poderia ser aplicado na solução de diversos outros problemas matemáticos complexos¹, muitos deles de natureza determinística como em [Marshall56], ao fazer uma revisão do Método de Monte Carlo.

Passado o momento de euforia inicial que caracteriza todo novo método, suas principais deficiências foram mais bem reconhecidas, notadamente o grande trabalho computacional envolvido e a baixa precisão dos seus resultados. Deve-se lembrar que nesta época no final da década de 40, os computadores começavam a tornar-se realidade, mas ainda com um desempenho bastante abaixo dos da máquinas atuais.

Sendo restritos os recursos computacionais, as atenções voltaram-se para a obtenção de resultados mais precisos, mas sem que se aumentassem, proporcionalmente, os tempos de processamento envolvidos. Este esforço resultou no desenvolvimento das técnicas de redução de variância, muitas delas objetivando um controle parcial do processo de amostragem.

A simulação computacional de sistema consiste na utilização de determinadas técnicas matemáticas empregadas em computadores digitais, as quais permitem imitar o funcionamento de, praticamente, qualquer tipo de operação ou processo do mundo real. Em [LaKe91] há uma definição mais completa, abrangendo todo o processo de simulação:

¹ “a idéia intrigante, descoberta ou promulgada por Von Neumann e Ulam, segundo a qual podemos utilizar métodos de amostragem aleatória para a solução de problemas matemáticos de natureza determinística.”

“Simulação é o processo de projetar um modelo de um sistema real e conduzir experimentos com este modelo com o propósito de entender seu comportamento e/ou avaliar estratégias para sua operação”. Entende-se simulação como um processo mais amplo, compreendendo não somente a construção de um modelo, que permita imitar o comportamento de um sistema real, mas também todo o método experimental que se segue, buscando sobretudo:

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

3.3.1 - Sistemas

Sistemas são definidos como “um conjunto de objetos, como pessoas ou máquinas, por exemplo, que atuam e interagem com a intenção de alcançar um objetivo ou um propósito lógico” [LaKe91]. Os objetivos de um particular estudo é que irão definir que objetos devem constituir um sistema.

3.3.2 - Modelos

No encaminhamento de um estudo de simulação, uma das principais etapas consiste na modelagem do sistema sob estudo, para que se possa observar seu comportamento sob determinadas condições, na busca de sua compreensão. Este processo de imitação e criação de uma história artificial da atuação e desempenho dos sistemas reais (modelagem, simulação e experimentação), pressupõe, na maioria das vezes, uma série de simplificações sobre seu funcionamento de forma que possamos, cientificamente, estudá-los e entendê-los.

3.4 - Terminologia de Modelagem e Simulação de Sistemas

Uma série de termos são usualmente utilizados quando da conceituação dos elementos básicos envolvidos na modelagem e na simulação de sistemas. Para uma melhor compreensão dos mesmos, as definições serão acompanhadas de exemplos de sistemas simples, com apenas um único servidor, como os que se apresentam em áreas tão distintas

como redes de computadores, serviços ou sistemas computacionais. Por exemplo: na visão de um administrador de redes de computadores, este sistema poderia ser uma rede de três computadores interligados entre si. O objetivo do administrador seria verificar o comportamento da rede diante de diferentes serviços solicitados como impressão, consulta a base de dados, todos esses serviços através da rede, de acordo com a carga de trabalho [Jain91].

3.4.1 - Variáveis de Estado

As variáveis cujos valores definem o estado do sistema são conhecidas como variáveis de estado. Se interrompermos uma rodada de simulação, só poderemos retomá-la, a partir do ponto em que foi interrompida se, e somente se, forem conhecidos os valores de todas as variáveis de estado no exato momento da interrupção. Desta forma, as variáveis de estado constituem o conjunto de informações necessárias à compreensão do que está ocorrendo no sistema, num determinado instante no tempo, com relação aos objetos de estudo. Variáveis de estado definidas numa determinada investigação podem ser completamente diferentes daquelas definidas em outro estudo, mesmo quando se trata do mesmo sistema [Jain91].

3.4.2 - Eventos

A ocorrência de uma mudança de estado em um sistema é chamada de evento. Num sistema de redes de computadores com três máquinas ligadas entre si através de um canal de comunicação com filas de espera de serviços, podemos considerar como exemplos de eventos: a emissão de requisições por uma máquina e a saída de serviços por outro computador [Pidd84].

3.4.3 - Modelos Discretos e Modelos Contínuos

Estes conceitos estão associados a idéia de sistemas que sofrem mudanças de forma discreta ou contínua ao longo do tempo. Esta classificação diz respeito às variáveis do modelo do sistema, e não ao sistema propriamente dito.

No modelo discreto, as variáveis de estado mantêm-se inalteradas ao longo de intervalos de tempo e mudam seus valores somente em pontos bem definidos, também

conhecidos como *tempo de ocorrência do evento*. Já nos modelos contínuos, as variáveis variam seus valores continuamente ao longo do tempo. Há também a possibilidade de modelagens mistas, com variáveis contínuas e discretas num mesmo modelo.

3.4.4 - Entidades e Atributos

Uma entidade representa um objeto. Esta entidade pode ser dinâmica, movendo-se através do sistema, ou estática, servindo outras entidades. Exemplo de entidades dinâmicas, são as requisições que chegam e saem da estação agente após serem executadas na MIT e MIB do agente. Já entidades estáticas podem ser, por exemplo os objetos gerenciados[Lucas79].

As características próprias das entidades, isto é, aquelas que as definem totalmente, são chamadas de atributos. Várias entidades semelhantes possuem os mesmos atributos. Os valores dos atributos é que as diferenciam entre si. Os atributos associados às entidades também dependem do tipo de investigação que está sendo levada a efeito.

3.4.5 - Recursos

Um recurso pode ter a capacidade de servir uma ou mais entidades dinâmicas ao mesmo tempo, ou seja, operando paralelamente. É possível que uma entidade dinâmica opere com mais de uma unidade do recurso ao mesmo tempo, ou com diferentes recursos ao mesmo tempo. Se uma entidade dinâmica não puder apoderar-se do recurso solicitado, ela deverá aguardar pelo mesmo em uma fila. O processamento de uma fila, isto é, a forma como a mesma será gerenciada depende, fundamentalmente, das políticas operacionais adotadas no sistema ou no modelo que o representa. A política de tratamento de fila mais comum é a FIFO [Tocher63](First In, First Out), onde o primeiro a chegar na fila será o primeiro a ser atendido pelo recurso. Outras formas podem ser adotadas.

3.5 - Justificativas para o uso de simulação

Dentre os motivos que justificam o uso de simulação, em sua maior popularidade, pode-se ainda citar [Morgan84]:

- Modelos mais realistas:

A maior liberdade de que dispõe-se na construção de um modelo de simulação é, certamente, um bom motivo. A simulação não obriga a enquadrar um problema em determinado molde para que se possa obter uma solução, como ocorre, por exemplo, no caso de programação linear. Assim, em lugar de soluções exatas para “problemas aproximados”, tem-se agora soluções aproximadas para “problemas mais reais”.

- Processo de modelagem evolutivo:

Importante benefício do processo de modelagem em simulação é o seu caráter evolutivo. Assim, começando com um modelo relativamente simples pode-se, aos poucos, identificar de maneira mais clara as peculiaridades do problema; em função deste aprendizado, tem-se também condições de aperfeiçoar este modelo, incorporando novas variáveis e/ou relações.

- Perguntas do tipo “what if ?” (“e se ?”)

Muitas vezes, em lugar de busca de uma solução, o objetivo resume-se em tomar mais claras as possíveis conseqüências de um conjunto de decisões. A simulação é propícia à formulação de perguntas do tipo “what if ?” que permitem avaliar, com base no modelo, o efeito de possíveis mudanças de cenário ou de diferentes decisões.

- Problemas “mal-estruturados”:

Muitos problemas da vida real são do tipo “mal-estruturados”, ou seja, referem-se a situações em que dispomos apenas de um conhecimento parcial sobre suas variáveis e/ou relações. A simulação é uma das poucas ferramentas para o estudo deste tipo de problema, no entanto, é também verdade lembrar que, em casos como este, a solução obtida deve ser vista com a máxima cautela. Nas palavras de [Simon69], “uma simulação nunca será melhor do que suas próprias premissas”.

- Facilidade de comunicação:

Outro bom motivo para sua maior utilização é a facilidade de comunicação proporcionada por um modelo de simulação, em geral, muito mais fácil de se compreender do que um conjunto de complicadas equações matemáticas. Atualmente, esforços de pesquisa vêm sendo dedicados ao desenvolvimento da simulação visual, mostrando-se num vídeo uma imagem animada do problema. Uma das principais vantagens da simulação visual consiste na melhor comunicação com

os demais elementos interessados ou que venham a ser afetados pelas decisões tomadas. Com isso, aumentamos a probabilidade de aceitação do estudo e de que sua implementação seja bem-sucedida.

- Soluções “rápidas, porém pobres” (“Quick and dirty”):

A simulação permite obter as chamadas soluções “rápidas, porém pobres”, fazendo com que, rapidamente, tenhamos uma noção da ordem de grandeza dos valores em jogo.

- Maior disponibilidade de recursos computacionais

Finalmente, a maior disponibilidade de recursos de hardware e software tem tornado bem mais simples a tarefa de construção e teste dos modelos de simulação, assim como sua programação e seu processamento. Além disso, a difusão de uso do computador no apoio à tomada de decisão tem também levado a um maior interesse em relação a técnicas de análise mais sofisticadas, como é o caso da simulação. Convém ainda notar que, exceção feita a problemas muito complexos, poderemos normalmente fazer um estudo de simulação num microcomputador de 16 ou 32 bits.

3.6 - Vantagens e Desvantagens da Simulação

A técnica da simulação e seus conceitos básicos são em geral, facilmente compreensíveis e justificáveis, tanto para usuários quanto para os gerentes que tomam a decisão de aplicá-la em seus problemas. Em contraste com os modelos de otimização, um modelo de simulação é executado ao invés de resolvido. As diferenças destas duas abordagens implicam que o modelo simulado permite análises quase que a todo instante. Quase todos os modelos de simulação são do tipo entrada-saída, isto é, são modelos interativos aos quais se fornece dados de entrada e obtém-se respostas específicas. Não são por natureza modelos de otimização. Não oferecem a possibilidade de busca de uma solução ótima, servindo, somente, para a análise do comportamento do sistema sob condições específicas [Fishman78].

Apesar da simulação ser uma excelente ferramenta de análise, é preciso conhecer com um pouco mais de profundidade tanto as suas vantagens, quanto a suas desvantagens. As duas listas abaixo apontam algumas delas. Inicialmente tem-se alguns de seus benefícios, listados por [Pedgen95] e por [LaKe91]:

- Uma vez criado, um modelo pode ser utilizado inúmeras vezes para avaliar projetos e políticas propostas;
- A metodologia de análise utilizada pela simulação, permite a avaliação de um sistema proposto, mesmo que os dados de entrada estejam, ainda, na forma de “esquemas” ou rascunho;
- A simulação é, geralmente, mais fácil de aplicar do que métodos analíticos;
- Hipóteses sobre como ou por que certos fenômenos acontecem podem ser testadas para confirmação;
- Podemos compreender melhor quais variáveis são as mais importantes em relação a performance e como as mesmas interagem entre si e com os outros elementos do sistema;
- A identificação de “gargalos”¹, preocupação maior no gerenciamento operacional de inúmeros sistemas, tais como fluxos de materiais, fluxo de informações e de produtos, pode ser obtida de forma facilitada, principalmente com a ajuda visual;
- Um estudo de simulação costuma mostrar como realmente um sistema opera, em oposição à maneira com que todos pensam que ele opera.

Embora sejam inúmeras as vantagens, o processo de simular apresenta algumas dificuldades, como as que estão listados em [BFS83]:

- A construção de modelos requer treinamento especial. Envolve arte e portanto o aprendizado se dá ao longo do tempo com a aquisição de experiência. Dois modelos de um sistema construídos por dois indivíduos competentes terão similaridades mas dificilmente serão iguais;
- Os resultados da simulação são, muitas vezes, de difícil interpretação. Uma vez que os modelos tentam capturar a aleatoriedade do sistema, muitas vezes existem dificuldades em determinar quando uma observação realizada durante uma execução se deve a alguma significativa relação no sistema ou a aleatoriedade construída no modelo;
- A modelagem e a experimentação associadas a modelos de simulação consomem muitos recursos, principalmente tempo. A tentativa de simplificação na modelagem ou nos experimentos objetivando economia de recursos costuma levar a resultados

¹ Gargalo é um ponto do sistema onde o tempo de chegada das entidades é menor que o tempo de atendimento pelo servidor. Neste caso, há um congestionamento de entidades esperando processamento, congestionamento este que aumenta com o passar do tempo.

insatisfatórios. Em muitos casos a aplicação de métodos analíticos (como a teoria das filas, por exemplo) pode trazer resultados menos ricos e mais econômicos.

3.7 - Passos em um Estudo de Simulação

Os passos envolvidos num estudo de simulação podem ser sumariamente descritos como na Figura 3.1. [Freitas97]

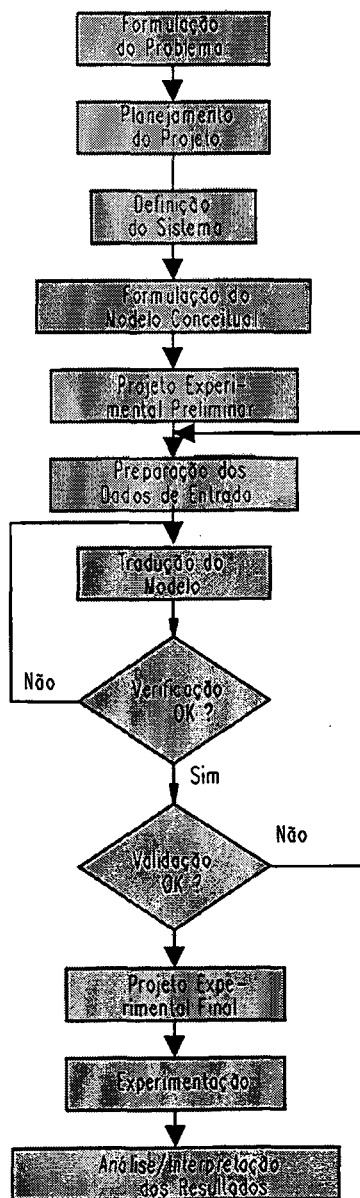


Figura 3.1 – Fluxograma de um Estudo de Simulação

- 1) **Formulação do Problema:** os propósitos e objetivos do estudo devem ser claramente definidos.
- 2) **Planejamento do Projeto:** os recursos necessários para a realização da simulação são determinados. Estes recursos incluem pessoal, suporte, gerência, hardware, software, etc.
- 3) **Definição do Sistema:** é necessário definir os limites e restrições dos sistema ou processo a ser simulado.
- 4) **Formulação do Modelo Conceitual:** traçar um esboço do sistema, de forma gráfica ou algorítmica, definindo componentes, descrevendo as variáveis e interações lógicas que constituem o sistema.
- 5) **Projeto Experimental Preliminar:** consiste numa seleção preliminar, que respostas serão esperadas do modelo, sob que forma e em que extensão.
- 6) **Preparação dos Dados de Entrada:** identificar e coletar os dados de entrada necessários ao modelo.
- 7) **Tradução do Modelo:** codificar o modelo numa linguagem de simulação apropriada.
- 8) **Verificação e Validação:** confirmar que o modelo opera de acordo com a intenção do analista, e que os resultados são confiáveis.
- 9) **Projeto Experimental Final:** projetar um conjunto de experimentos que produza a informação desejada, determinando as variáveis de respostas esperadas, em função da combinação dos níveis dos fatores escolhidos como relevantes.
- 10) **Experimentação:** executar as simulações para a geração dos dados desejados e para a realização das análises de sensibilidade.
- 11) **Análise e Interpretação dos Resultados:** traçar inferências sobre os resultados alcançados pela simulação.

