

Paulo Pinheiro da Silva

Um Estudo sobre a Migração de Sistemas Legados Centralizados
para Ambientes Distribuídos

Dissertação apresentada ao Curso de Pós-Graduação em Ciência da Computação da Universidade Federal de Minas Gerais, como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação.

Belo Horizonte

26 de junho de 1998

Resumo

Durante os últimos 20 anos, as organizações experimentaram um grande aumento no número de sistemas de informações. Muitos destes sistemas desempenham funções vitais para o bom funcionamento de suas organizações, embora estejam tecnologicamente desatualizados e esteja cada vez mais difícil mantê-los operacionais. Esse problema é chamado de *dilema dos sistemas legados* e a solução tradicional tem sido a migração destes sistemas chamados de *sistemas legados* para novos sistemas denominados de *sistemas alvo*. Esta dissertação analisa através de um estudo de caso o comportamento de um sistema alvo distribuído. Para realizar este estudo foram usadas técnicas de análise de desempenho e de planejamento de capacidade. A caracterização da carga de trabalho do sistema alvo foi modelada a partir de parâmetros obtidos do próprio sistema alvo ainda não operacional e do sistema legado centralizado e ainda operacional. Um modelo de simulação foi desenvolvido, e a partir deste foi verificado se o ambiente computacional do sistema alvo comportaria a carga de trabalho previamente modelada. Ainda foi desenvolvido um outro modelo de simulação para outro sistema alvo que indica que a opção cliente-servidor é melhor que a opção de sistemas distribuídos multiusuário, em termos de desempenho.

Abstract

In the last 20 years, many it was observed a great increase in the use of information systems. Many of these systems are essential to their organizations, although they can be said to be out-of-date and difficult to maintain operational. This problem is known as *the Legacy Systems Dilemma* and it is usually solved by the migration of the original systems, called *legacy systems*, to new systems, called *target systems*. This dissertation discusses, through a case study, the performance of a distributed target system. Performance analysis and capacity planning were used. The workload characterization of the target systems was modeled using data from the operational legacy system as well as from the not yet operational target system. The computational environment of the target system was assessed through the use of the modeled workload submitted to the developed simulation model. A second model was developed to simulate a client-server target system. It was verified that the client-server approach has a better performance than the distributed multi-user approach.

Sumário

Lista de Figuras	iv
Lista de Tabelas	vi
1 Introdução	1
1.1 Migração de Sistemas Legados	2
1.2 Análise de Desempenho para Migração de Sistemas	3
1.3 O Método de Análise de Desempenho e Planejamento de Capacidade	4
1.4 Organização da Dissertação	6
2 Caracterização do Estudo de Caso	7
2.1 Introdução	7
2.2 Características Gerais	8
2.3 Problemas Legados e a Necessidade de Evoluir	9
2.3.1 Projeto do Banco de Dados	10
2.3.2 Sistema Gerenciador de Banco de Dados	11
2.3.3 Manutenção	13
2.4 Estratégia de Migração	13
2.4.1 A Migração em Três Etapas	13
2.4.2 O SICOM/C	16
2.4.3 O SICOM/D	17
2.4.4 O SICOM/CS	19
3 Medição e Seleção de Serviços On-Line	21
3.1 Escopo do Estudo de Desempenho	21
3.2 Terminologia	24
3.3 O Ambiente On-Line	25
3.4 Mapeamento de Transações entre os Sistemas Legado e Alvo	26
3.5 Serviços Representativos da Carga de Trabalho On-Line do Sistema HP07	27
3.5.1 Monitoração das Transações On-Line	27
3.5.2 Representatividade das Transações On-Line Seleccionadas	28
3.5.3 Identificação dos Principais Serviços On-Line	30

4	Caracterização da Carga de Trabalho	33
4.1	Compreensão do Ambiente	33
4.1.1	Batch Jobs	33
4.1.2	Serviços ftp	34
4.2	Medições de Dados	34
4.2.1	Medições no Sistema Legado	34
4.2.2	Medições no Sistema Alvo	35
4.3	Modelagem da Carga de Trabalho	40
4.3.1	Intensidade da Carga de Trabalho	40
4.3.2	Uso de Recursos	42
5	Modelagem e Simulação do SICOM	47
5.1	O Simulador SES/workbench	47
5.2	O Modelo de Simulação do SICOM/D	50
5.2.1	Visão Geral	50
5.2.2	Os Submodelos da Carga de Trabalho	55
5.2.3	O Submodelo do Servidor de Banco de Dados	55
5.2.4	O Submodelo dos Processos Natural	57
5.2.5	O Submodelo do SGBD ADABAS	58
5.2.6	Os Submodelos da Rede de Computadores	60
5.3	O Modelo de Simulação do SICOM/CS	61
5.4	Validação do Modelo de Simulação do SICOM/D	62
5.4.1	Alternativas para Realizar as Medições Experimentais dos Tempos de Resposta	64
5.4.2	Medições Experimentais dos Tempos de Resposta	68
5.4.3	Calibragem e Validação	72
5.5	Resultados das Simulações	75
5.5.1	Análise Comparativa das Soluções	78
6	Conclusões	81
6.1	Revisão do Trabalho	81
6.2	Principais Resultados	82
6.3	Trabalhos Futuros	83
	Referências Bibliográficas	85

Lista de Figuras

2.1	Exemplo de reprojeto no banco de dados do HP07	11
2.2	Formas diferentes de um programa COBOL acessar o SGBD ADABAS. . .	12
2.3	Gerências regionais da COPASA-MG	15
2.4	Rede de computadores da GRNT	18
3.1	Esquema de mapeamento de transações <i>on-line</i>	26
3.2	Ferramenta de inspeção visual e quantitativa.	29
4.1	Relatório do utilitário ADACLIP do ADABAS	39
4.2	<i>Trace</i> do SGBD ADABAS gerado por um programa Natural	41
5.1	Nós usados no modelo de simulação do SICOM.	49
5.2	O modelo de simulação do SICOM/D.	51
5.3	Parâmetros característicos das transações no SES/ <i>workbench</i>	52
5.4	Parâmetros do modelo da carga de trabalho no modelo de simulação para serviços <i>on-line</i>	53
5.5	Parâmetros do modelo da carga de trabalho no modelo de simulação para serviços <i>batch</i>	53
5.6	Parâmetros do modelo da carga de trabalho no modelo de simulação para serviços <i>ftp</i>	53
5.7	Variáveis locais de cada instância de transação.	54
5.8	Constantes, variáveis e parâmetros do modelo de simulação.	55
5.9	O submodelo do servidor de banco de dados.	56
5.10	O submodelo dos processos Natural.	57
5.11	O submodelo do SGBD ADABAS.	59
5.12	O submodelo de rede (ethernet ou X.25).	60
5.13	O submodelo dos processos Natural do SICOM/CS.	62
5.14	Exemplo de <i>Script</i> do NetTerm.	65
5.15	Medidas de tempos de resposta da execução do serviço <i>pesquisa consumidor</i> com o uso do NetTerm.	67
5.16	Implementação do registro dos tempos de resposta no código fonte do ICS.	68

5.17	Medidas de tempos de resposta da execução do serviço <i>pesquisa consumidor</i> com o uso do ICS.	69
5.18	Gráfico dos tempos de resposta medidos experimentalmente.	71
5.19	Tempos de resposta obtidos durante a calibragem do modelo de simulação do SICOM/D.	72
5.20	Participação das transações <i>on-line</i> nos tempos de resposta dos serviços <i>on-line</i>	74
5.21	Tempos de resposta ano a ano do SICOM/D e do SICOM/CS.	79
5.22	Utilização do sistema computacional ano a ano.	80
5.23	Comparação da distribuição dos tempos de resposta médios do SICOM/D e do SICOM/CS.	80

Lista de Tabelas

3.1	Amostra das medições de parâmetros de desempenho de transações <i>on-line</i> .	27
3.2	Programas <i>on-line</i> mais usados.	30
3.3	Identificação dos programas mais utilizados.	31
4.1	Transações <i>on-line</i> no horário de pico na GRNT	35
4.2	Serviços <i>on-line</i> no horário de pico na GRNT	36
4.3	Serviços <i>on-line</i> na GRNT estimado para 1999	42
4.4	Parâmetros de serviços <i>on-line</i> relativos ao servidor de banco de dados. . .	46
4.5	Parâmetros de serviços <i>on-line</i> relativos à rede de computadores da GRNT.	46
4.6	Parâmetros de serviços <i>ftp</i>	46
4.7	Parâmetros de serviços <i>batch</i>	46
5.1	Frequência total de execução de serviços <i>on-line</i> na Gerência da Regional Norte.	70
5.2	Participação dos serviços <i>on-line</i> no experimento de medição dos tempos de resposta.	70
5.3	Distribuição dos serviços <i>on-line</i> por computadores clientes nos experimentos de medição de tempos de resposta	71
5.4	Tempos de resposta do SICOM/D medidos experimentalmente.	71
5.5	Tempos de resposta na simulação do SICOM/D.	76
5.6	Utilização do sistema computacional na simulação do SICOM/D.	77
5.7	Tempos de resposta na simulação do SICOM/CS.	78
5.8	Utilização do sistema computacional na simulação do SICOM/CS.	78

Capítulo 1

Introdução

Desde que foram desenvolvidos os primeiros computadores na década de quarenta, a indústria de computação vem suprindo a necessidade de processamento de dados das empresas essencialmente através da utilização de computadores centrais com grande capacidade de processamento (*mainframes*). Durante todo esse período, a maioria dos sistemas de informação foi desenvolvida para um ambiente de processamento de dados centralizado. Esses sistemas, atualmente denominados de sistemas legados, eram responsáveis pela execução de funções críticas dentro das empresas, e muitos deles continuam em operação até hoje [SRW97].

A partir de meados da década de oitenta, o cenário mundial da computação mudou significativamente com o surgimento das estações de trabalho [HP96] e a popularização das redes de computadores [Tan96] e dos sistemas gerenciadores de bancos de dados distribuídos [OV91]. Nos últimos anos, a capacidade de processamento das estações de trabalho se aproximou da capacidade de processamento dos *mainframes* [DLR95]. Além disso, as redes de computadores estão servindo de alicerce para a interconexão dessas estações de trabalho, surgindo assim computadores paralelos com capacidade de processamento superior a de qualquer *mainframe*.

A custos menores e com escalabilidade maior que a dos *mainframes* [HP96], o ambiente distribuído se tornou durante a década de 90 a melhor opção para o desenvolvimento de novos sistemas de informação. A indústria detectou nesse novo ambiente potencial para a abertura de novos mercados e negócios. Como resultado desta mudança de paradigma, a indústria tem convergido seus esforços de desenvolvimento de novas tecnologias para o ambiente distribuído, em detrimento do ambiente centralizado. Além disso, muitos produtos para ambientes centralizados foram descontinuados, ou seja, seus fabricantes deixaram de evoluir esses produtos e de prestar suporte técnico para os mesmos.

Considerando o cenário atual da computação, os sistemas legados provavelmente se tornaram um grande problema para muitas empresas [BLS97]. Com a evolução da computação, os usuários deste final de milênio passaram a exigir dos sistemas de informação, inclu-

sive dos sistemas legados, interfaces gráficas, tempos de resposta pequenos, extração *on-line* de relatórios gerenciais resultantes da consolidação de milhões de registros, etc. [Shn92]. Além de não oferecerem nenhuma dessas facilidades, os sistemas legados tornam-se ainda mais complexos em razão de terem sido desenvolvidos utilizando tecnologias que atualmente estão em desuso e desconhecidas para a maioria dos profissionais de informática disponíveis no mercado.

1.1 Migração de Sistemas Legados

Face aos problemas inerentes aos sistemas legados, é esperado que durante os próximos anos muitas empresas resolvam migrar esses sistemas para um ambiente distribuído, incorporando muitas das novas tecnologias disponíveis. A migração desses sistemas para um ambiente distribuído, entretanto, não é um problema trivial, pois são inúmeros os obstáculos a serem superados, tais como:

- a definição de estratégias de migração adequadas para cada tipo de sistema legado, bem como a avaliação do risco associado a cada estratégia de migração;
- a identificação ou desenvolvimento de ferramentas para facilitar o processo de migração de sistemas legados;
- a predição das características do comportamento dos novos sistemas derivados de sistemas legados, denominados sistemas alvos, quando de sua efetiva implantação em operação.

O processo de migração de sistemas legados além de possuir toda a complexidade dos sistemas de informação ainda agrega novos desafios a serem pesquisados e superados. Um estudo que tenta mostrar toda a amplitude do problema de se migrar um sistema legado é o de [BLS97], pois nele são descritos os vários componentes do processo de migração. Um outro trabalho que mostra bem a complexidade envolvida no processo da migração de sistemas legados é [AMR94] que descreve como o Departamento de Defesa dos Estados Unidos vem há mais de uma década tentando unificar os bancos de dados de seus sistemas legados.

A migração de sistemas legados nos últimos anos tem se tornado um dos principais assuntos de estudo da área de *engenharia de software* que aposta no uso de engenharia reversa como a melhor solução para o problema. O número de maio de 1994 da *Communication of the ACM* e o de janeiro de 1995 da *IEEE Software* são referências clássicas sobre o problema da migração de sistemas legados através do uso de técnicas de engenharia reversa. Nessas publicações encontramos estudos de casos [BH95, DV95, MNB⁺94, WTMS95] que relatam os problemas existentes e as soluções encontradas para se migrar sistemas legados, ficando clara a intenção dos autores em relatar as lições aprendidas. Na publicação da IEEE o enfoque é mais para as técnicas que estão sendo desenvolvidas para possibilitar

migrações mais fáceis no futuro. Em [NEK94] é apresentado um conjunto de ferramentas para identificar e extrair componentes de programas COBOL para serem reaproveitados no desenvolvimento dos sistemas alvos e [BMW94] descreve o uso de um conjunto de ferramentas denominado DESIRE que facilita a recuperação de informações referentes ao projeto conceitual do sistema legado.

Como sistemas de informação, os sistemas legados também utilizam da tecnologia de bancos de dados, de forma que esta também é uma área que se preocupa com o problema da migração de sistemas legados. Em [PB94] e [PH95] são descritos os problemas existentes na migração de aplicativos que utilizam bancos de dados relacionais, assim como são propostos métodos para a migração de bancos de dados relacionais.

Um livro específico sobre o problema de migração de sistemas legados é [BS95]. Nele os autores propõem um método genérico para se realizar a migração de sistemas legados, definindo a migração incremental como a forma mais segura para se sair de um sistema legado e chegar a um sistema alvo. A idéia básica apresentada é a de migrar o sistema por partes, sendo a interface, o banco de dados, o ambiente computacional e a aplicação os componentes do sistema legado. Assim, para se migrar o banco de dados, por exemplo, seria utilizado um *gateway* de banco de dados que faria a interface entre a aplicação e os bancos de dados legado e alvo. À medida que a aplicação e os dados fossem sendo migrados para o banco de dados alvo, menos se usaria o *gateway*, até que finalmente todos os dados e a aplicação tivessem sido completamente migrados, o que dispensaria o uso do *gateway*.

1.2 Análise de Desempenho para Migração de Sistemas

Como relatado na seção anterior, diversos trabalhos sobre o problema da migração de sistemas legados foram pesquisados, embora nenhum deles trate especificamente do problema de verificar se o ambiente computacional do sistema alvo comportará a carga de trabalho para o qual o ambiente está sendo dimensionado.

Por outro lado, de acordo com [MAD94] e [Jai91], estudos de planejamento de capacidade devem ser realizados de forma contínua durante todo o ciclo de vida de um sistema computacional, desde sua concepção, e enquanto ele se encontrar em operação.

Considerando a inexistência de estudos que descrevam como deve ser realizada a verificação do desempenho e da capacidade de um ambiente computacional durante a migração de sistemas legados, e a necessidade contínua de se realizar estudos de planejamento de capacidade para sistemas computacionais, esta dissertação tem como objetivo apresentar uma estratégia para se executar tal verificação. Um estudo de caso real é utilizado para demonstrar como a estratégia apresentada pode ser efetivamente utilizada para se realizar análises de desempenho e de planejamento de capacidade durante a migração de sistemas legados.

O estudo de caso realizado neste trabalho apresenta a Companhia de Saneamento de Minas Gerais (COPASA-MG) como uma empresa que se encontra perfeitamente ajustada para trabalhar com seu sistema comercial legado, denominado de Sistema HP07. O número de usuários, os recursos computacionais e o espaço físico da empresa se encontram dimensionados para os tempos de resposta das transações *on-line* do sistema legado. Assim, o conhecimento prévio do comportamento do novo sistema comercial, denominado de SICOM, possibilitará que a COPASA-MG se ajuste ao novo perfil de seu sistema, bem como possibilitará uma avaliação da migração antes que esta realmente ocorra.

Estando o sistema alvo completamente desenvolvido e seu ambiente computacional disponível é possível se obter experimentalmente algumas medidas de desempenho e de uso de recursos. Porém como o sistema alvo não se encontra operacional, não é possível se obter medidas de desempenho reais que capturem o efeito de múltiplos usuários concorrendo simultaneamente pelos recursos do sistema. Segundo [Jai91], entretanto, existem basicamente três técnicas para se analisar o desempenho de um sistema de computação e que solucionam o problema acima apresentado: uso de modelos analíticos, simulação e medições. Cada técnica apresenta seus requisitos, suas vantagens e suas desvantagens.

Neste estudo não foi cogitado o uso de medições no sistema com a finalidade de se realizar uma análise de desempenho da solução, pois tais medições além de serem onerosas e de difícil implementação (exige o uso de diversos equipamentos, usuários treinados e envolve o uso de redes remotas de computadores) não apresenta a flexibilidade necessária para o trabalho proposto. De fato, para se obter resultados confiáveis teriam de ser realizados inúmeros experimentos, em diversas condições, para, por exemplo, identificar o ponto de saturação do ambiente computacional. Assim restaram duas opções, o uso de modelos analíticos ou simulações.

O uso de modelos analíticos implicaria na necessidade de se adotar inúmeras simplificações, o que afastaria muito o modelo da situação real, comprometendo a validade dos resultados obtidos.

Logo será utilizada a simulação como a técnica para analisar o desempenho do sistema alvo. A técnica de simulação permite ao projetista do modelo especificar o nível de detalhamento necessário para se obter resultados confiáveis, além de poder ser repetido quantas vezes forem necessárias usando poucos equipamentos e sem envolver usuários.

1.3 O Método de Análise de Desempenho e Planejamento de Capacidade

Um método de análise de desempenho e planejamento de capacidade foi desenvolvido para ser usado durante a migração de sistemas legados para verificar se o sistema alvo funcionará de forma satisfatória no novo ambiente computacional. Mais especificamente, o método permite comparar os tempos de resposta das transações *on-line* dos sistemas

legado e alvo. Este método é composto das seguintes etapas:

1. Especificação da janela de medição.
2. Medições no sistema legado.
3. Identificação dos serviços relevantes do sistema legado.
4. Mapeamento de serviços do sistema legado para o sistema alvo.
5. Caracterização da carga de trabalho atual.
6. Modelagem do sistema alvo.
7. Calibragem e validação do modelo do sistema alvo.
8. Previsão da carga de trabalho futura.
9. Predição do desempenho do sistema.

Na primeira etapa do método é identificado no sistema operacional, ou observando as atividades da empresa em questão, um período no qual os serviços *on-line* devem possuir maior prioridade que os serviços de outras categorias, tais como os *batch jobs*. Este período é denominado de *janela de medição*.

Na segunda etapa são realizadas algumas medições do sistema legado com o uso de ferramentas adequadas, conhecidas como monitores. Dos dados obtidos são derivados dois parâmetros que compõem o modelo da carga de trabalho do sistema: a frequência de execução de cada categoria de serviço *on-line* sumarizado por hora e o tempo de resposta médio dos dos serviços *on-line*.

Na terceira etapa são selecionados os serviços que irão compor a carga de trabalho sintética do sistema alvo. Os serviços são selecionadas de uma lista dos serviços medidos na segunda etapa e ordenados pelas suas frequências de execução, de forma que um pequeno subconjunto dos serviços mais frequentemente executados represente um percentual significativo do total de serviços executados. O percentual depende da precisão desejada para os resultados obtidos.

Na quarta etapa os serviços são mapeados do sistema legado para o sistema alvo. Como somente parâmetros de transações são medidos nos sistemas legado e alvo, e transações normalmente não podem ser diretamente mapeadas entre os sistemas legado e alvo, então transações são mapeadas em serviços no sistema legado e serviços são mapeados em transações no sistema alvo. Serviços são conjuntos de ações convencionadas sobre as funções disponíveis no sistema, como será melhor descrito no Capítulo 3.

Na quinta etapa os demais parâmetros necessários para se caracterizar a carga de trabalho sintética do sistema alvo são obtidos diretamente do sistema alvo, monitorando-se individualmente cada transação.

Na sexta etapa o sistema alvo é modelado de forma que todos os aspectos a serem investigados estejam representados no modelo.

Na sétima etapa o modelo do sistema é resolvido através de técnicas de simulação, usando como parâmetro o modelo de carga de trabalho do sistema. Os resultados obtidos da simulação devem ser comparados com medições do sistema real e ajustados, quando for o caso. Este processo de ajuste do modelo em função de medições do sistema real é chamado de calibragem do modelo do sistema.

Uma vez que o modelo atinja um nível de precisão satisfatório em representar o sistema real, ele é classificado como validado. Assim, a partir de modelos validados é possível prever o comportamento que o sistema real terá, quando submetido a uma carga sintética. Desta forma, a oitava e a nona etapas consistem em modificar o modelo de carga de trabalho atual para uma suposta carga de trabalho ao qual o sistema será submetido no futuro. Usando esta carga como parâmetro no modelo validado, é possível prever como será o comportamento futuro do sistema, bem como comparar o tempo de resposta médio simulado com o tempo de resposta médio do sistema legado medido na segunda etapa.

1.4 Organização da Dissertação

Esta dissertação apresenta a seguinte organização. O Capítulo 2 apresenta as características dos sistemas legado e alvo usados como estudo de caso. O Capítulo 3 define o escopo e os elementos utilizados para se realizar o estudo de análise de desempenho do sistema alvo (etapas de 1 a 4 do método). O Capítulo 4 apresenta o estudo de caracterização da carga de trabalho prevista para o sistema alvo (etapa 5 do método). O Capítulo 5 descreve as simulações em ambiente descentralizado e cliente-servidor que foram realizadas usando a carga de trabalho previamente caracterizada (etapas de 6 a 9 do método). Neste capítulo ainda são apresentadas as análises comparativas dos resultados obtidos nas simulações. O Capítulo 6 conclui o trabalho apresentando uma revisão dos estudos realizados e dos principais resultados obtidos, e também indica quais os trabalhos futuros que poderão ser desenvolvidos a partir desta dissertação.

Capítulo 2

Caracterização do Estudo de Caso

2.1 Introdução

Esta dissertação tem como objetivo estudar os problemas existentes no processo de migração de um sistema legado centralizado para um ambiente distribuído. Seu principal enfoque está na caracterização do novo sistema de informação, que efetivamente ainda não se encontra em produção, a partir da análise do sistema legado do qual foi derivado, e que ainda se encontra em produção. Dentro deste escopo, esta dissertação utiliza como estudo de caso o atual Sistema Comercial da COPASA-MG, também conhecido por Sistema HP07, bem como o SICOM – Sistema Integrado Comercial que está sendo desenvolvido desde 1993 e que entrará em operação durante o ano de 1998.

O Sistema HP07 foi desenvolvido durante a primeira metade da década de 80 na PRODEMGE – Companhia de Processamento de Dados do Estado de Minas Gerais. Na época em que o Sistema HP07 foi desenvolvido, a cultura dominante era a dos *mainframes*. Assim, a PRODEMGE desenvolveu o Sistema HP07 neste ambiente e utilizando o sistema gerenciador de banco de dados (SGBD) ADABAS [ADA97].

A escolha do ambiente operacional do HP07 certamente foi uma escolha acertada para a época, pois o sistema se mostrou capaz de suportar a demanda de serviço desde que entrou em operação, bem como de suportar o crescimento da carga de trabalho durante mais de uma década. De fato, o sistema que foi implantado em 1985 se encontra em operação até hoje.

Um fato importante na história do Sistema HP07 foi a criação do Centro de Processamento de Dados (CPD) da COPASA-MG em 1989 [San95]. Nesse ano, a COPASA-MG achou que seria melhor se o processamento de dados fosse todo realizado dentro da própria empresa. Foi construído um prédio para abrigar o *mainframe* e seus dispositivos, assim como foram contratados novos analistas de sistemas e programadores para compor as equipes de desenvolvimento e manutenção de sistemas, sendo que uma destas equipes assumiu a manutenção do Sistema HP07.

Após cinco anos de operação, o Sistema HP07 começou a mostrar sintomas de que

não estava mais comportando a carga de trabalho demandada, bem como não mostrava a flexibilidade necessária para evoluir. Porém, mesmo não trabalhando conforme desejado pelos seus usuários, o sistema continua realizando todo o processamento necessário para que a COPASA-MG execute seus serviços comerciais.

Preocupada com a crescente obsolescência do Sistema HP07, a COPASA-MG resolveu avaliar a melhor forma de evoluí-lo de forma a agregar as novas demandas exigidas pela área comercial, bem como para processar a crescente demanda de serviços comerciais. Assim, em 1993 foi decidido que a melhor alternativa seria redesenvolver todo o Sistema HP07.

Face a uma proposta de parceria feita pela Consist Ltda., a empresa aceitou redesenvolver o Sistema HP07. Foi estabelecido o contrato de parceria entre as duas empresas no qual constava que o redesenvolvimento do Sistema HP07 seria realizado por uma equipe de analistas e programadores de ambas as empresas. Dentre as diversas condições firmadas no referido contrato, a Consist Ltda. exigiu que o sistema alvo fosse desenvolvido utilizando os produtos da Software AG, ou seja, que a linguagem de programação fosse a Natural e que o SGBD fosse o ADABAS (A Consist Ltda. é representante dos produtos da Software AG no Brasil). Assim começou a surgir o novo sistema comercial da COPASA-MG denominado de SICOM, abreviatura de Sistema Integrado Comercial.

Considerando que sistema legado é o sistema que resiste a mudanças e que a migração do sistema legado para um novo sistema adaptado aos requisitos e tecnologias atuais é uma forma eficaz de superar esta resistência, então a existência de um sistema legado implica na possível existência de um novo sistema denominado de sistema alvo. No estudo de caso que será realizado neste trabalho, o sistema legado em questão é o Sistema HP07 e o sistema alvo é o SICOM. Os termos HP07 e sistema legado, bem como SICOM e sistema alvo serão usados indistintamente neste trabalho.

2.2 Características Gerais

O Sistema HP07 é composto de 393 programas codificados nas linguagens COBOL e COBOL/CICS, 3.066 codificados em Natural e 213 procedimentos de execução de programas codificados em JCL [IBM97]. A maioria dos programas codificados em COBOL e COBOL/CICS e uns poucos programas codificados em Natural compõem o que pode ser chamado de núcleo do sistema. Mais precisamente, está sendo chamado de núcleo do Sistema HP07 o subconjunto de seus programas sem o qual o sistema é incapaz de executar os serviços considerados críticos para o funcionamento da empresa, tais como a emissão das contas de água, a consulta de débitos de clientes e do cadastramento de solicitações de serviços. Mesmo sendo o núcleo composto basicamente de programas codificados em COBOL, desde que o Sistema HP07 entrou em operação todos os novos serviços foram codificados em Natural.

O Sistema HP07 atualmente é responsável por inúmeros serviços fundamentais para o funcionamento da COPASA-MG. As principais funções do Sistema HP07 são: controlar

o cadastro de aproximadamente 1.900.000 ligações de água, controlar o processo de medição de consumo de água de cada ligação, e faturar e cobrar o consumo de água em cada uma dessas ligações. Além desse controle das ligações de água, o sistema ainda faz todo o controle do processo de execução de serviços realizados pela empresa, tais como o prolongamento das redes de água e esgoto, testes de vazamento de água, testes de análise química da qualidade da água, etc. Assim, o Sistema HP07 acompanha a execução de cada serviço desde a sua solicitação até a sua efetiva execução (que pode ser na área comercial, se for um serviço comercial, ou na área operacional, se for um serviço operacional). O módulo denominado de SATI – Sistema de Atendimento Integrado, está sendo considerado como um módulo do Sistema HP07. Além de todos esses serviços prestados diretamente para a área comercial, o Sistema HP07 ainda fornece informações para diversas áreas, tais como as áreas contábeis, financeira, operacional, planejamento, etc.

Considerando os serviços realizados pelo Sistema HP07, pode-se afirmar que esse é um sistema estratégico e essencial para a empresa. De fato, ocorrendo uma paralisação do sistema imediatamente os atendentes comerciais ficam incapacitados de prestar informações e de registrar as solicitações de serviços dos consumidores, bem como as áreas operacionais passam a controlar os serviços manualmente, o que pode levar à perda do controle de execução dos serviços. Persistindo a paralisação por mais de um dia, outros setores começam a ser afetados, tais como o de faturamento e de medição. À medida que o tempo de paralisação se prolongue, outros setores, não pertencentes às áreas comercial e operacional, param. De fato, paralisações longas não programadas poderiam comprometer até as atividades fins da empresa como a distribuição de água.

