

ABORDAGEM ORIENTADA A OBJETOS NO DESENVOLVIMENTO DE SIMULADORES DE REDES LOCAIS

Maria Madalena Dias
FUEM/CTC/DIN - Maringá, Pr

Maria Izabel Cavalcanti Cabral
UFPb/CCT/DSC-COPIN

Marcos Antonio Gonçalves Brasileiro
UFPb/CCT/DEE
58100 - Campina Grande - Pb
Fone: (083)333-1929
FAX : (083)333-1945
E-Mail:copin@brfapesp-bitnet

Sumário

Com a crescente demanda de utilização de Redes Locais de Computadores (RLCs), tem havido uma maior preocupação em avaliar o desempenho dessas redes. Este trabalho apresenta uma Ferramenta de Simulação para Avaliação de Desempenho de Redes Locais, denominada SIMILE. Essa ferramenta projetada para RLCs com Protocolos de Acesso ao Meio conforme Padrão IEEE 802, objetiva, também, servir como protótipo de simulação de RLCs que possa ser facilmente expandido para o estudo e a avaliação de outros protocolos de comunicação. Essa facilidade deve-se às características de modularidade, extensibilidade e reutilização de software propiciadas pela Abordagem Orientada a Objetos utilizada no desenvolvimento do SIMILE.

Abstract

The potencial use of Local Area Networks-LANs in many applications has made, presently, the performance evaluation of those LANs an important research topic. This paper addresses a simulation tool named SIMILE, developed for performance evaluation of LANs, in particular those LANs in IEEE 802 standard. The Object-Oriented Approach is considered in the implementation of SIMILE. As a result, characteristics such as modularity, extensibility and reusability are inherent to this simulation tool. These characteristics make SIMILE a versatile performance evaluation tool that can be easily adapted for performance evaluation of LANs and protocols other than those specified in IEEE 802 standard.

1 Introdução

A avaliação da habilidade de um sistema em suportar a demanda de sua utilização é essencial nas fases de projeto e operacionalidade. Essa demanda é caracterizada pelo tipo de aplicação (tráfego) na entrada do sistema.

Na avaliação de desempenho de um sistema, define-se um conjunto de medidas de desempenho de interesse (relevantes), e observa-se o comportamento destas medidas face às variações na demanda da sua utilização.

A avaliação de desempenho de Redes Locais de Computadores é feita tipicamente por três técnicas que obtêm relações quantitativas entre as medidas de interesse: Simulação Digital, Tratamento Analítico (Teoria dos Processos Estocásticos/Teoria das Filas) e Medição de Desempenho (tempos de resposta, vazão, etc.).

Nas técnicas Simulação Digital e Tratamento Analítico, usadas na avaliação de desempenho de RLCs, se faz necessária a construção de modelos para a observação do comportamento das medidas de desempenho relevantes.

Redes Locais de Computadores são sistemas que apresentam contenção de recursos. Desta forma, modelos de redes de filas se adequam à modelagem destes sistemas.

Apesar do tratamento analítico ser mais econômico e eficiente, muitas vezes sua aplicação é limitada pela complexidade do sistema que está sendo modelado. Para esse tipo de situação, na maioria das vezes, uma solução possível é a Simulação Digital.

A abordagem orientada a objetos facilita a solução de problemas de simulação por fornecer uma forma natural de representar objetos do mundo real. O problema é decomposto em um conjunto de objetos, onde cada objeto representa um objeto do modelo de simulação.

Pode-se observar através de experiências que projetos de simulação são evolutivos por natureza [4]. As características de modularidade, extensibilidade e reutilização de software, propiciadas pela programação orientada a objetos, facilitam a manutenção de programas de simulação.

Este artigo apresenta uma Ferramenta de Simulação Reutilizável para Avaliação de Desempenho de Redes Locais, com Protocolos de Acesso ao Meio conforme Padrão IEEE 802, denominada SIMILE. No desenvolvimento e implementação dessa ferramenta considerou-se a Abordagem Orientada a Objetos.

O SIMILE, projetado para RLCs com Protocolos de Acesso ao Meio conforme Padrão IEEE 802, objetiva, também, servir como protótipo de simulação de RLCs que possa ser facilmente expandido para o estudo e a avaliação de outros protocolos de comunicação. Essa facilidade deve-se às características de modularidade, extensibilidade e reutilização de software propiciadas pela Abordagem Orientada a Objetos.