3.8 - Linguagem de Simulação

Nas seções anteriores, tratou-se da formulação dos modelos e conceitos relacionados à avaliação de desempenho. Após a modelagem, é necessário implementar estes modelos numa linguagem de computador [CACI83].

Sem dúvida, uma maneira de programar experiências de simulação consiste em escrever um programa especial para a simulação de cada sistema a ser estudado, em uma das bem conhecidas linguagens gerais para computadores, tais como Fortran, Pascal, Algol, C, etc. Com certeza esta alternativa oferece ao programador a máxima flexibilidade em [MHBK62]:

- projeto e formulação do modelo do sistema em estudo;
- tipo e formato dos dados de saída gerados;
- espécies de experiências de simulação executadas com o modelo.

Contudo, a principal deficiência deste modo de ação é a dificuldade encontrada na confecção dos programas nestas linguagens de programação de propósito geral. Esta dificuldade levou ao desenvolvimento de um certo número das assim chamadas linguagens de simulação, que visam simplificar a tarefa de escrever programas de simulação para diferentes tipos de modelos e sistemas. Entre as linguagens de simulação que foram desenvolvidas encontram-se:

- GPSS [Purpose83];
- SIMSCRIPT [MHBK62];
- GASP [Kiviat63];
- SIMULATE [HCSR64];
- ARENA/SIMAN [ARENA95].

Estas linguagens foram desenvolvidos tendo em vista os seguintes objetivos:

- Produzir uma estrutura generalizada para escrever modelos de simulação;
- Fornecer uma maneira rápida de converter um modelo de simulação em um programa de computador;
- Fornecer um modo rápido de introduzir alterações no modelo de simulação que possam ser rapidamente refletidas no programa de máquina;
- Fornecer uma maneira flexível de obter saídas em forma útil para a análise.

As linguagens de simulação disponíveis hoje em dia diferem consideravelmente quanto às possibilidades que apresentam de serem aplicadas a tipos particulares de sistemas e também quanto à sua capacidade em tornar os processos de simulação mais ou menos automáticos [GoMi86].

O programa de simulação que melhor se adapta a um determinado estudo de simulação depende da natureza do sistema e da habilidade do programador que realiza o estudo. De uma maneira geral, um aumento na flexibilidade de um programa de simulação é obtido às custas de um estudo mais aprofundado dos processos de programação[BFS83].

A escolha do ambiente de simulação se deve por algumas facilidades em vistas: O fato de ter a linguagem adquirida pelo departamento e alguns estudos efetuados nas diversas linguagens de simulação. O ambiente Arena também é uma linguagem de simulação de propósito específico, voltada para transformar os sistemas em estudo em um modelo de simulação para a coleta de dados. Esta linguagem apresenta características que se adaptam perfeitamente ao presente trabalho.

3.8.1 - Características de um Programa para Simulação

Cada programa de simulação possui uma característica básica que o diferencia uns dos outros: “a visão do mundo”. Este termo significa a forma com que o programa foi concebido, ou como ele vê um sistema a ser simulado. Isto tem como consequência que a maneira como os dados serão fornecidos a cada programa é diferente dos outros.

3.8.2 - O ARENA

O ARENA foi criado pela empresa *Systems Modeling* em 1993 e ele é o sucessor de um outro produto de sucesso na mesma empresa, o SIMAN, o qual foi desenvolvido em 1982. Este produto foi uma evolução da arquitetura do GPSS que durante anos, foi líder entre os produtos de simulação-de-uso-geral. Em 1990 ele recebeu um complemento chamado de CINEMA que adicionava maiores habilidades gráficas. Este conjunto foi amplamente melhorado e, a partir de 1993, passou a se denominar ARENA.

O ARENA possui um conjunto de blocos (ou módulos) que são utilizados para descrever uma aplicação real. Estes blocos funcionam como comandos de uma linguagem de programação. Obviamente foram projetados sob a ótica de simulação e, por isso, facilitam muito esta tarefa de programação.

Além de permitir a construção de modelos de simulação, o ARENA possui ainda duas outras ferramentas muito úteis: o processador de dados de entrada (*Input Analyzer*) e o processador de resultados (*Output Analyzer*). Possuem papéis importantes no ambiente utilizado apresentam a parte de entrada/saída dos dados como ainda tem recursos de efetuar importantes comparações estatísticas.

Outro fator relevante para a escolha desta linguagem foi a sua disponibilidade no Curso de Pós-Graduação em Ciência da Computação da Universidade Federal de Santa

Catarina, onde o presente trabalho foi desenvolvido. Desta maneira, havia a possibilidade de consulta aos manuais, exemplos, e projetistas que já haviam utilizado a linguagem, facilitando assim o esclarecimento de possíveis dúvidas.

3.9 - Variáveis e Medidas de Desempenho

Para este estudo, ou qualquer outro caso, um conjunto de critérios de performance ou medidas devem ser escolhidas. Uma maneira de preparar este conjunto é fazer uma lista dos serviços que o sistema a ser avaliado oferece. Cada serviço requisitado pode ser realizado corretamente, realizado incorretamente ou totalmente rejeitado.

Se o sistema realizar o serviço corretamente, sua performance poderá ser medida pelo tempo gasto para realizar a operação, a taxa na qual o serviço é realizado, e os recursos consumidos para realização do serviço, como mostra a tabela 3.2.

Serviço	Medidas
TEMPO	Capacidade de resposta
TAXA	Produtividade
RECURSO	Utilização

Tabela 3.2 - Medidas relacionadas com a realização satisfatória do serviço

O tempo pode ser medido em relação à duração do tempo total de uma entidade no sistema, bem como o tempo necessário para o processamento num determinado recurso. Basicamente, quanto menor o tempo gasto no processamento, maior a capacidade de resposta do sistema ou entidade [Simon69].

A taxa de serviço representa a produtividade do sistema ou do recurso. Quanto maior o volume de entidades processadas pelo sistema ou recurso, maior a taxa de serviço.

A medida a ser tomada num recurso é basicamente o tempo em que ele permanece ocupado processando entidades. Quanto mais tempo é gasto no processamento, sem que o recurso fique ocioso, maior é a utilização do mesmo.

Os dados obtidos durante uma sessão de medidas de desempenho precisam ser tratados de alguma forma. É importante lembrar que estas medidas são as peças chave de todo o processo de avaliação de desempenho. A seguir temos as medidas comumente usadas para avaliar performance de sistemas. Em muitos casos, as definições dadas devem

adaptar-se à determinado sistema em particular. No caso deste trabalho as definições serão adaptadas para a aplicação de gerência de redes.

- Tempo de resposta é definido como o intervalo de tempo transcorrido desde que o usuário requisita um serviço e a resposta é obtida. No modelo de simulação de um sistema de gerenciamento de redes, o tempo de resposta será o tempo que uma entidade (requisição do gerente) vai permanecer no sistema ou seja, saindo do gerente, chegar ao agente e retornar ao gerente.
- Throughput - para redes de computadores, o throughput é medido em pacotes por segundo (pps) ou bits por segundo (bps). No caso de CPUs, esta medida pode ser dada em milhões de instruções por segundo (MIPS), ou milhões de pontos flutuantes por segundo (MFLOPS). O throughput cresce à medida que a carga no sistema cresce, até um certo limite. Este limite é o ponto onde a carga começa a degradar o serviço e o sistema não responde a todas as requisições.
- A utilização de um recurso é medida como fração de tempo. É a razão do tempo em que o recurso está ocupado pelo tempo total medido. O período em que o recurso não está sendo utilizado é chamado de tempo ocioso (idle). Em gerência de redes sempre existe o interesse em balancear este tempo para que um recurso seja usado tanto quanto outro recurso, ou seja, o agente sempre acessando sua base de informação.
- Confiabilidade do sistema é medida em termos de probabilidade de ocorrência de erros ou pelo tempo médio decorrido entre falhas.
- Disponibilidade de um sistema é definida como uma fração de tempo em que o sistema está disponível no momento da requisição de serviço feita pelo usuário. O tempo em que o sistema não está disponível é chamado de *downtime*, e o tempo em que o sistema está disponível é chamado de *uptime*.

3.10 - Projeto Experimental

Quando se deseja avaliar o desempenho de um sistema, mais de um fator precisa ser levado em consideração. Uma análise criteriosa requer que os efeitos de cada fator sejam isolados dos outros, de modo que conclusões apropriadas sejam feitas. Basicamente, num projeto experimental, deseja-se:

- projetar um conjunto de experimentos para a simulação;
- desenvolver um modelo que melhor descreva os dados obtidos;
- estimar a contribuição de cada alternativa;
- estimar o intervalo de confiança de cada alternativa;
- verificar se as alternativas apresentam diferenças significantes e se o modelo é adequado.

Alguns termos são freqüentemente utilizados no projeto e análise de experimentos. Os principais são [Jain91]:

- **Variável de resposta:** é uma variável cujo valor, na saída do sistema, contém um valor significativo para avaliação do desempenho do sistema.

- **Fatores:** cada variável do modelo que descreve o sistema, e que pode assumir diferentes valores relevantes, é denominada de fator.

- **Níveis:** são os valores que cada fator pode assumir. Num modelo de computador, um fator pode ser a CPU, e os níveis possíveis para este fator podem ser 8088, 8086, 80286, etc.

- **Replicações:** são as repetições de um determinado experimento. Se um experimento é realizado 3 vezes, diz-se que foram feitas 3 replicações.

- **Projeto:** é a especificação do número de experimentos, das combinações dos níveis dos fatores, e a quantidade de replicações para cada experimento.

3.10.1 - Tipos de Projetos Experimentais

Existem diversos tipos de projetos experimentais. Os três principais são:

Projeto simples: um dos fatores é escolhido, e são realizados experimentos variando os níveis deste fator, buscando determinar o nível ótimo. A seguir, este nível é fixado, e varia-se os níveis de outro fator, buscando novamente o nível ótimo. Este processo é repetido para todos os fatores. É importante ressaltar que não são exploradas todas as combinações de fatores e níveis possíveis, o que pode levar a conclusões erradas.

Projeto Fatorial Completo: utiliza todas as combinações de todos os níveis dos fatores. Por exemplo, se existirem três fatores, um com 2 níveis, um com 3 níveis e outro com 4 níveis, são necessários $2 \times 3 \times 4 = 24$ experimentos. Uma vantagem é que todas as combinações são exploradas, sendo possível determinar a combinação que melhor atende as características desejadas. Uma desvantagem é o rápido crescimento da quantidade de

experimentos necessários quando temos um crescimento do número de fatores ou fatores com muitos níveis. Um caso especial é quando todos os fatores possuem apenas dois níveis. Assim, com k fatores, teremos 2^k experimentos. Neste caso, o projeto é chamado de projeto fatorial completo 2^k .

Projeto Fatorial Fracionário: como o projeto fatorial completo necessita de um grande número de experimentos, pode-se retirar, com algum critério, algumas combinações de níveis dos fatores, sem que os resultados apresentem, na média, variações muito significativas. No caso de um projeto fatorial completo 2^k , pode-se retirar uma fração p de experimentos, sendo então necessários apenas 2^{k-p} experimentos. Desta forma, se um fração for retirada ($p = 1$), o número de experimentos já cai pela metade.

Maiores considerações sobre projetos experimentais podem ser encontrados em [LiWa95]. O objetivo desta seção é tão somente apresentar os conceitos e terminologia básica.

3.11 - Análise dos Resultados da Simulação

Toda a simulação de sistemas estocásticos gera resultados que contém variações. É possível fazer uma inferência sobre estes resultados ou, mais especificamente, sobre os resultados que medem o desempenho dos sistemas modelados, através da realização de análises apropriadas.

Algumas questões que devem inicialmente ser tratadas dizem respeito aos seguintes aspectos:

- Qual é a duração apropriada de uma rodada de simulação?
- Como interpretar corretamente seus resultados?
- Como analisar corretamente as diferenças obtidas em cada um dos experimentos?

Estas e outras questões são objeto dos tópicos que serão analisados a seguir. Várias técnicas são utilizadas ao longo deste trabalho, desde a avaliação visual de dados dispostos graficamente até técnicas estatísticas um pouco mais sofisticadas, como médias móveis, a definição de intervalos de confiança e análise de variância.

3.11.1 - Análise Estatística

A determinação de intervalos de confiança para as variáveis de interesse que medem o desempenho do sistema é um componente fundamental no processo de análise de resultados. Da estatística sabemos que o intervalo de confiança compreende um intervalo numérico que possui uma probabilidade igual a $1 - \alpha$ de incluir o verdadeiro valor da variável ou medida de desempenho sob análise, onde $1 - \alpha$ é denominado o nível de confiança do intervalo [Freitas97].

Desta forma, três fatores influenciam a largura do intervalo de confiança:

1. o número de replicações n (*quantas rodadas de experimentos serão realizadas*);
2. o nível de confiança ($1 - \alpha$) pré-definido pelo analista;
3. variação (S^2) associada à medida de desempenho sob análise.

O relacionamento entre estes três fatores ocorre da seguinte maneira:

1. na medida em que se aumenta o número de replicações, reduz-se a largura do intervalo de confiança;
2. na medida em que o nível de confiança aumenta, aumenta também a largura do intervalo de confiança. O intervalo com 99% de confiança será maior do que o aquele com 95% de confiança, uma vez mantidos os demais elementos;
3. na medida que a variação (S^2) aumenta, cresce a largura do intervalo de confiança.

A variância da medida de desempenho de um sistema, e por consequência de seu modelo, é dependente dos parâmetros deste sistema. Se ao analista não é permitida a modificação de elementos do modelo os quais alterem sua natureza e, por consequência, a variabilidade, sua liberdade de atuação fica restrita a possibilidade de lidar com alterações no número de replicações e no nível de confiança. Desta maneira, com base nas afirmações acima temos três situações que definem as relações entre o número de replicações e o nível de confiança:

1. Se o nível de confiança é fixo, um grande número de replicações resultará num menor intervalo de confiança.
2. Se o tamanho do intervalo de confiança é fixo, um grande número de replicações resultará em um maior grau de confiança.

3. Se o número de replicações é fixo, um alto nível de confiança resultará um grande intervalo de confiança.

Para fins de análise, os sistemas são classificados como *terminais* e *não-terminais*. Como pode ser visto na próxima seção, onde pode-se observar que a definição do número de replicações que devem ser realizadas em um estudo onde se emprega a técnica da simulação é um fator relevante na determinação do intervalo de confiança.