O SICOM foi inteiramente codificado em Natural, embora não possa ser visto apenas como uma recodificação do Sistema HP07 na linguagem Natural. Quando a equipe de desenvolvimento do SICOM fez o levantamento dos processos que seriam contemplados pelo sistema alvo, foram identificados processos informatizados por outros sistemas menores nas plataformas MS-DOS e UNIX, além daqueles já contemplados pelo Sistema HP07.

Após 6 anos de desenvolvimento (1993 a 1998), o SICOM foi concluído mesmo existindo alguns serviços a serem implementados. Como consequência deste esforço de desenvolvimento, foram codificados 6.528 programas Natural e 403 *shell scripts* do UNIX, totalizando 1.291.087 linhas de código.

2.3 Problemas Legados e a Necessidade de Evoluir

Atualmente muitas organizações estão substituindo sistemas de informação que se encontram operacionais há muitos anos por outros novos, desconhecidos e, provavelmente, muito menos estáveis do que os substituídos. Essas organizações certamente possuem bons motivos para justificar tais mudanças. Muitos desses motivos são devidos ao que está sendo chamado de *problemas legados*.

A evolução tecnológica na computação é muitas vezes mais rápida do que a capacidade

das organizações de se manterem atualizadas. Como consequência deste processo, muitos produtos, sejam equipamentos ou *software*, são rapidamente descontinuados pelos seus fabricantes. Na maioria das vezes, essas empresas oferecem para seus clientes um produto mais moderno para substituir o obsoleto, de forma a garantir uma assistência técnica para seus produtos, pois uma vez descontinuado o produto geralmente deixa de ter assistência técnica.

Desta rápida evolução tecnológica surgem outros problemas periféricos. Com a popularização do uso de microcomputadores, os usuários começam, por exemplo, a ter mais contato com computadores com interfaces gráficas, multimídia e tempos de resposta praticamente nulos. Confrontando o microcomputador com o *mainframe* o usuário deixa de aceitar com naturalidade a necessidade de ter de trabalhar com terminais monocromáticos e não gráficos e com sistemas com tempos de resposta elevados. Enfim, o usuário deixa de estar satisfeito com a qualidade dos serviços oferecidos pelo sistema computacional na empresa.

O caso em estudo neste trabalho apresenta alguns problemas que foram decisivos para que a COPASA-MG optasse pela migração do Sistema HP07. A seguir serão discutidos três problemas legados existentes no caso em estudo e que dão uma idéia das dificuldades que levam ao processo de migração de sistemas.

2.3.1 Projeto do Banco de Dados

Após mais de 10 anos de contínua manutenção no Sistema HP07, o projeto original do banco de dados do sistema foi consideravelmente modificado, resultando num banco de dados complexo e difícil de ser usado por programadores e analistas de sistema. De fato, o banco de dados do Sistema HP07 é repleto de atributos não usados na aplicação (e não documentados), assim como de atributos redundantes.

Em princípio, a afirmação acima pode parecer uma crítica à qualidade dos serviços realizados pelos profissionais que trabalharam com o Sistema HP07, mas não o é. O banco de dados se encontra com estes problemas principalmente devido à dificuldade de se modificar esquemas de bancos de dados ADABAS. Um banco de dados ADABAS é composto fisicamente de arquivos, denominados de arquivos ADABAS [ADA97]. Assim, verifica-se que para eliminar um atributo é necessário reconstruir o arquivo ADABAS no qual ele se encontra definido, o que na maioria das vezes é inviável devido ao custo e ao risco envolvido em tal procedimento. Estruturas do ADABAS como *grupos periódicos* [ADA97] também não podem ser modificadas sem que seu respectivo arquivo ADABAS seja reconstruído.

Assim, a Figura 2.1 mostra um exemplo desse tipo de problema no qual um grupo periódico PER-MULTA-Z é composto exclusivamente do atributo VALOR-MULTA-URV porque não foi possível criar um novo atributo dentro do grupo periódico PER-MULTA sem se exportar e importar o arquivo ADABAS HP07-CONTAS que é um dos maiores e um dos mais usados arquivos do banco de dados ADABAS da COPASA-MG.

```

DB 0      File 116  - HP07-CONTAS
T L  DB  Name                                     F Leng  S D
- -  - -  -----
  1  AB  TIPO-REGISTRO                             A    1  F
  1  AA  NUM-MATRICULA                             N   7,0  N
G 1  QD  GRUPO-CONTA
  2  HR  NUM-REG-DIST                             N   2,0  N
  2  HM  COD-LOCALIDADE                           N   4,0  N
  2  IH  NUM-SETOR-ROTA                           N   4,0  N
  2  AF  NUM-CONTA                                 N   5,0  N
  1  DH  DATA-REFERENCIA                         N   4,0  N
P 1  GM  PER-MULTA
  2  MR  DATA-REF-MULTA                           N   4,0  N
  2  MD  DATA-VENCTO-MULTA                       N   6,0  N
  2  MP  DATA-PAGAMENTO-MULTA                   N   6,0  N
  2  MQ  VALOR-CONTA-MULTA                       N   9,2  N
  2  MV  VALOR-MULTA                              N   9,2  N
P 1  S3  PER-MULTA-2
  2  S4  VALOR-MULTA-URV                          P   3,2  N
  1  AT  COD-FATURAMENTO                           N   2,0  N
  1  A1  COD-TIPO-DOCUMENTO                       A    1  N

```

Figura 2.1: Exemplo de reprojeto no banco de dados do HP07

Outro tipo de problema, mais específico dos sistemas legados, é da implementação ao longo do ciclo de vida da aplicação de novos serviços que exigem a modificação de entidades existentes. Devido à importância do Sistema HP07 e ao elevado número de programas que devem ser modificados, para se mudar entidades geralmente os desenvolvedores optam pela criação de uma pseudo-entidade. Por pseudo-entidades estão sendo chamadas as estruturas de dados com atributos redundantes acrescidas de alguns novos atributos em relação a entidades existentes no banco de dados. Como consequência do uso repetido desta estratégia, o banco de dados atualmente possui vários atributos redundantes, assim como existem processos usados para manter o sincronismo entre entidades e pseudo-entidades.

2.3.2 Sistema Gerenciador de Banco de Dados

O SGBD ADABAS é um produto muito robusto e com excelente desempenho, motivo pelo qual ele é um dos SGBDs mais utilizados até nossos dias, mesmo após a consagração dos SGBDs relacionais. Como qualquer outro produto, o ADABAS continua evoluindo constantemente e é deste processo de evolução que surgiu um tipo de problema no Sistema HP07.

A evolução de um SGBD muitas vezes ocorre dentro do próprio produto, tal como na implementação de algum algoritmo de gerência de *buffer pool* mais eficiente, por exemplo. Melhorias deste tipo são transparentes até mesmo para o administrador de banco de dados (ABD). Algumas outras somente não são transparentes para o ABD, que pode ter que ajustar um ou outro parâmetro para que o SGBD continue operando normalmente. Porém existem mudanças que não são transparentes para os programadores e analistas de sistemas.

Essas mudanças costumam ser de fácil solução mesmo num sistema do porte do HP07, porém existem algumas que são completamente desastrosas em termo de impacto na programação. Como muitas vezes não existem pessoas em número suficiente para implementar tais mudanças, então a empresa assume os riscos de manter produtos desatualizados em operação em troca do esforço de modificar centenas ou até mesmo milhares de programas.

```
MOVE ZEROS TO VALOR-MULTA78(01)
           VALOR-ENTRADA78(01)
           VALOR-ACRESCIMO78(01).
PERFORM INCLUSAO-ADABAS78.
```

(a)

```
EXEC ADABAS
  READ LOGICAL
  DECLARE USU1 CURSOR FOR
  SELECT TIPO-REGISTRO, NUM-MATRICULA, GRUPO-CONTA,
         TIPO-FATURAMENTO-DE, TIPO-LIGACAO-AGUA-DE,
         QTDE-RESIDENCIAL, QTDE-COMERCIAL, QTDE-INDUSTRIAL
  FROM HP07-USUARIOS USUL
  WHERE SUPER-GRUPO-CONTA >= :SUPER-GRUPO-CONTA-AUX
  OPTIONS HOLD
         SUFFIX=02
  ORDER BY SUPER-GRUPO-CONTA
END-EXEC
```

(b)

```
MOVE 'CL' TO COMMAND-CODE.
CALL 'ADABAS' USING CONTROL-BLOCK  FORMAT-BUFFER
                  RECORD-BUFFER   SEARCH-BUFFER
                  VALUE-BUFFER    ISN-BUFFER.
```

(c)

Figura 2.2: Formas diferentes de um programa COBOL acessar o SGBD ADABAS.

Como resultado desta contínua evolução do SGBD ao longo dos anos, e de repetidas decisões de não se modificar centenas de programas, existem programas COBOL que codificam as operações de banco de dados de diversas formas. A Figura 2.2 mostra exemplos de fragmentos de código de um programa COBOL que codifica e decodifica as operações a serem enviadas e recebidas do SGBD ADABAS de três formas diferentes. Na forma (a) é usado um comando MACRO ADAMINT. Na forma (b) é usado um comando ADABAS SQL, que é um pré-compilador COBOL que implementa comandos SQL embutidos. Na forma (c) é usada a forma nativa de se comunicar com o SGBD, denominada de *direct call* [ADA95a].

A forma de comunicação (a) é um exemplo do risco de se manter operacional sistemas legados. Muitos dos programas componentes do chamado núcleo do Sistema HP07 ainda hoje utilizam o ADAMINT, que é um produto descontinuado pelo seu fabricante há muito tempo. Isto significa que praticamente nenhum profissional é capaz de solucionar problemas resultantes do uso deste produto. Assim, da noite para o dia, a COPASA-MG pode se ver envolvida num processo de modificação de milhares de programas para manter seu principal sistema de informação operante.

2.3.3 Manutenção

Atualmente os profissionais que estão sendo formados possuem conhecimento de linguagens modernas que utilizam interfaces gráficas e técnicas de programação orientada a objetos, tais como o Delphi da Borland e o Visual BASIC da Microsoft. Este é um mundo completamente diferente e muito distante dos *mainframes* onde até hoje se programa em COBOL, FORTRAN, PL/1 e muitas outras linguagens obsoletas. Considerando este cenário, observa-se que não se formam profissionais aptos para trabalhar com tecnologias obsoletas. Por outro lado, os profissionais experientes e conhecedores das tecnologias legadas também estão migrando dos *mainframes* para o mundo dos PC's com Windows. Assim, a partir da Figura 2.2 é fácil constatar que será cada vez mais difícil e dispendioso para as organizações encontrarem profissionais que saibam dar manutenção em programas COBOL, isto sem considerar o uso de ADAMINT, ADABAS SQL, etc.

2.4 Estratégia de Migração

2.4.1 A Migração em Três Etapas

Para se compreender o processo de migração do Sistema HP07 é necessário se conhecer um pouco da estrutura organizacional da COPASA-MG.

Estrutura Organizacional da COPASA-MG

Como uma empresa estadual, a COPASA-MG atua em mais de 500 municípios de todo o Estado de Minas Gerais, sendo alguns destes municípios pertencentes à Região Metropolitana de Belo Horizonte (RMBH), e os demais pertencentes ao interior do Estado. Na RMBH a maioria das unidades administrativas da empresa estão reunidas num único local denominado de Regional Metropolitana, enquanto as unidades operacionais, denominadas de distritos operacionais, se encontram fisicamente distribuídas pela RMBH. No interior do Estado as unidades são organizadas hierarquicamente. No primeiro nível desta hierarquia se encontram as gerências regionais. No segundo nível e subordinados às gerências regionais se encontram os distritos. Cada distrito, por sua vez, é responsável pelas atividades administrativas e operacionais da COPASA-MG no conjunto de municípios sob sua responsabilidade. A Figura 2.3 apresenta o mapa do Estado de Minas Gerais composto por suas sete gerências regionais, bem como pela Região Metropolitana de Belo Horizonte. Ainda nesta figura estão identificadas as cidades que são sede de gerências regionais e/ou distritos do interior do Estado, e a cidade de Belo Horizonte.

A Estratégia de Migração Original

Atualmente todo o processamento do Sistema Comercial da COPASA-MG é realizado pelo *mainframe* localizado na RMBH. Neste trabalho este ambiente está sendo chamado de *centralizado*. O SICOM, entretanto, foi projetado para trabalhar de forma distribuída, ou seja, de forma que cada gerência regional possuísse seu próprio CPD no qual existiriam computadores servidores que processariam os serviços do SICOM. Neste trabalho este ambiente está sendo chamado de *descentralizado*.

A distribuição como proposta pelo SICOM não implica na existência de um SGBD distribuído para gerenciar transações de banco de dados distribuídas, como descrito em [OV91]. A estratégia de distribuição do SICOM assume algumas simplificações que, conforme estudos realizados pela sua equipe de desenvolvimento, não teriam maiores impactos na empresa. Para exemplificar uma dessas simplificações, um consumidor da COPASA-MG somente pode solicitar um serviço ou consultar seus débitos em cidades pertencentes à gerência regional na qual o consumidor se encontra cadastrado.

De acordo com a estratégia de distribuição do SICOM, os bancos de dados da empresa seriam fragmentados por gerência regional. Assim, os dados cadastrais de um consumidor estariam no fragmento de banco de dados da gerência regional da qual seu município pertence. A maioria das *views* (terminologia do ADABAS para arquivos lógicos do banco de dados) corporativas seriam regionalizadas, o que também é de interesse da empresa (tarifas de água regionalizadas, por exemplo). Algumas outras *views* corporativas poderiam ser mantidas somente na RMBH, por não serem usadas no âmbito das gerências regionais. Finalmente as *views* corporativas que não foram regionalizadas, e que são usadas por toda a empresa, seriam replicadas através de mecanismos de atualização assíncronos,



Figura 2.3: Mapa do Estado de Minas Gerais dividido em sete gerências regionais e a Regional Metropolitana de Belo Horizonte (RMBH). As cidades são sedes de distritos e/ou gerências regionais. A Gerência da Regional Norte (GRNT) é a que contém as cidades de Montes Claros, Januária e Janaúba.

assumindo que seus dados praticamente não sofrem atualizações. Para manter atualizadas as *views* corporativas existentes na RMBH (plano de contas da empresa, estatísticas de abastecimento, etc) seriam criadas transações de atualização do banco de dados da RMBH a partir de suas gerências regionais. Como o volume de transações desta natureza é insignificante perto do volume de transações locais às gerências regionais (além de serem todas processadas no *mainframe* da RMBH), então elas não estarão sendo consideradas neste estudo.

A Estratégia de Migração Vigente

Preocupados com o problema de migrar o Sistema HP07 diretamente para o ambiente descentralizado, os técnicos e gerentes da COPASA-MG se reuniram para discutir o assunto. Nesta reunião foi verificado que a estratégia de migração inicial apresentava muitos riscos. De fato, além de estarem sendo migrados a aplicação, o banco de dados da aplicação e os dados da aplicação, também estavam sendo modificados o sistema operacional e a forma de

teleprocessamento. Mais ainda, a empresa estaria pela primeira vez operando um sistema do porte do SICOM numa plataforma totalmente nova, da qual quase nada se sabe em termos práticos a respeito de desempenho de tais sistemas computacionais.

Como já era planejada a migração do Sistema HP07 para o SICOM/C na RMBH, a empresa decidiu mudar a estratégia de migração inicial migrando todo o Sistema HP07 para o SICOM/C, o que possibilitaria que o SICOM/D fosse implantado de forma gradativa em cada gerência regional da empresa. Essa mudança de estratégia possibilitaria uma pré-validação da aplicação SICOM antes de implantada nas gerências regionais, assim como minimizaria os riscos da migração tanto do SICOM/D como do próprio SICOM/C. De fato seriam preservados o sistema operacional e a estrutura de teleprocessamento existentes.

Conhecida a estratégia de migração do Sistema HP07, este trabalho também avaliará as vantagens e desvantagens, em termos de uso dos recursos computacionais e de desempenho, de se migrar do SICOM/D para o SICOM/CS. Desta forma, a estratégia de migração do Sistema HP07 ocorreria em três etapas: a etapa do SICOM/C, a do SICOM/D e a do SICOM/CS.

2.4.2 O SICOM/C

O ambiente computacional do SICOM/C é praticamente o mesmo do Sistema HP07. Ele é um ambiente multiusuário baseado num *mainframe* IBM 9121 com 512 Mbytes de memória RAM e 24 canais de comunicação de dados. O sistema operacional é o IBM MVS/ESA 5.1 e o monitor de TP é o IBM CICS/VS 2.1 [SC391]. Seu subsistema de E/S possui 180 Gbytes de espaço em disco usando RAID5 e *spare disks*. Conectados a ele existem 850 terminais IBM 3278 (*dumb terminals*) ou microcomputadores emulando terminais IBM 3278. A rede de teleprocessamento é controlada pelo IBM VTAM usando o protocolo de comunicação BSC3, e os *links* remotos usam o serviço de comutação de pacotes MINASPAC da Telemig. O SICOM/C usará o SGBD ADABAS, o ambiente Natural e o sistema de segurança Natural Security da Software AG, assim como o sistema Common Function da Consist Ltda., todos estes *softwares* atualmente em uso pelo Sistema HP07.

Em relação à migração do Sistema HP07 para o SICOM/C, a COPASA-MG está assumindo que a demanda de recursos computacionais do SICOM/C será três vezes e meia maior que a do Sistema HP07. Esta é uma estimativa informal feita pela equipe de desenvolvimento do SICOM. Este aumento na demanda de recursos possui algumas justificativas. Talvez o mais forte argumento a favor do aumento da demanda de recursos seja o fato da COPASA-MG e da Consist Ltda. terem decidido pela normalização do banco de dados, o que indubitavelmente aumentará de forma significativa o volume de dados no banco de dados devido à redundância de atributos em praticamente todas as *views* do sistema, assim como isso provavelmente irá aumentar a complexidade das consultas ao banco de dados, já que o SICOM não tirará proveito das estruturas não normalizadas do ADABAS como campos múltiplos e periódicos, que são atributos multivalorados. Além desta justificativa, pode ser verificado que vários novos serviços foram agregados ao sistema alvo, o que

aumentou indiretamente a complexidade de vários serviços existentes no HP07.

Considerando o aumento da demanda de recursos computacionais por parte do SICOM/C e a estratégia de inicialmente se migrar o HP07 inteiramente para o SICOM/C, a COPASA-MG teve de redimensionar temporariamente os recursos computacionais de seu *mainframe*, sendo a escalabilidade destes equipamentos fundamentais para que a migração ocorra sem maiores problemas devido a problemas de desempenho.

2.4.3 O SICOM/D

Sendo o ambiente do SICOM/D descentralizado, cada gerência regional pode ser vista (e estudada) isoladamente. Analisando as características de cada gerência regional, se verifica que cada uma possui um número médio de usuários ativos diferente, distribuídos por um número diferente de distritos e executando quantidades diferentes de serviços. Devido às características diversificadas, cada gerência regional deve ser estudada isoladamente. Como o ambiente do SICOM/D pode ser estudado por gerência regional e a COPASA-MG já havia definido que o SICOM/D seria implantado inicialmente na Gerência da Regional Norte (GRNT), então o ambiente computacional objeto de estudo neste trabalho será o da GRNT.

Conforme mostrado na Figura 2.3, a GRNT é composta dos distrito de Montes Claros, Januária e Janaúba. A sede da GRNT fica localizada em Montes Claros no mesmo prédio do Distrito de Montes Claros. Na GRNT foram implantadas três redes locais *ethernet* de 10 Mbits, uma em Januária, uma em Janaúba e outra em Montes Claros, assim como foram instalados dois *links* remotos X.25 entre Montes Claros e Januária e entre Montes Claros e Janaúba.

Com a migração do SICOM/C para o SICOM/D ocorre uma mudança significativa em termos de ambiente computacional que é a mudança de plataforma do servidor de banco de dados. Embora o SICOM/D também faça uso de um ambiente multiusuário como o SICOM/C, o servidor de banco de dados e o sistema operacional são outros. Como os recursos computacionais da GRNT já foram adquiridos, é possível se descrever detalhadamente os equipamentos e *softwares* que serão usados quando o SICOM/D entrar em operação, e que foram parcialmente usados para se realizar os experimentos de validação do modelo de simulação do SICOM/D, conforme descrito na Seção 5.4.

Na rede local de Montes Claros esquematizada na Figura 2.4 se encontram duas estações IBM RISC R/6000 com 96 Mbytes de memória real, 512 Kbytes de memória cache e 1 disco SCSI-II de 9,5 GBytes de capacidade de armazenamento. Essas estações estão sendo denominadas de servidores de banco de dados. Uma dessas estações foi escolhida para funcionar como o servidor de banco de dados principal, enquanto que a segunda funcionará como um servidor de banco de dados de reserva (*backup*) para ser usado em situações de contingência. Nesses servidores estarão instalados o sistema operacional IBM AIX 4.1, o SGBD ADABAS 2.1 e a linguagem Natural 2.2. Estarão conectados ao servidor de banco de dados principal duas impressoras laser e uma impressora de impacto. Um terceiro

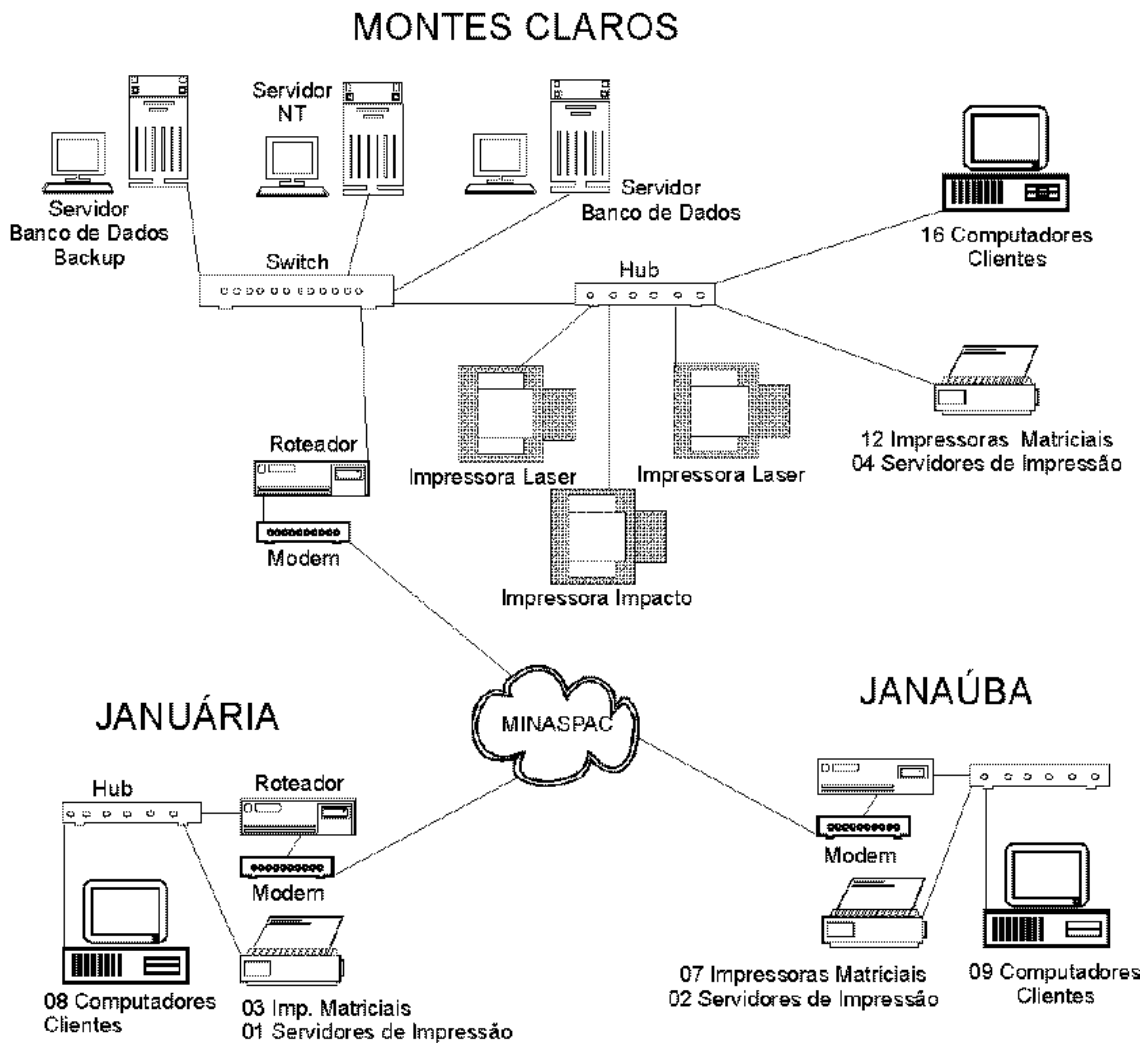


Figura 2.4: Rede de computadores da GRNT

servidor Pentium 266 MHz com 64 Mbytes de memória real e com 1 disco SCSI-II de 4,5 Gbytes, denominado de servidor de aplicativos, estará presente nessa rede. Seu sistema operacional será o Windows NT e sua função será a de armazenar utilitários e arquivos de trabalho de usuários usados nos computadores clientes, tais como pacotes gráficos, editores de texto, planilhas eletrônicas, etc. Quanto aos computadores clientes, existirão 16 computadores Pentium 166MHz com 32 Mbytes de memória real, todos usando o sistema operacional Windows 95. Para os serviços de impressão locais existirão na rede local 12 impressoras matriciais. Para segmentar o acesso aos servidores, existirá um chaveador (*switch*). Finalmente, para conectar a rede local de Montes Claros com a MINASPAC (e a seus distritos), existirá um roteador.

As redes locais de Janaúria e Janaúba são semelhantes entre si, diferenciando basicamente no número de computadores clientes, pois na primeira existirão 8 computadores clientes e na segunda 9, todos com a mesma especificação dos computadores clientes de Montes Claros, e também usando o sistema operacional Windows 95. Além dos computa-

dores clientes, cada rede local possuirá um servidor de aplicativo com a mesma especificação do servidor de aplicativo de Montes Claros e um roteador para conectar a rede local à MINASPAC. Em Janaúria existirão 3 impressoras matriciais e em Janaúba existirão 7 impressoras matriciais.

Quanto ao uso do SICOM/D, os usuários da GRNT inicialmente terão de solicitar uma conexão `telnet` entre o computador cliente e o servidor de banco de dados. Estes usuários poderão solicitar a conexão `telnet` de microcomputadores localizados na própria rede local do servidor de banco de dados, que no caso da Gerência da Regional Norte correspondem aos usuários do Distrito de Montes Claros e aos usuários da própria Gerência da Regional Norte, ou poderão ainda estar solicitando a conexão `telnet` de microcomputadores remotos, que na GRNT correspondem aos microcomputadores dos distritos de Janaúba e Janaúria. No caso das conexões remotas, estarão sendo usados os serviços da MINASPAC, que é um serviço de comutação de pacotes da Telemig que implementa o protocolo X.25, das gerências regionais aos seus respectivos distritos. Um vez estabelecida a conexão `telnet`, os usuários se conectam ao sistema operacional UNIX (AIX 4.1), que automaticamente chama o programa Natural responsável pelo menu principal do SICOM/D.

2.4.4 O SICOM/CS

No SICOM/CS ocorre a distribuição do processamento, em relação ao SICOM/D, visto que a aplicação será executada nos próprios computadores clientes, enquanto que o servidor de banco de dados da regional será usado exclusivamente para executar o processamento *batch* e para atender às consultas ao SGBD submetidas pela rede.

Existem duas diferenças básicas em termos de componentes do ambiente SICOM/D para o SICOM/CS. A primeira é que a aplicação SICOM não mais estará no servidor de banco de dados, mas nos próprios computadores clientes, assim como também deverá estar a linguagem Natural. A segunda diferença é a presença de um novo componente que deverá estar presente em todos os computadores clientes e nos servidores de banco de dados, que é o Entire Network [WCP94]. Este produto pertence a uma categoria de produtos denominados de *stub* e que basicamente fazem o roteamento de requisições e respostas entre um programa num computador cliente com o SGBD localizado num servidor de banco de dados. Assim, toda requisição de banco de dados passa a ser solicitada à cópia local do Entire Network, que se comporta para a aplicação como se fosse o próprio SGBD, resolvendo a requisição com a ajuda das demais cópias do Entire Network e dos SGBDs ADABAS presentes na rede de computadores. Desta forma, os usuários se conectarão diretamente à aplicação que estará disponível no próprio computador cliente sem terem de estabelecer uma conexão `telnet` com o AIX, assim como também não terem de se conectar ao AIX.

Uma vantagem esperada pelo SICOM/CS em relação ao SICOM/D é a redução da carga de trabalho do servidor no banco de dados, pois deixará de ser o responsável pela execução de toda a aplicação SICOM, passando a executar somente a cópia local do Entire

Network, que certamente demanda muito menos recursos que a aplicação SICOM, pois tem como única função rotear as requisições de banco de dados entre os computadores clientes e seus servidores.

Capítulo 3

Medição e Seleção de Serviços On-Line

3.1 Escopo do Estudo de Desempenho

Considerando que a *solução de um sistema alvo* abrange os recursos humanos, a infraestrutura (instalações físicas e elétricas), os recursos computacionais, os *softwares* básicos (sistema operacional, SGBD, utilitários, etc.), o treinamento dos usuários e a aplicação propriamente dita, o estudo da migração de um sistema legado deve abranger esta solução como um todo.