O SIMILE é, também, parte integrante do módulo solução do projeto denominado "SAVAD - Uma Ferramenta Inteligente para Avaliação de Desempenho de Sistemas Distribuídos" (proposta de pesquisa submetida ao CNPq e, aprovada em abril de 1989, maiores detalhes vide [3]).

Na especificação do SIMILE seguimos etapas apresentadas em [9]. No decorrer dessas etapas usamos representações definidas por outros autores [1, 8, 10].

O restante deste artigo é organizado em quatro seções. A seção 2 descreve sucintamente os modelos utilizados na definição do SIMILE. A seção 3 apresenta a

interface do SIMILE. A seção 4 discorre sobre o projeto orientado a objetos do SIMILE e faz alguns comentários sobre o seu ambiente de desenvolvimento. Finalmente, a seção 5 apresenta a conclusão deste artigo.

2 Descrição dos Modelos

Seguem descrições sucintas dos modelos de Redes Locais com Passagem de Ficha e em Barra com CSMA-CD e, também, a descrição do Modelo do Servidor Único utilizado na avaliação de desempenho de RLCs. Maiores informações sobre tais modelos poderão ser encontradas em [2, 3, 6, 7, 11, 12, 14].

2.1 Modelo de Redes Locais com Passagem de Ficha

O modelo de Redes Locais com Passagem de Ficha pode possuir topologia em anel ou em barra. Esse modelo consiste de um conjunto de interfaces conectadas serialmente por um meio de transmissão e uma ficha livre que é passada entre as interfaces ativas da rede. Quando uma interface detém a ficha livre, ela fica com o controle do meio de transmissão e pode transmitir suas informações.

Uma ficha livre é um sinal de controle compreendido de uma sequência única de sinalização que circula no meio de transmissão, seguindo cada transferência de informação. Ela é responsável pelo controle de acesso ao meio de transmissão.

A figura 2.1 apresenta o modelo de uma Rede Local com Passagem de Ficha em Anel. Nesta figura, f_i ($i = 1, 2, \dots, n$) representa a fonte de geração de pacotes em cada interface i . Cada interface possui um tipo de classe (k , $k \leq n$) e uma fila para armazenamento de pacotes, cujo comprimento (L) pode ser limitado ou ilimitado.

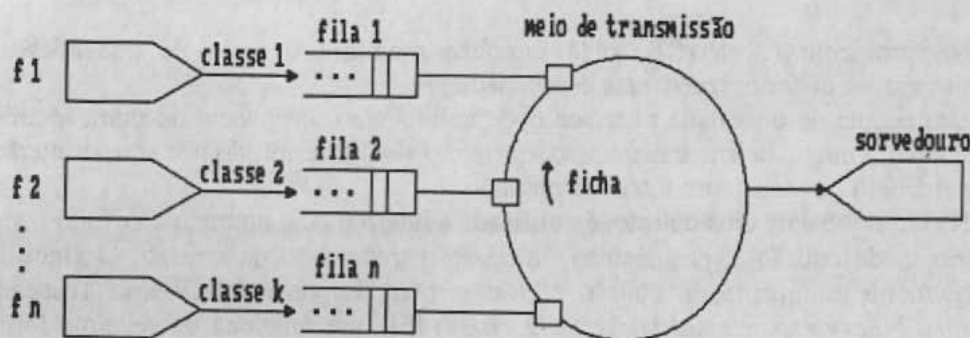


Fig. 2.1 - Rede Local com Passagem de Ficha em Anel

Todo pacote, após ser servido pelo meio de transmissão, entra no nodo sorvedouro, onde é eliminado do modelo, significando que esse pacote foi transmitido pela Rede Local e entregue ao seu respectivo destino.

O funcionamento de Redes Locais com Passagem de Ficha em Barra é similar ao funcionamento de Redes Locais com Passagem de Ficha em Anel. A principal diferença decorre do fato de que a barra não impõe uma ordenação natural entre as interfaces tal como no caso do anel, por este motivo torna-se necessária a criação de um anel lógico ou virtual sobre a estrutura em barra que possibilite a operação do protocolo. O anel lógico no modelo de uma Rede Local com Passagem de Ficha em Barra corresponde ao meio de transmissão do modelo de uma Rede Local com Passagem de Ficha em Anel.