3.11.2 - Verificação e Validação dos Modelos de Simulação

No desenvolvimento de um modelo de simulação é preciso estar seguro que o mesmo, esteja corretamente implementado, isto é sem erros de sintaxe e/ou lógica, bem como que ele seja representativo do sistema real (existente ou não). Estes dois passos são conhecidos como verificação e validação de um modelo e são os passos iniciais de um estudo completo de simulação.

A qualidade e a validade de um modelo de simulação é medida pela proximidade entre os resultados obtidos e comparado com os resultados obtidos do sistema real. Uma vez que uma série de pressupostos e simplificações sobre o comportamento do sistema real são aplicados no desenvolvimento do modelo, qualquer tomada de decisão com base em seus resultados deve ser precedida de uma avaliação de sua qualidade e apropriação.

Esta avaliação esta subdividida em duas etapas. A primeira consiste em avaliar se, apesar do pressupostos e das simplificações implementadas, o modelo ainda é válido, isto é, comporta-se à semelhança do sistema real. A segunda é saber se estes pressupostos e simplificações foram corretamente implementadas no modelo computacional. As duas etapas são chamadas de validação e verificação, respectivamente.

Em outras palavras, validação relaciona-se com a representatividade dos pressupostos enquanto que verificação diz respeito a correção, isto é, ausência de erros, das implementações computacionais. Pode-se dizer também que verificação é a etapa na qual assegura-se que o modelo realiza o que se pretende que seja realizado

3.11.3 - Tipos de Sistemas

Os sistemas chamados de terminais apresentam condições iniciais fixas (condições estas que o sistema volta a assumir ao início de cada rodada de simulação) e um evento que determina um fim natural para o processo de simulação (isto é, pode-se programar o fim de uma rodada de simulação). Como exemplos podemos citar os restaurantes, bancos, lojas comerciais, etc. [Freitas97].

As condições iniciais da maioria dos sistemas terminais apontam para um sistema *vazio e disponível*, isto é, os recursos do sistema estão disponíveis e não existem entidades ou *clientes* esperando em filas. Para os sistemas comerciais, as condições de término geralmente constituem-se de um evento que aponta para o fim de um dia de trabalho com 8 a 10 horas de funcionamento.

Já os sistemas considerados não-terminais não possuem condições iniciais fixas nem um evento que determina o fim do processo de simulação. Como exemplos mais comuns deste tipo de sistemas podemos citar os serviços de 24 horas, os hospitais, os sistemas de comunicação, etc..

3.11.3.1 - Sistema Terminais

O objetivo de simular sistemas terminais é compreender seu comportamento ao longo de um período predeterminado e com duração fixa. Uma vez que as condições iniciais e o período simulado são fixos, o único fator controlável é o número de replicações. Desta forma, os procedimentos de análise para sistemas terminais consistem em realizar um determinado número replicações, verificar a variância da medida de desempenho selecionada e determinar se o intervalo de confiança resultante encontra-se dentro de limites aceitáveis.

3.11.3.2 - Sistemas Não-Terminais

A maioria dos sistemas modelados são rotulados como não-terminais. Como já foi dito, diferentemente dos sistemas terminais, não temos um estado inicial predefinido nem tampouco um evento caracterizando o encerramento do período de simulação. Além disso, dois problemas básicos devem ser contornados. O primeiro deles trata do descarte das observações que pertencem ao período transiente, isto é, aquelas observações que são

fortemente influenciadas pelo estado inicial do sistema. O segundo problema diz respeito ao período de simulação.

Uma vez que não mais temos um evento característico de término da simulação, para podermos avaliar a variância da medida de interesse, considerando os descartes da fase transiente, as técnicas de avaliação exigem longas simulações. O tempo total de uma rodada de simulação. Este é o segundo problema a ser tratado. O que pode ser resolvido utilizando técnicas de sistemas não-terminais, facilmente encontrada em [Freitas97].

3.12 - Avaliação de Desempenho em Redes de Computadores

Outros trabalhos efetuados na área de avaliação de desempenho em redes de computadores buscando obter informações atualizadas nesta área. Foi feito um estudo relatando alguns destes trabalhos que fazem parte do estado da arte deste assunto.

Em [LiDu95] foi desenvolvido um trabalho o qual busca a análise de desempenho de três protocolos ponto-a-multiponto que provêem uma transferência de dados confiável em ambientes de satélite a alta velocidade. As estratégias de recuperação de erros se servem de mecanismos de retransmissões continuamente repetidas e dotam os receptores de memória a fim de obter uma alta eficiência mesmo em condições bastante desfavoráveis em relação a taxa de erro. Os resultados numéricos mostram que os protocolos apresentam um bom desempenho e que as modificações melhoram o rendimento através do uso da análise.

O desenvolvimento de ferramentas didáticas foi um trabalho elaborado por [BROO96], para a formação de recursos humanos nesta área com o uso de simulação. O sistema PAPO (*Prêt-à-Porter*), um modelo de computação associado a uma metodologia pedagógica construtiva, destinado ao ensino de redes de computadores e sistemas distribuídos. O PAPO integra um simulador de redes a uma estrutura LAN/WAN, permitindo um mapeamento da teoria à aplicação prática.

Já em [MSG96] a avaliação de desempenho de redes locais é alvo de atenções neste texto, onde o objetivo não se limita a fazer um apanhado do material bibliográfico sobre o assunto, mas se entender e por em prática ferramentas que se engaje na avaliação de desempenho da arquitetura de redes locais.

O ambiente VISE (*Visual Interactive Simulation Environment*) [LiWa95], em desenvolvimento na UFRGS, é composto de um conjunto integrado de ferramentas que permite a simulação de maneira visual e interativa. A linguagem utilizada pelo sistema para modelagem, é a linguagem SIMSCRIPT.

Em se tratando de programas de simulação voltados para redes de computadores, temos em [GoMi86], um produto da COMNET III, para o desempenho de redes de computadores de comunicação utiliza análise de simulação orientado ao objeto. Na programação é recomendado o uso de padronização de bibliotecas e modelagem *off-the-shelf*. Para situações onde o modelo desenvolve ajuda de baixo custo ou *fine-tune* de mesmo aspectos de um modelo padronizado um MODSIM baseado em ambiente de desenvolvimento de programação orientado ao objeto é provido para *overriding* e comportamento do objetos de COMNET III. O modelo construído pode adicionar objetos parametrizados ou seus próprios custos de objetos e objetos de bibliotecas. COMNET III traz a integração envolvendo gráficos para criação de modelos, execução e análise a qual permite a interação com o modelo enquanto este é executado.

O trabalho desenvolvido em [FHC96] trata de um método de avaliação de performance do uso recente de um programa de comunicação. Possíveis técnicas para performance são examinadas e a descrição de um método particular são apresentados. Resultados selecionados de um estudo de benchmark são dados para estabelecer a validação do modelo de performance. Um tema de um estudo é a comparação com uma prévia implementação baseada em transpor.

3.13 - Sumário

Neste capítulo foi tratado de conceitos e termos relacionados com técnicas de avaliação de desempenho. Dentre as alternativas propostas, foi justificada a escolha da simulação para a realização deste trabalho, que utiliza a avaliação de desempenho na comparação e avaliação de alternativas de projetos de sistemas de gerenciamento de redes de computadores.

Em relação à simulação, que é a técnica utilizada neste trabalho, foram vistos conceitos básicos, linguagens, e as melhores formas de aplicar simulação em ambientes de sistemas computacionais. Demonstrou-se a necessidade de um bom tratamento de dados de entrada e de fontes seguras, associadas com a aleatoriedade dos processos e eventos envolvidos nos sistemas simulados.

A definição do projeto de experimentos também é abordada, uma vez que um bom projeto é crucial para a confiabilidade dos resultados obtidos. As formas de análise dos resultados também foram apresentadas, para fornecer mecanismos de avaliar os resultados obtidos posteriormente neste trabalho [OMS96].

O próximo capítulo irá utilizar os conceitos apresentados neste capítulo, bem como no capítulo anterior, para construir os modelos de simulação dos componentes da plataforma de gerenciamento a serem comparados.

Capítulo IV

MODELAGEM DOS AMBIENTES SIMULADOS

4.1 - Introdução

A aplicação de avaliação de desempenho em sistemas de computação tem apresentado um crescimento considerável. Este fator é justificado pela possibilidade de melhorias nos sistemas novos e atuais através da avaliação, fornecendo resultados importantes que podem ser usados no projeto e desenvolvimento de novos sistemas.

Este capítulo apresenta os modelos de simulação que foram desenvolvidos para a comparação de duas abordagens de gerenciamento de redes de computadores, descrevendo os elementos que serão avaliados (simulados), com o uso de técnicas e medidas. São descritos, de forma detalhada, os dois modelos que serão comparados. As medidas de avaliação bem como os fatores que serão comparados estão descritos no projeto de experimento.

O capítulo está organizado da seguinte maneira: descrição geral do trabalho, apresentação detalhada dos modelos de objetos passivos e ativos e projeto de experimento.

4.2 - Visão Geral dos Modelos Desenvolvidos

O estudo em consideração é uma avaliação de desempenho de dois modelos de implementação de agentes. Um modelo utiliza a implementação de agente com objetos passivos, representando, genericamente, a forma como a maioria das plataformas atuais implementam os seus agentes. O outro utiliza a implementação adotada na plataforma de gerência de redes[Matias96], na qual o agente é implementado como objetos ativos [JoO196].

Neste trabalho foram modelados dois tipos de gerentes, de acordo com o seu modo de operação. Um gerente pode ser síncrono ou assíncrono. Um gerente que opera de modo

síncrono espera a chegada da resposta da requisição anterior antes de emitir uma nova requisição. No modo assíncrono, a emissão de novas requisições é feita mesmo que as respostas das requisições anteriores tenham chegado.

Os dois tipos de agentes foram modelados de modo a representar as características mais relevantes de cada modelo de agente. Com os resultados da simulação destes dois modelos, é possível comparar o desempenho dos dois modelos de implementação de agentes.

A seguir os modelos dos gerentes e agentes são apresentados, com a descrição de suas principais características. Este modelos foram construídos no Arena 2.1 [ARENA95].

4.2.1 - Modelagem dos Gerentes

Nesta seção serão descritos a modelagem das entidades gerentes, os quais estão divididos em dois tipos: síncronos e assíncronos.

4.2.1.1 - Gerente Síncrono

O sistema com um gerente síncrono é modelado como um mecanismo de criação de entidades, e um processo (recurso²) com uma fila associada. Esta construção pode ser vista na Figura 2.1.

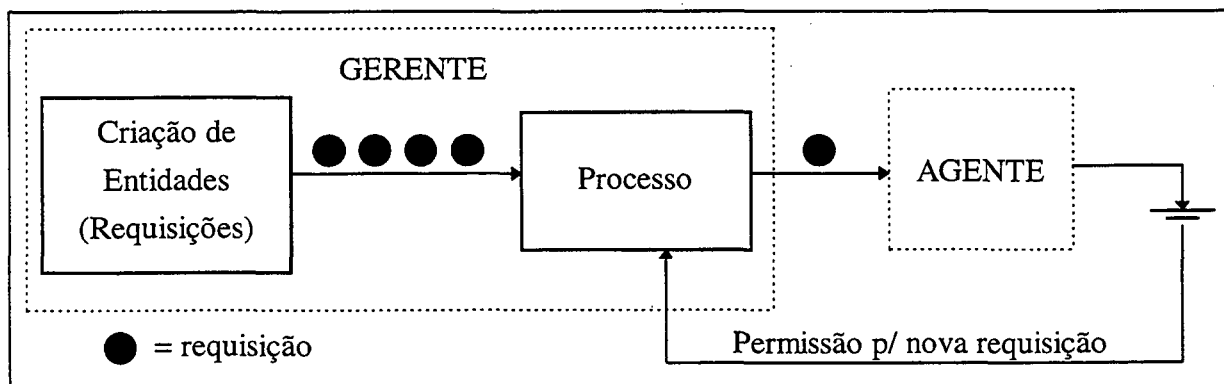


Figura 2.1 - Modelagem do Gerente Síncrono

² Recurso, no presente contexto, representa um processo, na linguagem de simulação, o qual consome um determinado tempo no processamento de uma entidade. Por exemplo, uma CPU (recurso) consome um determinado tempo (processamento) para executar cálculos sobre um determinado registro (entidade).

Uma requisição num sistema de gerenciamento é representada no modelo, por uma entidade que se desloca dentro do sistema. Um gerente síncrono irá enviar uma nova entidade (requisição) somente quando a entidade anterior deixar o sistema. O fato da entidade anterior deixar o sistema (modelo) pode ser interpretado como o recebimento da resposta da requisição pelo gerente num sistema real. Neste momento, uma nova entidade pode ser criada e tomada pelo recurso (nova requisição).

4.2.1.2 - Gerente Assíncrono

Um gerente assíncrono é modelado de forma muito semelhante ao gerente síncrono. O que diferencia um do outro é a forma de liberação do recurso que modela a emissão de requisições pelo gerente. A modelagem de um gerente assíncrono pode ser visto na Figura 2.2.

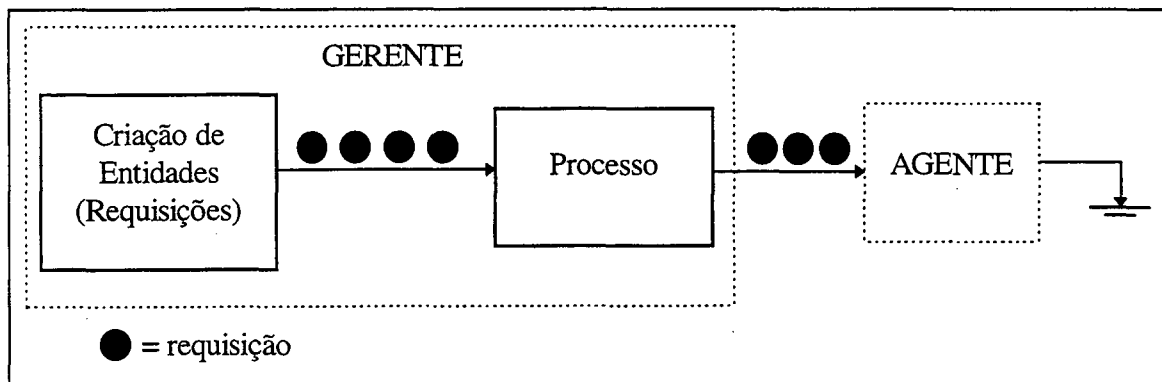


Figura 2.2 - Modelagem do Gerente Assíncrono

Como já mencionado anteriormente, o gerente não precisa esperar a resposta da requisição prévia para emitir uma nova. Assim, novas entidades são criadas no sistema mesmo que as anteriores ainda não o tenham deixado. Em outras palavras, o recurso é liberado logo após a passagem da entidade, mesmo que a entidade não tenha saído do sistema (recebimento da resposta pelo gerente).