Durante o processo de migração de sistemas legados as organizações geralmente focalizam sua atenção nas aplicações, deixando num segundo plano todos os demais componentes das soluções. Entretanto todos os componentes são necessários para que o sistema alvo funcione, pois de nada adianta que todos os componentes estejam disponíveis se não existir eletricidade para ligar os computadores. Ou ainda, mesmo estando os recursos computacionais bem dimensionados, a solução não irá funcionar se os usuários não souberem operar o sistema por falta de treinamento. Enfim, as organizações que estejam migrando sistemas legados devem se preocupar em tornar disponíveis todos os componentes das soluções dos sistemas alvos, para que eles funcionem como esperado.

Os componentes da solução do sistema alvo podem ser classificados em administrativos e computacionais. Neste sentido, estão sendo classificados como administrativos os componentes da solução que são de responsabilidade do corpo gerencial da organização, tal como o treinamento dos usuários para operarem o sistema. Os demais componentes estão sendo classificados de computacionais por estarem sob a responsabilidade do corpo técnico da área de informática da organização.

Os componentes computacionais da solução do sistema alvo, por sua vez, são compostos de *softwares* e de equipamentos. Desconsiderando a própria aplicação que é um *software*, verifica-se que todos os demais componentes que são *software* são usados mundialmente

e em ampla escala, o que garante a todos eles uma qualidade de serviço internacional. De fato, é esperado que esses componentes não falhem quando submetidos à carga de trabalho gerada pelo uso do sistema alvo. No caso dos *softwares* estão sendo considerados como falhas, problemas que não possibilitem o seu uso, como um erro de programação interno. No caso dos equipamentos as falhas são físicas e em geral facilmente identificadas. Assim, mesmo que nenhum componente falhe ainda é possível que a solução como um todo falhe. Um exemplo disto é o subdimensionamento de equipamentos. Se não existir memória suficiente, por exemplo, para que uma quantidade mínima necessária de processos executem simultaneamente, então a solução como um todo poderá falhar. O estudo de capacidade de um sistema de computação é chamado de planejamento de capacidade, e neste trabalho estarão sendo usadas técnicas de planejamento de capacidade, aliadas às técnicas de análise de desempenho, para verificar se não ocorrerão problemas decorrentes do subdimensionamento de recursos computacionais quando da migração do SICOM/C para o SICOM/D.

Assim, o escopo deste trabalho é analisar o comportamento da solução SICOM/D, bem como da solução SICOM/CS, considerando que todos os componentes administrativos destas soluções foram providenciados, e que todos os componentes computacionais da solução funcionem sem apresentar falhas. Como a principal diferença entre o SICOM/C e o SICOM/D é o ambiente computacional – mudança do ambiente proprietário do *mainframe* para um ambiente aberto como o UNIX, assim como o uso de uma arquitetura cliente-servidor – então pode ser dito que este estudo analisa os aspectos tecnológicos da migração do sistema legado HP07.

Uma ressalva sobre o correto funcionamento da aplicação SICOM se faz necessária. Revendo a estratégia de migração do SICOM, descrita na Seção 2.4, verifica-se que a migração do sistema legado para o sistema alvo centralizado validará grande parte dos serviços do SICOM. Assim, assume-se que a aplicação SICOM funcione corretamente, ou seja, que os programas realizem as operações para as quais foram desenvolvidos, assim como o projeto físico do banco de dados consiga atender às operações de banco de dados requisitadas pela aplicação. Certamente um sistema com as dimensões do SICOM apresenta falhas de codificação, bem como devem existir melhores estratégias de projeto de banco de dados para alguma parte do esquema de banco de dados, porém está sendo considerado que essas imperfeições serão corrigidas com a evolução do sistema, e que nenhuma delas comprometerá de forma decisiva o bom funcionamento do sistema alvo.

Lembrando que o sistema alvo é complexo e que possui mais de 1 milhão de linhas de código fonte, que a migração do Sistema HP07 para o SICOM/C já é uma realidade para a COPASA-MG, e que os equipamentos necessários para implantar o SICOM/D se encontram instalados na Gerência da Regional Norte, constata-se que um trabalho que estude todos os aspectos do comportamento do SICOM não é viável considerando o cronograma de migração do sistema. Surge então a necessidade de reduzir o escopo deste trabalho para adequá-lo ao cronograma de implantação do SICOM.

Continuando o estudo exploratório do sistema alvo, foram identificados três tipos de

serviço no SICOM: o *on-line*, o *batch* e o *ftp*. Os serviços *on-line* e *batch* são naturalmente esperados para um sistema do porte do SICOM, e serviços interativos não foram detectados no conjunto de serviços disponibilizados pelo SICOM. A necessidade de se definir uma classe de serviços *ftp* se deve à sua característica híbrida, apresentando algumas características dos serviços *batch* e outras de serviços *on-line*. De fato, ao se definir os parâmetros de classe no Capítulo 4, constatou-se que seus parâmetros não são iguais aos dos demais serviços.

Preliminarmente podemos desconsiderar o estudo dos serviços *ftp*, pois existindo na solução SICOM os recursos computacionais necessários para se executar os serviços *ftp*, certamente estes recursos comportarão a carga de trabalho resultante. Em outras palavras, está sendo considerado que a utilização de recursos computacionais pelos serviços *ftp* é mínima, de forma a não comprometer o desempenho da solução SICOM como um todo. Por outro lado, o comportamento da solução SICOM em termos da utilização de recursos pelos serviços *batch* e *on-line* é totalmente desconhecida para a própria COPASA-MG. Como é esperado que a utilização de recursos seja elevada tanto para um serviço quanto para o outro, vemos que estes serviços são o nosso objeto de estudo.

Como este estudo visa validar a solução tecnológica do SICOM na migração do SICOM/C para o SICOM/D, a escolha é pelo estudo dos serviços *on-line*, pois o maior impacto tecnológico nesta migração está no ambiente *on-line*. De fato, pela primeira vez o Sistema Comercial da COPASA-MG estará sendo executado fora de uma plataforma *mainframe* e utilizando a rede de computadores da COPASA-MG.

Problemas referentes ao desempenho dos serviços *batch* geralmente podem ser resolvidos através da melhoria do desempenho do servidor de banco de dados, tal como a substituição de um processador por outro com maior capacidade de processamento. Neste caso estamos supondo que existe a capacidade de melhorar o servidor de banco de dados, devido ao fato de ser a tecnologia RISC altamente escalável [HP96], assim como pelo fato da CPU adquirida para a Gerência da Regional Norte suportar uma expansão de sua capacidade de processamento, de memória e de subsistema de E/S. O desempenho dos serviços *on-line*, por outro lado, além de depender do desempenho do servidor de banco de dados ainda depende do desempenho da rede de computadores, ou seja, do desempenho das redes locais, dos *links* remotos, dos dispositivos de rede, etc. Porém os recursos da rede de computadores não apresentam a mesma escalabilidade do servidor de banco de dados, além de poderem surgir limitações fora do controle da própria COPASA-MG. Caso se verifique que algum *link* remoto esteja subdimensionado, por exemplo, será necessário solicitar à Telemig, empresa responsável pelo serviço MINASPAC, o aumento da taxa de transmissão do *link* remoto. Este incremento da taxa de transmissão, entretanto, depende da existência de recursos computacionais da Telemig, que podem simplesmente não existir.

Concluindo, este estudo irá se restringir aos aspectos tecnológicos envolvidos na migração do SICOM/C para o SICOM/D, e para realizar este trabalho será estudado o ambiente *on-line* do SICOM/D.

3.2 Terminologia

Neste trabalho existem termos utilizados em áreas distintas da computação com sentidos bem diferentes. Para evitar problemas de interpretação desses termos, eles são definidos a seguir:

- **Transação On-Line** – Corresponde a cada interação do usuário com a aplicação através de terminais¹, tanto para solicitar serviços, quanto para receber as respostas destas solicitações. Para que seja possível esse tipo de interação usuário/aplicação, as transações *on-line* geralmente processam pequenos volumes de dados, o que possibilita que sejam executadas em poucos segundos. De forma mais prática, uma transação *on-line* num *mainframe* corresponde a uma transação do CICS/VS. Em particular, os nomes de transações *on-line* que empregaremos neste capítulo e na Seção 4.2.1 correspondem aos nomes das transações CICS, compostos de quatro caracteres.
- **Serviço** – Uma aplicação é composta de diversos serviços, sendo cada serviço um conjunto de ações convencionadas sobre as funções disponíveis no sistema (como as opções nos menus, por exemplo). Neste sentido diz-se que um usuário de uma aplicação a utiliza serviço a serviço. Como é de se supor, serviços dentro de uma mesma aplicação podem apresentar características bem diferenciadas. Num sistema de folha de pagamento, por exemplo, um serviço pode ser simplesmente a consulta ao salário de um empregado, como pode ser a execução de toda a folha de pagamento, imprimindo os comprovantes de pagamento e gerando as movimentações bancárias dos salários para as contas dos empregados. Esta definição também aceita que um sistema operacional seja visto como uma aplicação, assim os serviços do sistema operacional também podem ser vistos como serviços.
- **Serviço On-Line** – São serviços compostos de uma seqüência de transações *on-line*. Essa seqüência de transações *on-line* não é rígida, porém é bem comportada, ou seja, para um dado serviço *on-line* é possível se identificar um número finito de combinações de transações *on-line* que o compõem.
- **Serviço Batch ou Batch Job** – Os serviços que processam grandes volumes de dados não são adequados para serem utilizados de forma *on-line*, pois seu processamento pode durar horas, ou mesmo dias, o que tornaria indisponível o terminal por igual período. Para evitar esse tipo de problema existem os *batch jobs* que são serviços solicitados através de formulários, ou mesmo através de serviços *on-line*, que são executados desassociados de terminais, com prioridades menores, e cujas respostas são normalmente apresentadas através de relatórios impressos.

¹Estão sendo chamados de *terminais* os clássicos terminais TTY (*dumb terminals*) e os computadores clientes de redes de computadores cliente-servidor.

- **Serviço ftp** – Está sendo chamado de serviço *ftp* o próprio serviço *ftp* do TCP/IP. Ele está sendo citado nesta terminologia para enfatizar que este tipo de serviço se adapta à definição de *serviço* acima e não é um serviço *on-line* e nem *batch*.
- **Carga Transacional ou Carga On-Line** – Segundo [FSZ83], uma carga de trabalho é toda a requisição de processamento submetido a um sistema, por uma comunidade de usuários, durante um dado período de tempo. Assim, chama-se de carga transacional à carga de trabalho resultante do processamento de todas as transações *on-line* num dado período de tempo.

3.3 O Ambiente On-Line

Como estamos interessados na carga *on-line* da COPASA-MG, devemos conhecer o comportamento da aplicação como um todo em função do tempo para podermos identificar o período de uso no qual a carga *on-line* é máxima. De fato, devemos não somente compreender como a carga *on-line* se comporta em função do tempo, mas como toda a carga do sistema se comporta em função do tempo, pois a qualquer instante podem existir serviços de todos os tipos sendo executados.

O processamento da carga de trabalho pelo servidor de banco de dados seguirá um esquema de prioridade tradicionalmente adotado pelo Centro de Processamento de Dados (CPD) da COPASA-MG. Neste esquema as horas dos dias úteis são agrupadas em períodos definidos como janela *on-line* e janela *batch*. Essas denominações vêm do fato de serem as transações *on-line* prioritárias na janela *on-line*, enquanto os *batch jobs* prioritários na janela *batch*. Na realidade, as transações *on-line* sempre são prioritárias em relação aos *batch jobs*, pois seus tempos de resposta devem ser sempre os menores possíveis, porém durante a janela *batch* o número de transações *on-line* em execução é extremamente reduzido em relação à janela *on-line*. Desta forma, como a prioridade é para todo o conjunto de transações, o que ocorre em termos práticos é o aumento do número de *batch jobs* executados (*throughput*) sem a redução do tempo de resposta das transações *on-line*.

A janela *on-line*, embora também chamada de horário comercial, não corresponde exatamente ao horário comercial oficial da COPASA-MG que é definido como sendo o período das 8:00 às 18:00 horas, de segunda a sexta-feira, enquanto que a janela *on-line* é definida como sendo o período das 7:00 às 19:00 horas, também de segunda a sexta-feira. Essa extensão no início e no fim do horário comercial é devida à necessidade de diversas áreas da COPASA-MG executarem serviços *on-line* imediatamente antes e/ou logo após o horário comercial oficial da empresa.

A janela *batch* corresponde a toda a semana, excluída a janela *on-line*. Os fins-de-semana e feriados também estão dentro da janela *batch*.

Como estamos particularmente interessados na carga *on-line* do sistema legado, então devemos manter o monitor de transações *on-line* ativo durante todo o período em que durar a janela *on-line*.

3.4 Mapeamento de Transações entre os Sistemas Legado e Alvo

Uma tarefa fundamental no estudo proposto é a realização do mapeamento entre as transações *on-line* do sistema legado e as transações *on-line* do sistema alvo. A necessidade deste mapeamento é óbvia, pois se estamos interessados em comparar tempos de resposta entre as transações dos sistemas legado e alvo, devemos conseguir identificar que uma determinada transação *on-line* do sistema alvo é equivalente a uma dada transação *on-line* do sistema legado. Infelizmente, comparando as transações *on-line* de ambos os sistemas, verificamos que não existe tal mapeamento direto entre elas.

Para resolver este problema foram identificados os serviços *on-line* que são realizados em ambos os sistemas. O cadastramento de um consumidor é um exemplo de um serviço *on-line* existente tanto no sistema legado quanto no sistema alvo. Esses serviços *on-line* possuem duas características importantes para se construir o mapeamento de transações *on-line* entre os sistemas legado e alvo: são elementos comuns e facilmente identificados em ambos os sistemas, e como conjuntos de transações *on-line* permitem identificar como e quais são as suas transações.



Figura 3.1: Esquema de mapeamento de transações *on-line*.

Definido que o mapeamento de transações *on-line* será realizado usando o mapeamento de serviços *on-line*, podemos completar nosso esquema de mapeamento conforme mostra a Figura 3.1. Dada uma transação *on-line* no sistema legado é possível determinar a qual serviço *on-line* a transação pertence. Conhecido o serviço *on-line* no sistema legado automaticamente é conhecido o serviço *on-line* no sistema alvo. Finalmente, conhecido o serviço *on-line* no sistema alvo é possível se identificar as transações *on-line* do sistema alvo que são equivalentes às transações *on-line* do sistema legado.

Na Seção 3.5 são identificados os principais serviços *on-line* do sistema legado. No Capítulo 4 é apresentado o mapeamento existente entre os serviços *on-line* dos sistemas legado e alvo de nosso estudo de caso, assim como são identificadas e caracterizadas as transações *on-line* das categorias de serviços representativos da carga de trabalho *on-line* do SICOM.

3.5 Serviços Representativos da Carga de Trabalho On-Line do Sistema HP07

3.5.1 Monitoração das Transações On-Line

Com o objetivo de se medir a frequência de utilização dos serviços *on-line* do sistema legado foi utilizado o monitor CICS MANAGER da Boole & Babbage [TD-95]. Este monitor armazena em arquivos VSAM [SC294] todas as informações disponíveis de cada transação *on-line* sendo executado sob um determinado monitor de teleprocessamento, ou monitor de TP. No caso da COPASA-MG, esse monitor de TP é o IBM CICS/VS 2.1.2 [SC391].

Uma vez que o CICS MANAGER tenha sido ativado é possível consultar o arquivo VSAM através de utilitários do CICS MANAGER que geram relatórios pré-definidos, ou ainda através de programas escritos em Performance Reporter Language (PRL) [TD-94] que geram relatórios personalizados selecionando, consolidando e mostrando os dados da forma desejada.

Inicialmente foram coletados dados sobre todas as transações *on-line* da COPASA-MG. A Tabela 3.1 mostra uma pequena parte dos dados coletados das transações *on-line* do Sistema HP07. Porém, ao se analisar esses dados foi detectado que a consolidação das medições por transação *on-line* não era suficiente para fornecer os dados necessários. De fato, dentro do conceito de transação do IBM CICS/VS existem transações responsáveis por diversos serviços, como é o caso da transação SATI responsável por todos os serviços de atendimento do sistema legado, enquanto existem serviços que são executados através de inúmeras transações, como é o caso da maioria das antigas transações COBOL/CICS. O fato de ser um serviço realizado por diversas transações será melhor discutido quando formos descrever o modelo de simulação.

Transação On-line	Número Transações	Tempo CPU	Tempo Resposta	Tempo Espera E/S	Número Chamadas E/S
HPAT 08	69	0.622	11.560	10.851	556
HPAX 08	91	1.107	18.991	17.578	575
HPPC 08	266	6.627	85.907	78.773	3656
HPSS 08	504	9.432	129.246	119.362	6823
SATI 08	166	4.562	555.242	546.986	3525
HPAT 09	12	0.094	2.014	1.915	84
HPAX 09	30	0.419	11.276	10.777	192
HPPC 09	229	5.623	72.939	66.982	2580
HPAT 10	39	0.497	8.149	7.674	560
HPAX 10	90	1.357	23.803	22.377	1053
SATI 10	241	5.266	86.649	79.401	4055

Tabela 3.1: Amostra das medições de parâmetros de desempenho de transações *on-line*.

Considerando os problemas acima diagnosticados, foi necessário se descartar as me-

dições realizadas, refazer o programa de consulta às medições do CICS MANAGER, e novamente medir toda a carga *on-line* da COPASA-MG, desta vez consolidando as medições por transação *on-line* e programas.

Com o uso dessa estratégia de medição foi resolvido o problema de saber quais serviços estão sendo executados dentro de uma transação *on-line* que torna disponível diversos serviços, porém ainda teríamos de depurar manualmente os programas componentes das transações *on-line* para identificar os serviços mais representativos.

3.5.2 Representatividade das Transações On-Line Seleccionadas

As transações *on-line* da COPASA-MG foram monitoradas no período de 25 a 31 de janeiro de 1998 durante o horário comercial. Foram executados 2.982 programas em 2.022.335 transações *on-line*. O desafio desta etapa do trabalho é, portanto, identificar neste conjunto de 2.982 programas o menor subconjunto de programas representativo da carga de trabalho *on-line* do Sistema HP07.

Conforme a Tabela 3.1, foram coletados e consolidados por transação e programas os seguintes dados sobre as transações *on-line* do HP07: frequência de execução, tempo de uso de CPU, tempo de espera pelo subsistema de E/S, número de chamadas ao SGBD e tempo de resposta.

Escolher dentre os dados disponíveis qual é o mais relevante em relação à carga de trabalho não é uma decisão óbvia. Programas mal codificados podem, por exemplo, usar CPU e subsistema de E/S por períodos prolongados sem serem representativos da carga de trabalho, assim como o número de chamadas ao SGBD pode ser nulo no caso de programas intensamente utilizados que não usam dados armazenados em bancos de dados. Mais ainda, no caso da migração de sistemas legados nem sempre existe uma correlação entre parâmetros de desempenho de ambientes diferentes [Gra93], como os ambientes legado e alvo. De fato, a necessidade de comparar ambientes computacionais diferentes tem levado a indústria a criar métricas tais como o TPC-C, SPECint92 [Gra93], etc.

Embora não seja um dado isento de problemas, consideramos a frequência de execução de programas como uma bom ponto de partida na identificação dos programas representativos da carga de trabalho *on-line* do HP07. Assim, a estratégia adotada consiste em ordenar todas as consolidações de transações/programas por sua frequência de execução, para posteriormente se verificar se os programas selecionados também são representativos em termos dos demais dados medidos, corrigindo as eventuais distorções que apareçam. Tratadas todas as consolidações, montamos uma tabela ordenada pelas frequências com 2.982 linhas. As transações *on-line* da semana de 18 a 24 de janeiro de 1998 também foram monitoradas, porém foram desconsideradas devido à perda das medições do dia 21. Desta forma, aproveitamos as medições dos demais dias desta semana para comparar com os resultados da semana de referência, da qual foi verificada a precisão das medições realizadas, pois as tabelas apresentavam diferença em apenas uma única transação de uso menos freqüente que as demais.

O próximo problema era identificar onde fazer o corte na tabela de consolidações de modo a obter o menor número de programas representativos da carga de trabalho. Para auxiliar nesta decisão, foi desenvolvido um programa na linguagem Delphi [Bor97], que manipulando a tabela de consolidações de transações calculava e mostrava de forma gráfica e quantitativa a representatividade dos programas selecionados em relação ao total de programas executados, isto para cada um dos tipos de dados medidos, conforme mostra a Figura 3.2.

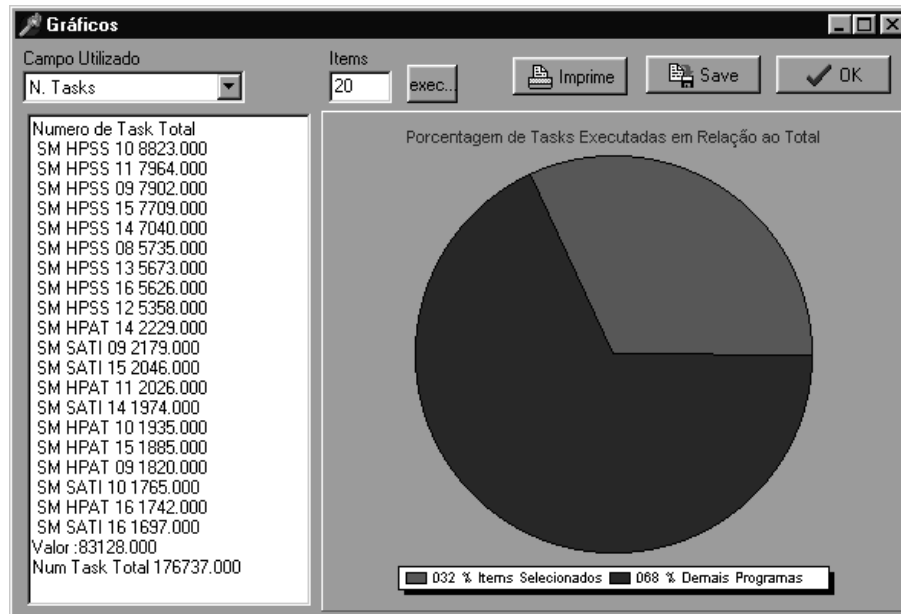


Figura 3.2: Ferramenta de inspeção visual e quantitativa.

Utilizando essa ferramenta de inspeção visual, concluímos que o corte deveria ser feito no vigésimo nono programa porque o impacto da inclusão dos programas seguintes no conjunto selecionado não iria aumentar de forma significativa sua representatividade (0,61% na inclusão de cada um dos próximos três programas). Ainda a partir do programa de inspeção visual, pôde-se observar que os 29 programas selecionados representavam:

- 0,73% de todos os programas executados durante a semana de referência;
- 31% do tempo de uso de CPU;
- 30% do tempo de uso do subsistema de E/S.

A Tabela 3.2 mostra alguns parâmetros de desempenho medidos para os 29 programas mais utilizados pelas transações *on-line* durante a semana de referência. Das transações *on-line* desses programas verificamos que duas são usadas pelos utilitários de impressão de relatórios *on-line* (PRNT e NAFF), três são usadas pelo próprio monitor de TP (CSAC, CSPK e CSGM), uma é usada pelo Sistema de Ponto Eletrônico (NATU VHP16094) e uma é usada

Transação On-line	Programa	Número Transações	Tempo Uso CPU	Tempo Resposta	Tempo Espera E/S	Número Chamadas E/S
PRNT	PGP0473	208464	790,818	70473,765	0,000	0
HPSX	VHP07421	117403	1218,112	9707,754	7748,069	343223
HPSX	VHP07428	64528	982,083	19718,011	18397,552	728954
HPSX	VHP07426	58854	1252,999	41987,416	40493,066	1058181
HPPX	VHP07470	50283	706,273	1857,144	654,275	22074
SATX	ST123100	49616	1013,638	32129,566	30805,674	662768
NATU	NHP07REC	49585	450,373	4524,805	3964,778	3473575
HPAT	PHP07401	45592	580,772	16464,873	15897,208	622680
HPCS	PHP07442	44581	297,559	15029,908	14679,365	254479
HPAL	PHP07417	44070	1400,490	41045,259	39981,415	2129423
HPSX	NHP07424	43496	791,976	30443,598	29587,506	1070484
HPPX	VHP07472	42921	1423,65	38517,662	36843,399	956995
NAFP	NC226RE	39799	5575,035	775804,638	31378,809	1974724
HPSX	VHP07427	33685	695,416	15297,981	14476,616	750295
CSAC	DFHACP	31711	51,475	205,186	0,000	0
HPSX	NC226RE	22166	209,283	3690,458	3164,433	78749
HPIC	PHP07443	21208	148,954	8301,019	8130,588	145578
HPSS	PGP07NAT	20769	919,881	12160,508	11226,445	749745
HPIA	PHP07403	19395	183,840	4414,019	4193,197	97382
CSPK	DFHPRK	18901	28,740	56,069	0,000	0
HPSX	VHP07472	17996	576,028	13953,888	13314,696	366744
HPCA	PHP07411	16401	81,114	2726,01	2616,885	60652
NATU	VHP16094	16319	984,875	14271,665	13375,49	772553
SATX	ST112000	15475	389,037	13445,204	13017,946	574058
NATU	NHP07TRT	15100	301,307	6057,615	5734,558	317118
CSGM	DFHGMM	14197	20,518	4673,459	0,000	0
HPL1	PHP07433	14122	112,121	5925,111	5803,026	114020
HPSX	VHP07423	13168	170,346	4533,086	4328,013	138933
NATU	NC226RE	13143	126,097	3498,501	3230,583	75080

Tabela 3.2: Programas *on-line* mais usados.

pelo núcleo do Natural (NATU NC226RE). Todos os demais programas são do Sistema HP07, o que mostra a importância da carga de trabalho *on-line* do Sistema HP07 na carga de trabalho *on-line* da COPASA-MG, assim como a relevância deste trabalho.

3.5.3 Identificação dos Principais Serviços On-Line

Conhecidos os 22 programas, e suas respectivas transações, representativos da carga de trabalho *on-line* do Sistema HP07, devemos identificar de quais serviços são essas transações *on-line*.

Tradicionalmente sistemas legados não possuem documentação atualizada [WTMS95], problema este presente no Sistema HP07. Para contornar este problema foi necessária a leitura do código fonte desses programas, assim como eventuais entrevistas com analistas de sistemas, analistas comerciais e demais usuários do sistema. Como resultado deste esforço, foram identificadas as funções de cada programa componente das transações *on-*

line selecionadas, conforme mostrado na Tabela 3.3.

Transação <i>on-line</i>	Programa	Descrição
HPSS	VHP07421	Identificação de consumidor
	VHP07423	Pesquisa logradouro (<i>help</i>)
	VHP07424	Pesquisa bairro (<i>help</i>)
	VHP07426	Pesquisa SS por consumidor
	VHP07427	Pesquisa SS por código solicitação
	VHP07428	Gera SS
SATI	ST123100	Baixa OS
	ST112000	Recebimento de OS
HPCS	PHP07442	Identificação do consumidor e pesquisa de serviços comerciais
	PHP07443	Alteração de OS comerciais
HPAT	PHP07401	Identificação do consumidor
	PHP07403	Demonstrativo de débitos de consumidores
HPCA	PHP07411	Identificação do consumidor
	PHP07417	Alteração de dados cadastrais
HPPC	VHP07470	Identificação do consumidor
	VHP07472	Relatório com dados cadastrais, de faturamento e medições
HPLC	PHP07433	Lançamento em conta

Tabela 3.3: Identificação dos programas mais utilizados.

Comparando esta tabela com a Tabela 3.2, verifica-se que não foram relatadas as funções dos programas NHP07REC, NHP07TRT, PHP07NAT e NC226REC. De fato, os dois primeiros programas são usados no processo de transmissão de arquivos entre Belo Horizonte e os distritos do interior e serão substituídos pelo serviço *ftp* do TCP/IP. O que é identificado como o programa PHP07NAT na realidade é uma tabela residente no CICS e que associa transações *on-line* com os primeiros programas Natural a serem executados pela transação, correspondendo, portanto, a nenhum serviço *on-line*. Finalmente, o programa NC226RE é o próprio núcleo do Natural do qual nada pode ser dito a respeito da execução de funções do Sistema HP07.

Das funções identificadas para os programas vemos que algumas representam o próprio serviço *on-line*, tais como os programas ST123100 e PHP07403 que representam, respectivamente, os serviços de *baixa de ordem de serviço* e *alteração cadastral de consumidores*. Outros programas, entretanto, apenas identificam parte do processamento de um serviço *on-line*, tal como o programa VHP07423 que ajuda o usuário a identificar o código de um logradouro.

A COPASA-MG é uma empresa prestadora de serviços e, como tal, ela recebe, programa e executa serviços demandados pela população. Alguns dos serviços *on-line* identificados nesta seção são referentes a serviços executados pela empresa. Assim, para diferenciar o serviço prestado pela empresa do serviço prestado pelo sistema de informação, e melhorar o entendimento desta dissertação, as solicitações de serviço recebidas pela COPASA-MG serão chamadas de SS, e as ordens de serviço, documentos criados para controlar a execução

dos serviços realizados pela COPASA-MG, serão chamadas de OS.