2.2 Modelo de Redes Locais em Barra com CSMA-CD

A figura 2.2 apresenta o modelo de uma Rede Local em Barra com CSMA-CD. Nesta figura, f_i ($i = 1, 2, \dots, n$) representa a fonte de geração de pacotes em cada interface i . Cada interface possui um tipo de classe (k , $k \leq n$) e uma fila para armazenamento de pacotes, cujo comprimento (L) pode ser limitado ou ilimitado.

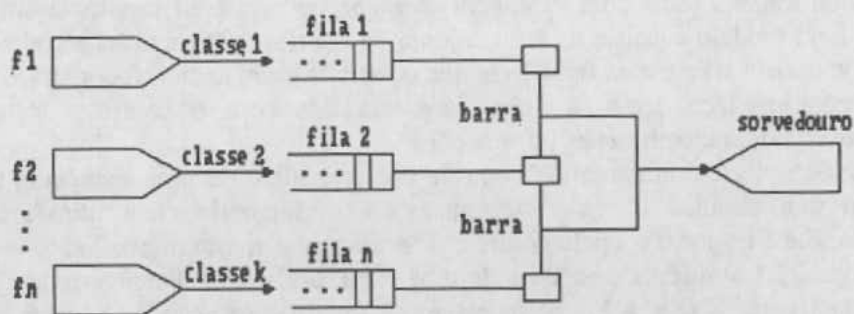


Fig. 2.2 - Rede Local em Barra com CSMA-CD

No protocolo CSMA-CD, cada interface monitora o meio de transmissão e transmite apenas quando o meio está desocupado [7].

Um pacote de uma dada interface pode colidir com um pacote de outra interface durante a parte inicial de sua transmissão, quando o sinal transmitido por aquela interface não é percebido por esta outra interface no meio.

Na ocorrência de uma colisão, é notificada a interferência no meio e ativado o sinal de detecção de colisão e, em seguida, inicia-se o tratamento da colisão. O algoritmo empregado no tratamento de colisão é o algoritmo "Exponencial Binário Truncado" (*truncated binary exponential backoff algorithm*) [6], que funciona da seguinte forma: primeiro, uma sequência de bits chamada *jam* é transmitida para assegurar que a duração da colisão seja suficiente para ser notada pelas outras interfaces transmissoras envolvidas na colisão. Depois que o *jam* é enviado, a transmissão é encerrada e é preparada uma tentativa de retransmissão num tempo aleatoriamente selecionado. Uma vez que repetidas colisões indicam um meio sobrecarregado, as retransmissões são retardadas (*backing off*) para reduzir a carga ao meio. Isto é conseguido expandindo o intervalo no qual o período de espera para retransmissão é selecionado em cada tentativa. A cada tentativa é incrementado o número de tentativas e testado se foi atingido o seu limite. Eventualmente, a transmissão consegue ser efetivada sem colisão, ou as tentativas são

abandonados na hipótese de que o meio falhou ou tornou-se bastante sobrecarregado (caso em que estourará o limite de tentativas) [14].

Todo pacote, após ser servido pelo meio de transmissão, entra no nodo sorvedouro, onde é eliminado do modelo, significando que esse pacote foi transmitido pela Rede Local e entregue ao seu respectivo destino.

2.3 Modelo do Servidor Único

O modelo mais comumente utilizado para a avaliação do desempenho de protocolos de acesso ao meio em redes locais é o modelo de Servidor Único mostrado na Fig. 2.3 [7].

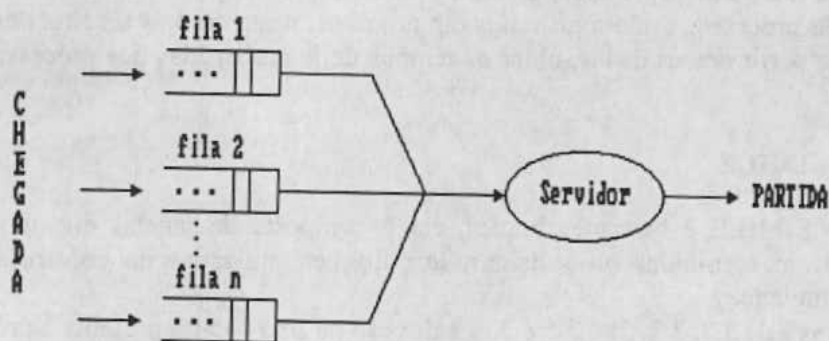


Fig. 2.3 - Modelo do Servidor Único

Neste modelo, o servidor representa o meio de transmissão, os clientes representam pacotes, e a fila i ($i=1,2,\dots,n$) corresponde ao armazenamento de pacotes na interface i .