4.2.2 - Modelagem dos Agentes

Como no caso da modelagem dos gerentes, para os agentes foram construídos dois modelos diferentes, sendo um modelo para o agente com objetos passivos e um modelo para o agente como objetos ativos. A seguir cada um dos modelos é explicado.

4.2.2.1 - Agente com Objetos Passivos

O agente com objetos passivos é modelado com diversos recursos. O principal deles é o gerenciador de objetos (OM), o qual recebe as requisições dos gerentes. O processamento da requisição por este recurso representa a pesquisa na MIT no agente real. Outros recursos modelam os objetos, cujo processamento representando a comunicação dos objetos com os recursos reais³ do sistema. Esta modelagem pode ser vista na Figura 2.3.

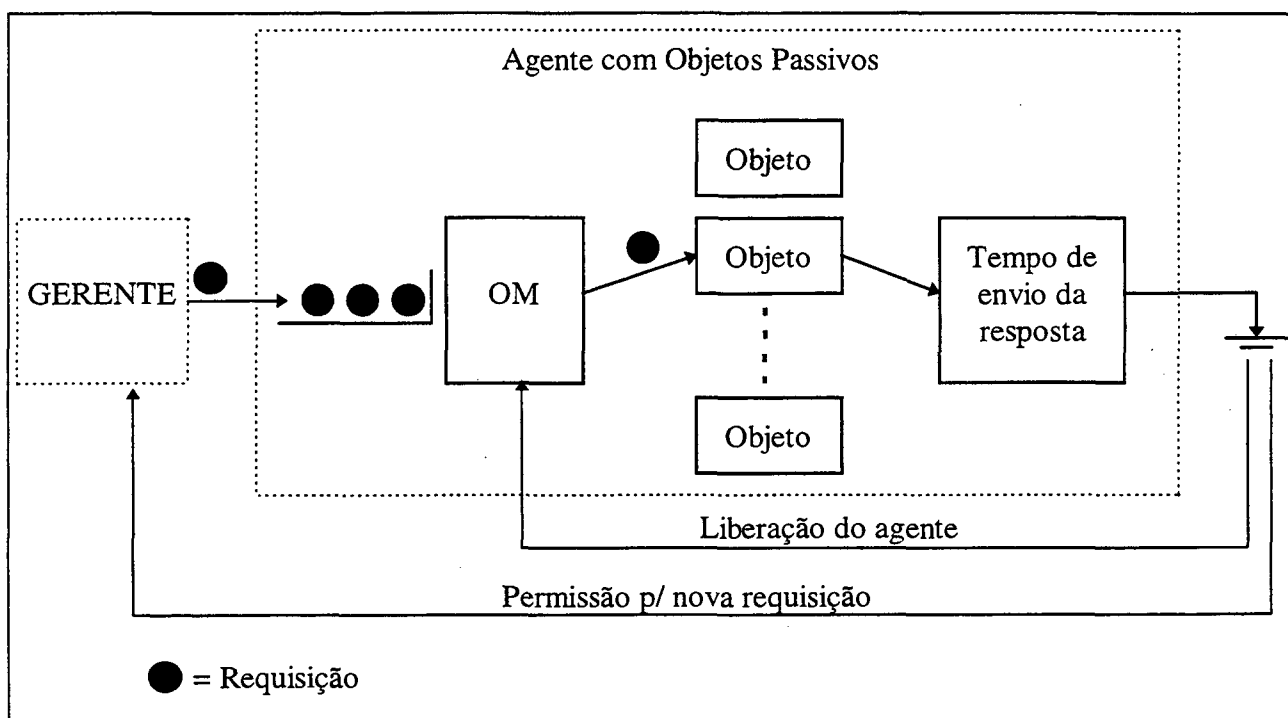


Figura 2.3 - Modelagem do Agente com Objetos Passivos

Como no sistema real, o agente com objetos passivos pode receber apenas uma requisição de cada vez, na modelagem foi associada uma fila ao recurso OM. A retirada de uma entidade desta fila representa, no sistema real, a chegada de uma nova requisição ao agente. O processamento da entidade pelo OM consome um determinado tempo, que representa a pesquisa na MIT. A seguir, a requisição é enviada para o recurso que modela o objeto gerenciado.

³ Recurso real é um recurso computacional sendo gerenciado. Exemplos de recursos reais são impressoras, hubs, switches, estações de trabalho, etc.

Cada objeto gerenciado, como já mencionado, é representado por um recurso. A simulação da sua interação com o recurso real é feita através da passagem de tempo até a sua liberação. Após este tempo (término da interação), a entidade que representa a requisição é enviada para um outro recurso, que novamente simula o tempo de envio da resposta ao gerente.

Uma vez transcorridos estes retardos a entidade deixa o sistema, sinalizado para o modelo do gerente que houve uma resposta para sua requisição. Esta sinalização é feita somente quando o gerente tem o modo de interação síncrono. Da mesma forma, há uma sinalização para o recurso OM dizendo que o processamento da requisição terminou, podendo assim, iniciar o processamento da próxima requisição pendente.

As notificações são geradas na modelagem dos objetos. Para modelar a prioridade das notificações sobre as requisições no sistema real, quando uma notificação é gerada por um objeto, a mesma é enviada para a fila do OM, sendo inserida no início da mesma, sendo assim processada antes de qualquer requisição pendente, conforme pode ser visto na Figura 2.4.

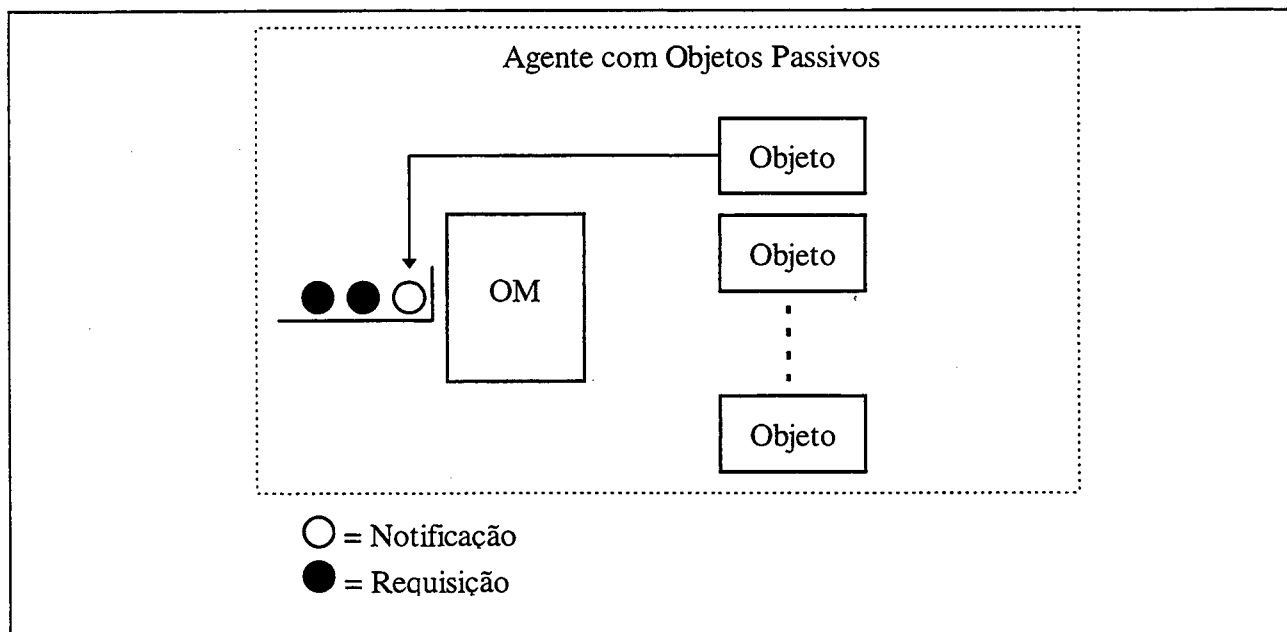


Figura 2.4 - Tratamento das Notificação na Modelagem do Agente com Objetos Passivos

Uma notificação, na sua criação, não consome tempo, o que não condiz com o sistema real, onde um tempo de *polling* é necessário. A solução encontrada foi enviar a

notificação novamente para o objeto, sendo tratada da mesma forma que uma requisição, da maneira descrita acima.

4.2.2.2 - Agente com Objetos Ativos

Na modelagem do agente com objetos ativos, é necessário modelar o processamento paralelo dos diversos objetos e do gerenciador de objetos do sistema real. Este paralelismo é real, proveniente da implementação *multithreaded* do agente [MaSp97]. Devido as diferenças conceituais deste agente em relação ao agente com objetos passivos, o modelo também é diferente, como pode ser visto na Figura 2.5.

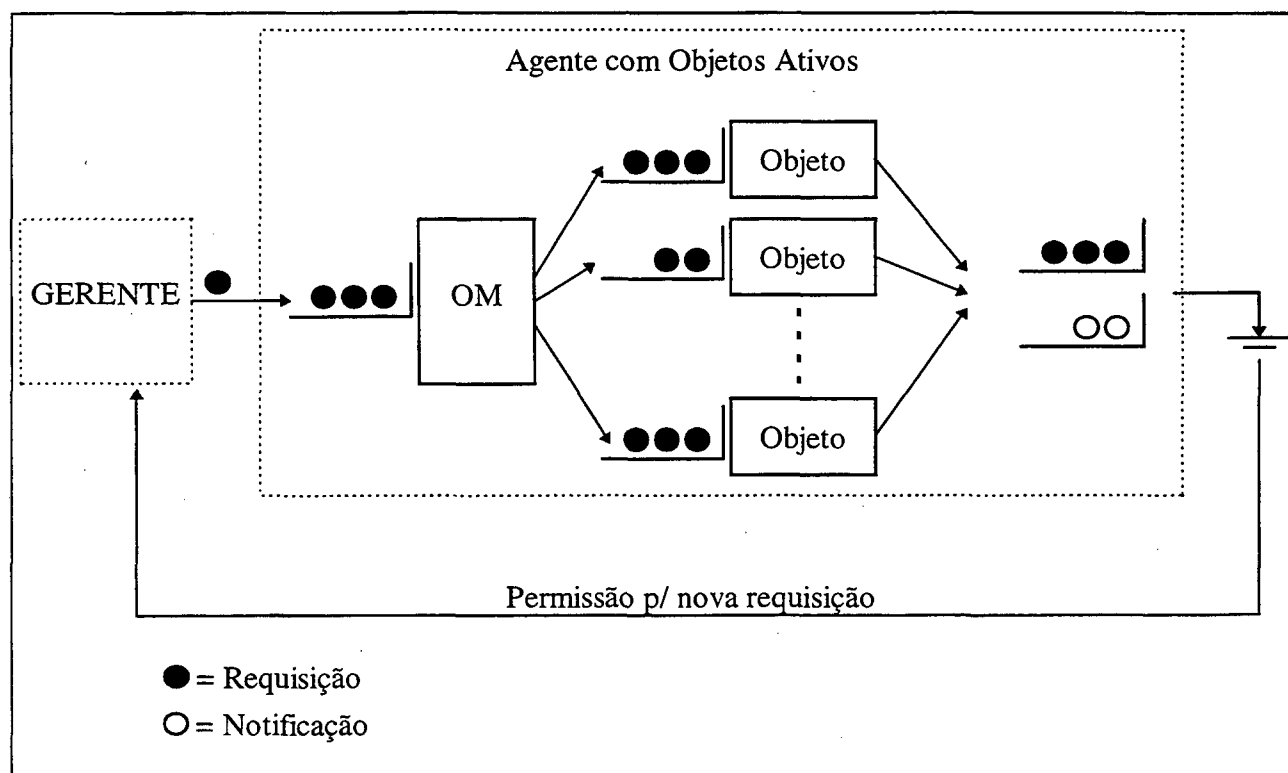


Figura 2.5 - Modelagem do Agente com Objetos Ativos

O recurso que modela o gerenciador de objetos, neste caso, não precisa esperar que pelo processamento de uma requisição antes de atender a próxima. Assim, este recurso retira constantemente entidades (requisições de gerenciamento) de sua fila de entrada, e após um atraso que representa o processamento da MIT, envia a entidade para a fila do recurso que modela o objeto.

O recursos que modelam os objetos, neste modelo, têm associados a si uma fila de requisições pendentes. Cada recurso que modela um objeto retira a próxima entidade da fila

de entrada, processa por um determinado tempo (comunicação com o recurso real), e envia uma entidade, que representa a resposta, para uma fila de respostas.

No caso de notificações, uma nova entidade que representa uma notificação é criada pelo modelo do objeto. Novamente, para modelar o tempo de processamento da notificação, a entidade é inserida na fila do objeto, na primeira posição, já que uma notificação tem prioridade sobre requisições. Este tratamento das notificações pode ser visto na Figura 2.6. Após o processamento, esta entidade é enviada para uma fila de notificações, diferente da fila de respostas.

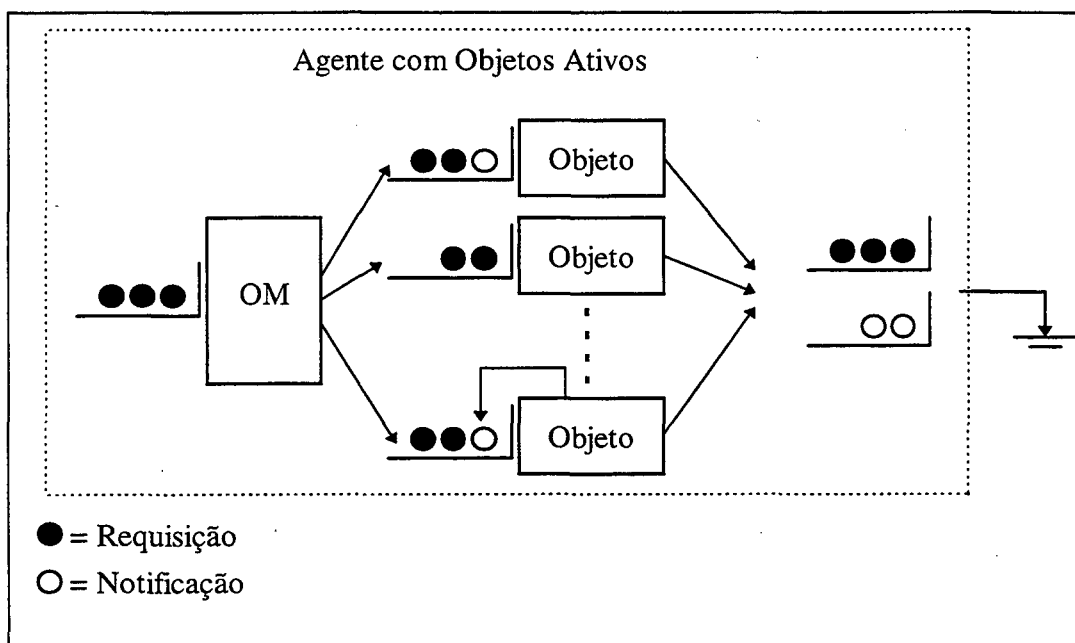


Figura 2.6 - Tratamento das Notificação na Modelagem do Agente com Objetos Ativos

As filas de respostas e de notificações são processadas por recursos, cujos tempos de processamento modelam o tempo de envio das mesmas para o gerente. Após este processamento, se o modelo do gerente tiver uma modo de interação síncrono, é feita uma sinalização para o mesmo, informando o final do processamento da requisição, permitindo o envio de outra.