Após conhecidas as funções de todos os 22 programas, esses foram os serviços identificados como representativos da carga de trabalho *on-line* do Sistema HP07:

- **Alteração Cadastral** – Realiza alterações nos diversos dados cadastrais do consumidor no sistema, tais como dados pessoais, pontos de serviço servidos, situação dos hidrômetros, etc.
- **Alteração de OS** – Este serviço procura corrigir algum dado incorreto entrado durante a geração de uma SS.
- **Baixa de OS** – Após a execução de um serviço, é executada esta função com intuito de informar ao sistema o resultado da execução deste, assim como fornecer dados adicionais como, por exemplo, se é necessária a abertura de uma nova OS.
- **Consulta Dados Cadastrais** – Realiza uma consulta completa aos dados cadastrais do consumidor.
- **Consulta Débitos** – No caso de existirem débitos, este serviço consulta todos os dados sobre cada um dos débitos existentes.
- **Gera Lançamento em Conta** – Caso o consumidor solicite algum serviço que seja cobrado pela COPASA-MG, utiliza-se este serviço para gerar um lançamento em sua respectiva conta de água.
- **Consulta Dados Consumidor** – Fornece um resumo das informações cadastrais do cliente para uma rápida consulta aos seus dados. São apresentados dados pessoais, sobre faturamento, medição do consumo de água e pontos de serviço.
- **Gera SS** – Este serviço identifica o consumidor e registra no banco de dados uma SS na qual estão descritos os dados necessários para se realizar o serviço demandado. Cada SS gera automaticamente uma OS, que pode gerar outra OS quando baixada, até que não seja mais necessária a geração de outras OS, o que resulta na baixa tanto da OS, quanto da SS. Desta forma, vemos que uma SS gera uma cadeia de OSs composta no mínimo de uma OS.
- **Pesquisa SS** – Pesquisa a SS em andamento para determinado consumidor, informando o seu estado atual e a previsão para conclusão.
- **Recebimento OS** – Interface criada na implantação do SATI para inserir no seu banco de dados informações referentes às ordens de serviço comerciais. Também é através deste serviço que é realizada a programação das OS a serem executadas.
- **Verifica Débitos** – Informa ao consumidor a existência de débitos pendentes, se existirem.

Capítulo 4

Caracterização da Carga de Trabalho

4.1 Compreensão do Ambiente

Muitas características do SICOM foram apresentadas nos Capítulos 2 e 3. A composição da rede de computadores da Gerência da Regional Norte, incluindo as características dos computadores servidores e clientes, foi descrita nas Seções 2.4.3 e 2.4.4. Os serviços *on-line* a serem utilizados foram descritos na Seção 3.5.3. Assim, resta verificar como serão os *batch jobs* e os serviços *ftp* para compreendermos como será o ambiente operacional do SICOM.

4.1.1 Batch Jobs

A carga *batch* será intensa durante a janela *batch*, porém como o escopo de nosso estudo se restringe ao horário comercial, então teremos somente que compreender como serão os *batch job* durante este período.

O ideal seria que não existissem *batch jobs* sendo executados durante o horário comercial, ou seja, que todo o processamento *batch* fosse executado durante a janela *batch*. Isto não ocorrerá porque a rotina de impressão, que é um *batch job*, estará submetendo os *spools* das contas de água para duas impressoras laser e os relatórios *batch* para uma impressora de impacto durante todo o horário comercial. Assim esses *batch jobs* concorrerão com as transações *on-line* no uso da CPU e do subsistema de E/S. Isto se deve ao fato de serem impressoras dispositivos lentos e exigirem um acompanhamento constante de operadores para suprir formulários, papéis, *toners*, fitas de tinta, etc. É importante observar que tanto os *spools* das contas de água, quanto os relatórios *batch* que serão impressos durante o horário comercial foram processados e armazenados no subsistema de E/S durante a última janela *batch*.

4.1.2 Serviços ftp

A necessidade de transmitir arquivos de dados entre o *mainframe* e seus terminais, no caso do Sistema HP07, ou entre o servidor e suas estações clientes, no caso do SICOM, se deve à existência de uma rotina que transmite para os distritos do interior os dados necessários para se emitir e consolidar os *boletins de leitura*, documentos usados pelos leituristas para registrar as medições de consumo de água. Esta rotina foi redesenvolvida dentro da mesma sistemática de utilização do Sistema HP07, o que significa que continuará existindo a necessidade da transmissão de arquivos de dados entre o computador servidor e os computadores clientes.

O serviço *ftp* do TCP/IP surgiu como solução natural para um problema de transmissão de arquivos de dados na COPASA-MG. No Sistema HP07, por exemplo, a transmissão ocorre através do uso de transações *on-line* que consultam arquivos ou o banco de dados no *mainframe*, mostrando na tela do microcomputador remoto os dados desejados. Já no microcomputador, um programa captura os dados exibidos na tela armazenando-os num arquivo local. Embora primitivo, este procedimento é uma solução estável para o problema apresentado e é utilizado por todos os distritos da COPASA-MG.

4.2 Medições de Dados

Identificados os principais serviços responsáveis pela carga de trabalho *on-line* do SICOM, devemos continuar a caracterização da carga de trabalho através de medições, tanto no sistema legado quanto no sistema alvo, dos parâmetros que caracterizam os serviços das classes *on-line*, *batch* e *ftp*.

4.2.1 Medições no Sistema Legado

Neste momento estamos interessados em determinar o número de vezes que cada uma das transações identificadas na Seção 3.5 foi executada no horário de pico semanal na Gerência da Regional Norte. Desta forma, foram adotados os mesmos critérios adotados para se utilizar o CICS MANAGER descritos na Seção 3.3.

Como estávamos interessados em saber o número de transações *on-line* em cada distrito/regional, tivemos de mapear, dentro de um programa codificado em Performance Report Language (PRL) [TD-94], os 860 terminais existentes na COPASA-MG. Para isto, foi necessário associar o código de cada terminal ao respectivo distrito/regional a partir do Terminal Control Table do CICS/VS. Este programa PRL não somente mapeava os terminais por distrito/regional, assim como consolidava as transações/programas hora a hora.

O CICS MANAGER coletou dados sobre as transações *on-line* executadas no período de 16 a 20 de fevereiro de 1998. Após consolidar a cada hora o número total de transações executadas na semana, foi verificado que na Gerência da Regional Norte o horário de pico

era das 09:00 às 10:00 horas do dia 17 de fevereiro, uma quinta-feira. Este é um dado considerado consistente devido ao fato histórico de ser a quinta-feira o dia da semana em que o *mainframe* é mais utilizado na COPASA-MG.

Montes Claros			Januária			Janaúba		
HPSX	VHP07426	196	HPPX	VHP07470	59	NATU	NHP07REC	209
HPSX	VHP07421	184	HPAL	PHP07417	51	HPAT	PHP07401	23
HP1C	PHP07443	167	HPPX	VHP07472	49	HPPX	VHP07470	13
HPSX	VHP07428	130	HPL1	PHP07433	49	HPPX	VHP07472	12
SATX	ST123100	123	NATU	NC226RE	9	HP1A	PHP07403	12
HPCS	PHP07442	112	HPCA	PHP07411	7	HPAL	PHP07417	9
HPSX	NHP07424	109	HPPX	NC226RE	6	NATU	NC226RE	8
HPSX	VHP07472	74	HPAT	PHP07401	4	HPPX	NC226RE	7
HPSX	VHP07427	68	HP1A	PHP07403	2	HPL1	PHP07433	5
SATX	ST112000	67				HPCS	PHP07442	4
HPSX	NC226RE	56				HPCA	PHP07411	4
HPAT	PHP07401	48						
HPSS	PGP07NAT	44						
HPCA	PHP07411	35						
HPAL	PHP07417	21						
HP1A	PHP07403	19						
NATU	NC226RE	17						
HPPX	VHP07472	12						
HPPX	VHP07470	11						
HPPX	NC226RE	11						
HPL1	PHP07433	10						
HPSX	VHP07423	2						

Tabela 4.1: Transações *on-line* do HP07 na Gerência da Regional Norte, no horário de pico.

Durante o horário de pico da Gerência da Regional Norte foram encontradas transações de todos os tipos anteriormente selecionados, o que confirma a representatividade das transações selecionadas. A distribuição das transações *on-line* por distrito é apresentada na Tabela 4.1 onde são identificados seus respectivos códigos no CICS/VS (código de quatro dígitos), o código (oito dígitos) e a frequência de execução de cada programa. Aplicando o mapeamento de transações *on-line* para serviços *on-line* descrito na Seção 3.4 obtivemos o número de serviços *on-line* executados no horário de pico na Gerência da Regional Norte, conforme Tabela 4.2.

4.2.2 Medições no Sistema Alvo

Um serviço, seja *batch* ou *on-line*, irá usar uma certa quantidade de memória durante sua execução, irá usar a(s) CPU(s) por um determinado período de tempo e provavelmente irá buscar e armazenar dados nos discos. Analisando o funcionamento dos serviços existentes num sistema computacional se verifica que todos eles consomem recursos. Assim, a próxima etapa para caracterizar a carga de trabalho do SICOM é conhecer o tempo de uso de recursos de cada um dos serviços representativos da carga de trabalho.

Categoria	Frequência		
	Montes Claros	Januária	Janaúba
Gera SS	130	0	0
Consulta SS	68	0	0
Baixa OS	123	0	0
Recebimento OS	67	0	0
Consulta Cadastro Consumidor	28	17	1
Altera Cadastro Consumidor	7	0	3
Pesquisa Consumidor	86	59	12
Consulta Débito Consumidor	29	2	11
Detalhe Débito Consumidor	19	2	12
Lançamento em Conta	10	49	5
Altera Serviço Comercial	84	0	0
Consulta Serviç Comercial	28	0	4
FTP	0	0	1

Tabela 4.2: Número de serviços *on-line* executados no horário de pico do período de 16 a 20 de fevereiro de 1998 na Gerência da Regional Norte.

A partir do sistema legado foi possível identificar os serviços representativos da carga *on-line*, assim como a taxa de chegada desses serviços por unidade organizacional. Entretanto, será necessário medir o tempo de uso de cada recurso do sistema computacional por cada serviço representativo da carga de trabalho.

Conforme o modelo de simulação, que será detalhadamente descrito no Capítulo 5, são os seguintes os parâmetros a serem medidos:

- **Tempo de Uso de CPU** – Todos os programas, inclusive os que compõem o sistema operacional consomem CPU. No modelo a ser simulado os seguintes componentes usarão CPU: os programas Natural, os serviços *telnet* e *ftp* do TCP/IP, o SGBD ADABAS e a rotina de impressão *batch*.
- **Tempo de Uso do Subsistema de E/S** – No caso em estudo, os programas Natural e o serviço *telnet* não fazem uso do subsistema de E/S, o que é aceitável para o modelo de simulação proposto. De fato, todas as operações de E/S em disco realizadas pelo *telnet* ocorrem na inicialização de uma sessão, assim como os programas Natural usam o subsistema de E/S na primeira execução de um programa. Como estamos supondo que durante a janela de tempo usada na simulação os usuários não necessitarão de entrar e/ou sair do ambiente Natural e nem do *telnet*, então esses programas realmente não realizarão operações de E/S.

Por outro lado, tanto os serviços *batch* e *ftp* e o SGBD ADABAS realizam operações de E/S, sendo necessário especificar o tempo de uso do subsistema de E/S para cada um desses serviços e processos.

- **Tempo de Uso de Canal de Comunicação Local e Remoto** – Ao se supor a existência de uma rede de computadores, também está sendo suposta a existência de um meio de comunicação que estamos chamando de canal de comunicação. A análise

em relação aos canais de comunicação é quanto à sua utilização, pois eles possuem uma capacidade limitada de transferência de dados [Tan96], limitação esta que pode comprometer não só o desempenho mas o próprio funcionamento do SICOM. No caso estudado, conforme visto na Seção 2.4.3, existem cinco canais de comunicação a serem considerados: as redes *ethernet* da Gerência da Regional Norte e dos distritos de Janaúria e de Janaúba, e os dois *links* remotos entre Montes Claros e Janaúria e Montes Claros e Janaúba. Assim, a pergunta básica em relação aos canais é a seguinte: comportarão eles o tráfego de dados demandado pelo SICOM?

- **Número de chamadas ao SGBD** – Os programas de uma aplicação de banco de dados normalmente solicitam inúmeras operações ao SGBD. Nesta situação o programa que estava usando a CPU é momentaneamente suspenso à espera da resposta pelo SGBD, que acessa uma ou mais vezes o subsistema de E/S. Devido a esse procedimento é importante conhecermos o número de vezes que um determinado serviço solicita operações de banco de dados ao SGBD.
- **Número de transações on-line por serviço** – Em termos de carga *on-line*, o que chega ao servidor de banco de dados são serviços *on-line*, conforme descrito na Seção 3.3. Logo se faz necessário conhecer o número de transações *on-line* que compõem cada serviço *on-line*, já que o estudo é baseado nos tempos de resposta das transações *on-line*.

Conhecidos os parâmetros necessários para se caracterizar a carga de trabalho do SICOM, resta medi-los. Para determinar quantas transações *on-line* compõem um serviço *on-line* basta se utilizar o sistema alvo e contar quantas vezes o usuário interage com o mesmo (ou quantas vezes as teclas ENTER e de funções foram acionadas). Porém, para determinar os valores para os demais parâmetros novamente serão usadas ferramentas de monitoração. Alguns tempos de uso de recursos são medidos diretamente com a ajuda de monitores adequados, porém outros tempos de uso de recursos devem ser calculados a partir de dados medidos no sistema computacional.

Tempo de Uso de CPU

Para medir o tempo de uso da CPU foi utilizado o sistema de *accounting* do próprio AIX. Este monitor foi adotado por ele ser capaz de apresentar o tempo de uso de CPU por processo. Enquanto ativado, o sistema de *accounting* armazena num arquivo específico, e de forma compactada, todas as informações sobre todos os processos que terminaram. Resultados parciais de processos que terminaram após o período de medição não são registrados, enquanto que informações sobre processos que começaram antes do período de medição, mas que acabaram antes do término da medição, são registrados. Esse comportamento é devido ao fato do sistema de *accounting* armazenar em disco os dados estatísticos armazenados dentro dos próprios processos, sempre que algum processo termina.

Assim, usando o sistema de *accounting* do AIX foi possível se medir o tempo de uso de CPU para todos os serviços. Entretanto, um tratamento especial foi dado a essas medições devido ao fato do tempo de uso de CPU poder variar para duas medições aparentemente idênticas. Para contornar este problema foi adotada a seguinte estratégia:

- Foi medido o tempo de uso de CPU dos processos sem executarem nenhum serviço, ou seja, foram medidos os tempos de uso de CPU dos processos Natural, do SGBD ADABAS, do *telnet* e do *ftp* para serem ativados e imediatamente desativados.
- Foram medidos os tempos de uso de CPU para cada processo ser ativado, executar um serviço dez vezes e ser desativado.
- Subtraindo-se a primeira medição da segunda e dividindo o resultado por dez obtém-se o tempo de uso de CPU para cada processo executar um serviço uma única vez.

Tempo de Uso do Subsistema de E/S

O processo de medir o tempo de uso do subsistema de E/S não é um processo tão direto quanto medir o tempo de uso de uma CPU. O dado a ser medido é o número de operações de E/S, já que os demais parâmetros componentes do tempo de uso do subsistema de E/S são constantes e determinados pelas características físicas dos discos, conforme a Equação 4.1.

$$\text{Tempo de Uso de Disco} = \text{Operações} \times (\text{Latência} + \text{seek} + \text{Transferência}) \quad (4.1)$$

Conforme descrito na Seção 2.4.3, o servidor de banco de dados de Montes Claros possui apenas um disco de 9.1 Gbytes. Assim, segundo o fabricante, este disco opera a 7.200 rpm, possui um tempo de *seek* médio de 8,5ms, e a taxa de transferência varia entre 10,2 e 15,4 Mbytes/segundo. Da velocidade de rotação derivamos um tempo de latência calculando o valor da metade da frequência de rotação, de onde obtemos 4,17 ms. Como a maioria das operações de E/S será realizada pelo SGBD, e este está configurado para trabalhar com blocos de 2 Kbytes, podemos calcular o tempo de transferência para uma operação de E/S como sendo o tempo necessário para transferir 2 Kbytes de dados, o que resulta num tempo de transferência entre 0,13 e 0,19 ms.

Considerando a questão de medir o número de operações de E/S surge um problema: o SGBD ADABAS armazena seus dados em *raw devices* para manter sob seu controle o processo de gravação dos dados em disco, ou seja, fora dos *file systems* do AIX. Como os monitores de E/S do AIX somente monitoram operações de E/S em *file systems*, logo eles não são úteis nessa situação. Assim, para monitorar o número de operações de E/S em *raw devices* sob o controle do SGBD foi utilizado o *log* de comandos do SGBD ADABAS. Neste caso, para cada operação solicitada ao SGBD, é gravado um arquivo com a relação de todas as operações de E/S realizadas. Com a ajuda do utilitário ADACLIP [ADA95b] do ADABAS foi gerado um relatório, conforme mostra a Figura 4.1.

```

%ADACLPI-STARTED,          3-MAY-1998 16:03:33, Version 2.2.1.8, (AIX)
IO ( 1) : 7436/RD
IO ( 1) : 694/RA
IO ( 3) : 14563/RA, 6418/RA, 4023/RD
IO ( 1) : 14511/RD
IO ( 1) : 19369/RA
IO ( 5) : 160/RA, 802/RA, 6599/RA, 6601/RA, 6549/RA
IO ( 2) : 12421/RA, 3996/RD
IO ( 4) : 4126/RA, 4070/RA, 2160/RD, 2161/RD
IO ( 8) : 1949/RD, 10065/RA, 4750/RA, 1993/RD, 1945/RD, 2034/RD,
        1995/RD, 4751/RA
IO ( 6) : 1935/RD, 4745/RA, 1973/RD, 1974/RD, 10495/RA, 2019/RD
IO (10) : 2068/RD, 4746/RA, 2050/RD, 1933/RD, 10149/RA, 3933/RD,
        2066/RD, 4752/RA, 2004/RD, 1932/RD

```

Figura 4.1: Relatório do utilitário ADACLPI que mostra para cada operação de banco de dados requisitada ao SGDB ADABAS quantas operações de E/S foram realizadas. Assim o valor 7436/RD significa uma operação de leitura (R) na área de dados (D) no bloco de número 7436.

Uma questão relacionada com a medição do número de operações de E/S realizadas por uma operação do SGBD é o uso do *buffer pool*, pois o SGBD não funciona sem este recurso, o que altera consideravelmente o número de operações de E/S realizadas. Cada vez que um item de dados é lido do subsistema de E/S ele permanece no *buffer pool* que se encontra em memória, e toda a vez que este mesmo item de dados é requisitado ele é inicialmente procurado no *buffer pool* diretamente. Desta forma, para cada item de dados utilizado que se encontre no *buffer pool* o SGBD realiza uma ou mais operações de E/S a menos. Como o modelo de simulação considera a existência do *buffer pool*, então neste momento devemos tentar medir o número real de operações de E/S que seriam realizadas caso não existisse o *buffer pool*. Para minimizar o efeito do uso do *buffer pool*, foi adotada a estratégia de se desativar e reativar o SGBD a cada medição do número de operações de E/S realizadas. Desta forma, o *buffer pool* é destruído e reconstruído sem dados.

Os serviços de *ftp* e de impressão realizam operações de E/S em *files system* do AIX. Assim, o utilitário *iostat* do próprio AIX é capaz de medir tais operações de E/S. Sua utilização é simples, bastando executar o comando *iostat* antes e depois de se realizar o serviço. A cada execução o *iostat* informa o total de operações de entrada e saída que foram realizadas. Para se obter o total de operações resta subtrair os valores obtidos na segunda execução dos valores obtidos na primeira execução. O problema do uso do *iostat* está no fato dele contabilizar todas as operações de E/S realizadas no *files systems* do AIX, e não somente as operações realizadas por um processo em especial. Mas como o ambiente de medição estava totalmente isolado e sob controle, não tivemos de tomar nenhum cuidado em especial nesse sentido.

Tempo de Uso de Canais de Comunicação

A medição do uso do canal de comunicação local é semelhante ao processo de medição do número de operações de E/S utilizando o `iostat`, pois o utilitário `entstat` deve ser executado antes e depois da execução de um serviço, também com a preocupação de trabalhar num ambiente isolado para não alterar o valor real. No caso, a rede utilizada para medir o uso do canal de comunicação local estava isolada do resto da rede de computadores da COPASA-MG. O `entstat` fornece o número de pacotes e *bytes* enviados e recebidos. O número de pacotes enviados e recebidos e o tamanho médio dos pacotes em *bytes* são parâmetros do modelo de simulação. Logo, como também sabemos quantos *bytes* foram enviados e recebidos, então podemos saber qual o tamanho médio dos pacotes.

A medição do uso do canal de comunicação remoto é mais complexa do que a do canal local devido ao fato da medição ter de ser comandada pela Telemig, empresa responsável pelo serviço MINASPAC e ao fato de somente ser possível obter dados estatísticos desses *links* de 15 em 15 minutos. Mais uma vez surge o problema de se obter dados estatísticos de recursos compartilhados por diversos aplicativos. Assim, para obtermos dados isentos de interferências externas foi necessário realizar essas medições em horários em que esses *links* não são utilizados.

Número de chamadas ao SGBD por serviço

Para obtermos o número de chamadas ao SGBD recorreremos ao Natural que possui uma opção de depuração denominada de NAT_TRACE. O funcionamento do NAT_TRACE é simples. O NAT_TRACE é uma variável de ambiente que quando definida deve possuir valores entre 1 e 6. Quando esta variável de ambiente possui o valor 6 todas as operações solicitadas ao SGBD são gravadas num arquivo texto conforme mostrada na Figura 4.2.

Trabalhando com o NAT_TRACE, percebemos que toda vez que qualquer programa Natural é executado ele realiza uma seqüência de comandos para estabelecer e finalizar a comunicação com o SGBD ADABAS. Assim, para se medir o número de solicitações foram executados os programas e subtraídos os comandos de estabelecimento e finalização de comunicação entre o Natural e o SGBD ADABAS.

4.3 Modelagem da Carga de Trabalho

4.3.1 Intensidade da Carga de Trabalho

De acordo com o esquema de mapeamento apresentado na Seção 3.4, vemos que é possível definir a intensidade da carga de trabalho *on-line* do sistema alvo a partir do número de serviços executados no sistema legado, pois estamos supondo que todo serviço realizado no sistema legado também tenha de ser executado de alguma forma no sistema


```

*****
*   NATURAL trace file for ADABAS / ENTIRE DB calls
*
*   Version : V 2.2.1 Pl 5 Software AG 1995
*   DATE    : Sep  2 1995
*   Level   : 6 (6)
*   User    : PP
*   Date    : 19/03/1998 14:22:07
*****

      Count Cmd DBID FNR   Rsp     ISN Quantity   Cmd ID Opt  FB  RB  SB  VB  IB
A       0 OP  156   0   0         0         1 00000000      0   4
A       1 S1  156  82   0    48025         1 00002002     40 220   7   8   4
A       2 S1  156  82   0    48025         1 00002006     40 220   7   8   4
A       3 RC  156   0   0         0         0 00002002   F
A       4 RC  156   0   0         0         0 00002006   F
A       5 S1  156  82   0    48025         1 00003002     24  10   7   8   4
A       6 S1  156  82   0     1212         1 00004002     36 607   7   7   4
A       7 RC  156   0   0         0         0 00004002   F
A       8 S1  156  82   0     1215         1 00004002     36 607   7   7   4

```

Figura 4.2: Arquivo texto contendo a descrição das operações de banco de dados requisitadas ao SGBD ADABAS por um programa Natural.

alvo. Para verificarmos a validade de tal suposição basta imaginarmos o caso de um consumidor que procura a COPASA-MG para saber se ele possui algum débito pendente. A empresa, ou mais especificamente, o sistema comercial – seja legado ou alvo – terá de prestar tal informação ao consumidor. Se formos analisar os serviços identificados e descritos na Seção 3.5.3, verificamos da mesma forma como descrito acima que essa suposição se aplica a todos os casos.

Vemos que a intensidade da carga de trabalho *on-line* do sistema alvo pode ser dada em função da frequência de uso dos serviços *on-line* do sistema legado, que foi determinada na Seção 3.5. Porém, essa frequência de uso de serviços *on-line* foi derivada de medições realizadas em fevereiro de 1998, embora a migração para o SICOM/D esteja planejada para o segundo semestre de 1998, ou primeiro semestre de 1999. Assim, devemos tratar os valores referentes à frequência de uso de serviços para que a carga a ser submetida ao modelo de simulação espelhe a situação a ser encontrada quando da migração do SICOM/D.

Para projetar valores estamos utilizando como índice de crescimento de solicitações de serviços a taxa de crescimento do número de ligações de água e esgoto, visto que ligação de água ou esgoto é a unidade de negócio da área operacional da COPASA-MG.

De acordo com a Coordenadoria de Informações Comerciais da COPASA-MG existiam 61.713 ligações de água e 54.571 de esgoto em Montes Claros em fevereiro de 1998, sendo que a taxa de crescimento dessas ligações durante o ano de 1997 foi de 4,68%. Em Janaúria existiam 8.153 ligações de água e 922 de esgoto, com uma taxa de crescimento de 3,31% ao ano. Em Janaúba existiam 11.462 ligações de água e nenhuma de esgoto, com uma

taxa de crescimento de 5,97% ao ano. Aplicando essas taxas de crescimento aos valores da Tabela 4.2 projetamos a frequência de uso dos serviços *on-line* para fevereiro de 1999, ou seja, uma provável data para a migração do SICOM/C para o SICOM/D.

Antes de serem utilizados os valores projetados da Tabela 4.2, ainda foi feita mais uma modificação, pois foram incluídos de forma proporcional, e considerando a projeção anteriormente descrita, serviços do Módulo de Atendimento nas frequências de serviços executados em Janaúria e Janaúba. Isto se deve ao fato destes distritos atualmente não utilizarem a versão legada do Módulo de Atendimento denominada de SATI – Sistema de Atendimento Integrado. Porém como existe a intenção da COPASA-MG de informatizar o atendimento de todos os seus distritos, então é razoável supormos a presença desses serviços naqueles distritos. Logo, após todas estas considerações é esperado que o número de serviços executados no horário de pico da Gerência da Regional Norte em fevereiro de 1999 seja o descrito na Tabela 4.3.

Categoria	Frequência		
	Montes Claros	Januária	Janaúba
Gera SS	137	11	14
Consulta SS	72	6	13
Baixa OS	129	10	13
Recebimento OS	71	6	8
Cancela OS	88	0	0
Consulta Cadastro Consumidor	30	18	2
Altera Cadastro Consumidor	8	0	0
Pesquisa Consumidor	91	61	13
Consulta Débito Consumidor	31	3	12
Detalhe Débito Consumidor	20	3	13
Lançamento em Conta	11	51	6
FTP	0	0	1

Tabela 4.3: Número estimado de serviços *on-line* a serem executados na Gerência da Regional Norte em fevereiro de 1999 no horário de pico.

4.3.2 Uso de Recursos

Conhecida a intensidade da carga de trabalho *on-line* do SICOM devemos então conhecer como é o desempenho de uma transação *on-line* no sistema alvo. Neste ponto, vale ressaltar a necessidade de um estudo preliminar do funcionamento global do sistema alvo, pois conhecidas as suas principais características não é difícil identificar como um serviço é executado no sistema. Um estudo preliminar do sistema alvo [GdSL98] foi realizado, assim como foi consultado [Com97] no qual existe uma descrição dos programas do Módulo de Atendimento, assim como diagramas E-R dos quais foi derivado parte do esquema do banco de dados do SICOM.

Um serviço *on-line*, em nosso estudo de caso, sempre é executado por um conjunto de transações *on-line*. Portanto estamos desconsiderando a possibilidade de uma transação

on-line iniciar um *batch job* ou um serviço *ftp* com o objetivo de processar uma consulta *on-line*. Desta forma, um serviço *on-line* pode variar tanto no número de transações executadas, quanto no seu custo, em termos de utilização de recursos, das transações executadas.

Considerando as várias formas de se executar um serviço, foi feito um levantamento das alternativas mais comuns para cada um deles. Para exemplificar essa diversidade de formas de se executar um serviço podemos comparar o serviço *Alterar Cadastro Consumidor* com o serviço *Cancelar Solicitação de Serviço*.

Um atendente pode efetuar uma alteração cadastral não possuindo a matrícula do consumidor, o que implica em fornecer seu endereço. Independente da forma de identificação, o consumidor pode ainda solicitar a alteração de seu endereço de correspondência, ou simplesmente solicitar uma modificação em seu nome. Para alterar o endereço, pode ser necessário que o atendente consulte uma tela auxiliar para ajudar no preenchimento de informações tais como logradouros, bairro, etc. Enfim, a alteração de cadastro é um serviço que pode ser realizado de diversas formas, e com objetivos ligeiramente diferentes, o que resulta na execução de um número variável de transações, com custos igualmente variáveis. Por outro lado, para se cancelar uma solicitação de serviço, ou SS, basta o atendente fornecer o número da solicitação, o que resulta numa seqüência de transações *on-line* bem determinada e constante.

Neste estudo não estamos considerando as situações de falha que podem resultar no término prematuro e incompleto de um serviço. De fato, é esperado que um percentual insignificante de transações *on-line* sejam abortadas.