Um pacote, ao chegar à interface, é transmitido imediatamente se o meio de transmissão estiver disponível ou, caso contrário, aguarda em fila. Este sistema de fila corresponde, em geral, a uma fila G/G/1 [2]. A solução analítica desta fila poderá ser extremamente difícil, dependendo da natureza dos processos de chegada e serviço de clientes. Na inviabilidade de uma solução analítica, a utilização da técnica de Simulação Digital é geralmente recomendada.

A disciplina de atendimento às filas pelo servidor é estabelecida pelos detalhes operacionais do protocolo de acesso ao meio. A seguir apresentam-se algumas considerações sobre as características gerais do modelo do Servidor Único utilizado pelo SIMILE.

A disciplina de liberação de pacotes em cada interface pode ser: a) não exaustiva: transmissão de apenas um pacote por vez; b) exaustiva: transmissão de todos os pacotes existentes na fila da interface; c) limitada: transmissão de um número "x" de pacotes ($x \leq L$) cada vez que a interface detém o meio de transmissão.

A disciplina de escalonamento das filas das interfaces pode ser de um dos seguintes tipos: FCFS (*First Come First Served*), LCFS (*Last Come First Served*) e Aleatória.

O modelo do servidor único, aqui definido, prevê a utilização de multiclasses. Uma classe representa o tipo de informação que é transmitida na rede, que pode ser voz, dados, imagens, etc. Esse modelo prevê, também, os mecanismos de prioridade adotados no Padrão IEEE 802 [6, 12].

E, para finalizar a especificação do modelo, o usuário define as distribuições dos tempos médios de interchegadas de pacotes e dos tempos médios de serviço (tempo médio de transmissão de pacotes no meio de comunicação). O usuário pode escolher para cada classe uma das seguintes funções de distribuição de probabilidade previstas no modelo: Uniforme, Exponencial, Normal e Geral (definida pelo usuário).

O SIMILE também permite que os tempos de interchegadas dos pacotes sejam definidos previamente e gravados em um arquivo, neste caso o usuário informa o nome desse arquivo. Em algumas aplicações de controle de processos, por exemplo, os tempos de ocorrência de processos estão registrados em arquivos, neste caso, o sistema deve ter facilidades de, a partir desses dados, obter os tempos de interchegadas dos processos em consideração.

3 Interface do SIMILE

A interface do SIMILE é bastante simples, ela é composta de janelas que mostram opções para serem escolhidas ou solicitam informações utilizadas na construção do modelo a ser simulado.

Nas figuras 3.1, 3.2, 3.3, 3.4, 3.5 e 3.6, a direção da seta (-->) representa a próxima janela que será aberta após a entrada da informação solicitada pelo SIMILE ou após a escolha da opção de menu na janela anterior. A escolha de uma opção pode ser feita através das setas de direção e acionando a tecla <ENTER> ou apenas teclando a 1ª letra ou número da opção.

A janela (1) da fig. 3.1 é aberta quando o SIMILE é executado, nela o usuário é questionado se deseja ou não carregar um arquivo. Se ele responder que sim, então é aberta a janela (2), onde é solicitado o nome do arquivo que contém as informações necessárias à simulação do modelo do usuário. Caso contrário, é dado início à entrada dessas informações.