Um fator relevante neste sistema, como dito acima, é o fato de que cada processo (OM, objetos gerenciados e as filas de respostas e notificações) são implementados em *threads*, que executam concorrentemente através de *time-slice*. No modelo de simulação, é possível introduzir esta característica através de tempos de processamento para cada recurso.

Cada recurso processa sua fila de entrada durante uma fatia de tempo. Terminado este tempo, outro recurso inicia seu processamento, e assim por diante, de maneira circular. Recursos cuja fila de entrada está vazia passa o processamento para outro, mesmo que sua fatia de tempo não tenha ainda terminado.

Como cada entidade tem um tempo de processamento no recurso, caso a fatia de tempo do recurso termine antes que a entidade seja processada, a mesma é re-inserida no início da fila do recurso, com o tempo ajustado para que seu processamento continue na próxima fatia de tempo do recurso.

É importante ressaltar que a serialização das requisições, requerida pelo CMIP, é mantida pela própria política de gerenciamento das filas dos recursos. Estas filas seguem a política FIFO (*First In, First Out*). Desta forma, requisições oriundas de um mesmo gerente para um mesmo objeto serão respondidas na ordem exata da sua emissão.

4.3 - Principais Diferenças entre os Modelos

A tabela 4.1 apresenta as principais diferenças entre os modelos de objetos ativo e modelo de objetos passivos.

Diferenças	Modelo Objeto Passivo	Modelo Objeto Ativo
Operação com diferentes objetos gerenciados	Sequencial	Operam concorrentes
Quantidade de Requisições em trânsito no Agente	Apenas 1	Múltiplas
Estado do Gerente Assíncrono após cada requisição	Bloqueado	Ñ Bloqueado
Dependência de execução entre os objetos da MIB	Dependente	Independente

Tabela 4.1 - Principais diferenças entre agentes com objetos ativos e com objetos passivos

4.4 - Sumário

Neste capítulo foram descritos os modelos, para fins de simulação, dos componentes da plataforma de gerenciamento a serem comparados.

Foram descritos os mecanismos dos modelos para os dois tipos de gerentes envolvidos: um para o gerente síncrono e um para o gerente assíncrono. Também os modelos dos agentes foram descritos, com modelos distintos para o agente com objetos passivos e para o agente com objetos ativos. As formas de simulação das principais características dos componentes foi abordada, de forma a refletir o funcionamento dos componentes do sistema real.

No próximo capítulo serão apresentados os projetos experimentais, os resultados obtidos com as simulações, e as análises destes resultados.

Capítulo V

EXPERIMENTAÇÃO E RESULTADOS DA SIMULAÇÃO

5.1 - Introdução

Neste capítulo serão apresentados os parâmetros dos experimentos realizados, e os resultados obtidos com a simulação dos modelos descritos no capítulo anterior. Através da análise destes resultados, a comparação dos modelos de gerenciamento com objetos passivos e ativos será possível. A análise dos resultados será feita através de análises estatísticas e pela inspeção visual.

Os pontos a serem avaliados são a performance de cada sistema modelado em relação à alguns critérios a serem estabelecidos, bem como avaliar a importância de cada fator dentro do sistema. Será também verificado se os resultados da simulação condizem com os resultados esperados no sistema real, de modo a validar os modelos.

Primeiramente serão descritos os parâmetros de simulação dos dois modelos estudados neste trabalho. Em seguida serão apresentados os resultados obtidos com a simulação. Após a apresentação dos resultados, serão feitas as análises propostas.

5.2 - Técnicas e Ferramentas de Avaliação de Desempenho Utilizadas

Como visto na seção 3.1, as técnicas de simulação contribuem e proporcionam a validação do modelo em estudo. Como o uso de avaliação de desempenho é importante para a comprovação dos resultados na busca de prováveis gargalos do sistema, serão aplicados dois métodos de avaliação de desempenho: medições e simulação.

O presente trabalho utilizou dados provenientes de medições realizadas nas implementações feitas em [Matias96], portanto, foram inseridos nas implementações do agente com objetos ativos pontos de coletas de dados. Neste mesmo trabalho, medições são feitas nos tempos dos agentes com objetos passivos. Estes dados foram utilizados para alimentar o processo de simulação.

Para a aplicação da implementação na linguagem de simulação foi necessário definir os dados que serviram como parâmetros dos modelos. Estes parâmetros foram coletados da implementação do sistema real feitos em [Matias96], são os seguintes:

- 85 % das operações de gerência serão requisições e 15% serão notificações.
- Tempo de processamento do gerente de objetos (OM):
 - 0,073 s Objetos Passivos
 - 0,075 s Objetos Ativos
- Tempo de processamento dos objetos gerenciados:
 - 2,0094 s Objetos Passivos
 - 2,0508 s Objetos Ativos
- Tempo de processamento das requisições e notificações:
 - 0,033 s (para ambos)

Na construção dos modelos, como o objetivo deste trabalho é mostrar o tempo de resposta das operações de gerência, optou-se por modelar apenas as operações que envolvem monitoração. Em outras palavras, operações que exigem que o gerente envie uma requisição e que esta resulte no envio de uma resposta ao gerente.

Como as notificações influenciam no tempo de resposta das requisições, devido a prioridade das notificações, foi necessário modelá-las também. Com base nos dados dos sistemas reais, foi modelado um fluxo de operações de gerenciamento com 85% de requisições e 15% de notificações.

A escolha da técnica de simulação para aplicar nos modelos descritos nas seções 4.2.2.1 e 4.2.2.2 foi baseado em estudos feitos em [Ferrari78], [Jain91], [Pedgen95] e [LaKe91].

5.2 - Experimentos

Nesta seção é descrito o projeto de experimentos utilizado para a avaliação e comparação do desempenho dos dois sistemas modelados. Como descrito na seção 3.5, para construir um projeto de experimento, é necessário determinar os fatores relevantes no modelo, seus níveis, as variáveis de resposta e o número de experimentos necessários.

Para avaliar ambos os modelos foi necessário a escolha de alguns critérios, que representam parâmetros equivalentes que possam ser avaliados, tanto no modelo ativo, quanto o passivo. Assim, foram utilizados os mesmo fatores para todos os experimentos. Também as variáveis de respostas utilizadas em ambos os casos foram as mesmas, sendo descritas a seguir.

5.2.1 - Fatores e Níveis.

Estes fatores representam as variáveis de análise escolhidas para representar ambos os modelos. Os níveis são as variações que estas variáveis podem sofrer. Para ambos os modelos, foram identificados os seguintes fatores e seus respectivos níveis:

Fator 1 – Número de Gerentes

Neste fator busca-se estudar o impacto do número de gerentes emitindo requisições sobre os sistemas em estudo. Basicamente, um número maior de gerentes irá gerar uma carga maior de requisições.

Para fins de simulação, o nível máximo para o número de gerentes escolhido é 5, número considerável, possível de ser encontrado num sistema de gerenciamento real. O nível mínimo, logicamente, é de apenas 1 gerente no sistema.

Número de Gerentes

- Nível (+): 5 gerentes
- Nível (-): 1 gerente

Fator 2 – Número de Objetos

Este fator pode ser visto como um dos mais importantes na modelagem dos sistemas em estudo. Basicamente, o que se pretende mostrar é a diferença de desempenho dos agentes ativos e passivos quando do tratamento de um fluxo considerável de requisições. Aumentando-se o número de objetos associados aos agentes, espera-se que

haja um fluxo maior de informações. Pela natureza paralela dos objetos ativos, um melhor desempenho é esperado neste modelo, com menores tempos de resposta e maior número de requisições atendidas.

Os níveis máximos e mínimos utilizados foram de 30 e 10 objetos, respectivamente. Estes números foram obtidos através de estudos que demonstraram haver pouca diferença entre os modelos, quando o número de objetos é pequeno (abaixo de 10). O número máximo representa um número considerável de objetos conectados a um agente num sistema de gerenciamento real.

Número de Objetos

- Nível (+): 30 objetos
- Nível (-): 10 objetos

Fator 3 – Carga de Requisições

Este fator determina a quantidade de requisições em trânsito no sistema. Na prática, o que determina a quantidade de requisições no sistema é a frequência de envio das mesmas pelo gerente. Quanto maior for o intervalo entre os envios de requisições, menor será a carga de requisições no sistema, e vice-versa. Portanto, este fator determina o intervalo de tempo entre o envio de duas requisições por um gerente.

Para fins de simulação, foram gerados valores diferentes para o modelo ativo e para o modelo passivo, com base nos tempos de processamento das requisições. Isto se deve ao fato de que os tempos médios de resposta são menores no modelo ativo, já que o agente processa requisições para diferentes objetos em paralelo. Já no modelo passivo, o agente atende às requisições de maneira seqüencial, mesmo para objetos diferentes. Na simulação, uma distribuição exponencial foi utilizada para caracterização da variabilidade deste fator.

Analisando-se os modos de processamento das requisições pelos agentes, pode-se dizer que há diferentes pontos de gargalo nos modelos passivo e ativo. No modelo passivo, o agente irá atender uma requisição de cada vez, podendo assim, haver um congestionamento de requisições esperando atendimento pelo agente. No caso do modelo ativo, não há espera da requisição para ser processada pelo agente. Porém, se mais de uma requisição chegar para um mesmo objeto, haverá um congestionamento de requisições esperando por processamento, já que um objeto pode atender apenas uma requisição de cada vez.

Os níveis para este fator foram calculados com o objetivo de se alcançar uma carga do sistema próxima do máximo⁴, de modo a poder avaliar o desempenho dos modelos da melhor forma possível (sem sobrecarga nem ociosidade). Para uma simulação com nível baixo neste fator, foi utilizada a metade do valor que gera a carga máxima. Os valores estão discriminados abaixo.

Carga de Requisições

- Nível (+): 1.1 s (objetos ativos) e 12.44 s (objetos passivos)
- Nível (-): 2.2 s (objetos ativos) e 24.88 s (objetos passivos)

Fator 4 - Tipo de Interação

Este fator determina qual é o tipo de interação do gerente com o agente. Existem dois modos de interação do gerente: assíncrono e síncrono. O modo de interação determina qual o comportamento do gerente em relação às respostas das requisições enviadas, descritos a seguir.

No modo assíncrono, após o envio de uma requisição, o gerente não aguarda a chegada da resposta para o envio de outra requisição. Assim, pode haver mais de uma requisição de um mesmo gerente em trânsito no sistema. Outra consequência é que a relação de ordem entre o envio das requisições e a chegada das respostas não é necessariamente respeitada.

Já no modo síncrono o gerente só pode enviar uma nova requisição depois do recebimento da resposta da requisição anterior. Desta forma, para um determinado gerente, só haverá uma requisição em trânsito no sistema. Além disso, a relação de ordem entre o envio das requisições e suas respostas é obviamente respeitada. Assim, os possíveis níveis para este fator são definidos abaixo

Interação

- Nível (+): Assíncrono
- Nível (-): Síncrono

Uma vez que foram identificados quatro fatores relevantes comuns aos dois modelos, e cada fator apresenta dois níveis distintos, o projeto de experimentos adotado

⁴ Uma carga é considerada como máxima se a mesma gerar uma taxa de ocupação dos pontos de gargalo próxima de 100%.

será do tipo fatorial completo 2^k , com $k = 4$. Desta forma, serão necessários 16 experimentos, para que todas as possíveis interações entre os fatores possam ser avaliados. Como este é um número relativamente pequeno, não foi necessário utilizar um projeto fatorial fracionário.

Também garante-se que não serão perdidas informações relevantes, de modo que todas as combinações de níveis são exploradas, permitindo uma avaliação completa dos modelos. Como as combinações de níveis terão que ser simuladas para ambos os modelos, o presente trabalho apresentará resultados de simulação de 32 experimentos.

5.2.2 - Variáveis de Resposta

Para fins de análise dos resultados das simulações, serão avaliadas algumas variáveis de resposta, que determinam os aspectos críticos dos sistemas simulados. O que se deseja avaliar, basicamente, é o desempenho de cada modelo de agente (ativo e passivo), no que diz respeito à quantidade de requisições atendidas e o tempo necessário para o processamento completo de uma requisição, desde seu envio até o recebimento de sua resposta.

Assim sendo, duas variáveis de resposta foram escolhidas como relevantes para o presente estudo, as quais são:

- Tempo total gasto por uma requisição no sistema;
- Número de requisições atendidas em ambos modelos.

O tempo total de uma requisição no sistema é o tempo gasto, em média, para o processamento de uma requisição, desde a sua geração até o recebimento da resposta pelo gerente. No modelo, este tempo é calculado, para o agente com objetos passivos, como a soma dos tempos de processamento da entidade (requisição) pelos recursos. Descarta-se o tempo de espera das entidades nas filas. Isto porque no sistema real não existem filas. Apenas uma entidade está no sistema. Já no caso do agente com objetos ativos, como existem filas no sistema real, os tempos de espera nas filas dos recursos são computados no tempo de processamento da entidade (requisição).

Já o número de requisições atendidas diz respeito à quantidade de requisições efetivamente atendidas pelos agentes. No final da simulação, este total quer dizer, o total

das requisições que deixaram o sistema (respostas recebidas pelos gerentes). No caso, as requisições que ficaram pendentes nas filas do modelo não são computadas.

Além destas duas variáveis de resposta, consideradas as mais relevantes, serão apresentadas duas outras medidas: o número de notificações e a razão entre as duas variáveis de desempenho.

O número de notificações é calculado da mesma maneira que o número de requisições. Como as notificações têm prioridade sobre as requisições, elas têm mais chances de deixar o sistema, pois são inseridas no início das filas dos modelos.

A razão entre o número das requisições atendidas e tempo médio de atendimento permite a comparação entre sistemas que apresenta valores distintos para ambas as variáveis. O objetivo é atender o maior número de requisições possível, gastando o menor tempo possível de processamento para cada requisição. Assim, quanto maior for o valor resultante nesta razão, melhor o desempenho do sistema modelado.

Existem outras variáveis de respostas que poderiam ser coletadas. Entre elas pode-se mencionar o tempo de atendimento de uma requisição pelo gerenciador de objetos, o tempo de interação agente-recurso real, e o tempo de processamento de uma requisição pelo objeto gerenciado. Algumas destas variáveis são dependentes do congestionamento dos dois sistemas, os quais estão sendo medidos pelo tempo de resposta das requisições, números de requisições atendidas e número de notificações.

Embora possam ser relevantes para o estudo de um determinado modelo de gerenciamento, não se mostraram relevantes para o objetivo deste trabalho, que é a comparação entre os modelos ativo e passivo. Estas variáveis mostraram variação idêntica nos dois modelos, sendo portanto inúteis como fator de comparação.

5.3 - Análise dos Resultados da Simulação

Os sistemas estudados neste trabalho podem ser classificados como sistemas não-terminais. Isto se deve ao fato de não haver um ponto de parada, podendo haver o trânsito de requisições, respostas e notificações ao longo de um tempo infinito. Desta forma, será utilizada a metodologia de simulação para sistemas não-terminais descrita na seção 3.6.3. A seguir serão tratados os procedimentos para a obtenção dos resultados da simulação dos dois modelos estudados neste trabalho.