Como não são conhecidas as transações *on-line* do sistema alvo, mas apenas seus serviços *on-line*, surge a necessidade de se especificar um conjunto de transações resultantes da execução dos serviços *on-line* representativos da carga de trabalho *on-line* do SICOM, e que contemple as variações mais prováveis de cada serviço. A definição deste conjunto é necessária por não existir uma única transação que represente este serviço. Assim, chegamos ao seguinte conjunto de transações *on-line* do sistema alvo sobre as quais foram realizadas as medições dos parâmetros da classe *on-line*:

Dossiê

- **Doss-min** – Somente consulta o dossiê.
- **Doss-fat-med** – Consulta o dossiê, os dados de faturamento e as medições de consumo de água.
- **Doss-idmin** – Identifica consumidor no módulo de *Atendimento e Serviços*, e consulta dossiê.
- **Doss-idmax** – Identifica consumidor no módulo de *Atendimento e Serviços* consultando telas auxiliares para identificar logradouros e bairros, consultando em seguida o seu dossiê.
- **Doss-max** – Identifica consumidor com a ajuda de telas auxiliares para identificar logradouros e bairros no *Atendimento e Serviços*, e consulta dossiê, dados

de faturamento e medições de consumo de água.

Gera SS

- **GeraSS-min** – Gera uma SS de ligação de água sem consulta a telas auxiliares e com fatura mensal.
- **GeraSS-med** – Gera uma SS de ligação de água com consulta às telas auxiliares.
- **GeraSS-al-med** – Gera uma SS de análise de água com consulta às telas auxiliares.
- **GeraSS-vp-med** – Gera uma SS de vistoria de padrão com consulta às telas auxiliares.
- **GeraSS-idend-max** – Gera SS com identificação através de endereço e com consulta às telas auxiliares.

Consulta SS

- **Cons-SS-and-min** – Consulta clientes com somente uma SS em andamento.
- **Cons-SS-and-max** – Consulta clientes com mais de uma SS em andamento.
- **Cons-SS-and-idos** – Consulta SS através de seu número.
- **Cons-SS-det-min** – Consulta detalhes da SS de consumidores com somente uma SS em andamento.
- **Cons-SS-det-med** – Consulta detalhes de uma SS de consumidores com quatro SS em andamento.
- **Cons-SS-det-max** – Consulta detalhes das SS de consumidores com mais de uma SS em andamento.
- **Cons-SS-det-idos** – Consulta detalhes da SS acessando-a através do seu número.

Cancela SS

- **Cancela-SS** – Cancela uma SS.

Altera cadastro

- **cad-alt-pessoais** – Altera dados pessoais do consumidor.
- **cad-alt-endcor-min** – Altera endereço de correspondência.
- **cad-alt-endcor-max** – Altera endereço de correspondência com consulta às telas auxiliares.
- **cad-alt-endimo-min** – Altera endereço do imóvel.
- **cad-alt-endimo-max** – Altera endereço do imóvel com consulta às telas auxiliares.
- **cad-alt-situ-ag-min** – Altera situação água/esgoto.
- **cad-alt-situ-ag-max** – Altera situação água/esgoto com consulta às telas auxiliares.

- **cad-alt-ps-ag-min** – Altera ponto de serviço de água.
- **cad-alt-ps-ag-max** – Altera ponto de serviço de água com consulta às telas auxiliares.

Baixa solicitação serviço

- **BaixaSS-min** – Baixa solicitação de serviço a partir do número da solicitação.
- **BaixaSS-max** – Baixa solicitação de serviço com consulta às telas auxiliares.

Verifica débitos de consumidor

- **Verifica-deb** – Consulta os débitos de um consumidor.

Consulta débito de consumidor

- **Consulta-deb-min** – Consulta consumidores com somente um débito.
- **Consulta-deb-max** – Consulta consumidores com vários débitos.

Seleciona serviços

- **Sele-SS-um** – Seleciona um serviço.
- **Sele-SS-todos** – Seleciona todos os serviços.

Consulta dados de consumidor

- **Consulta-cons-pess** – Consulta dados pessoais de um consumidor.
- **Consulta-cons-endcor** – Consulta endereço de correspondência de um consumidor.

Considerando a necessidade de se determinar um conjunto único de parâmetros por serviço *on-line*, as medições dos tempos de uso dos recursos das diversas transações componentes de um serviço *on-line* foram agrupadas. Mas tanto existem serviços compostos de poucas transações, assim como existem serviços compostos de várias transações e com valores bem diferentes. Segundo [Jai91], deve ser utilizada a média aritmética para calcular os parâmetros de utilização de recursos para os serviços compostos de transações homogêneas, enquanto deve ser utilizada a mediana para representar os parâmetros de utilização dos serviços compostos de transações heterogêneas.

O modelo de carga final é composto do conjunto de serviços executados segundo o número de vezes especificado na Tabela 4.3 num período de 1 hora, com os parâmetros dos serviços *on-line* especificados conforme as Tabelas 4.4 e 4.5, e dos serviços *batch* e *ftp* conforme especificados nas Tabelas 4.7 e 4.6, respectivamente.

Serviços	Trans.	Número Chamadas ao SGBD	Operações E/S SGBD	Demanda de Processador	
				Natural + telnet (ms)	SGBD (ms)
Gera-SS	14	115	262	1470	600
Consulta-SS	8	38	89	680	230
Baixa-OS	15	86	153	1490	360
Recebimento-OS	5	47	61	1090	330
Cancela-OS	9	20	68	590	260
Consulta-Cadastro	9	34	9	390	240
Altera-cadastro	9	33	61	620	260
Pesquisa-Consumidor	7	76	185	1270	400
Verifica-Débito	7	6	39	590	190
Consulta-Débito	8	15	43	440	240

Tabela 4.4: Parâmetros de serviços *on-line* relativos ao servidor de banco de dados.

Serviços	Ethernet		X.25		
	Qtde. pacotes	Tam. médio pacotes (bytes)	Qtde. quadros	Tam. médio quadros entrada (bytes)	Tam. médio quadros saída (bytes)
Gera-SS	222	483.24	4290	7,06	42,44
Consulta-SS	82	481.95	1857	6,89	43,60
Baixa-OS	197	480.41	3231	10,29	41,96
Recebimento-OS	90	475.95	1589	7,29	43,09
Cancela-OS	103	480.41	2259	7,29	42,29
Consulta-Cadastro	54	657.78	1336	7,32	42,50
Altera-cadastro	108	486.07	1692	7,89	41,41
Pesquisa-Consumidor	172	480.93	1755	6,81	44,44
Verifica-Débito	76	489.00	1209	7,80	49,23
Consulta-Débito	79	475.95	1549	7,63	42,54

Tabela 4.5: Parâmetros de serviços *on-line* relativos à rede de computadores da GRNT.

Parâmetro	Valor
Tempo de uso de CPU (ms)	256
Número de operações de E/S	1320
Número de pacotes <i>ethernet</i>	183
Tamanho pacotes <i>ethernet</i> (byte)	1500
Número quadros X.25	2286
Tamanho quadros X.25 (byte)	120

Tabela 4.6: Parâmetros de serviço *ftp* considerando-se a transferência de um arquivo de 268 Kbytes.

Parâmetro	Valor
Tempo de uso de CPU (ms)	13,75
Número de operações de E/S	14
Tempo impressão – laser (ms)	4653,3
Tempo impressão – impacto (ms)	37200,0

Tabela 4.7: Parâmetro de serviços *batch* de impressão considerando-se os recursos necessários para imprimir uma página.

Capítulo 5

Modelagem e Simulação do SICOM

5.1 O Simulador SES/workbench

Este trabalho usa técnicas de simulação para analisar como será o comportamento do SICOM. O produto usado para realizar estas simulações foi o SES/*workbench* da Scientific and Engineering Software Inc. [Sci96c] que é um simulador de eventos discretos. A escolha do SES/*workbench* foi devida principalmente a quatro de suas características:

- É um simulador de uso genérico, o que garante a flexibilidade necessária para modelar todos os aspectos de um sistema de computação complexo como o SICOM;
- Possui modelos básicos (*templates*) de diversos componentes de um sistema de computação (sistemas operacionais, sistemas de banco de dados, redes locais, redes remotas, etc) no sítio WWW da Scientific and Engineering Software Inc. que podem ser aproveitados no desenvolvimento do modelo de simulação do sistema alvo;
- Possibilita a criação de modelos de simulação com o auxílio de interfaces gráficas, o que significa a geração rápida de tais modelos sem a necessidade de programação;
- Provê mecanismos de depuração.

O SES/*workbench* [Sci96a] é composto de três módulos. O primeiro módulo é o SES/*design* através do qual são criados graficamente os modelos de simulação. O segundo módulo é o SES/*sim* responsável pela simulação propriamente dita, assim como pela tradução dos modelos gerados graficamente no SES/*design* para a linguagem SES/*sim*. O terceiro módulo é o SES/*scope* responsável pelos mecanismos de depuração dos modelos de simulação. De dentro do ambiente de trabalho do SES/*design* é possível se executar o SES/*sim* e o SES/*scope* como se fossem um único módulo.

A linguagem SES/*sim* é uma extensão das linguagens C e C++, o que facilita o uso de rotinas C e C++ para descrever mais detalhadamente os componentes do modelo.

Neste capítulo serão apresentados os modelos que foram utilizados na simulação do SICOM/D e do SICOM/CS. Para isto será apresentado o funcionamento do ambiente gráfico do SES/*design*, assim como dos componentes visuais usados na construção dos modelos.

O SES/*workbench* foi desenvolvido baseado na idéia de representar os sistemas de computação através de grafos. Nesses grafos os nós são conectados dois a dois por arcos topológicos ou simplesmente arcos. Os nós representam os componentes do sistema e os arcos representam os relacionamentos existentes entre os componentes. Adicionalmente, as transações¹ representam os dados sendo processados pelo sistema de computação. Desta forma, num modelo de simulação podem existir diversas transações sendo executadas simultaneamente, sendo possível que múltiplas transações permaneçam num único nó.

Para manipular esses grafos, o SES/*workbench* implementa uma estrutura hierárquica de ambientes gráficos. A tela principal apresenta o catálogo do diretório no qual o SES/*workbench* está sendo executado. Não existindo nenhum arquivo de catálogo do SES/*workbench* no diretório corrente, o SES/*workbench* solicita que seja criado um, ou que seja informado o nome e o caminho de um catálogo existente. A estrutura de catálogos do SES/*workbench* é equivalente à estrutura de diretórios do UNIX: num diretório do UNIX podem existir inúmeros arquivos, bem como outros diretórios, assim como num catálogo do SES/*workbench* podem existir inúmeros modelos, bem como outros catálogos de modelos.

Uma vez selecionado o ícone de um modelo no catálogo do SES/*workbench*, este exhibe o grafo do modelo, composto de um ou mais submodelos. Um submodelo pode referenciar outros submodelos, e submodelos sempre são visualizados tanto no modelo quanto nos seus submodelos. Assim, não existe razão para que um submodelo possua um sub-submodelo.

Como foi dito, um modelo é composto de arcos e nós, sendo estes os componentes dos submodelos (o próprio submodelo é um nó). Existe ainda um terceiro tipo de componente num modelo SES/*workbench* chamado de **Resource**. Este componente é usado para representar recursos passivos tais como memória, itens de dados num SGBD e páginas de uma memória virtual, por exemplo. Nós do tipo **Resource** podem ser declarados no modelo. Os demais nós, apresentados a seguir, são declarados nos submodelos, pois arcos somente são criados dentro de submodelos.

Além de conhecer a estrutura hierárquica de catálogos, modelos e submodelos apresentada acima, é necessário se conhecer as funções dos tipos de nós usados no modelo para se compreender os modelos de simulação do SICOM/D e do SICOM/CS.

Os nós usados nos modelos de simulação deste trabalho estão representados graficamente na Figura 5.1 e possuem as seguintes funções:

- **Service** – Este tipo de nó representa os recursos ativos do sistema de computação, tais como processadores e discos. A cada nó deste tipo está associada uma fila infinita

¹O termo transação neste capítulo se refere a uma unidade de processamento dentro do modelo de simulação. O termo *transação on-line* usado nos primeiros capítulos deste trabalho e definido na Seção 3.2 continuará a ser empregado sempre com o adjetivo “on-line”, o que o diferencia do termo anterior.

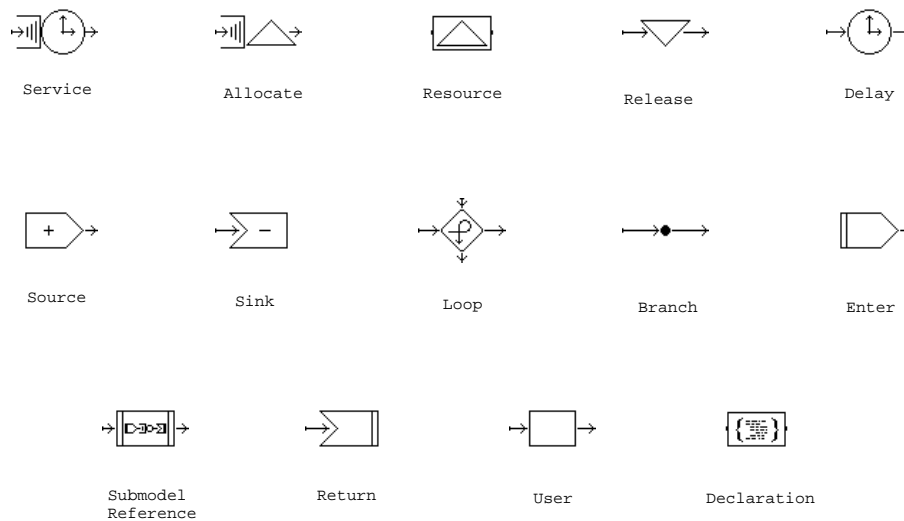


Figura 5.1: Nós usados no modelo de simulação do SICOM.

na qual as transações aguardam pelo serviço. Enquanto atendidas, as transações ficam residentes no recurso pelo tempo de serviço especificado.

- **Allocate** – Ao passar por este tipo de nó, uma transação reserva para si uma quantidade específica de um dado recurso. A este tipo de nó também está associada uma fila infinita, pois pode ser que não existam os recursos necessários para que a transação continue processando, o que faz com que ela aguarde pelos recursos.
- **Resource** – Declara um conjunto (*pool*) de recursos passivos cujas quantidade e tipo devem ser especificados, ou seja, se um recurso é endereçável como um item de dados num arquivo ou não-endereçável como uma ficha numa rede *token ring*. Os recursos serão reconhecidos em quaisquer submodelos estando o nó declarado no modelo. Caso contrário, serão enxergados somente no escopo do submodelo.
- **Release** – As transações devolvem para o *pool* de recursos passivos os recursos que estavam em uso pela transação correspondente.
- **Delay** – A transação fica retida neste tipo de nó pelo tempo de espera nele especificado.
- **Source** – As transações nascem neste tipo de nó. Neles são especificados o tempo entre cada transação submetida ao modelo. Adicionalmente, ainda neste tipo de nó são especificadas a categoria e a quantidade das novas transações.
- **Sink** – Tipo de nó onde as transações morrem.
- **Loop** – Este nó serve para auxiliar no controle do fluxo de uma transação. Uma vez que a transação tenha entrado no *loop*, ela irá permanecer lá por um número

determinado de vezes ou até que uma condição seja atingida.

- **Branch** – Tipo de nó geralmente usado para organizar graficamente os arcos. Inúmeros arcos podem ser agregados num único arco com a ajuda de um nó deste tipo.
- **Enter** – Ponto onde uma transação entra no submodelo. Desta forma, um submodelo não pode possuir mais do que um nó deste tipo.
- **Submodel Reference** – Ao chegar a um nó deste tipo, a transação é redirecionada para o submodelo referenciado, entrando pelo nó **Enter** e retornando quando chegar a um nó do tipo **Return** do submodelo referenciado.
- **Return** – Uma transação sai de um submodelo quando chega a este tipo de nó, retornando para onde o submodelo em uso foi chamado.
- **User** – Sendo o *SES/sim* uma extensão da linguagem C, este tipo de nó possibilita que seja inserida uma rotina C para executar funções não contempladas pelos demais nós.
- **Declaration** – Tipo de nó onde são declaradas as variáveis, as constantes e os parâmetros usados na simulação, bem como as categorias de transações existentes no modelo. Estando este nó no modelo as variáveis serão consideradas como variáveis globais. Estando este nó num submodelo as variáveis declaradas serão consideradas como variáveis locais, não sendo enxergadas em outros submodelos.

Conhecida a estrutura hierárquica do *SES/design* e as funções de seus principais componentes, se torna possível descrever o modelo de simulação do *SICOM/D*. Os detalhes específicos do *SES/workbench* necessários para se compreender um modelo de simulação serão fornecidos durante a descrição do modelo.

5.2 O Modelo de Simulação do *SICOM/D*

5.2.1 Visão Geral

O modelo de simulação do *SICOM/D* é composto de dezessete submodelos e um nó do tipo **Declaration**, conforme mostra a Figura 5.2. Os submodelos no *SICOM/D* representam os componentes do sistema de computação. Este sistema foi modelado a partir da carga de trabalho, que é o componente mais abstrato do modelo, até chegar em seus recursos primários, que neste trabalho são o processador (subsistema de CPU) e o disco (subsistema de E/S).

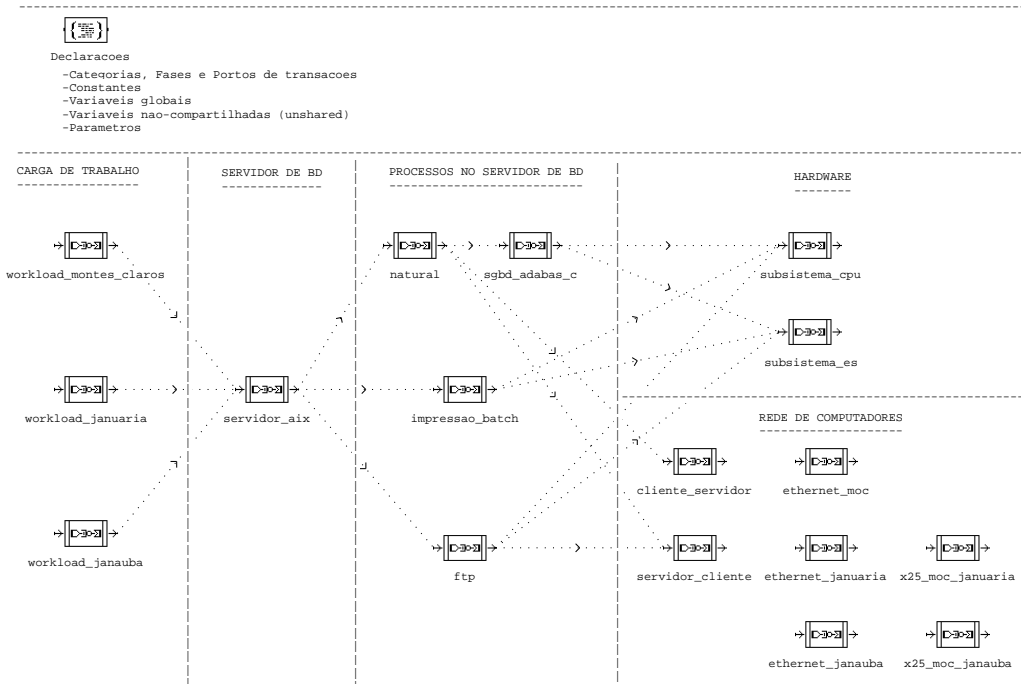


Figura 5.2: O modelo de simulação do SICOM/D.

Os Submodelos

A carga de trabalho foi modelada por distrito/gerência regional, existindo assim três submodelos que representam, respectivamente, os serviços submetidos a partir de Montes Claros, denominado de `workload_montes_claros`, os serviços submetidos a partir de Janaúria, denominado `workload_januaria`, e os os serviços submetidos a partir de Janaúba, denominado `workload_janauba`. Os serviços representados pelos submodelos de carga de trabalho são enviados ao servidor de banco de dados localizado em Montes Claros, representado pelo submodelo `servidor_aix`. Uma vez no servidor de banco de dados, esses serviços gerarão uma carga de trabalho distribuída pelos processos existentes no servidor de banco de dados, que são o Natural e o telnet representados pelo submodelo `natural`, o SGBD ADABAS representado por `sgbd_adabas_c`, o serviço `ftp` do TCP/IP representado por `ftp` e o escalonador (*scheduler*) de impressão *batch* representado por `impressao_batch`. Como esses processos estão no servidor de banco de dados, eles usam processador e discos do servidor, motivo pelo qual existem respectivamente outros dois submodelos que representam estes recursos, que são o `subsistema_cpu` e o `subsistema_es`. O uso de submodelos para representar os subsistemas de CPU e E/S é extremamente vantajoso em trabalhos deste tipo, pois permite que análises do impacto de modificações complexas, como a inclusão de um segundo processador, sejam facilmente realizadas modificando-se um único submodelo.

Os serviços *on-line* e `ftp` executados no modelo de simulação do SICOM/D usam a rede de computadores. Desta forma, para representar a rede de computadores existem três

modelos de rede *ethernet* e dois modelos de rede X.25, representando, respectivamente, as redes locais de Januária, Janaúba e Montes Claros, e os *links* remotos entre Montes Claros e Januária, e entre Montes Claros e Janaúba.

Na Figura 5.2 existem alguns arcos orientados que conectam os submodelos. Esses arcos são *arcos de dependência* que informam para o compilador do SES/*sim* as dependências entre os submodelos.

O Nó Declarações do Modelo

Além dos submodelos, o modelo do SICOM possui ainda um nó denominado de *Declarações* no qual estão descritas características importantes do modelo de simulação. Todo o modelo da carga de trabalho do SICOM, as constantes, as variáveis globais, as variáveis não-compartilhadas (*unshared*) que são específicas de cada transação, além dos atributos primários de uma transação do SES/*workbench* estão declarados neste nó. A seguir, são descritas as características contidas no nó *Declaracoes*.

```

category const gera_ss;
category const consulta_ss;
category const baixa_os;
category const recebimento_os;
category const cancela_os;
category const consulta_cadastro;
category const altera_cadastro;
category const pesquisa_consumidor;
category const verifica_debito;
category const consulta_debito;
category const transferencia_ftp;
category const impressao_laser;
category const impressao_impacto;
phase const buffered;
phase const notbuffered;
port const montes_claros;
port const januarua;
port const janauba;

```

Figura 5.3: Parâmetros característicos das transações no SES/*workbench*.

O SES/*sim* classifica as transações em função de três atributos básicos: sua *categoria*, sua *fase* e seu *porto*. A Figura 5.3 mostra como estão declarados os valores possíveis para os atributos básicos das transações no modelo do SICOM. O uso desses atributos é facultado ao projetista do modelo de simulação. No caso do modelo do SICOM, toda transação possui uma *categoria* que é uma das 13 definidas na Figura 5.3. As transações

geradas a partir dos submodelos de carga de trabalho usam o atributo `porto` para fazer o seu roteamento pelas redes locais e os *links* remotos existentes. O atributo `fase` é usado no submodelo do SGBD ADABAS e seu uso está descrito na Seção 5.2.5.

```

double tempo_entre_gera_ss_moc      = 26277.0;
double tempo_entre_gera_ss_ja      = 327272.0;
double tempo_entre_gera_ss_jb      = 257142.0;
int    adabas_calls_gera_ss        = 115;
int    adabas_es_gera_ss           = 262;
int    transacoes_gera_ss          = 8;
double nattel_consumo_cpu_gera_ss  = 1470.0;
double adabas_consumo_cpu_gera_ss  = 600.0;
int    pacotes_eth_gera_ss         = 222;
double tam_pacote_eth_gera_ss      = 483.24;
int    pacotes_x25_gera_ss         = 0;
double tam_pacote_x25_gera_ss      = 0.0;

```

Figura 5.4: Parâmetros do modelo da carga de trabalho no modelo de simulação para serviços *on-line*.

```

double tempo_entre_impressao_impacto = 120000.0;
double impacto_consumo_cpu           = 10.0;
double impacto_consumo_es            = 100.0;
double impacto_tempo_impressao       = 210000.0;

```

Figura 5.5: Parâmetros do modelo da carga de trabalho no modelo de simulação para serviços *batch*.

```

double tempo_entre_transferencia_ftp = 200000.0;
double ftp_prob_es                   = 0.80;
double ftp_consumo_cpu               = 0.5;
double ftp_consumo_es                = 18.0;
int    pacotes_eth_ftp               = 0;
double tam_pacote_eth_ftp            = 0;
int    pacotes_x25_ftp               = 0;
double tam_pacote_x25_ftp            = 0;

```

Figura 5.6: Parâmetros do modelo da carga de trabalho no modelo de simulação para serviços *ftp*.

Os parâmetros da carga de trabalho estão agrupados por categoria de transações. Existem parâmetros específicos para cada categoria, dependendo se a transação se refere a um

serviço *on-line*, como mostra a Figura 5.4, um serviço *batch*, como mostra a Figura 5.5 ou um serviço *ftp* como mostra a Figura 5.6. O uso de cada parâmetro da carga de trabalho será naturalmente entendido ao serem descritos os submodelos componentes do modelo de simulação do SICOM/D nas seções seguintes.

```

unshared int    adabas_calls;
unshared int    adabas_es_fisico;
unshared double adabas_consumo_cpu;
unshared double adabas_consumo_es;
unshared int    adabas_lock_registro;
unshared double nattel_consumo_cpu;
unshared int    transacoes_servico;
unshared double consumo_cpu;
unshared double consumo_es;
unshared int    pacotes_ethernet;
unshared double tam_pacote_ethernet;
unshared int    pacotes_x25;
unshared double tam_pacote_x25;

```

Figura 5.7: Variáveis locais de cada instância de transação.

Uma característica importante do SES/*sim* é a capacidade de cada instância de transação possuir um conjunto específico de variáveis, denominadas de variáveis não-compartilhadas. Assim, uma transação leva consigo todos os dados necessários para o seu processamento, tais como os parâmetros que especificam o tempo de uso de recursos ativos. A Figura 5.7 mostra as variáveis não-compartilhadas das transações do modelo do SICOM. Assim, cada nova transação gerada num nó do tipo `Source` possui um conjunto dessas variáveis não compartilhadas às quais são atribuídos os parâmetros de carga. Para exemplificar, ao se gerar uma transação da categoria `gera_ss` a variável não-compartilhada `adabas_calls`, mostrada Figura 5.7, recebe o valor do parâmetro de carga de trabalho `adabas_calls_gera_ss`, mostrada na Figura 5.4. De forma análoga, os valores dos demais parâmetros de carga são atribuídos às suas respectivas variáveis não-compartilhadas.

Finalmente, dentro do nó `Declaracoes` existem ainda constantes, variáveis e parâmetros usados durante a execução do modelo de simulação, sendo a sintaxe de declaração dessas constantes e variáveis igual à sintaxe da linguagem C. Assim, na Figura 5.8, se vê como essas declarações são usadas no modelo de simulação do SICOM. As constantes `disco_min` e `disco_max` são as taxas de transferência mínima e máxima do disco do servidor de banco de dados, enquanto a `taxa_ethernet` e a `taxa_x25` são respectivamente as taxas de transmissão de pacotes *ethernet* e *X.25* das redes locais e dos *links* remotos. As variáveis `calls_medio` e `es_medio` são variáveis globais enxergadas em todo o submodelo. Neste caso, essas variáveis são usadas para calcular o número de chamadas que um programa Natural faz ao submodelo do SGBD ADABAS e quantas operações de E/S serão realiza-

```

#define disco_min      13.87
#define disco_max      14.57
#define taxa_ethernet  10000
#define taxa_X25       64

float calls_medio;
float es_medio;

parameter double adabas_eficiencia_buffer_pool = 0.875;
parameter double cpu_power = 1.0;

```

Figura 5.8: Constantes, variáveis e parâmetros do modelo de simulação.

das por chamada ao SGBD ADABAS por uma transação *on-line*. O parâmetro `cpu_power` é solicitado toda vez que o modelo de simulação é executado e significa o quanto do processador estará disponível para a carga de trabalho modelada. O segundo parâmetro `adabas_eficiencia_buffer_pool` também é solicitado toda vez que o modelo é executado, sendo o seu uso descrito na Seção 5.2.5.

5.2.2 Os Submodelos da Carga de Trabalho

Como visto no Capítulo 4, a carga de trabalho do SICOM é gerada por serviços das categorias *on-line*, *ftp* e *batch*. Nos submodelos de carga de trabalho, entretanto, somente estão modelados os serviços *on-line* e *ftp*, pois os serviços *batch* são submetidos diretamente dentro do submodelo do servidor de banco de dados, conforme descrito na Seção 5.2.3.

Dentro de cada submodelo de carga de trabalho existe um conjunto de nós do tipo `source`, um para cada categoria de serviço. Em cada um desses nós `Source` está especificado o tempo entre cada geração de uma nova transação, assim como são atribuídas a categoria e o porto dessas transações. Após geradas, essas transações são enviadas ao submodelo do servidor de banco de dados no qual são efetivamente processadas. Ao final de seu processamento, as transações retornam ao submodelo de carga de trabalho original e morrem num nó do tipo `Sink`.

5.2.3 O Submodelo do Servidor de Banco de Dados

Observando o submodelo do servidor de banco de dados mostrado Figura 5.9, se nota a existência de dois grafos distintos, um aberto e outro fechado.