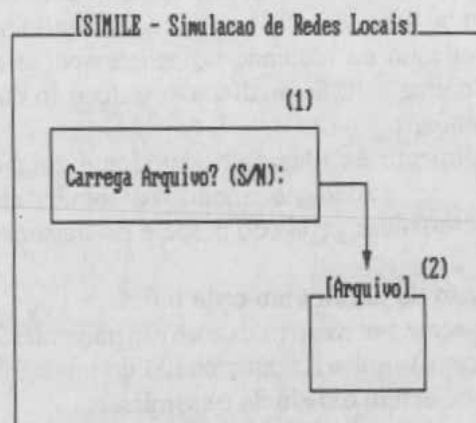


Fig. 3.1 Início de Execução do Simulador

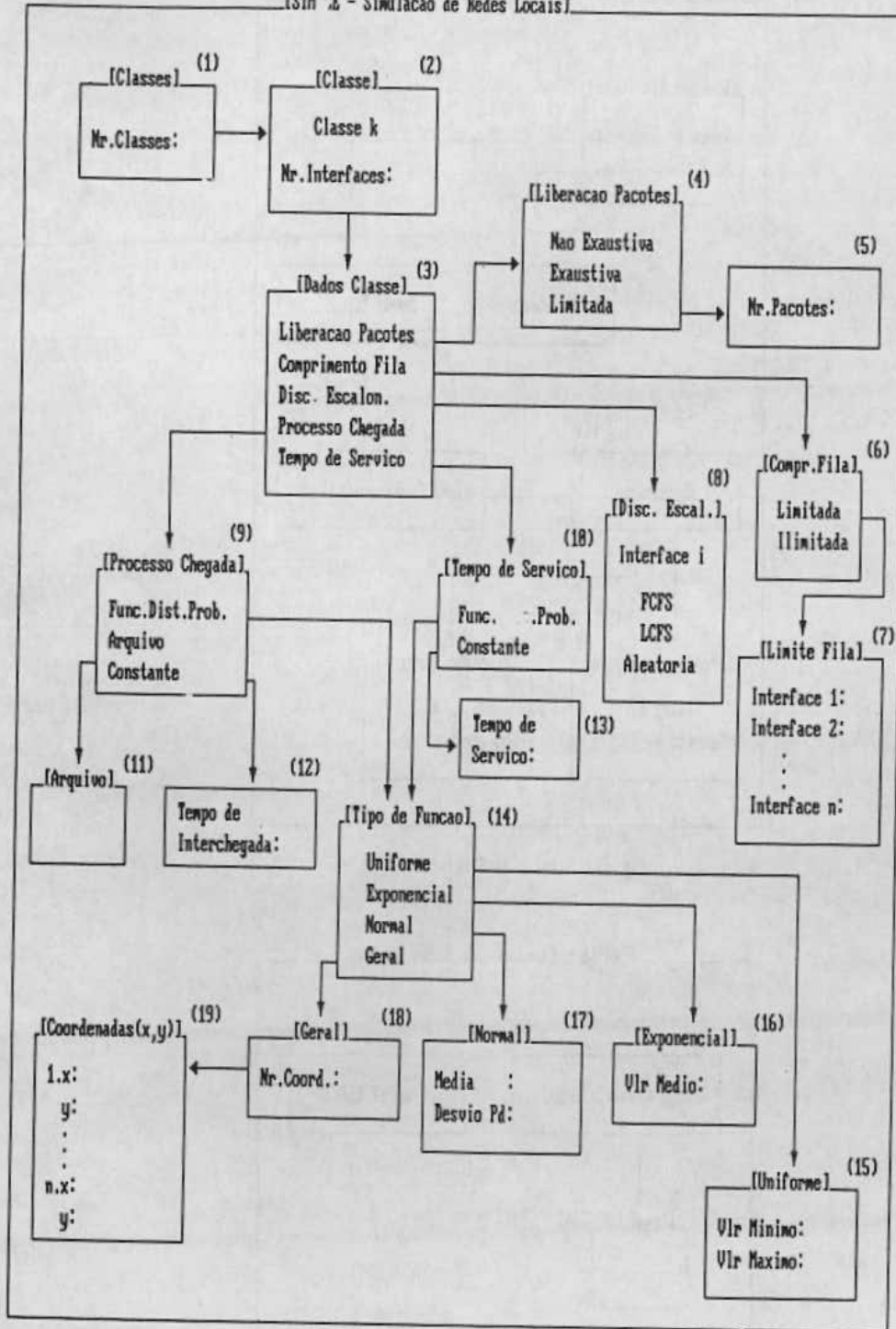


Fig. 3.2 Informacoes sobre as Classes Simuladas

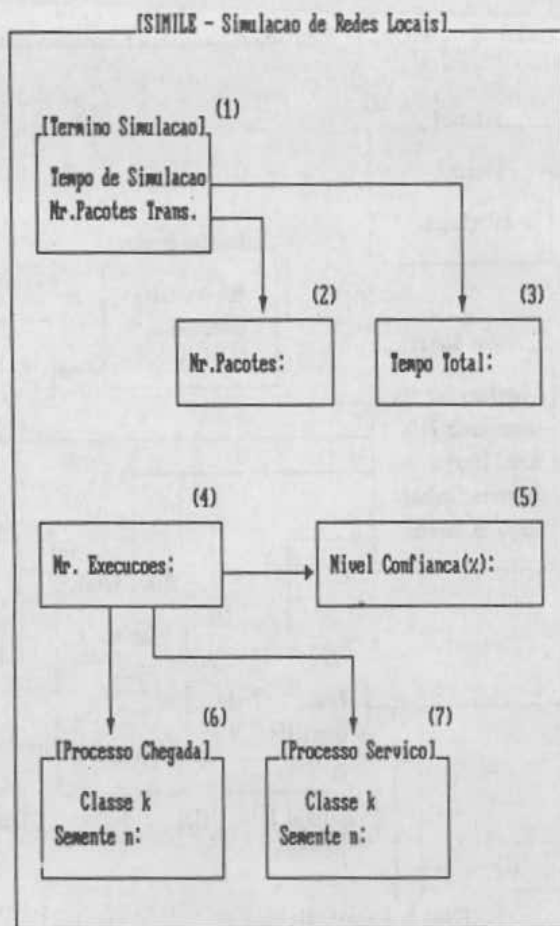