5.3.1 - Caracterização das Replicações

Para a determinação do período de simulação de cada um dos experimentos os seguintes procedimentos foram adotados:

- 1) realização de uma longa rodada de simulação (execução de um lote com 10.000 segundos).
- 2) observação visual dos gráficos que mostram o comportamento das variáveis de desempenho, objetivando a determinação dos períodos transientes e de estabilidade.

Em ambos os modelos, observou-se que a estabilidade ocorre após cerca de 50 segundos de simulação, período adotado como *warm-up*⁵ para as replicações executadas. Desta forma os dados utilizados na análise dos resultados, foram apenas após este período de *warm-up*. Os resultados apresentados pelos dois modelos foram distintos.

Uma vez que o número de observações geradas no modelo de agente de objetos passivos mostrou-se bem inferior ao do modelo de agente de objetos ativos. Os tempos de cada replicação foram diferentes, visando atender um número mínimo necessário de observações, de forma que obter-se intervalos de confiança aceitáveis para as variáveis sob estudo.

Desta forma, as replicações foram realizadas com períodos de 500 segundos (objetos ativos) e de 1000 segundos (objetos passivos). Os períodos de *warm-up* adotados forma de 10% do total simulado, ou seja, 50 segundos e 100 segundos respectivamente.

Foram realizadas 10 replicações, para cada um dos experimentos sendo que os resultados apresentados são valores médios das respostas destas replicações. Não foi necessário utilizar o método de loteamento [Freitas97], uma vez que os períodos das simulações são perfeitamente toleráveis.

Nas próximas seções serão mostrados os dados obtidos e os resultados serão avaliados.

⁵ *Warm-up*: é conhecido como aquele tempo inicial onde o sistema não está normalizado. Não possuem condições iniciais fixas até que o sistema entre em normalidade.

5:4 - Resultados do Modelo de Agente com Objetos Passivos

A simulação do modelo de agentes com objetos passivos, como já descrito nas seções anteriores, foi feita com a combinação dos níveis dos fatores escolhidos. Os fatores utilizados são mostrados, de forma resumida, na tabela 5.1.

Fatores	Níveis	
	(+)	(-)
Nº de Gerentes	5	1
Nº de Objetos	30	10
Carga de Requisições	12.44 s	24.88 s
Modo de Interação	assíncrono	síncrono

Tabela 5.1 - Fatores utilizados na simulação de objetos passivos.

Os resultados obtidos são apresentados na tabela 5.2. Cada um dos 16 experimentos mostra os resultados da simulação com uma combinação dos níveis dos fatores citados acima. Os resultados das simulações são apresentados nas 4 últimas colunas da tabela. Os níveis dos fatores estão representados na tabela com os valores: 1, 2, 3 e 4. As variáveis de resposta já foram todas explicadas na seção 5.2.2, sendo as seguintes:

- NRA: Número de requisições atendidas pelo agente;
- TTR: Tempo médio de existência de uma requisição;
- NN: Número de requisições atendidas, gerado por seus respectivos objetos;
- NRA/TTR: Razão entre o as variáveis de resposta NRA e TTR, acima citadas.

Experimentos	1	2	3	4	NRA	TTR	NN	NRA/TTR
1	+	+	+	+	403	2,115	69	190,54
2	+	+	+	-	404	2,115	68	191,02
3	+	+	-	+	223	2,115	59	105,44
4	+	+	-	-	223	2,115	59	105,44
5	+	-	+	+	386	2,115	29	182,51
6	+	-	+	-	386	2,115	28	182,51
7	+	-	-	+	183	2,115	26	86,52
8	+	-	-	-	180	2,115	27	85,11
9	-	+	+	+	70	2,115	67	33,10
10	-	+	+	-	70	2,115	67	33,10
11	-	+	-	+	40	2,115	64	18,91
12	-	+	-	-	40	2,115	64	18,91
13	-	-	+	+	70	2,115	18	33,10
14	-	-	+	-	70	2,115	18	33,10
15	-	-	-	+	31	2,115	24	14,66
16	-	-	-	-	31	2,115	24	14,66

Tabela 5.2 – Resultados obtidos do modelo Objetos Passivos

5.4.1 - Tempo médio de uma requisição no sistema (TTR)

Observando o gráfico da Figura 5.1, pode-se ver que o tempo de uma requisição no sistema, no modelo da agente com objetos passivos, é constante. Isto se deve ao fato de que somente uma requisição é atendida pelo agente de cada vez. Como todas as operações de gerência foram modeladas como tendo o mesmo tempo de processamento (tempo médio das medições obtidas), todas as requisições têm o mesmo tempo de vida no sistema.

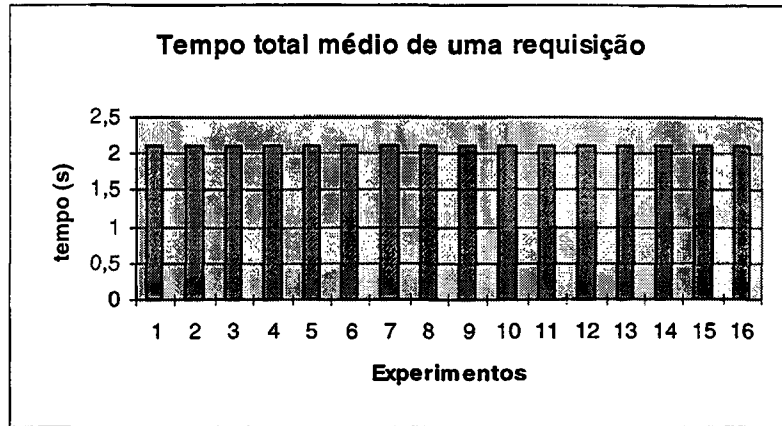


Figura 5.1 - Gráfico de Tempos Médios de Requisição - Objetos Passivos

Nestes tempos, são ignorados os tempos de espera nas filas presentes no modelo, já que no sistema real o gerente não envia as requisições se o agente está ocupado com outra, mesmo o gerente operando no modo assíncrono.

5.4.2 - Número de Requisições Atendidas (NRA)

Observando-se o gráfico da Figura 5.2, pode-se ver que quando há um maior número de gerentes, há um maior número de requisições atendidas (experimentos de 1 a 8). Note-se que quando o intervalo de envio das requisições está no nível máximo, a quantidade de requisições atendidas cai pela metade, já que metade das requisições são geradas. Isto pode ser visto claramente comparando-se os experimentos 3 e 4 com os experimentos 1 e 2, e os experimentos 7 e 8 com os experimentos 5 e 6.

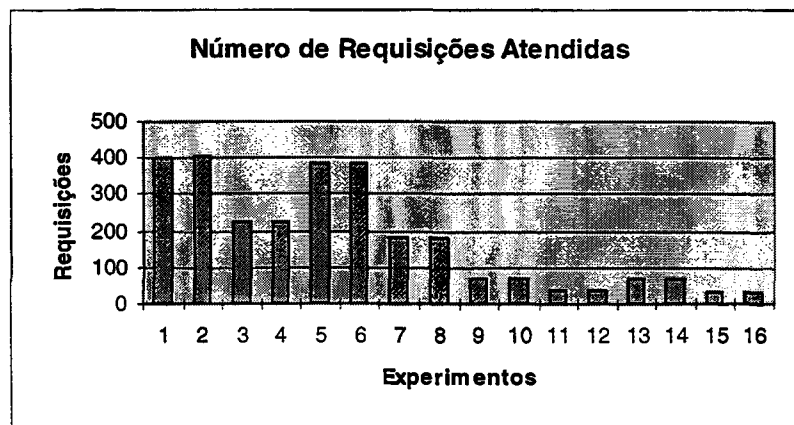


Figura 5.2 - Gráfico dos Números de Requisições Atendidas - Objetos Passivos

A carga máxima do agente foi calculada para 30 objetos. Os experimentos 1 e 2 mostram os melhores desempenhos, exatamente com 30 objetos. Se a taxa de envio de requisições aumentar, o número de requisições atendidas não passará do máximo, já que o agente não poderá processá-las. Em termos de modelo, a simulação terminaria com a quantidade excedente de requisições esperando na fila de entrada do OM.

Nos experimentos com apenas um gerente, pode-se afirmar que o sistema fica ocioso grande parte do tempo (experimentos de 9 a 16), como também pode ser visto no gráfico.

5.4.3 - Taxa de Atendimento das Requisições (NRA/TTR)

Novamente, observando-se o gráfico da Figura 5.3, pode-se chegar às mesmas conclusões da seção anterior, bastando para isto observar que, como o tempo de atendimento das requisições é fixo, o denominador da razão é fixo. Desta forma a razão varia na mesma proporção que o numerador.

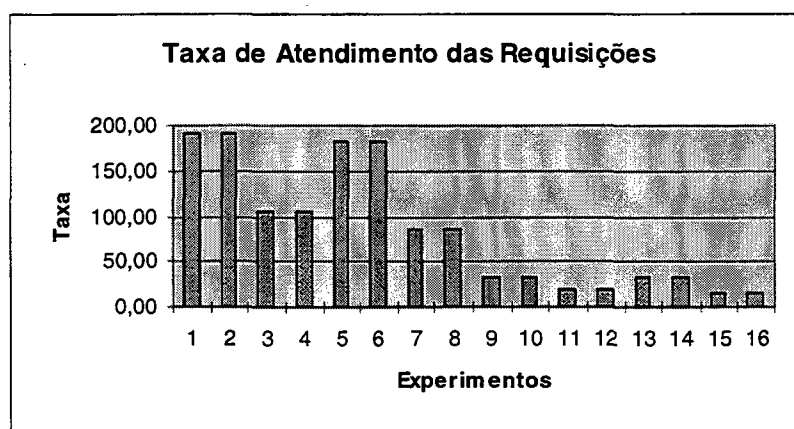


Figura 5.3 - Gráfico das Taxas de Atendimento das Requisições - Objetos Passivos

5.5 - Resultados do Modelo de Agente com Objetos Ativos

Na simulação do modelo de objetos ativos, analogamente ao modelo com objetos passivos, foram feitas simulações com todas as combinações possíveis dos níveis dos fatores. Alguns valores diferenciam-se dos valores utilizados nos objetos passivos, como explicado nas seções anteriores.

Resumidamente, os valores utilizados são mostrados na tabela 5.3.

Fatores	Níveis	
	(+)	(-)
Nº de Gerentes	5	1
Nº de Objetos	30	10
Carga de Requisições	1.1 s	2.2 s
Modo de Interação	assíncrono	síncrono

Tabela 5.3 - Fatores utilizados na simulação de objetos ativos

Os resultados obtidos pela simulação estão discriminados na tabela 5.4. Novamente, cada um dos 16 experimentos expressa uma combinação dos níveis dos diferentes fatores. As quatro últimas colunas expressam os resultados obtidos com cada combinação, sendo utilizadas as mesmas variáveis de resposta do modelo de objetos passivos.

Experimentos	1	2	3	4	NRA	TTR	NN	NRA/TTR
1	+	+	+	+	2280	3,139	97	726,25
2	+	+	+	-	1019	2,448	97	416,26
3	+	+	-	+	1104	2,573	98	429,07
4	+	+	-	-	1008	2,443	98	412,61
5	+	-	+	+	2146	15,211	34	141,08
6	+	-	+	-	855	2,904	34	294,42
7	+	-	-	+	1118	3,798	37	294,35
8	+	-	-	-	813	3,058	37	265,86
9	-	+	+	+	466	2,393	97	194,70
10	-	+	+	-	229	2,177	97	105,19
11	-	+	-	+	219	2,281	107	96,03
12	-	+	-	-	214	2,198	107	97,36
13	-	-	+	+	455	2,707	32	168,09
14	-	-	+	-	228	2,190	32	104,11
15	-	-	-	+	233	2,412	35	96,61
16	-	-	-	-	277	2,184	35	126,83

Tabela 5.4 – Resultados obtidos do modelo Objetos Ativos

5.5.1 - Tempo médio de uma requisição no sistema (TTR)

Os tempos médios das requisições no sistema mostram-se estáveis, a não ser para o experimento 5, onde este cresce, como pode ser visto no gráfico da Figura 5.4.

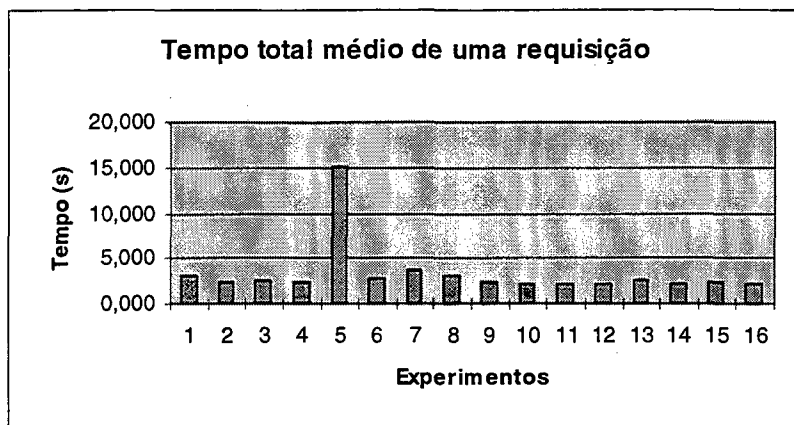


Figura 5.4 - Gráfico dos Tempos Médios de Requisição - Objetos Ativos

Isto se deve ao fato de que o intervalo de tempo entre o envio das requisições (carga de requisições), foi calculado para 10 objetos. A mesma taxa, utilizada no experimento 1, apresentou uma folga razoável no gargalo do sistema (objetos gerenciados). Para que se pudesse observar uma equivalência entre os experimentos 1 e 5, será realizada uma rodada de simulações do experimento 1, com uma carga 3 vezes maior, esperando-se um comportamento semelhante ao experimento 5 acima.

Na Tabela 5.4 e na Figura 5.5 pode-se observar os resultados para o experimento com 5 gerentes, 30 objetos, carga de requisições (intervalo) de 370 ms (3 vezes em relação a carga de 1100 ms, para 10 objetos).