O grafo aberto, na parte superior da figura, se inicia num nó do tipo `Enter` e termina num nó do tipo `Return`. Pelo nó `Enter` chegam os serviços *on-line* e *ftp* que passam pelo nó (do tipo `User`) `parametros_transação`, responsável pela execução de duas funções:

- atribuir os parâmetros da carga de trabalho da categoria do serviço para as variáveis

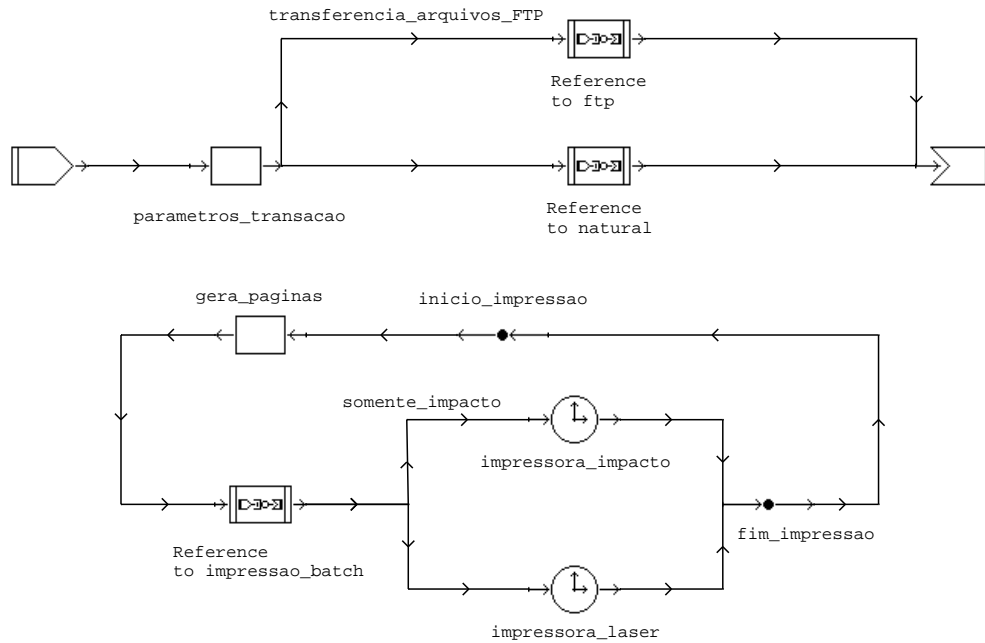


Figura 5.9: O submodelo do servidor de banco de dados.

não-compartilhadas das transações. Para explicar a necessidade deste nó é melhor exemplificar o seu uso. Todos os serviços usam o processador por um período de tempo definido pela variável não-compartilhada `consumo_cpu`. Assim, para que a transação *on-line* da categoria `gera_ss` (Figura 5.3) use o processador conforme especificado no modelo da carga de trabalho é feita a seguinte atribuição: `consumo_cpu = consumo_cpu_gera_ss`, ou seja o parâmetro do modelo da carga de trabalho é atribuído à variável não-compartilhada.

- atribuir nome às transações de acordo com o nome de sua categoria, o que facilita a depuração do modelo.

Ajustados os valores das variáveis não-compartilhadas dos serviços *on-line* e *ftp*, estes seguem por dois caminhos diferentes. Os serviços *ftp* são enviados para o submodelo `ftp`, enquanto os serviços *on-line* são enviados para o submodelo `natural`. Concluídos os processamentos dos serviços *on-line* e *ftp* nos seus respectivos submodelos, eles retornam ao submodelo do servidor de banco de dados para serem enviados aos seus respectivos submodelos de carga de trabalho.

No grafo fechado, por sua vez, ficam sendo executados continuamente três serviços *batch* referentes ao uso constante de três impressoras conectadas ao servidor de banco de dados. Dois destes serviços *batch* são da categoria `impressao_laser`, enquanto o outro é da categoria `impressao_impacto`. Independente da categoria do serviço *batch*, eles são criados no nó `inicio_impressao` seguindo imediatamente para o nó `gera_paginas` cuja função é gerar um número aleatório entre 50 e 100 que representa o número de páginas

que cada serviço *batch* terá de imprimir. Em seguida, os serviços *batch* passam pelo submodelo *impressao_batch*, que representa o uso do escalonador (*scheduler*) de impressão do sistema UNIX por estes serviços. Como qualquer outro processo, o escalonador de impressão usa o processador. Além disso, ele ainda usa o subsistema de E/S para ler e atualizar o *spool* de impressão que se encontra gravado em disco. Saindo do submodelo *impressao_batch*, a transação do serviço que é da categoria *impressao_impacto* segue pelo arco *somente_impacto*, enquanto as transações dos outros dois serviços seguem pelo arco que leva à *impressora_laser*. As impressoras são representadas por nós do tipo *Delay*, ou seja, é um recurso que sempre está disponível (existe um serviço *batch* para cada impressora). Cada serviço permanecerá nos nós *impressora_laser* e *impressora_impacto* por um tempo proporcional ao número de páginas que deverão ser impressas, retornando para o nó *inicio_impressao* e recomençando o ciclo de impressão.

5.2.4 O Submodelo dos Processos Natural

Este é o principal submodelo em termos de processamento de transações *on-line*, pois todas passam por ele. Em termos do modelo de simulação cada transação que chega ao submodelo *natural* representa um serviço *on-line* que vai ser executado através de um número bem definido de transações *on-line*. Quando uma transação passa pela primeira vez pelo *contador_transacoes*, conforme Figura 5.10, ela deixa de representar um serviço *on-line* e passar a representar uma transação *on-line*. Assim, a transação que representava um serviço *on-line* irá passar por *contador_transacoes* quantas vezes for o valor do parâmetro *transacoes_servico*, cada vez representando uma das transações *on-line* do serviço *on-line*. Para facilitar a descrição do modelo chamaremos à primeira transação *on-line* de um serviço de transação *on-line* 1, à segunda transação *on-line* de transação *on-line* 2, e assim para as demais transações *on-line*.

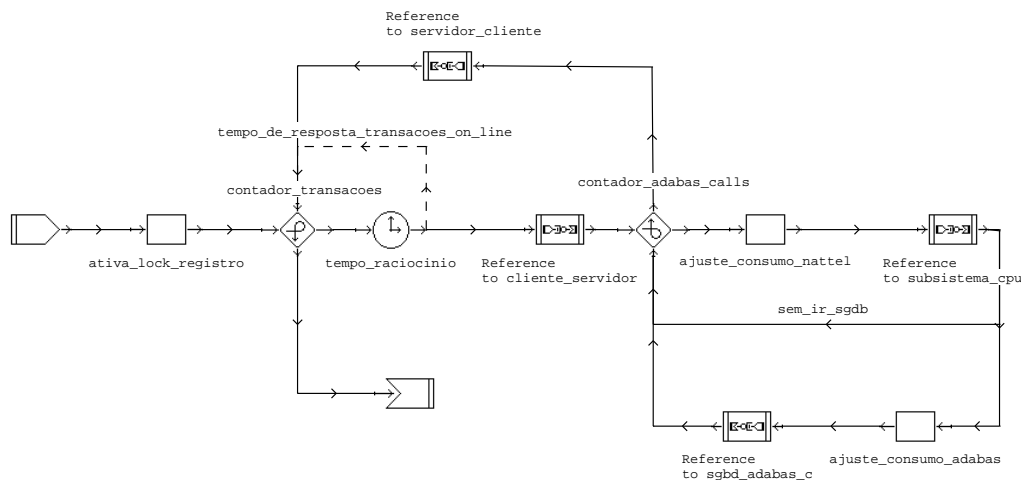


Figura 5.10: O submodelo dos processos Natural.

Como o usuário demora um certo tempo analisando os dados recebidos e decidindo as próximas ações, período de tempo este chamado de *tempo de raciocínio* [Shn92], foi incluído no modelo um nó do tipo `Delay` que retém as transações pelo tempo especificado no parâmetro `tempo_raciocinio`.

Uma vez que o usuário tenha submetido um comando para a aplicação, uma transação *on-line* é gerada. No modelo isto é representado pelas transações deixando o nó `tempo_raciocinio`. Como essas transações estão no computador cliente que tanto pode ser local quanto remoto ao servidor de banco de dados, os dados da transação devem ser transmitidos do computador cliente para o servidor, o que é modelado pelo nó `cliente_servidor`, explicado em detalhe na Seção 5.2.6. Chegando os dados ao servidor de banco de dados, as transações passam pelo `contador_adabas_calls`, quantas forem as operações de banco de dados realizadas pela transação *on-line*. Conforme verificado durante o processo de validação descrito na Seção 5.4.3, no caso do SICOM a maioria das operações de banco de dados são requisitadas pelas terceiras e quartas transações *on-line* dos serviços. Por isso, no modelo do SICOM as transações *on-line* que não são 3 ou 4 passam uma única vez pelo `contador_adabas_calls`. As transações *on-line* 3 e 4, por sua vez, passam (`adabas_calls / 2`) vezes pelo `contador_adabas_calls`, representando cada interação do programa Natural com o SGBD ADABAS.

As transações ao passarem pelo `contador_adabas_calls` são submetidas ao subsistema de CPU representando o tempo de uso do processador pelos processos Natural. Sendo a transação *on-line* 3 ou 4, esta será submetida ao submodelo `sgbd_adabas_c` descrito na próxima seção.

Concluído o processamento das transações *on-line* no servidor de banco de dados, elas são enviadas aos computadores cliente através da rede de computadores. Neste sentido a rede é representada pelo nó `servidor_cliente`. Caso a transação em questão represente a última transação *on-line* do serviço, ela sai do nó `contador_transacoes` já como um serviço *on-line* que é enviado ao nó `Return` do submodelo `natural`. Não sendo a transação a última transação *on-line* do serviço ela permanecerá no submodelo passando novamente pelo `tempo_raciocinio` e repetindo todo o ciclo de uma transação *on-line*.

5.2.5 O Submodelo do SGBD ADABAS

O submodelo do SGBD ADABAS é simples. Um dos fatores que mais contribui para isto é a baixa concorrência para atualização dos itens de dados. De fato, a maioria dos serviços *on-line* identificados são apenas para consulta. Nos serviços que atualizam o banco de dados não existe dependência entre os dados a serem atualizados durante a janela de tempo da simulação. Esta independência pode ser melhor compreendida com um exemplo: é praticamente nula a probabilidade de que os dados cadastrais de um consumidor específico sejam atualizados simultaneamente em mais do que um computador cliente.

Existe, entretanto, uma exceção a esta regra: toda SS gerada recebe um código que deve ser único na COPASA-MG. Para minimizar a concorrência por este número que é

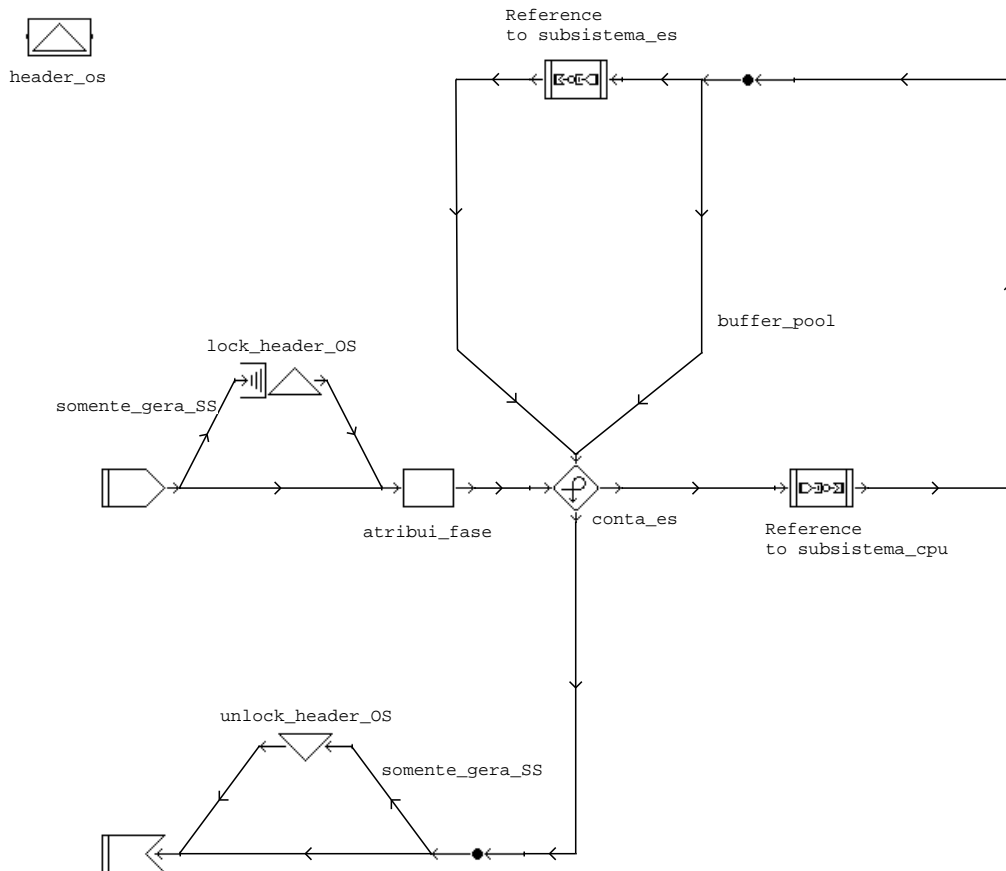


Figura 5.11: O submodelo do SGBD ADABAS.

armazenado no banco de dados a equipe de desenvolvimento do SICOM regionalizou seu uso. Cada gerência regional da empresa trabalha com uma faixa de valores para identificar suas SS, porém dentro da gerência continua existindo uma concorrência por tal número. Assim, no submodelo `sgbd_adabas_c` existe uma *ficha* identificada por `header_os` que possibilita à transação registrar o uso do próximo código de SS disponível, conforme mostra a Figura 5.11. Desta forma, toda terceira transação da categoria `gera_ss` inicialmente passa pelo `lock_header_os` para obter a *ficha* de atualização que será liberada ao sair do submodelo do SGBD quando passar pelo `unlock_header_os`. As demais transações simplesmente não passam por este nó indo direto para o nó `atribui_fase`.

No nó `atribui_fase` existe uma função que verifica se um número gerado aleatoriamente entre 0.0 e 1.0 é maior que o parâmetro `adabas_eficiencia_buffer_pool`. Se for, o atributo `fase` da transação passa a ser do tipo `buffered`, senão passa a ser do tipo `notbuffered`. Assim, cada vez que uma transação *on-line* passa pelo submodelo do SGBD ela está requisitando uma operação ao SGBD. Como os dados de uma operação podem estar no *buffer pool* do SGBD, pode ser que o SGBD não necessite acessar o subsistema de E/S para obter os dados requisitados. Logo, uma operação requisitada ao SGBD sempre irá usar o processador, porém nem sempre usará o subsistema de E/S.

Para executar uma operação de banco de dados o SGBD realiza uma certa quantidade de operações de E/S. O nó *conta_e/s* é o responsável por contar quantas operações de E/S uma operação de banco de dados irá realizar. Para cada operação de E/S, o SGBD usa o processador, o que é representado pela referência ao *subsistema_cpu*, assim como pode passar um tempo realizando uma operação de E/S, sendo o atributo *fase* da transação do tipo *notbuffered*. O tempo de uso do subsistema de E/S para realizar uma operação de E/S é representado neste submodelo pela referência ao submodelo *subsistema_es*.

5.2.6 Os Submodelos da Rede de Computadores

Na Gerência da Regional Norte existem três redes locais *ethernet*, uma para Montes Claros, uma para Janaúria e outra para Janaúba, além de dois *links* remotos². Todas estas redes locais e *links* remotos estão sendo representados por um modelo extremamente simples do tipo mostrado na Figura 5.12. Nele as transações *on-line* chegam basicamente com dois dados: o número de pacotes e o tamanho médio em *bytes* a serem transmitidos, no caso das redes locais, ou o número de quadros e o tamanho médio dos quadros em *bytes*, no caso dos *links* remotos.

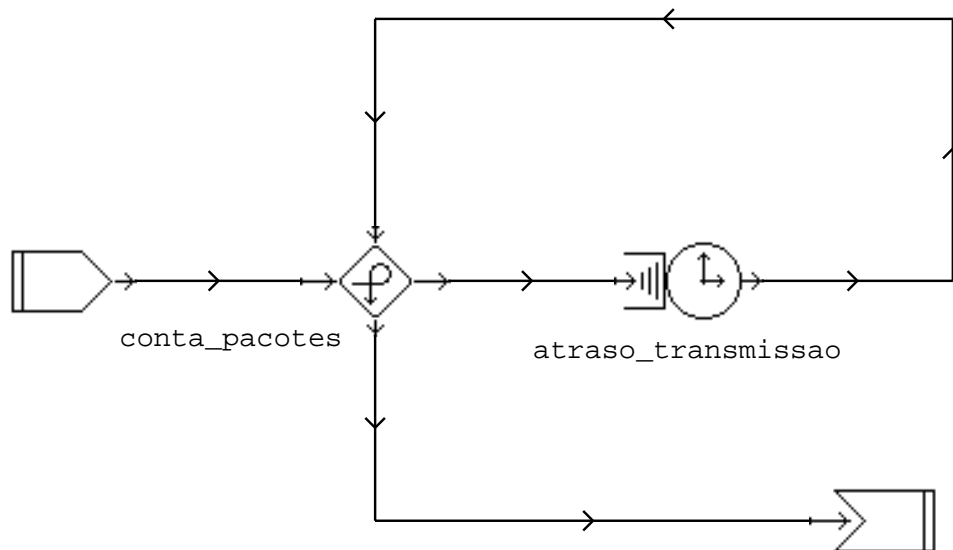


Figura 5.12: O submodelo de rede (ethernet ou X.25).

Para mostrar como funcionam os modelos de rede local e *links* remotos adotados no modelo de simulação do SICOM, descrevemos a seguir a forma como um computador cliente se comunica com o servidor localmente. O computador cliente envia uma mensagem para o

²O serviço MINASPAC está sendo chamado de *link* remoto por ser esta a forma com que a COPASA-MG contrata este tipo de serviço. Pois, independente de quais recursos serão necessários, a TELEMIG, empresa responsável pelo serviço MINASPAC, deverá prover um acesso de 64 Kbps entre Janaúria e Montes Claros e entre Janaúba e Montes Claros.

servidor composto de 50 pacotes com o tamanho médio de 400 bytes. Ao chegar no modelo da rede local a transação irá passar pelo `contador_pacotes_ethernet` para logo em seguida passar pelo nó `transmissao_moc`. Após esperar na fila do nó de transmissão a transação irá usar o nó de transmissão pelo tempo necessário para se transmitir um pacote com 400 bytes. Concluída a transmissão, a transação retorna ao `contador_pacotes_ethernet` que irá enviá-la novamente ao nó `transmissao_moc` caso não tenham sido transmitidos os 50 pacotes da mensagem.

Conhecido o funcionamento individual das redes, basta entender como funcionam os submodelos `cliente_servidor` e `servidor_clientes` para se compreender como a rede de computadores da GRNT foi representada no modelo de simulação do SICOM. Basicamente, estes dois submodelos identificam o `porto` da transação, roteando-as de sua origem até o servidor de banco de dados, no caso do submodelo `cliente_servidor`, ou então roteando-as do servidor de banco de dados para a origem, no caso do submodelo `servidor_cliente`. Assim, uma mensagem de um computador cliente em Januária, por exemplo, primeiro passa pelo submodelo `cliente_servidor` que faz com que a mensagem passe pela rede *ethernet* de Januária, para em seguida passar pelo *link* X.25 entre Januária e Montes Claros, para passar pela rede *ethernet* de Montes Claros e finalmente chegar ao servidor de banco de dados. Uma vez processada a mensagem a resposta é enviada ao submodelo `servidor_cliente` que faz com que a transação passe pela rede *ethernet* de Montes Claros, para em seguida passar pelo *link* X.25 entre Januária e Montes Claros, para passar pela rede *ethernet* de Januária e chegar ao computador cliente.

5.3 O Modelo de Simulação do SICOM/CS

O modelo de simulação do SICOM/CS não apresenta grandes mudanças em relação ao modelo do SICOM/D, estando todas as diferenças implementadas no submodelo `natural`, conforme Figura 5.13, e nos parâmetros de carga da trabalho referente à rede de computadores.

Em relação ao processamento de uma transação *on-line*, o que muda é a localização da aplicação que deixa de estar no servidor de banco de dados para ser implantada em cada computador cliente. Desta forma, a aplicação irá usar a rede de computadores e o servidor de banco de dados para transmitir e processar as operações de banco de dados.

Existe, entretanto, um novo componente no modelo do SICOM/CS que é o roteador (*stub*) de operações de banco de dados Entire Network da Software AG [WCP94]. Este produto funciona da seguinte forma. Existe uma cópia do Entire Network em cada computador cliente, assim como uma cópia do Entire Network no servidor de banco de dados em Montes Claros. Cada cópia do Entire Network, seja cliente ou servidor, sabe através de tabelas internas onde se localizam as demais cópias com as quais deve se comunicar (isto ocorre porque as cópias do Entire Network são previamente configuradas com as tabelas dos endereços das demais cópias). Assim, quando um usuário executa uma transação *on-*

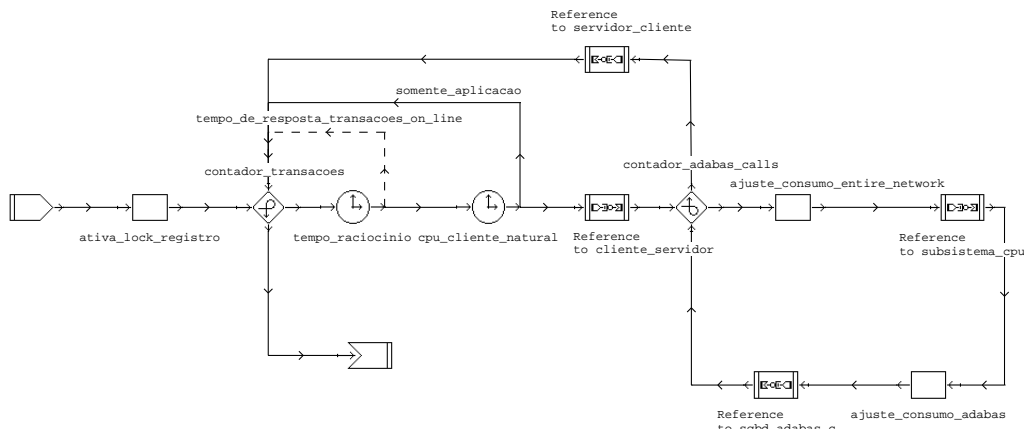


Figura 5.13: O submodelo dos processos Natural do SICOM/CS.

line que requisita operações ao SGBD ADABAS, o programa Natural enxerga a cópia do Entire Network local como se fosse um SGBD e passa para ela a operação desejada. Essa cópia do Entire Network, com o auxílio de suas tabelas, identifica onde se encontra a cópia do Entire Network responsável pela execução da operação requisitada e usando os serviços TCP/IP envia a mensagem. A cópia do Entire Network de destino recebe a mensagem contendo a operação, repassa para o SGBD ADABAS local, recebe a resposta da operação e envia para a cópia do Entire Network de origem. Ao receber a resposta, a cópia do Entire Network do computador cliente a repassa para o programa Natural.

No caso do modelo SICOM/CS, foi retirado do submodelo *natural* o tempo de uso do processador pela aplicação SICOM, que foi substituída pelo tempo de uso do processador pelo Entire Network. Uma segunda mudança neste submodelo é que somente as terceiras e quartas transações *on-line* trafegarão pela rede de computadores, já que as demais são resolvidas localmente, ou seja, dentro do próprio computador cliente.

Em relação à rede, os pacotes *ethernet* e os quadros *X.25* também mudaram em relação ao modelo do SICOM/D, pois antes estavam trafegando pela rede mensagens contendo telas (com os dados embutidos em tais telas) do serviço *telnet* do TCP/IP, enquanto no modelo do SICOM/CS somente trafegam requisições ao SGBD ADABAS e respostas dessas requisições (que contêm essencialmente dados).

5.4 Validação do Modelo de Simulação do SICOM/D

Para validar o modelo de simulação do SICOM/D, há necessidade de se comparar os tempos de resposta de suas transações *on-line* medidos experimentalmente, com os tempos de resposta de suas transações *on-line* obtidos da execução de seu modelo de simulação. Assim, existem três etapas a serem cumpridas: a de medição experimental dos tempos de respostas das transações *on-line* do SICOM/D, a de execução do modelo de simulação e a

de calibragem do modelo de simulação.

A etapa de medição experimental dos tempos de resposta do SICOM certamente é uma das tarefas mais delicadas de todo este trabalho. De fato, não é uma tarefa simples criar um ambiente experimental de onde seja possível capturar nos tempos de resposta das transações *on-line* os efeitos da concorrência pelos recursos computacionais .

Neste trabalho está sendo considerado que a validação do modelo de simulação não será precisa o suficiente para se obter resultados finais com os quais a COPASA-MG irá trabalhar. De fato, a própria estratégia de migração do SICOM minimiza a necessidade de uma validação precisa neste momento. Essa estratégia, como visto na Seção 2.4, prevê inicialmente a migração do Sistema HP07 para o SICOM/C, que posteriormente será migrado, regional a regional, para o SICOM/D. Assim, logo que ocorrer a migração para o SICOM/D na Gerência da Regional Norte, será possível medir os tempos de resposta das transações *on-line* num ambiente operacional real, de forma que ainda haverá tempo suficiente para reavaliar o dimensionamento dos recursos computacionais da própria Gerência da Regional Norte, bem como para dimensionar com precisão os recursos computacionais necessários para as demais regionais da COPASA-MG. Este estudo, entretanto, exige uma primeira validação do modelo de simulação, para que sejam detectadas eventuais falhas no modelo de simulação que não foram observadas durante o seu desenvolvimento.

Sobre a etapa de execução do modelo de simulação, deve ser verificado que todos os detalhes observados na experimentação serão implementados também no modelo de simulação. Assim, se forem utilizados dez computadores clientes executando cada um o serviço *on-line Pesquisa de Consumidor*, com um tempo de raciocínio médio de 1000 ms, então o modelo também deverá ser submetido à uma carga de trabalho equivalente à gerada por estes mesmos dez clientes executando este mesmo serviço.

A terceira e última etapa do processo de validação do modelo de simulação do SICOM/D consiste no processo de comparar os tempos de resposta obtidos experimentalmente e através da execução do modelo de simulação e em ajustar os parâmetros, quando não o próprio modelo, de forma a aproximar o máximo possível os valores dos tempos de resposta. O processo de ajustar os parâmetros e o modelo de simulação é chamado de calibragem do modelo de simulação. O modelo será considerado validado quando ele for capaz de representar situações reais medidas experimentalmente, sem que seja necessária nenhuma calibragem.

Conhecidas as etapas do processo de validação, iremos relatar como foram realizadas as etapas de medição experimental dos tempos de resposta e de calibragem do modelo de simulação. Porém antes iremos relatar quais foram as alternativas cogitadas para se realizar a medição, assim como quais problemas foram identificados ou enfrentados em cada uma destas alternativas.

5.4.1 Alternativas para Realizar as Medições Experimentais dos Tempos de Resposta

Existem diversas formas de se medir o tempo de resposta de transações *on-line* num sistema multiusuário sendo executado numa rede como a do SICOM, cada uma com suas vantagens e desvantagens. Desta forma, foram identificadas quatro formas de realizar essas medições que serão descritas a seguir.

Instrumentalização do Código Fonte do SICOM

Esta alternativa consiste em modificar o código fonte do SICOM de forma a inserir em todos os pontos que sejam o início e o fim de uma transação *on-line* uma rotina que grava num arquivo a hora (*timestamp*) em que os referidos pontos do código foram executados.

Esta alternativa apresenta uma série de problemas. Inicialmente, ela não captura nos tempos de resposta medidos o tempo de rede das transações *on-line*, já que esses códigos são executados após a transação *on-line* já estar no servidor de banco de dados e antes das respostas terem efetivamente retornado aos usuários. Um segundo problema é que o próprio processo de registrar a hora faz com que cada transação aumente sua utilização de CPU e do subsistema de E/S.

Ainda baseado na idéia de instrumentalizar o código fonte, foi cogitada a modificação do código de forma a registrar nas telas do sistema a hora em que elas foram executadas, para gravar, já no computador cliente, todas as telas que chegam. Esta variante também não captura o tempo de rede nos tempos de resposta medidos e ainda cria o problema de tratar um enorme volume de dados com o intuito de recuperar as horas que foram exibidas nas telas.

Inserção de um Espião entre o Telnet Cliente e o Telnet Servidor

De acordo com [Ste94], dispositivos que usam o protocolo TCP/IP, tais como computadores, roteadores e impressoras de rede são identificados por um número de 4 bytes denominado de endereço IP. Um serviço TCP/IP, num dado endereço IP, é identificado por um número de 16 bits ao qual é dado o nome de porta. O par endereço IP e número da porta é usualmente chamado de *socket*. Como o serviço `telnet` do TCP/IP é baseado numa conexão TCP, então ele usa um *socket* do cliente e um *socket* do servidor.

O que está sendo chamado de espião é um programa que monitora a conexão TCP entre o `telnet` cliente e o `telnet` servidor. O espião deve possuir dois *sockets*: um que oferece um serviço `telnet` cliente e outro que oferece um serviço `telnet` servidor. Desta forma, seria configurado o `telnet` cliente do computador cliente para se conectar ao *socket* que oferece o serviço `telnet` servidor do espião, assim como seria configurado o `telnet` cliente do espião para se conectar ao *socket* servidor do computador servidor. O funcionamento do espião consiste em repassar os pacotes recebidos do *socket* cliente para o *socket* servidor,

os pacotes recebidos do *socket* servidor para o *socket* cliente e registrar a hora em que cada pacote passou pelo espião. De acordo com o esquema de funcionamento do espião, ele poderia estar localizado em qualquer computador da rede.

Esta alternativa também apresenta problemas. Um serviço `telnet` pode operar caractere a caractere ou linha a linha. Desta forma, o número de pacotes TCP transmitidos durante uma sessão `telnet` é elevado, não existindo nenhuma distinção entre esses pacotes que identifique qual está efetivamente enviando o comando a ser processado pelo servidor. Assim, teria de ser criado um procedimento de análise dos conteúdos dos pacotes para identificar quais são os primeiros e os últimos pacotes de uma transação *on-line*.

Uso do *Script* de Conexão do Telnet Cliente

Esta alternativa é de fácil implementação, computa o tempo de rede nos tempos de resposta das transações *on-line* e não causa nenhum *overhead* na utilização dos recursos computacionais do servidor de banco de dados e da rede de computadores; porém existem problemas difíceis de serem diagnosticados numa primeira inspeção dos resultados obtidos. Acreditando nesta alternativa, foram codificados os *scripts* de todos os serviços *on-line* geradores da carga de trabalho sintética. Infelizmente todas as medições realizadas foram inúteis para se calibrar o modelo de simulação.

Os problemas desta alternativa são sutis e para serem compreendidos é necessário se conhecer um pouco do funcionamento do *script* do NetTerm [Int96] da InterSoft International, que foi o utilitário `telnet` usado na tentativa frustrada de medição dos tempos de resposta.

```

expect 10 "padrao :"
output "1^M"
expect 10 "Cod:"
output "SC^M"
TESTE:
# Executando DOSSIE (1a vez)
expect 10 "13-24"
exec time2.bat
output " DOSSIE^M"
exec time2.bat
expect 10 "...:"
exec time2.bat
output "100000886^M"
exec time2.bat
expect 10 "13-24"

```

Figura 5.14: Exemplo de *Script* do NetTerm.

O interpretador de *script* do NetTerm oferece o seguinte conjunto de recursos: a detecção do recebimento de uma seqüência de caracteres; o envio de uma seqüência de caracteres; o desvio do fluxo de execução do *script*; o atraso programado por um período determinado de tempo; e a execução de programas. A Figura 5.14 mostra parte de um *script* que foi criado para se realizar as medições. Nesta figura o comando `expect 10 "padrao:"` espera por até 10 segundos o recebimento da seqüência de caracteres "padrao:", passando a execução para o próximo comando do *script*. O output "1^M" envia o caracter "1" seguido de um ENTER, representado neste caso pelo "^M". O `exec` executa um programa (ou mais especificamente, um arquivo executável). Neste caso, ele executa o comando TIME do DOS indiretamente através da execução de um arquivo *batch* do DOS, gravando o resultado num arquivo local. Este arquivo local será chamado de arquivo de *log* das transações *on-line*, ou simplesmente *log*. Assim, o comando `exec time2.bat` é executando imediatamente antes e depois de cada comando `output` contendo um ENTER ou uma tecla de função. Supunha-se que seria possível calcular o tempo de resposta de cada transação *on-line* a partir deste *log* gerado subtraindo dois a dois os tempos registrados. Como o NetTerm funciona assíncrono ao serviço `telnet`, foi verificado que a estratégia correta para calcular os tempos de resposta seria subtrair a hora registrada antes do comando `output` da hora registrada após o comando `expect`, assumindo como desprezível o tempo de *parse* que o NetTerm usa para avaliar o casamento de padrão existente no comando `expect`.

O principal problema desta solução está no escalonamento dos processos no sistema operacional Windows 95 dos computadores clientes. Observando o comportamento do Windows 95 em sua configuração padrão foi verificado que ele prioriza um processo em detrimento dos demais. Assim, se forem submetidos dois NetTerm, executando cada um um *script*, é possível verificar que apenas um executa normalmente, enquanto o segundo permanece inativo. Do *script* da Figura 5.14 verifica-se que cada comando `exec time2.bat` inicia um processo que fica inativo por longos períodos de tempo. Mas como são justamente estes os processos que registram o tempo de resposta das transações, então conclui-se que as medições realizadas não são válidas.

A Figura 5.15 mostra que os tempos de resposta de três medições da execução de um mesmo serviço não se comportam de forma similar. Comparando-se os dois primeiros lotes de medições verificamos uma variação de até 6.700. Até este momento se acreditava que esta variação estava ocorrendo exclusivamente devido ao escalonamento de processos do Windows 95, porém quando aumentou-se o *time out* do comando `expect` foi verificado que o tempo de resposta também aumentou, ou seja, o tempo de *parse* do NetTerm não é desprezível como havia sido suposto.

Certamente existem parâmetros para ajustar a configuração do Windows 95, o que implica em alterações no conteúdo do *registry* do sistema operacional, que é justamente o arquivo de parâmetros do Windows 95. Como na COPASA-MG o Windows 95 é instalado com a configuração padrão (*default*), o *registry* se encontra protegido e o experimento de medição exige a participação de até 24 computadores clientes que não estavam autorizados a serem reconfigurados, então resolveu-se não alterar a configuração de nenhum computador

```

Expect de 10 seg.
14:33:35,78 ~ 14:33:36,05  270 msec
14:33:43,69 ~ 14:33:46,10  410 msec
14:33:57,09 ~ 14:33:57,31  220 msec
14:34:05,60 ~ 14:34:10,22 4620 msec

Expect de 10 seg.
14:54:31,92 ~ 14:54:34,62 2700 msec
14:55:22,62 ~ 14:55:25,37 2750 msec
14:56:15,46 ~ 14:56:16,23  770 msec
14:57:04,67 ~ 14:57:07,09 2420 msec

Expect de 50 seg.
15:26:06,03 ~ 15:26:06,19  160 msec
15:26:06,74 ~ 15:26:11,41 4670 msec
15:26:11,80 ~ 15:26:15,75 3950 msec
15:26:16,30 ~ 15:26:18,88 2580 msec

```

Figura 5.15: Medidas de tempos de resposta da execução do serviço *pesquisa consumidor* com o uso do NetTerm.

cliente, assim como não usar esta alternativa de medição de tempos de resposta.

Instrumentalização do Telnet Cliente

A instrumentalização do próprio `telnet` provavelmente é a melhor alternativa para se realizar medições de tempos de resposta, pois além de todos os benefícios identificados no uso de um `telnet` com um *script*, ela ainda possibilita que os tempos sejam registrados em memória e pelo mesmo processo que executa o serviço `telnet`, o que elimina todos os problemas previamente identificados.

Para se instrumentalizar um `telnet` cliente do Windows 95, foi utilizado o utilitário `telnet` de um pacote de componentes para a linguagem Delphi denominado de Internet Component Suite [Pie98], ou ICS. Deste pacote de componentes foram utilizados dois componente que implementam o serviço `telnet` do TCP/IP com a facilidade de submeter um *script*.

O registro dos tempos de resposta é todo processado dentro do próprio código fonte do componente e armazenado temporariamente em memória, conforme mostra a Figura 5.16. Sempre que um evento que compõe o *script* do ICS é executado, a hora do sistema é armazenada num vetor e é atribuído o valor TRUE à variável booleana `MarcaTempo`. Toda vez que chega algum dado através do *socket* do `telnet` cliente o componente aciona a rotina `TriggerDataAvailable`. Para registrar o fim de uma transação *on-line*, foi incluído

```

PEvent := PEventDescriptor(FEventList.Items[Item]);
I := FindEventString(PEvent^.Search);
if I <> -1 then begin
  if Length(PEvent^.ToSend) > 0 then
    begin
      SendStr(PEvent^.ToSend);
      ListaTempos[ContaTempo].HoraInicio := DateTimeToTimeStamp(Time);
      MarcaTempo := true;
    end;
  (...)
end;
(...)
procedure TInScript.TriggerDataAvailable(Buffer: PChar; Len: Integer);
begin
  if MarcaTempo then
    begin
      ListaTempos[ContaTempo].HoraFim := DateTimeToTimeStamp(Time);
      MarcaTempo := false;
    end;
  ProcessInputData(Buffer, Len);
  inherited TriggerDataAvailable(Buffer, Len);
end;

```

Figura 5.16: Implementação do registro dos tempos de resposta no código fonte do ICS.

nesta rotina uma condição que verificando que a variável `MarcaTempo` é verdadeira registra a hora do fim da resposta.

A Figura 5.17 mostra medidas de tempo de resposta do serviço *pesquisa consumidor* com o uso do ICS e em situações diferentes. Em todas as situações o tempo de resposta é praticamente o mesmo. As duas primeiras medições correspondem a uma amostra de uma seqüência de execuções do serviço *pesquisa consumidor*, sendo que em todas elas os valores apresentam uma variação insignificante. Nas duas últimas medições foram inseridos pequenos atrasos entre a execução de cada transação *on-line*, o que não afetou em nada o tempo de resposta das transações, como era esperado. Assim, considerou-se esta alternativa confiável para se realizar as medições de tempo de resposta das transações *on-line* do SICOM/D.

5.4.2 Medições Experimentais dos Tempos de Resposta

Foram realizados três experimentos utilizando respectivamente 8, 14 e 24 clientes submetendo serviços *on-line* do SICOM constantemente a um sistema computacional equivalente ao sistema computacional em questão. Esses valores foram escolhidos por representarem o uso do SICOM em três situações distintas: com uma carga de trabalho leve (8 usuários), uma carga de trabalho intermediária (14 usuários) e uma carga de trabalho pesada (24 usuários). A carga de trabalho resultante do uso do SICOM com 24 usuários

```

----- Novo Servico -----
21:49:22 Transacao 17 = 1760 msec
21:49:23 Transacao 18 = 660 msec
21:49:24 Transacao 19 = 1760 msec
21:49:25 Transacao 20 = 710 msec
----- Novo Servico -----
21:49:27 Transacao 21 = 1810 msec
21:49:28 Transacao 22 = 720 msec
21:49:29 Transacao 23 = 1750 msec
21:49:30 Transacao 24 = 720 msec
----- Novo Servico -----
22:28:34 Transacao 13 = 1860 msec
22:28:35 Transacao 14 = 720 msec
22:28:37 Transacao 15 = 1870 msec
22:28:37 Transacao 16 = 710 msec
----- Novo Servico -----
22:31:01 Transacao 17 = 1820 msec
22:31:02 Transacao 18 = 720 msec
22:31:04 Transacao 19 = 1810 msec
22:31:05 Transacao 20 = 770 msec

```

Figura 5.17: Medidas de tempos de resposta da execução do serviço *pesquisa consumidor* com o uso do ICS.

está sendo considerada uma carga de trabalho pesada pelo fato de ser a taxa de submissão de serviços *on-line* ao sistema de computação bem maior que a taxa de submissão real, assim como pelo fato de ser toda a submissão de serviços *on-line* na mesma rede local do servidor de banco de dados, o que elimina o atraso na transmissão de pacotes X.25.

Um dos problemas enfrentados na realização deste experimento foi determinar uma forma de automatizar a execução do serviço `telnet` de forma a representar a presença de um usuário efetivamente usando o SICOM/D. Este procedimento se faz necessário para que o experimento seja reproduzível, pois desta forma é possível que cada um dos serviços `telnet`, que serão chamados de usuários virtuais, mantenha um tempo de raciocínio constante. Conforme visto na Seção 5.4.1, a melhor alternativa identificada para se implementar usuários virtuais usando o SICOM/D foi usar o pacote de componentes Internet Component Suite para a linguagem Delphi da Borland [Bor97].

Para se realizar as medições experimentais, entretanto, se faz necessário definir uma carga de trabalho sintética para ser usada de forma a capturar nos tempos de resposta o efeito da concorrência por recursos em função do uso simultâneo de serviços *on-line* do SICOM. Neste sentido, foram consolidadas as frequências de execução dos serviços *on-line* da Gerência da Regional Norte a partir da Tabela 4.3, da qual foi obtida a Tabela 5.1.

Categoria	Frequência
Gera SS	162
Consulta SS	121
Baixa OS	152
Recebimento OS	85
Cancela OS	88
Consulta Cadastro Consumidor	50
Altera Cadastro Consumidor	12
Pesquisa Consumidor	165
Consulta Débito Consumidor	46
Detalhe Débito Consumidor	36

Tabela 5.1: Frequência total de execução de serviços *on-line* na Gerência da Regional Norte.

Conhecida a intensidade desejada da carga sintética, o próximo passo seria operacionalizar o experimento. Infelizmente não foi possível executar de forma automatizada (usando os usuários virtuais) e contínua os dez serviços *on-line* identificados. De fato, os serviços que trabalham com OSs e débitos não podem ser continuamente executados. Uma OS, por exemplo, não pode ser baixada ou cancelada mais de uma vez. Mais ainda, mesmo antes de se baixar ou cancelar uma OS deve ser verificado se ela pode ser efetivamente baixada ou cancelada, o que na maioria das vezes não é possível.

Categoria	Participação (%)
Gera SS	31,8
Consulta SS	23,7
Consulta Cadastro Consumidor	9,8
Altera Cadastro Consumidor	2,3
Pesquisa Consumidor	32,3

Tabela 5.2: Participação dos serviços *on-line* no experimento de medição dos tempos de resposta.

Face a esses problemas, os serviços *Recebimento OS*, *Baixa OS*, *Cancela OS*, *Consulta Débitos* e *Verifica Detalhe Débito* não foram usados no experimento de medição do tempo de resposta das transações *on-line*. Assim, a partir dos serviços restantes foi gerada a Tabela 5.2 que identifica o percentual de participação de cada um dos serviços na composição da carga de trabalho sintética a ser gerada experimentalmente. Do número de usuários virtuais e da Tabela 5.2 foi gerada a Tabela 5.3 que descreve a distribuição dos serviços *on-line* pelos computadores clientes.

Neste experimento foi adotada a estratégia de se executar um único serviço por computador cliente, o que simplificou o processo de medição e tratamento dos dados, pois cada *log* de transação *on-line* era composto de uma única categoria de serviço. É importante ressaltar que os serviços executados durante o horário de pico da Gerência da Regional Norte não têm correlações dentro da janela de tempo de 1 hora, conforme detectado nos estudos dos sistemas legado e alvo [GdSL98]. Não existindo correlações entre os serviços, eles puderam

Exp.	Clientes	Serviços
1	3	Gera SS
	2	Consulta SS
	3	Pesquisa Consumidor
2	5	Gera SS
	3	Consulta SS
	1	Consulta Cadastro Consumidor
	5	Pesquisa Consumidor
3	8	Gera SS
	5	Consulta SS
	2	Consulta Cadastro Consumidor
	9	Pesquisa Consumidor

Tabela 5.3: Distribuição dos serviços *on-line* por computadores clientes nos experimentos de medição de tempos de resposta .

ser executados livremente, sem a preocupação de executar um serviço específico após a execução de um determinado serviço.

Após consolidados os dados contidos no *log* de transações *on-line* de cada computador participante do experimento, finalmente foram obtidos os resultados descritos na Tabela 5.4 e mostrados graficamente na Figura 5.18.

Usuários Simultâneos	Tempo de Resposta (ms)				Média
	Gera SS	Consulta SS	Pesquisa Consumidor	Pesquisa Cadastro	
8	231	271	386		299,12
14	296	241	852	234	478,35
24	446	358	1.537	328	826,96

Tabela 5.4: Tempos de resposta do SICOM/D medidos experimentalmente.

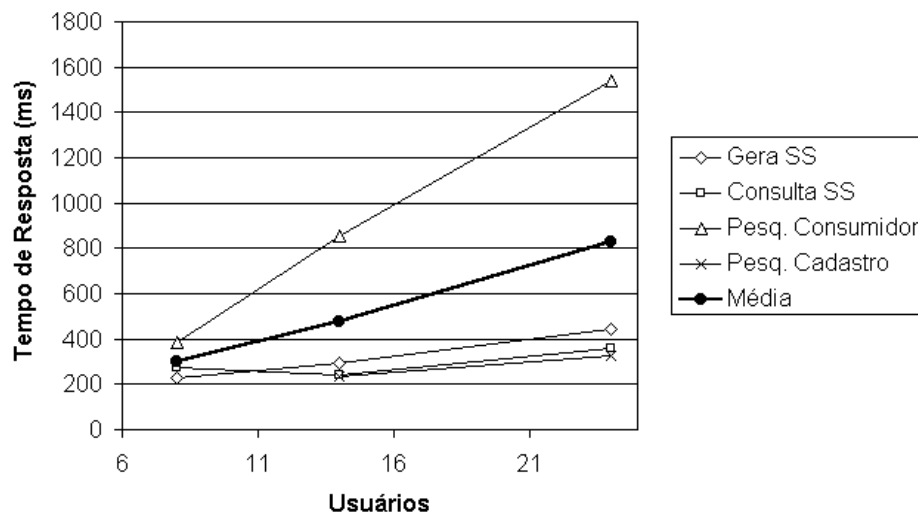


Figura 5.18: Gráfico dos tempos de resposta medidos experimentalmente.

5.4.3 Calibragem e Validação

O modelo de simulação deve ser calibrado para cada uma das medições experimentais realizadas. Por calibragem está sendo chamado o processo de comparação dos tempos de resposta obtidos experimentalmente e através de simulação, assim como do ajuste do modelo de simulação de forma que a diferença dos tempos de resposta obtidos seja satisfatoriamente pequenos (Uma variação de 5% geralmente é usada para validar modelos, porém como o modelo de simulação apresentado neste trabalho terá de ser novamente calibrado quando o SICOM/D entrar em operação na Gerência da Regional Norte, então estamos adotando uma variação de até 10% como sendo suficiente para validar o modelo). O modelo de simulação será considerado validado após estar simultaneamente calibrado para os tempos de resposta obtidos experimentalmente com 8, 14 e 24 usuários virtuais usando o SICOM.

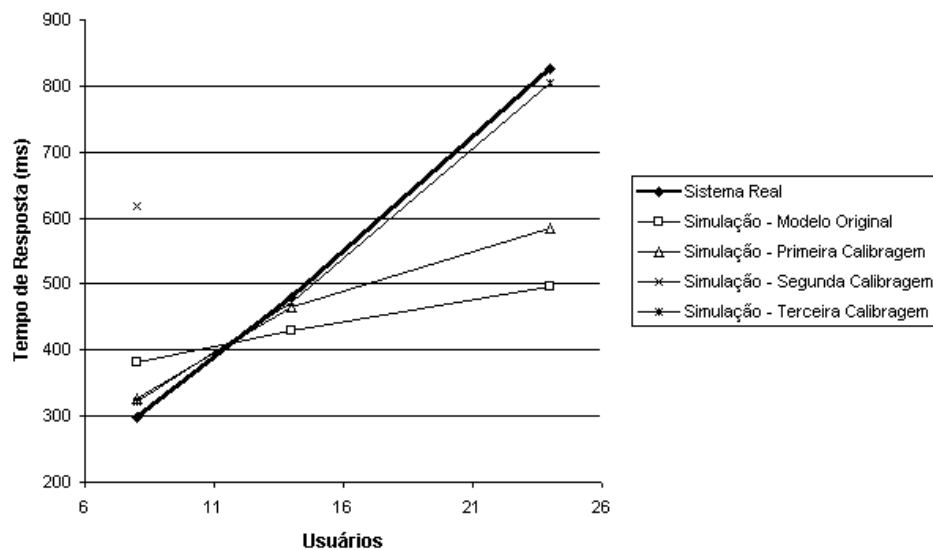


Figura 5.19: Tempos de resposta obtidos durante a calibragem do modelo de simulação do SICOM/D.

O modelo de simulação descrito na Seção 5.2 é o modelo validado do SICOM/D, diferente do modelo inicialmente proposto. A Figura 5.19 mostra o gráfico comparando os tempos de resposta médios das transações *on-line* em função do número de usuários simultaneamente submentendo serviços *on-line* ao SICOM/D, medidos experimentalmente e através de simulação, durante cada etapa de calibragem do modelo de simulação. A Figura 5.19 mostra ainda que foram realizadas três modificações no modelo proposto até se chegar no modelo validado. Assim, a seguir serão descritas as modificações realizadas em cada etapa do processo de calibragem, relatando a motivação para se realizar cada modificação.

O Modelo de Simulação Proposto e a Primeira Tentativa de Calibragem

O modelo de simulação inicialmente proposto não diferenciava muito do modelo validado, o que mostra que o conhecimento do ambiente computacional do SICOM/D adquirido principalmente durante o processo de caracterização da carga de trabalho foi decisivo no processo de criação do modelo de simulação.

Ao se executar o modelo de simulação proposto, entretanto, foram obtidos três tempos de resposta bem diferentes dos tempos de resposta medidos experimentalmente, sendo de 27,20% a diferença para 8 usuários e de 66,58% a diferença para 24 usuários. O resultado obtido, além de não satisfazer ao requisito de validação do modelo, apresentava o agravante do tempo de resposta simulado para 8 usuários ser bem maior que o tempo de resposta experimental, enquanto o tempo de resposta simulado para 24 usuários ser bem menor que o tempo de resposta obtido experimentalmente.

Deste resultado foram identificados dois possíveis problemas: (1) o modelo não estaria reproduzindo de forma satisfatória o efeito da concorrência de recursos pelos serviços, o que justificaria o fato do tempo de resposta não crescer tanto quanto o medido experimentalmente, e (2) a existência de parâmetros da carga de trabalho superdimensionadas, o que justificaria o tempo de resposta elevado obtido da simulação com 8 usuários virtuais.

Verificando o modelo, foi identificado um erro básico: os pacotes *ethernet* não estavam sendo distribuídos entre as transações *on-line* quando da execução dos serviços, assim como não estava sendo considerado que os pacotes transmitidos eram utilizados nos dois sentidos, ou seja, tanto na submissão quanto na resposta da transação *on-line*. Assim, por exemplo, se para um serviço que executava 6 transações *on-line* foi medido que 120 pacotes *ethernet* eram transmitidos, o modelo proposto estava transmitindo $120 \times 6 \times 2$ pacotes. Como consequência deste erro, as transações *on-line* estavam com tempos de resposta simulados maiores que os medidos experimentalmente de uma forma geral (o que inclui o caso de 8 usuários virtuais). Entretanto, como as transações *on-line* ficaram mais na rede de computadores que o necessário, foi reduzida a carga de trabalho no servidor de banco de dados, o que ficou claramente refletido nos tempos de resposta simulados com 24 usuários virtuais.

A Segunda Tentativa de Calibragem

Sanado o erro de modelagem no uso do parâmetro *número de pacotes ethernet* foram realizadas novas simulações, das quais foram obtidos tempos de resposta mais próximos dos tempos medidos experimentalmente, sendo de 8,98% a diferença para 8 usuários, de 2,80

Com base nesses novos resultados, foi verificado que o modelo de simulação ainda não estava capturando o efeito da concorrência pelos recursos resultante de 24 serviços *on-line* sendo executados simultaneamente. Desta forma, foi cogitada a possibilidade de se simplificar o modelo e ao mesmo tempo aumentar o efeito da concorrência pelos recursos no tempo

de resposta. Assim, no submodelo `natural` foi eliminado o nó `contador_adabas_calls` responsável indireto pelo escalonamento das transações *on-line* quando da chamada do SGBD ADABAS pelos programas Natural.

Ao se remover o nó `contador_adabas_calls`, foi necessário se ajustar alguns outros parâmetros. No nó `ajuste_consumo_nattel`, ao invés de se dividir o tempo de uso do processador pelos processos Natural pelo número de chamadas ao SGBD, passou-se a dividi-lo pelo número de transações *on-line* do sistema. No submodelo `sgbd_adabas_c`, ao invés de calcular o número de operações de E/S por chamada ao SGBD, passou-se a calculá-lo pelo número de transações *on-line* por serviço.

Ao se executar a simulação com 8 usuários virtuais, verificou-se que a preempção dos processos Natural pelo sistema operacional não estava sendo tão eficaz quanto o escalonamento obtido com a chamada ao SGBD. De fato, conforme mostra a Figura 5.19, o tempo de resposta médio simulado foi maior que o dobro do tempo de resposta médio obtido experimentalmente. Logo, foi desconsiderada esta alternativa para calibrar o modelo do SICOM.

A Terceira Tentativa de Calibragem

Como havia sido diagnosticado na primeira tentativa de calibragem do modelo, ele não estava representando as transações *on-line* de forma que a concorrência pelos recursos fosse modelada como realmente ocorre com as transações *on-line*.

Para melhorar esta representação das transações *on-line*, foi feito um estudo estatístico dos tempos de resposta das transações *on-line*, bem como do código fonte dos programas responsáveis pelos serviços *on-line* em estudo.

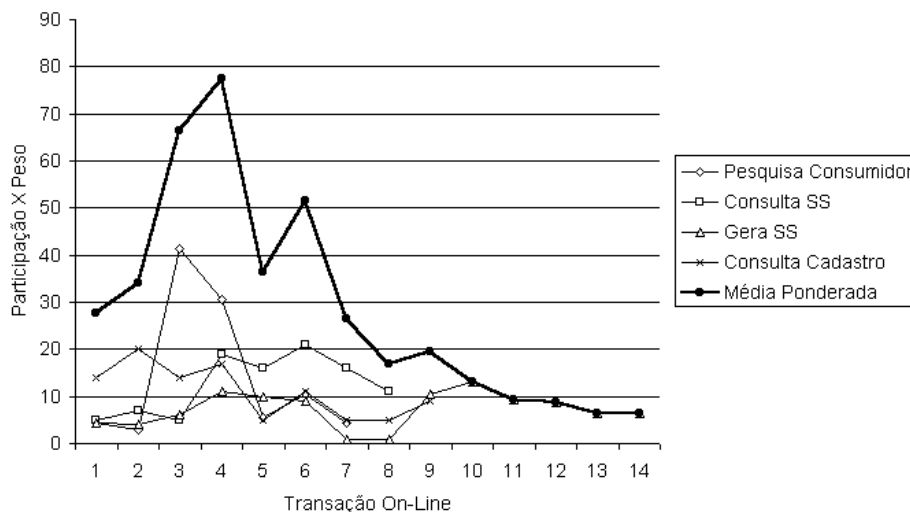


Figura 5.20: Participação das transações *on-line* nos tempos de resposta dos serviços *on-line*.

Do estudo estatístico, foi produzido um gráfico, mostrado na Figura 5.20, que descreve

o percentual de duração de cada transação *on-line* em relação do tempo de duração total do serviço *on-line*. Calculando a média de participação de cada conjunto de transações (transações *on-line* 1, transações *on-line* 2, etc) e atribuindo um peso 1 para cada serviço participante, foi obtida uma média ponderada que mostra que as transações *on-line* 3 e 4 são as que possuem os maiores tempos de resposta.

Do estudo do código fonte dos programas, foi verificado que muitas transações não fazem requisições ao SGBD ou, se fazem, são poucas as requisições. Por outro lado, poucas transações *on-line*, geralmente duas, realizam quase todas as requisições ao SGBD realizadas por todo o serviço *on-line*.

Confrontando essas duas informações, chegou-se à conclusão de que apenas as transações *on-line* 3 e 4 deveriam fazer muitas requisições ao SGBD. Assim, foi inserido um arco após o nó de referência ao submodelo `subsistema_cpu` que redireciona para o `contador_adabas_calls` todas as transações que não representam as transações *on-line* 3 e 4.

Como consequência desta calibragem, foram obtidas as seguintes diferenças: 7,6% para a simulação com 8 usuários, 1,2% para com 14 usuários e 2,8% para com 24 usuários. Esses valores foram considerados satisfatórios (todos menores que 10%), o que significa que o modelo de simulação pode ser considerado validado para as necessidades deste estudo.

5.5 Resultados das Simulações

Validado o modelo de simulação do SICOM, resta agora executá-lo. Na realidade serão realizadas duas simulações, uma com o modelo do SICOM/D e outra com o modelo do SICOM/CS derivado do modelo do SICOM/D. Como as mudanças são mínimas, o modelo de simulação do SICOM/CS está sendo considerado um modelo de predição [MAD94, Jai91] do modelo do SICOM/D.

Para realizar essas simulações, foi usado o módulo `confidence.grf` do SES/*workbench* [Sci96c] que implementa uma rotina que faz com que o SES/*workbench* forneça resultados confiáveis a partir do uso de intervalos de confiança [Mey83]. Alguns parâmetros devem ser fornecidos para se executar modelos com o módulo `confidence.grf`:

- **Período de observação** – Número de unidades de simulação que serão executadas. No caso dos modelos do SICOM, a unidade de simulação foi o milisegundo e o período de observação foi de 1 hora. Assim, será fornecido um período de observação de 3.600.000 unidades de simulação.
- **Período de aquecimento** – Período de tempo, em unidades de simulação, que o modelo será executado antes de efetivamente começar a registrar as medições. O uso deste período de aquecimento é aconselhado para que ocorra uma estabilização da simulação. De fato, quando as transações são submetidas, elas começam a usar recursos nos quais ainda não existe concorrência, ou seja, é uma situação que não

reflete a realidade de um sistema que supostamente já estaria em plena atividade. Conforme sugerido em [Sci96b], será usado um período de aquecimento igual ao período de observação.