Experimentos	1	2	3	4	NRA	TTR	NN	NRA/TTR
1	+	+	+	+	6471	22,351	68	289,51

Tabela 5.5 - Resultados do Experimento 1 com Intervalo de Requisições de 367 ms

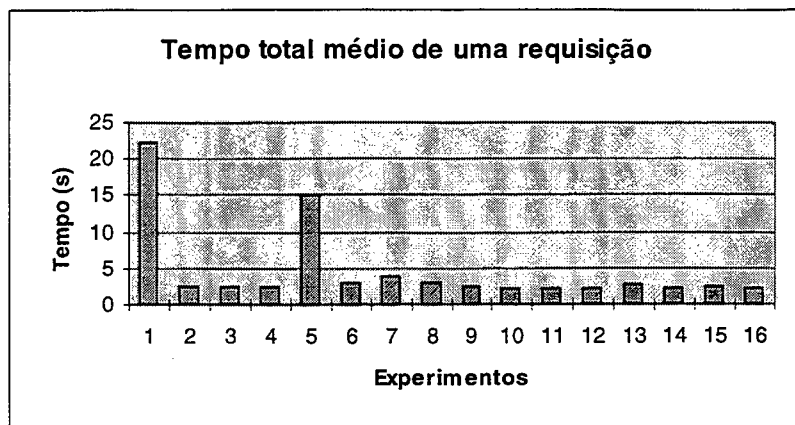


Figura 5.5 - Gráfico dos Tempos Médios de Requisição para o Experimento 1 Modificado

Como pode-se ver nos resultados apresentados acima, quando a carga de requisições é triplicada para o experimento 1, o comportamento é semelhante ao do experimento 5. O tempo médio cresce, já que o gargalo do sistema (objetos gerenciados) fica também sobrecarregado. Este tempo é ligeiramente superior ao do experimento 5, já que, como a carga é muito maior, a quantidade de requisições nas filas dos objetos aumenta proporcionalmente. Note-se que cada requisição na fila do objeto aumenta o tempo médio em 2,0508 segundos (tempo de processamento do objeto). Assim, guardadas as respectivas proporções, pode-se dizer que o experimento 1 e 5 comportam-se de maneiras semelhantes.

Ainda com relação ao experimento 5, pode-se fazer algumas comparações com os experimentos 6 e 13. No caso do experimento 6, pelo fato dos gerentes operarem de modo síncrono, não há a emissão de novas requisições antes do atendimento das requisições já enviadas, o que evita o congestionamento das filas dos objetos. E no experimento 13 há somente um gerente enviando requisições, o que causa um baixo congestionamento das filas, resultando num baixo tempo de resposta.

Outra observação relevante é que, no modelo, as requisições geradas pelos gerentes são distribuídas entre os objetos de forma probabilística, balanceando o número de requisições nas filas dos objetos. No sistema real, entretanto, pode haver casos em que há um grande número de requisições para um mesmo objeto, enquanto outros objetos ficam ociosos, o que determinará tempos diferentes para os tempos médios.

Nos experimentos de 9 a 16, os tempos médios caem devido ao fato de haver somente um gerente enviando requisições para o agente, mesmo variando os demais fatores. Os tempos permanecem próximos do tempo total de processamento estimado para o sistema real, já que não há grande espera nas filas.

É importante ressaltar que neste modelo, ao contrário do modelo de agente com objetos passivos, os tempos de espera nas filas são computados, uma vez que as filas existem no sistema real.

5.5.2 - Número de Requisições Atendidas (NRA)

Como pode ser visto no gráfico da Figura 5.6, há uma variação no número de requisições atendidas nos diferentes experimentos. Os experimentos 1 e 5 são os que apresentaram um maior número de requisições atendidas.

Pelo fato da carga de requisições ter sido estimada para 10 objetos, não há uma variação muito significativa nos valores desta variável de resposta nos experimentos 1 e 5. No caso do experimento 1, a diferença diz respeito ao número de requisições pendentes nas filas dos objetos no experimento 5. Caso não restassem requisições nas filas dos objetos do experimento 5, a quantidade de requisições atendidas em ambos os experimentos tenderia a ser igual.

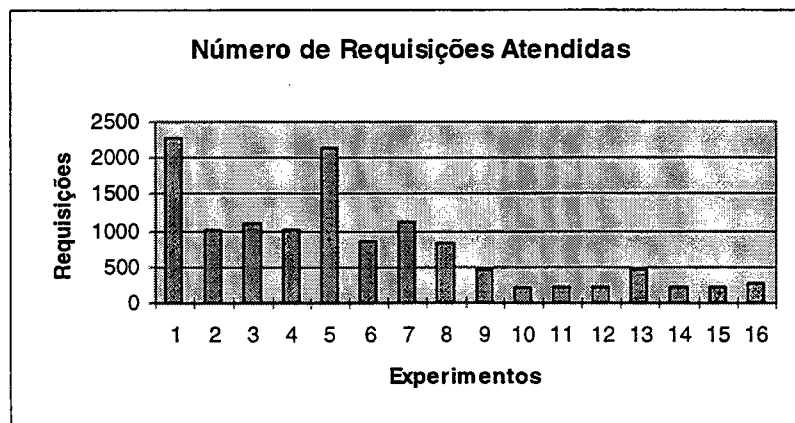


Figura 5.6 - Gráfico dos Números de Requisições Atendidas - Objetos Passivos

É possível ver no gráfico que, quando o número da carga de requisições cai pela metade, o número de requisições também cai na mesma proporção, pois todas as requisições são processadas, sem acúmulos, e até com certa ociosidade dos objetos. Quando há apenas um gerente, ou quando os gerentes operam no modo síncrono, também há uma queda sensível no número de requisições atendidas pelo modelo de agente com objetos ativos.

5.5.3 - Taxa de Atendimento das Requisições (NRA/TTR)

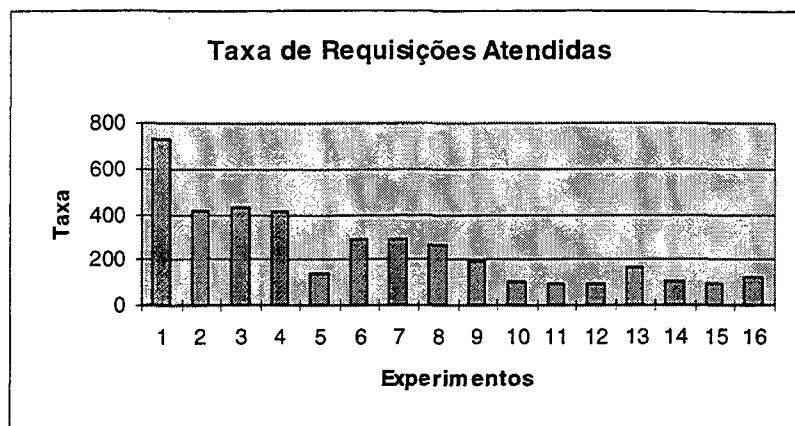


Figura 5.7 - Gráfico das Taxas de Atendimento das Requisições (NRA/TTR)

Esta medida pode ser considerada a melhor para avaliar o desempenho do modelo nos diversos experimentos. Como pode ser visto no gráfico da Figura 5.7, a melhor taxa verificada é a apresentada pelo experimento 1, onde o maior número de requisições foi atendida, e o tempo médio para o atendimento de uma requisição foi relativamente baixo.

O experimento 5 apresentou uma das piores taxas pois, mesmo atendendo um número grande de requisições, o tempo médio para o atendimento de cada requisição foi muito alto, em relação aos demais experimentos.

Nos experimentos de 9 a 16, as taxas foram menores pois havia apenas um gerente para enviar requisições. Mesmo sendo baixos os tempos de atendimento de cada requisição, o gerente não foi capaz de gerar uma carga de requisições que pudesse fazer esta taxa aumentar.

É importante ressaltar que, no caso do experimento 1 com a taxa de requisições triplicadas, a taxa de requisições atendidas será menor. Porém não será próxima do experimento 5, já que o tempo médio de atendimento das requisições aumenta, porém o número de requisições atendidas aumenta mais (aproximadamente 3 vezes).

5.6 - Comparação dos Modelos

Os resultados apresentados nas seções anteriores foram obtidos com parâmetros diferentes para cada modelo, no que diz respeito à carga de requisições. Porém, uma

comparação de dois modelos só pode ser considerada válida quando é feita com base em resultados obtidos com parâmetros exatamente iguais [Lucas79].

Para tanto, ambos os modelos foram simulados com as cargas de requisições do outro modelo. Em outras palavras, o modelo de agentes com objetos passivos foi simulado com as cargas do modelo de agentes com objetos ativos, e vice-versa, conforme pode ser visto na Tabela 5.6.

Fatores	Níveis Ag. Obj. Ativo		Níveis Ag. Obj. Passivo	
	(+)	(-)	(+)	(-)
Nº de Gerentes	5	1	5	1
Nº de Objetos	30	10	30	10
Carga de Requisições	12,44 s	24,88 s	1,1 s	2,2s
Modo de Interação	assíncrono	síncrono	assíncrono	síncrono

Tabela 5.6 - Parâmetros iguais utilizados na simulação tanto em agentes com objetos passivos quanto em agentes com objetos ativos

Os resultados obtidos com estas simulações podem ser vistos na tabela 5.7.

Além das variáveis de resposta já mencionadas nas seções anteriores, para uma melhor comparação dos dois modelos, outra variável de resposta é apresentada NMIROM, a qual representa o número médio de requisições esperando na fila do gerenciador de objetos.

Resultados Objetos Ativos				Resultados Objetos Passivos			
NMROM	NRA	NN	TTR	NMROM	NRA	NN	TTR
0,0001720	381	75	2,1997	1077,4	139	97	2,1154
0,0003740	397	69	2,3711	5,7196	139	97	2,1154
0,0000033	216	59	2,1649	516,5	139	97	2,1154
0,0000032	216	59	2,2733	6,2678	139	97	2,1154
0,0004460	387	29	2,341	1020,6	208	28	2,1154
0,0004500	422	27	2,5789	4,3425	202	34	2,1154
0,0000271	183	26	2,2677	475,85	199	37	2,1154
0,0000000	183	26	2,3107	4,3277	199	37	2,1154
0,0000000	70	67	2,1759	167,85	139	97	2,1154
0,0000000	70	67	2,2822	1,6537	139	97	2,1154
0,0000000	40	64	2,1743	53,76	130	106	2,1154
0,0000000	40	64	2,2228	2,1375	130	106	2,1154
0,0000093	70	18	2,2892	133,82	204	32	2,1154
0,0000000	70	18	2,3555	0,2787	204	32	2,1154
0,0000000	36	37	2,3189	23,779	201	35	2,1154
0,0000000	31	21	2,3155	0,29915	201	35	2,1154

Tabela 5.7 - Resultados da Simulação dos Modelos com Comparação

Como pode ser observado na tabela acima, em relação ao modelo de agentes com objetos passivos, ao ser utilizada a carga de requisições do modelo de agentes com objetos ativos, o número de requisições atendidas não aumenta significativamente. Isto porque o modelo já está no seu limite máximo de utilização, quando na sua carga máxima de requisições. Todas as requisições geradas além desta capacidade terminam na fila do OM, o que pode ser visto nos tempos médios de espera na fila do OM.

Já no caso do modelo de agentes com objetos ativos, quando simulado com a carga de requisições do modelo de agentes com objetos passivos, pode-se ver que não há um crescimento no número de requisições atendidas. Isto porque a carga é muito baixa para a capacidade de processamento do agente. Não há espera das entidades na fila do OM.

Pode-se concluir assim que a capacidade de processamento do agente com objetos ativos é muito maior, tanto no número de requisições atendidas como na taxa de atendimento de requisições.

5.7 - Análise Estatística

Nesta seção será feita uma análise estatística dos resultados obtidos com a simulação de cada um dos modelos. Esta análise é baseada na comparação dos efeitos dos diversos fatores sobre os resultados obtidos. Estes efeitos são analisados sobre os fatores individualmente, agrupados dois a dois, três a três e a de todos os fatores em conjunto.

Todas as possíveis combinações dos efeitos do fatorial completo de 2^k com $k=4$ é apresentado na Tabela 5.6. Admitindo os valores desta tabela, que os efeitos principais e as interações de dois fatores são suficientes para descrever adequadamente os resultados para validação dos sistemas em estudo. De acordo com [NSB95], a expansão em série pode ser truncada depois dos termos de segunda ordem. Assim, serão mostrados as interações de todos os fatores, porém somente os principais e a interação de dois fatores serão analisados.

5.7.1 - Cálculo dos Efeitos

A tabela 5.8, apresenta todos os sinais necessários para o cálculo dos efeitos. O divisor é 16 para a média e 8 para cada um dos efeitos. Empregando os sinais apropriados como coeficientes dos resultados médios observados (resultados), e em seguida aplicado os divisores calcula-se os 15 efeitos e a média global [BHH86].

As colunas da Tabela 5.8 apresentam os sinais para o cálculo de cada efeito. O efeito "G" diz respeito ao número de gerentes, o efeito "O" ao número de objetos, o efeito "C" à carga de requisições e o efeito "I" ao modo de interação do agente. Os efeitos compostos pela interação de dois ou mais fatores são representados pelo agrupamento das iniciais de cada fator, como apresentados anteriormente.

C	I	GO	GC	GI	OC	OI	CI	GOC	GOI	GCI	OCI	GOCI
1	1	1	1	1	1	1	1	1	1	1	1	1
1	-1	1	1	-1	1	-1	-1	1	-1	-1	-1	-1
-1	1	1	-1	1	-1	1	-1	-1	1	-1	-1	-1
-1	-1	1	-1	-1	-1	-1	1	-1	-1	1	1	1
1	1	-1	1	1	-1	-1	1	-1	-1	1	-1	-1
1	-1	-1	1	-1	-1	1	-1	-1	1	-1	1	1
-1	1	-1	-1	1	1	-1	-1	1	-1	-1	1	1
-1	-1	-1	-1	-1	1	1	1	1	1	1	-1	-1
1	1	-1	-1	-1	1	1	1	-1	-1	-1	1	-1
1	-1	-1	-1	1	1	-1	-1	-1	1	1	-1	1
-1	1	-1	1	-1	-1	1	-1	1	-1	1	-1	1
-1	-1	-1	1	1	-1	-1	1	1	1	-1	1	-1
1	1	1	-1	-1	-1	-1	1	1	1	-1	-1	1
1	-1	1	-1	1	-1	1	-1	1	-1	1	1	-1
-1	1	1	1	-1	1	-1	-1	-1	1	1	1	-1
-1	-1	1	1	1	1	1	1	-1	-1	-1	-1	1

Tabela 5.8 - Coeficientes de Contraste para um fatorial 2^4

Todas as colunas da Tabela 5.8, exceto a primeira, tem oito sinais positivos e oito negativos. Qualquer efeito portanto, pode ser interpretado como a diferença entre duas médias, cada uma das quais contendo metade das observações. Chamando de X a matriz 16×16 com sinais algébricos da tabela dos coeficientes de contrastes e com elementos $+1$ ou -1 , os efeitos serão dados, a menos dos divisores, pelo produto $X^t \times y$, onde y é o vetor coluna contendo os resultados médio dos experimentos. É importante ressaltar que a Tabela 5.8 já é a matriz X^t .

Dividindo o primeiro elemento por 16 e os demais por 8 obtém-se o vetor dos efeitos (arredondados para a segunda casa decimal) [NSB95]. Os valores dos efeitos aparecem no vetor na mesma ordem das colunas da matriz da Tabela 5.8.