- **Componentes e precisão do intervalo de confiança** – Ao se executar um modelo de simulação, algumas medições são registradas durante a sua execução. Em termos de intervalos de confiança, entretanto, faz-se necessário especificar quais medições devem ser confiáveis e com qual precisão. Neste trabalho, foi usado o tempo médio de resposta das transações *on-line* para o cálculo do intervalo de confiança, sendo de 95% a precisão desejada.
- **Número mínimo de observações** – Limite mínimo de execuções consecutivas, desprezando o período de aquecimento, para se atingir a precisão desejada no cálculo do intervalo de confiança. Segundo [Sci96b], este número deve ser igual a 5.
- **Número máximo de observações** – Limite máximo de execuções consecutivas para que o modelo construa o intervalo de confiança especificado com a precisão desejada. Este número, segundo [Sci96b], deve ser igual a 30.

Fornecidos os parâmetros acima, foram realizadas as simulações do SICOM/D e do SICOM/CS.

Para a simulação do SICOM/D foi necessário simular um período de 1,8e+07ms (de 3,6e+06ms a 2,16e+07ms) no qual foram executados 4.743.146 eventos. Durante este período de simulação, 95% dos valores obtidos para o tempo de resposta médio das transações *on-line* estavam no intervalo de confiança [587.235, 636.906], sendo de 612,071ms o tempo de resposta médio.

Serviço <i>On-Line</i>	Tempo de Resposta (ms)
Altera Cadastro	506,65
Baixa OS	458,91
Cancela OS	558,13
Consulta Cadastro	415,92
Consulta Débito	419,38
Consulta SS	437,10
Gera SS	553,20
Pesquisa Consumidor	1150,80
Recebimento OS	1060,70
Verifica Débito	557,61
Média Geral	612,06

Tabela 5.5: Tempos de resposta na simulação do SICOM/D.

A Tabela 5.5 mostra os tempos de resposta médios das transações *on-line*, por categoria de serviços *on-line*. Os tempos de resposta indicam que o sistema alvo atenderá os níveis de serviço almejados pela COPASA-MG, visto que os tempos de resposta são menores do que os tempos de resposta médios medidos do sistema legado, conforme Tabela 3.2.

Adicionalmente, foram verificadas algumas medidas de desempenho do SICOM/D tais como utilização do subsistema de processamento, das redes locais e dos *links* remotos. Essas medidas de desempenho, embora não forneçam um resultado tão imediato em termos de satisfação para o usuário como tempos de resposta médios, são bons para indicar quais os componentes mais próximos de saturarem. A Tabela 5.6 mostra a utilização dos componentes do sistema computacional implantado na GRNT executando a carga de trabalho prevista do SICOM/D. Desses dados, observa-se que todos os componentes do sistema são pouco utilizados, sendo o subsistema de processamento o mais usado. Por outro lado, componentes como as redes locais e o subsistema de E/S certamente serão os últimos a apresentarem problemas de desempenho na GRNT.

Componente do Sistema	Utilização (%)
Subsistema de Processamento	48,184
Subsistema de E/S	1,557
Rede local de Montes Claros	0,071
Rede local de Janaúria	0,010
Rede local de Janaúba	0,007
<i>Link</i> remoto entre MOC e Janaúria	18,882
<i>Link</i> remoto entre MOC e Janaúba	20,347

Tabela 5.6: Utilização do sistema computacional na simulação do SICOM/D.

Para a simulação do SICOM/CS, também foi necessário simular um período de $1,8e+07$ ms (de $3,6e+06$ ms a $2,16e+07$ ms) no qual foram executados 1.351.327 eventos, ou seja, bem menos que os 4.743.146 da simulação do SICOM/D. Esta redução no número de eventos necessários para se realizar a simulação do SICOM/CS, foi devido principalmente à redução na utilização da rede de computadores. Durante este período de simulação, 95% dos valores obtidos para o tempo de resposta médio das transações *on-line* estavam no intervalo de confiança [211.746, 220.491], sendo de 216,118ms o tempo de resposta médio.

Assim, como para os resultados da simulação do modelo do SICOM/D, os da simulação do modelo do SICOM/CS também foram dispostos em duas tabelas. A Tabela 5.7 mostra os tempos de resposta médio das transações *on-line* e a Tabela 5.8 mostra a utilização de cada componente do sistema computacional.

Realizadas essas simulações, foi feita uma análise para se identificar quando cada uma das duas alternativas, SICOM/D e SICOM/CS, irão saturar. Desta forma, o número de serviços executados foram acrescidos ano a ano pela taxa de crescimento vegetativo da unidade de negócio³ em cada uma das unidades. Em termos dos modelos de simulação isto foi implementado aumentando-se a taxa de chegada de transações nos submodelos de carga de trabalho proporcionalmente às taxas acumuladas ano a ano das unidades envolvidas, o que reduziu o tempo entre a submissão de transações *on-line* e aumentou a carga de trabalho. Desta projeção, foram obtidos os tempos de resposta ano a ano

³Na COPASA-MG usa-se como unidade de negócio o número de ligações de água ou esgoto. Desta forma, todos os recursos que a empresa torna disponível para uma localidade, tais como funcionários e equipamentos, são proporcionais ao número de ligações de água ou esgoto nessa localidade.

Serviço <i>On-Line</i>	Tempo de Resposta (ms)
Altera Cadastro	176,04
Baixa OS	147,67
Cancela OS	187,92
Consulta Cadastro	133,90
Consulta Débito	163,01
Consulta SS	164,57
Gera SS	190,41
Pesquisa Consumidor	396,25
Recebimento OS	440,13
Verifica Débito	194,48
Média Geral	216,12

Tabela 5.7: Tempos de resposta na simulação do SICOM/CS.

Componente do Sistema	Utilização (%)
Subsistema de Processamento	6,368
Subsistema de E/S	1,486
Rede local de Montes Claros	0,015
Rede local de Janaúria	0,002
Rede local de Janaúba	0,003
<i>Link</i> remoto entre MOC e Janaúria	4,880
<i>Link</i> remoto entre MOC e Janaúba	4,871

Tabela 5.8: Utilização do sistema computacional na simulação do SICOM/CS.

do SICOM/D e do SICOM/CS, a partir dos quais foi gerado um gráfico mostrado na Figura 5.21. A Figura 5.22 mostra o gráfico comparativo entre a utilização do subsistema de processamento pelo SICOM/D e o SICOM/CS. Nela observa-se que ambos os sistemas alvo não irão saturar durante os próximos 8 anos. Esta projeção de 8 anos se baseia na hipótese de que o sistema não sofrerá nenhuma alteração funcional durante todo este período, bem como todos os recursos analisados permanecerão de uso restrito ao SICOM.

5.5.1 Análise Comparativa das Soluções

O SICOM/CS é a melhor alternativa para o projeto SICOM em termos de desempenho conforme mostram os dados das Tabelas 5.5, 5.6, 5.7 e 5.8, mas principalmente em termos de planejamento de capacidade. A utilização de todos os recursos computacionais envolvidos no processamento do SICOM é menor no ambiente cliente-servidor do que no ambiente multiusuário. De fato, a utilização do subsistema de processamento foi reduzida de 48,184% para 6,368%, o que significa uma sobrevida para o SICOM usando o mesmo servidor de banco de dados.

A Figura 5.23 mostra como se distribuem os tempos de resposta das transações *on-line* do SICOM/D e do SICOM/CS durante o período de observação. O gráfico mostra que a maioria das transações *on-line* do SICOM/CS são executadas em menos de 300ms, enquanto que as mesmas transações *on-line* no SICOM/D apresentam tempos de resposta

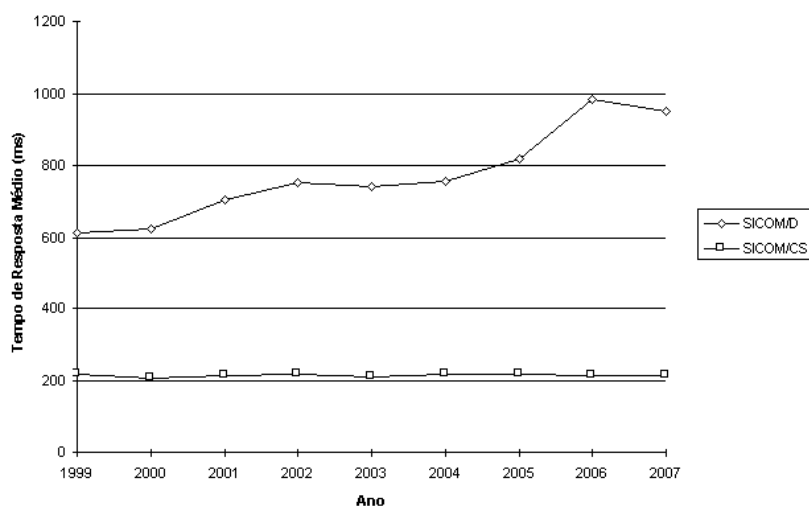


Figura 5.21: Tempos de resposta ano a ano do SICOM/D e do SICOM/CS.

bem distribuídos dentro da faixa de 100ms a 2s. Adicionalmente, verifica-se que não existem transações com tempo de resposta acima de 2s no SICOM/CS enquanto no SICOM/D 2% das transações apresentam tempo de resposta acima de 5s. Mesmo apresentando esta diferença significativa de tempo de resposta, ambos os sistemas alvo atendem aos requisitos da COPASA-MG. De fato, esta diferença no tempo de resposta é praticamente desprezível por estarem as médias abaixo de 1s.

Um segundo parâmetro a ser considerado nesta análise comparativa das soluções é o custo operacional associado com alguns recursos componentes do ambiente SICOM. No SICOM/D, em particular, o uso dos *links* remotos é 4,02 vezes maior que no SICOM/CS. Como o serviço MINASPAC é pago por volume trafegado, caso o SICOM/D venha a ser usado pela COPASA-MG pelos próximos 15 anos, como foi com o HP07, certamente a diferença acumulada do custo de manutenção dos *links* remotos de todas as gerências regionais não seria desprezível.

Quanto à utilização dos servidores de banco de dados, fica evidente que os equipamentos que estão da GRNT foram corretamente dimensionados. Entretanto, a migração do SICOM/D para o SICOM/CS permitiria que esses equipamentos fossem utilizados por outros sistemas de porte equivalente ao do SICOM sem que os novos tempos de resposta se tornem um problema.

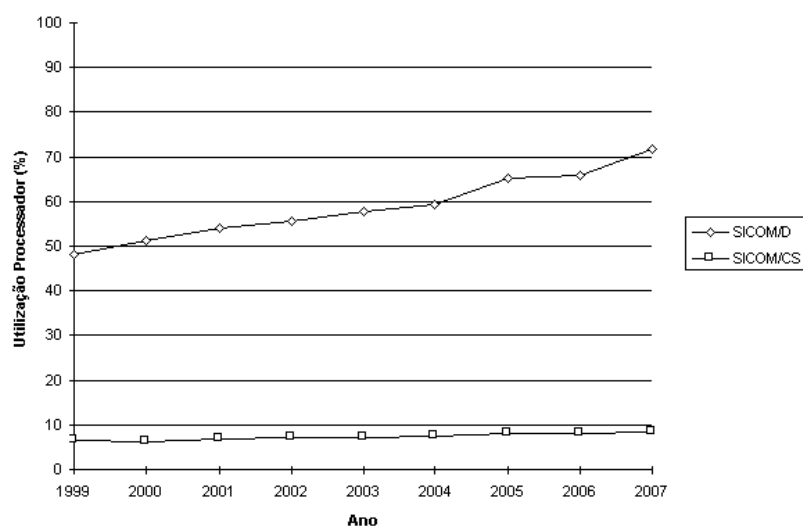


Figura 5.22: Utilização do sistema computacional ano a ano.

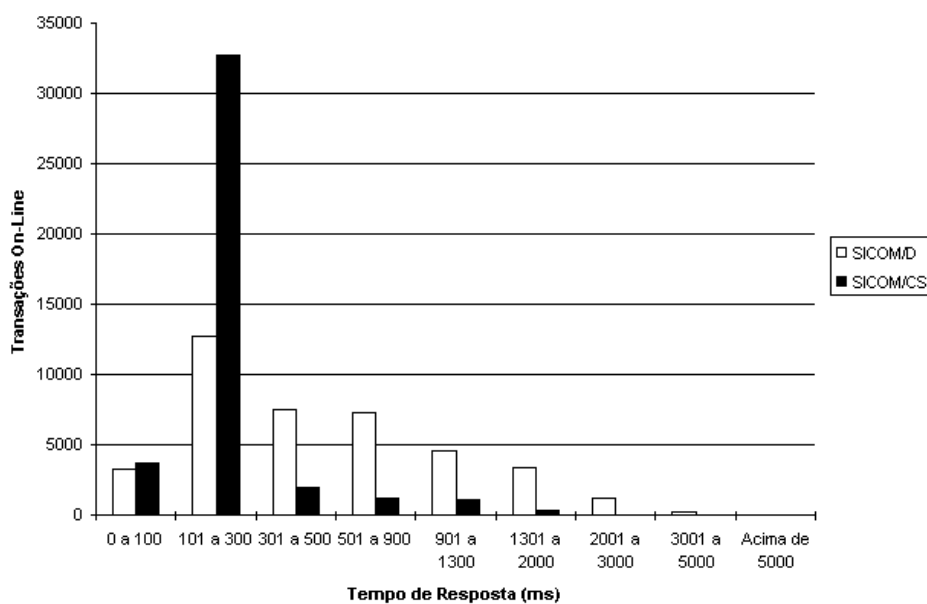


Figura 5.23: Comparação da distribuição dos tempos de resposta médios do SICOM/D e do SICOM/CS.

Capítulo 6

Conclusões

6.1 Revisão do Trabalho

Esta dissertação apresentou um estudo de caso no qual foi verificado se o ambiente computacional de um sistema alvo ainda não operacional suportaria a carga de trabalho prevista. Para que este estudo fosse possível, foi desenvolvido um método de análise de desempenho e planejamento de capacidade para ser usado durante a migração de sistemas legados. Uma descrição resumida do método foi apresentada no Capítulo 1. A partir do método foi desenvolvido um estudo de caso no qual foi verificado como seria o comportamento de um sistema alvo distribuído, derivado de um sistema legado centralizado, antes de sua efetiva implantação em operação. O estudo de caso apresentou detalhes do método, bem como algumas técnicas desenvolvidas para se solucionar problemas que surgiram durante a sua utilização.

O Capítulo 2 contextualizou o problema da migração de sistemas legados apresentando um histórico dos sistemas legado e alvo usados como estudo de caso. Foram relatados alguns dos problemas existentes no sistema legado que contribuíram para a decisão da COPASA-MG de migrar o sistema legado. Também foi apresentada a estratégia de migração que está sendo adotada para migrar o sistema legado em estudo.

O Capítulo 3 mostrou que o ambiente estudado era complexo e que um estudo completo do mesmo seria inviável devido ao cronograma de migração do sistema legado, pois as principais respostas somente seriam obtidas após verificadas na prática. Neste sentido foi definido um escopo menos amplo, no qual seria estudado o comportamento do sistema alvo durante o horário comercial. De fato, verificou-se que os serviços *on-line* eram responsáveis pela imagem da empresa perante seus clientes, como também demandavam mais recursos do ambiente computacional que os demais serviços. De fato, para processar um serviço *on-line* é necessário que todos os componentes do ambiente computacional funcionem corretamente. Foi verificado que mesmo reduzindo o escopo do trabalho o estudo ainda era representativo dos principais problemas tecnológicos que a COPASA-MG enfrentará para migrar o sistema legado em estudo.

Após definido o escopo do estudo, ainda no Capítulo 3, foi apresentada a estratégia de medição, seleção e mapeamento de transações *on-line* do sistema legado para o sistema alvo adotada neste trabalho. Como resultados preliminares foram identificados os principais serviços *on-line* do sistema legado em estudo.

O Capítulo 4 relatou como foram obtidos todos os parâmetros necessários para se modelar a carga de trabalho que o sistema alvo virá a ter em fevereiro de 1999. Para isto foram realizadas medições no sistema legado de onde foi obtida a intensidade da carga de trabalho. Já do sistema alvo foram medidos diversos dados tais como tempo de uso de processador, discos, canal de comunicação, e *links* remotos para os componentes das transações *on-line*, bem como de alguns serviços *batch* e de *ftp* que seriam executados de forma a concorrer com as transações *on-line*. Destes dados coletados dos sistemas legado e alvo foram derivados os parâmetros da carga trabalho do sistema alvo em questão.

O Capítulo 5 apresentou o modelo de simulação que foi desenvolvido para verificar o comportamento do sistema alvo ainda não operacional. Também foi descrito o procedimento que foi adotado para se validar o modelo de simulação, o que implicou no desenvolvimento de um experimento para representar o funcionamento do sistema alvo num ambiente operacional real. Calibrações foram realizadas até que o modelo de simulação pudesse ser classificado como um modelo de simulação validado.

Após modelados o ambiente computacional e a carga de trabalho os modelos de simulação do SICOM/D e do SICOM/CS foram executados gerando os resultados necessários para se verificar se os ambientes computacionais propostos atenderiam aos requisitos de desempenho da COPASA-MG. Adicionalmente, foram comparados os resultados obtidos na simulação dos dois ambientes.

A grande maioria dos problemas que se apresentaram, entretanto, provavelmente também estará presente em muitos sistemas legados que algum dia terão de ser migrados. Mais ainda, além do método, muitas das soluções descritas nesta dissertação são bem genéricas podendo facilmente serem aproveitados em estudos similares.

6.2 Principais Resultados

Considerando a existência dos sistemas legados e a evolução tecnológica pelo qual o mundo está passando, pode-se afirmar que inúmeras organizações enfrentam, ou enfrentarão, o problema de migrar tais sistemas. Assim, o resultado mais importante deste trabalho é a apresentação de um método de análise de desempenho e planejamento de capacidade desenvolvido especificamente para ser usado durante o processo de migração de sistemas legados. De fato, métodos tradicionais de análise de desempenho normalmente não trabalham com dois sistemas distintos, tais como os sistemas legado e alvo. Mais ainda, o método aqui apresentado não exige uma completa compreensão dos sistemas que estão sendo avaliados, como normalmente é requerido pelos métodos tradicionais. Assim, o tempo total para se realizar um estudo de análise de desempenho, usando o método apresentado

neste trabalho, tende a ser menor do que quando usando os métodos tradicionais.

Igualmente importante, é o relato dos problemas enfrentados e das soluções encontradas durante a execução de cada etapa do método, conforme descrito no estudo de caso. De fato, na caracterização da carga de trabalho foi importante o uso do conceito de serviços *on-line* para fazer o mapeamento entre as transações *on-line* dos sistemas legado e alvo. Ainda na caracterização da carga de trabalho, foram descritas as técnicas usadas para se obter os parâmetros do modelo da carga de trabalho. Na calibragem do modelo de simulação surgiram vários problemas práticos tais como a definição do processo de medição de tempo de resposta no ambiente do sistema alvo e a própria estratégia de validação do modelo de simulação que foram detalhadamente descritos.

Considerando o estudo de caso deste trabalho, constata-se que a COPASA-MG sabe hoje, com meses de antecipação, que o sistema computacional que em breve será responsável pelo atendimento de seus clientes será capaz de suprir os serviços sob sua responsabilidade dentro dos níveis de serviço necessários para o bom funcionamento da empresa. De fato, caso o resultado fosse outro, a empresa poderia ter que rever o dimensionamento dos equipamentos adquiridos, bem como o projeto da aplicação antes de efetivamente migrar o sistema. Em ambos os casos este estudo indica quais os componentes do sistema computacional que estão mais próximos de se saturarem (*bottleneck*), que no caso do SICOM/D é o subsistema de processamento, assim como quais os principais serviços *on-line* do SICOM que realmente influenciam na composição da carga de trabalho. No caso da Gerência da Regional Norte (GRNT), foi constatado que o servidor de banco de dados não irá saturar nos próximos 8 anos, estando ainda baixa a utilização do sistema após este período.

Um segundo resultado importante referente ao estudo de caso é o que compara o SICOM/D com o SICOM/CS. De fato, foi verificado que a migração do SICOM/D para o SICOM/CS representa uma melhoria de todas as medidas de desempenho do sistema, mas que tal melhoria não teria um impacto tão significativo para a empresa devido ao fato do desempenho do próprio SICOM/D já ser bem satisfatório. Conhecedora destes fatos, a empresa pode assim cogitar a hipótese de implantar novas aplicações usando do mesmo ambiente computacional do SICOM/D, como pode até cogitar um redimensionamento dos servidores para equipamentos de menor porte, principalmente se estiver nos seus planos tal migração.

6.3 Trabalhos Futuros

A migração de sistemas de informação complexos, segundo [BS95], normalmente demora anos para ocorrer. O próprio estudo de caso deste trabalho é uma prova desta afirmativa, pois está ocorrendo desde 1993.

Considerando o longo período necessário para migrar sistemas legados e a rapidez com que a tecnologia vem evoluindo, é aceitável supor que muitos dos sistemas alvo que estão surgindo já sejam sistemas legados. Por mais dura que seja esta constatação, as orga-

nizações devem estar preocupadas com este processo contínuo de evolução dos sistemas legados para agirem o mais cedo possível de forma a minimizar os esforços de migração. Quanto maior a defasagem tecnológica, maiores os problemas.

No caso do SICOM/D, fica evidente que a aplicação mesmo antes de entrar em operação começa a mostrar sintomas de ser este um *novo sistema legado*. Uma amostra deste problema é a interface do SICOM/D baseada em caracteres. Este tipo de interface poderia ainda ser considerado satisfatório para um produto do início da década de 90, porém é definitivamente desatualizada para os atuais padrões de interfaces. Assim, como explicar que um produto que se diz “novo em folha” seja incapaz de apresentar relatórios de forma gráfica? Provavelmente não seria este um problema grave do sistema alvo, porém não deixa de ser um sintoma de estar o produto tecnologicamente desatualizado. Para minimizar este problema, o ideal seria realizar um estudo para identificar uma forma de evitar que sistemas alvo rapidamente se tornem sistemas legados. Uma solução para este problema seria um processo de migração constante e gradual do sistema, como sugerido em [BS95].

No caso do SICOM, por exemplo, poderia ser cogitado um estudo de como migrá-lo de forma gradual para um ambiente baseado na tecnologia WWW. Assim, os seus usuários passariam a usar *browsers* tais como o Netscape ou Internet Explorer para acessar os serviços da aplicação. Mais do que isto, poderia ser feito um estudo para tornar o atendimento a clientes disponível diretamente através da Internet. Neste caso surgiriam novos aspectos a serem estudados tais como o dimensionamento de servidores WWW e das linhas de acesso Internet.

Um estudo complementar ao estudo do uso da Internet para prover serviços do SICOM aos clientes da COPASA-MG seria o de avaliar o impacto na carga de trabalho existente, pois parte da demanda de atendimento deixaria de ser realizada através dos meios atualmente disponíveis (escritórios comerciais e atendimento telefônico) para ser realizada através da Internet.

Referências Bibliográficas

- [ADA95a] Software AG, Darmstadt, Germany. *ADABAS Command Reference Manual*, May 1995.
- [ADA95b] Software AG, Darmstadt, Germany. *ADABAS UNIX Utilities Manual*, May 1995.
- [ADA97] Software AG, Darmstadt, Germany. *ADABAS Concepts and Facilities Manual, Version 6.2*, August 1997.
- [AMR94] P. Aiken, A. Muntz, and R. Richards. DoD Legacy Systems: Reverse Engineering Data Requirements. *Communications of the ACM*, 37(5):26–41, 1994.
- [BH95] O. Bray and M. Hess. Reengineering a configuration-management system. *IEEE Software*, 12(1):19–23, January 1995.
- [BLS97] J. Bergey, L. Northrop, and D. Smith. Enterprise framework for the disciplined evolution of legacy systems. Technical Report CMU/SEI-97-TR-007, Carnegie Mellon University, Pittsburgh, PA, October 1997.
- [BMW94] T. Biggerstaff, B. Mitbender, and D. Webster. Program understanding and the concept assignment problem. *Communications of the ACM*, 37(5):72–82, 1994.
- [Bor97] Borland International, Scotts Valley, CA. *Borland Delphi 3 for Windows 95 and Windows NT: User's Guide*, 1997.
- [BS95] M. Brodie and M. Stonebraker. *Migrating Legacy Systems: Gateways, Interfaces, and the Incremental Approach*. Morgan Kaufmann, San Francisco, CA, 1995.
- [Com97] Companhia de Saneamento de Minas Gerais, Belo Horizonte, MG. *Documentação do Módulo de Atendimento e Serviços do SICOM – Sistema Integrado Comercial*, 1997.
- [DLR95] C. Delobel, C. Lécluse, and P. Richard. *Databases: From Relational to Object-Oriented Systems*. International Thomson, London, UK, 1995.

- [DV95] G. Dedene and J. De Vreese. Realities of off-shore reengineering. *IEEE Software*, 12(1):34–45, January 1995.
- [FSZ83] D. Ferrari, G. Serazzi, and A. Zeigner. *Measurement and Tuning of Computer Systems*. Prentice Hall, Englewood Cliffs, NJ, 1983.
- [GdSL98] P. Golgher, P. Pinheiro da Silva, and A. Laender. Mapeamento de serviços entre os sistemas hp07 e sicom. Technical report, Convênio COPASA-MG – DCC/ICEx/UFMG, Belo Horizonte, MG, 1998.
- [Gra93] J. Gray. *The Benchmark Handbook: For Database and Transaction Processing Systems*. Morgan Kaufmann, San Francisco, CA, second edition, 1993.
- [HP96] J. Hennessy and D. Patterson. *Computer Architecture: A Quantitative Approach*. Morgan Kaufmann, San Francisco, CA, second edition, 1996.
- [IBM97] IBM Corporation, Poughkeepsie, NY. *MVS JCL Reference*, fourth edition, September 1997.
- [Int96] InterSoft International. Guide to using netterm. WWW:<http://starbase.neosoft.com/zkrr01/html/netterm.html>, 1996.
- [Jai91] R. Jain. *The Art of Computer System Performance: Analysis, Techniques for Experimental Design, Measurement, Simulation and Modeling*. Wiley, New York, NY, 1991.
- [MAD94] D. Menascé, V. Almeida, and L. Dowdy. *Capacity Planning and Performance Modeling: From Mainframes to Client-Servers Systems*. Prentice-Hall, 1994.
- [Mey83] P. Meyer. *Probabilidade: Aplicações à Estatística*. Livros Técnicos e Científicos, Rio de Janeiro. RJ, 1983.
- [MNB⁺94] L. Markosian, P. Newcomb, R. Brand, S. Burson, and T. Kitzmiller. Using an enabling technology to reengineer legacy systems. *Communications of the ACM*, 37(5):58–70, 1994.
- [NEK94] J. Ning, A. Engberts, and W. Kozaczynski. Automated support for legacy code understanding. *Communications of the ACM*, 37(5):50–57, 1994.
- [OV91] M. Özsu and P. Valduriez. *Principles of Distributed Database Systems*. Prentice-Hall, Englewood Cliffs, NJ, 1991.
- [PB94] W. Premerlani and M. Blaha. An approach for reverse engineering of relational databases. *Communications of the ACM*, 37(5):42–49, 1994.

- [PH95] G. Pernul and H. Hasenauer. Combining reverse with forward database engineering - a step forward to solve the legacy system dilemma. In *Proc. of 6th DEXA Conference*, pages 177–186, London, UK, 1995.
- [Pie98] F. Piette. nternet component suite. WWW:<http://www.torry.ru/vcl/internet/ics.zip>, 1998.
- [San95] F. Santos. Introdução à informática. Technical report, Superintendência de Informática da COPASA-MG, 1995.
- [SC294] IBM Corporation, San Jose, CA. *DFSMS/MVS Version 1 Release 2, Using Data Sets*, second edition, June 1994.
- [SC391] IBM Corporation, Mechanicsburg, PA. *CICS/MVS Version 2.1.2 Facilities and Planning Guide*, second edition, March 1991.
- [Sci96a] Scientific and Engineering Software, Austin, TX. *SES/workbench Creating Models*, 1996. Release 3.1.
- [Sci96b] Scientific and Engineering Software, Austin, TX. *SES/workbench Simulating Models*, 1996. Release 3.1.
- [Sci96c] Scientific and Engineering Software, Austin, TX. *SES/workbench Technical Reference*, 1996. Release 3.1.
- [Shn92] B. Shneiderman. *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Addison-Wesley, Reading, MA, second edition, 1992.
- [SRW97] I. Sommerville, J. Ransom, and I. Warren. A method for assessing legacy systems for evolution. Technical report, Lancaster University, Lancaster, UK, 1997.
- [Ste94] W. Stevens. *TCP/IP Illustrated, Volume 1: The Protocols*. Addison-Wesley, Reading, MA, 1994.
- [Tan96] A. Tanenbaum. *Computer Networks*. Prentice-Hall, Upper Saddle River, NJ, third edition, 1996.
- [TD-94] Boole&Babbage, San Jose, CA. *CICS Manager Performance Reporter User Guide*, 3 edition, July 1994.
- [TD-95] Boole&Babbage, San Jose, CA. *CICS Manager Performance Manager User Guide*, 3 edition, June 1995.
- [WCP94] Software AG, Darmstadt, Germany. *Entire Network for UNIX Reference Manual*, October 1994.

- [WTMS95] K. Wong, S. Tilley, H. Müller, and M. Storey. Structural documentation: A case study. *IEEE Software*, 12(1):46–54, January 1995.