O resultados dos efeitos calculados para o planejamento fatorial 2⁴, são mostrados a seguir:

Ativo	Passivo	Res. Ativo	Res. Passivo	Efeitos
-------	---------	------------	--------------	---------

Média dos Efeitos

2280	403	791,50	175,63	M
------	-----	--------	--------	----------

Efeitos Principais

1019	404	1002,75	245,75	G
1104	223	51,75	17,00	O
1008	223	336,50	113,50	C
2146	386	422,25	0,25	I

Interações de Dois Fatores

855	386	68,00	12,50	GO
1118	183	227,75	79,00	GC
813	180	239,75	-0,50	GI
466	70	25,75	-8,25	OC
229	70	-22,50	-0,50	OI
219	40	331,75	-0,50	CI

Interações de Três Fatores

214	40	3,50	-3,75	GOC
455	70	-37,25	-0,50	GOI
228	70	206,00	-0,50	GCI
233	31	17,50	0,25	OCI

Interações de Quatro Fatores

277	31	27,25	0,25	GOCI
-----	----	-------	------	-------------

5.7.2 - Interpretação dos Resultados

Os valores da tabela indicam a porcentagem da variação causada, nos diversos resultados, da variação de cada fator, nos seus respectivos níveis. A variável de resposta tomada como referência para a elaboração das tabelas acima foi o número de requisições atendidas.

Para o modelo com objetos ativos, na análise individual dos fatores, o fator que apresenta maior relevância é o número de gerentes, seguido do modo de interação dos gerentes. Desta forma, pode-se concluir que, individualmente, os fatores relacionados com os gerentes são os mais relevantes.

Já analisados dois a dois, os fatores que mais influenciam o modelo com objetos ativos são a carga de requisições e a interação dos gerentes. Os fatores que combinam o número de gerentes com a carga de requisições e a interação também apresentam uma variação considerável.

De maneira geral, pode-se concluir assim que o agente tem capacidade de lidar com diferentes cargas, interações dos gerentes e número de gerentes. Isto quer dizer que, nos experimentos realizados, o agente não mostrou-se como fator limitante do desempenho do sistema.

Para o modelo com objetos passivos, os fatores individuais que mostraram maior variação foram o número de gerentes e a carga de requisições. Combinando os fatores dois a dois, novamente a combinação do número de gerentes e da carga de requisições apresenta a maior variação.

O que é possível concluir com estes dados é que o agente representa um gargalo do sistema, já que as variações são pequenas, em média. Isto porque as variações dos parâmetros não alteram a principal característica do agente com objetos passivos, que é o atendimento das requisições em série, sem mais de uma requisição no sistema em um dado instante.

5.8 - Sumário

Neste capítulo foram apresentados os projetos de experimentos para as simulações dos modelos dos agentes com objetos ativos e passivos. Após a definição dos experimentos e parâmetros das simulações, os resultados foram apresentados. Para ambos os modelos, cada variável de resposta foi analisada individualmente, para determinar o desempenho de cada modelo individualmente.

Para efeitos de comparação dos modelos, foram realizadas novas simulações, com as mesmas condições em ambos os modelos, para estudar o comportamento de cada modelo com as cargas do outro. Os resultados obtidos foram utilizados para comparar os modelos em termos de desempenho.

Por fim, foi feita uma análise estatística dos resultados dos modelos, para determinar, de modo confiável, a importância dos fatores nos resultados, seja individualmente, ou combinados entre si.

No próximo capítulo, as conclusões finais do presente trabalho são apresentadas, bem como sugestões de trabalhos futuros.

Conclusão

Este trabalho apresentou a comparação de desempenho de duas formas de implementação de agentes de aplicações de gerenciamento de redes de computadores. As formas de implementação comparadas foram os agentes com objetos passivos e os agentes com objetos ativos.

Primeiramente foram apresentados os conceitos relativos ao modelo de gerenciamento OSI. Foi descrito o modelo de gerenciamento OSI, plataformas de gerenciamento, aplicações de gerenciamento, e as formas de implementação de agentes. Foram descritos também aspectos relevantes sobre as principais características de ambas as formas de implementação de agentes, bem como a forma de comunicação entre os componentes de uma aplicação de gerenciamento de redes de computadores.

Em seguida foram descritos os conceitos e termos relacionados com técnicas de avaliação de desempenho. Dentre as alternativas propostas, foi justificada a escolha da simulação para a realização deste trabalho, que utiliza a avaliação de desempenho na comparação e avaliação de alternativas de projetos de sistemas de gerenciamento de redes de computadores. A definição do projeto de experimentos também foi abordada, uma vez que um bom projeto é crucial para a confiabilidade dos resultados obtidos. As formas de análise dos resultados também foram apresentadas, para fornecer mecanismos de avaliar os resultados obtidos neste trabalho.

Por último, foram apresentados os projetos de experimentos para as simulações dos modelos dos agentes com objetos ativos e passivos. Após a definição dos experimentos e parâmetros das simulações, os resultados foram apresentados, com análises de desempenho individuais para ambos os modelos, com base nas variáveis de resposta. Também foi feita uma comparação dos dois modelos, com parâmetros iguais. Por fim, uma análise estatística dos resultados foi realizada, para determinar a importância dos fatores sobre uma das variáveis de desempenho.

O presente trabalho representa uma importante contribuição para o projeto da plataforma OSI de gerenciamento de redes de computadores desenvolvido no LISHA, da

UFSC, uma vez que apresenta resultados sobre a abordagem de implementação de agentes proposta neste projeto.

A comparação entre *multithreaded* e *single-thread* apresentado neste trabalho, mostrou a introdução de conceito de objetos ativos e objetos passivos. Uma vantagem da implementação de objetos ativos sobre objetos passivos é a grande independência entre execuções dos comportamentos dos objetos gerenciados, o que propicia um alto grau de tolerância a faltas para o processo como um todo, protegendo-o de faltas que possam ocorrer no escopo de um objeto e que venham interferir em outras atividades do agente.

Os resultados obtidos neste trabalho permitem concluir que o modelo de agente com objetos ativos apresenta um melhor desempenho em relação ao modelo com objetos passivos. Este melhor desempenho se deve ao fato de poder atender concorrentemente as requisições para diferentes objetos gerenciados.

O modelo com objetos ativos foi avaliado neste trabalho com os parâmetros de simulação, baseados no sistema real, procurando a ocupação máxima do agente. Foi visto que o desempenho nestas condições é bem melhor que o desempenho do modelo com objetos passivos. Isto porque quando o modelo com objetos passivos foi submetido à mesma carga, não apresentou melhoria no desempenho, pois o agente já se encontrava no seu limite máximo.

Mesmo demonstrando ser muito mais eficiente o modelo com objetos ativos, foi observado que as operações de gerência funcionam até um certo limite encontrado nos modelos de simulação, assim como cada recurso possui sempre seu limite superior e inferior.

A maior performance na execução das atividades intra-agente também é outro importante fator presente na proposta de objetos ativos. Os resultados obtidos a partir dos modelos simulados, comprovam o melhor desempenho desta proposta, quando comparada com o modelo tradicional que implementa objetos passivos.

Outro resultado deste trabalho são os próprios modelos de simulação. Os resultados obtidos validam os modelos dos agentes elaborados. Desta forma, os modelos podem ser utilizados para simulações com objetivos diferentes dos deste trabalho.

Como sugestões para trabalhos futuros, pode-se citar a realização de novos estudos em relação aos modelos implementados neste trabalho, com diferentes parâmetros. Estes estudos podem visar tanto a comparação dos dois modelos, como avaliar o desempenho de um ou outro sob diferentes condições.

Também como trabalho futuro, poderá ser melhorado os pontos de gargalos em ambos os modelos, sendo que em cada modelo (objetos passivos e ativos), possuem pontos de gargalos em diferentes situações como pode ser visto nos resultados obtidos.

Referências Bibliográficas

[ARENA95] ARENA 2.0 Reference guide; 1995; Systems Modelling Corp; PA,USA.

[BFS83] Bratley, P, Fox, B.L., Schrage, L.E.; "A guide to simulate", New York, Springer-Verlag, 1983.

[BHH86] Box, G.E.P., Hunter, W.G. and Hunter, J.S. (1986); "Statistics for experimenters"; J. Wiley; New York;

[BRISA93] BRISA "Gerenciamento de Redes - Uma Abordagem de Sistemas Abertos", Makron Books, 1993.

[BRISA94] BRISA "Arquitetura de Redes de Computadores - OSI e TCP/IP", Makron Books, 1994.

[BROO96] Barroso, G.; Rodrigues, A. W. O.; Oliveira, Á. E. e Oliveira, M.; "PAPO - Um Ambiente Integrado de Simulação de Redes de Computadores Para Fins Didáticos", XXIII Seminário Integrado de Software e Hardware, Agosto 1996.

[CACI83] CACI (1983); SIMSCRIPT II.5; Programming Language, CACI; Los Angeles; CA.

[Ferrari78] Ferrari, D.; "Computer Systems Performance Evaluation". Englewood Cliffs: Prentice-Hall. 1978.

[FHC96] Fleury, M.; Hayat, L. e Clark, A. F.; "Evaluating the Performance of Parallel Programs in a Distributed Environment", IEE Proc.-Comput. Tech, Vol 143 N. 2 March 1996.

[Fishman78] Fishman, G.S.; "Principles of discrete event simulation", New York, J. Wiley, 1978.

[Freitas97] Freitas, P.,J., "Introdução a modelagem e a simulação de sistemas discretos", INE, UFSC, 1997.

[GoMi86] Goble, J. G. e Mills, Dr, R.; " COMNET III: OBJECT-ORIENTED NETWORK PERFORMANCE EVALUATION";1986.

[GoNo95] Goulart, C. de C. e Nogueira, J. M. S.; "Utilização do Modelo TMN no Gerenciamento de Redes ATM". 13º Simpósio Brasileiro de Redes de Computadores, Maio 1995.

[HCSR64] Holt, Charles C., Shirey, Robert W, Steward, Donald V., Midler, Joseph L., and Stround, Arthur. "Program SIMULATE, a User's and Programmer's Manual", Social Systems Research Institute, University of Wisconsin, May 1964.

[ISO/TEC91a] ISO/IEC 9596, Information Technology - Open Systems Interconnection - Common management information protocol - part 1: Specification, 1991.

[ISO/TEC91b] ISO/IEC 9595 / CCITT Rec. X.710 - Information Technology - Open Systemd Interconnection - Common Management Information Service Definition, março de 1991.

[ISO/TEC92a] ISO/IEC 7498-4, Information Technology - Open Systems Interconnection - Management framework, 1992,

[ISO/TEC92b] ISO/IEC 10040, Information Thecnology - Open Systems Interconnection - Systems management overview, 1992

[Jain91] Jain, R.; "The Art of Computer Systems Performance Analysis". John Wiley & Sons, Inc. 1991.

[JoOI96] Jodai, J. R; Oliveira, J. D.; "Suporte para Implementação de Objetos Concorrentes em uma Plataforma de Gerenciamento OSI", trabalho conclusão de curso da UFSC, julho de 1996.

[Kiviat63] Kiviat, Philip J.; "GASP – A General Activity Simulation Program", Project N. 90. 17-019(2), Applied Reserch Laboratory, United State Steel, Monroeville, Pennsylvania, july-1963.

[LaKe91] Law, E. M. e Kelton, W. D.; "Simulation Modeling and Analysis", McGraw-Hill, USA, 1991.

[LiDu95] Lima, H. M. e Duarte, O. C. M. B., "Protocolos Ponto-a-Multiponto com Retransmissão de Múltiplas Cópias Adaptados ao Ambiente de Satélite a Alta Velocidade". 13º Simpósio Brasileiro de Redes de Computadores, Maio 1995.

[LiWa95] Lindstaedt, E. e Wagner, F. R.; "Um Ambiente de Simulação Visual Interativa", Trabalho apresentado no XXII Seminário Integrado de Software e Hardware, durante o XV Congresso da Sociedade Brasileira de Computação, em Canela-RS, 1995.

[Lucas71] Lucas, H.C.Jr; (1971); "Performance evaluation and monitoring"; computing surveys, 79-91.

[Marshall56] Marshall, A.W.; An introductory note.; In: MEYER, H.A., ed. Symposium on Monte Carlo methods, New York, J. Wiley, 1956. P. 1-14

[MaSp97] Matias, Jr R., Specialski, E.S.;"Managed objects as active objects: a multithreaded approach"IS&N'97,Italia.

[Matias96] Matias, R. Júnior. Uma Infra-Estrutura Multithreaded para a Implementação de Objetos Gerenciados em Agentes de uma Plataforma de Gerenciamento de Redes OSI/ISO. Trabalho Individual/CPGCC, Abril 1996.

[MHBK62] Markowitz, H.M., Hausner, Bernard and Karr; H.W. SIMSCRIPT: A Simulation Programming Languages, The RAND Corporation, RM-3310 (Nov/1962).

[MMS96] May, K.; Matias, R. Jr e Specialski, E. S. "Especificação e Implementação dos Tipos Construtores ASN.1", trabalho conclusão de curso da UFSC, julho de 1996.

[Morgan84] Morgan, B.J.T.; "Elements of simulation"; London, Chapman and Hall, 1984.

[MSG96] Moura, J. A. B.; Sauv e, J. P.; Giozza, W. F. e Ara jo, J. F. M.; "Redes Locais de Computadores - Protocolos de Alto N vel e Avalia o de Desempenho", McGraw-Hill, 1986.

[NaCa89] Navaux, P. O. A. e Cavalheiro, G. G. H.; "Introdu o aos M todos de Avalia o de Desempenho".1989

[NoMe96] Nogueira, J. M. S. e Meira, D. M.; "The SIS Project: A Distributed Platform for the integration of Telecommunication Management System". In 1996 IEEE Network Operations and Management Symposium , pages 175-185, April 1996.

[OMS96] Ould-Khaoua, M.; Mackenzie, L. M. and Sotuder, "Comparative Evaluation of Hypermesh and Multi-stage Interconnection Networks", The Computer Journal, Vol 39 N.3, 1996.

[Pedgen95] Pedgen, C. D. e outros; "Introduction to Simulation Using SIMAN V", McGraw-Hill, 1995.

[Pidd84] Pidd, M. "Computer simulation in management science"; Chichester, J. Wiley, 1984.

[Purpose83] General Purpose Systems Simulator, Reference Manual, International Business Machines Corporation.,1983

[Simon69] Simon, H.; The science of the artificial. Cambridge, Mass., MIT Press, 1969.

[SiNo96] Silva, R. G. R. e Nogueira, J. M. S.; "Sobre a Integra o de Gerenciamento de Redes   Plataforma SIS atrav s do Modelo Gerente/Agente". XXIII Semin rio Integrado de Software e Hardware, Agosto 1996.

[SLC95] Soares, L. F. G.; Lemos, G. e Colcher, S.; "Redes de Computadores Das LANs, MANs e WANs às Redes ATM", Editora Campus, 1995.

[SOS95] Souza, J. N.; Oliveira, A. M. B. e Staton, M. A.; "Migrando uma Solução de Interoperabilidade entre Sistemas TMNs baseada na Metodologia AIDE para o Ambiente OSIMIS", Maio 1995.

[Tanenbaum88] Tanenbaum, A. (1988), Computer Networks, Prentice-Hall; Englewood Cliffs, NJ.

[Tocher63] Tocher, K.D.; "The art of simulation"; English Universities Press, 1963.

[VaBB96] Van den Berg, K. G. and Van den Broek, P. M.; "Programmer's Performance on Structured versus nonstructured function definitions". Information and Software Thecnology n.38, pags 477-492, 1996.