Fig. 3.5 Termino da Simulacao

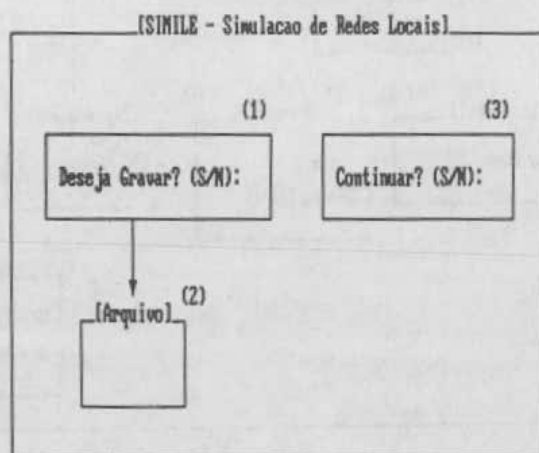


Fig. 3.6 Fim da Entrada de Dados

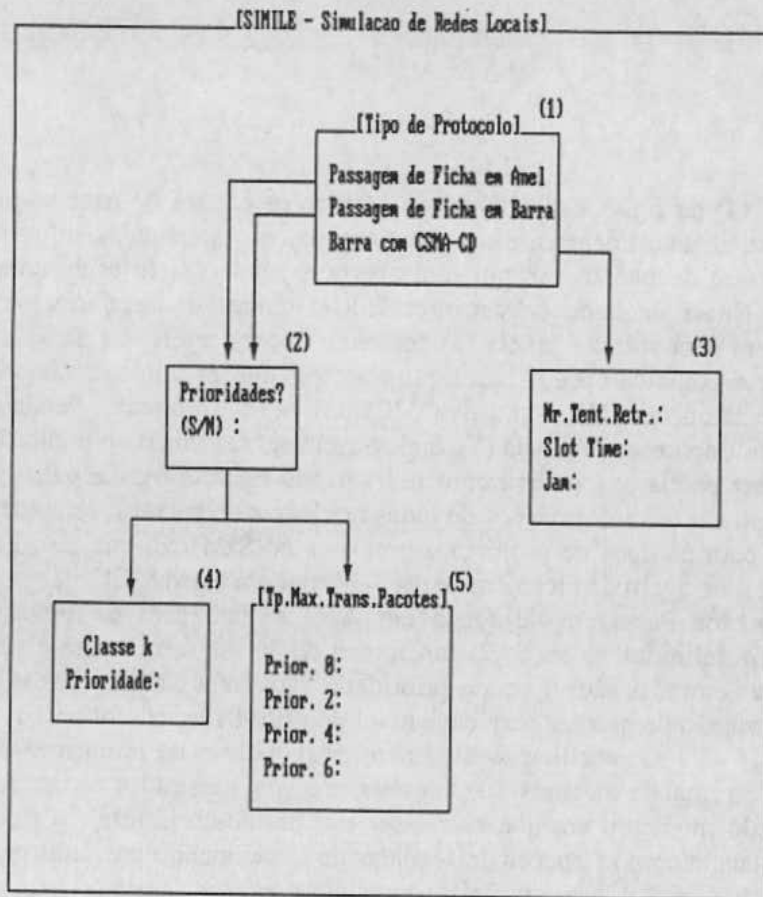


Fig. 3.3 Informacoes sobre os tipos de protocolos

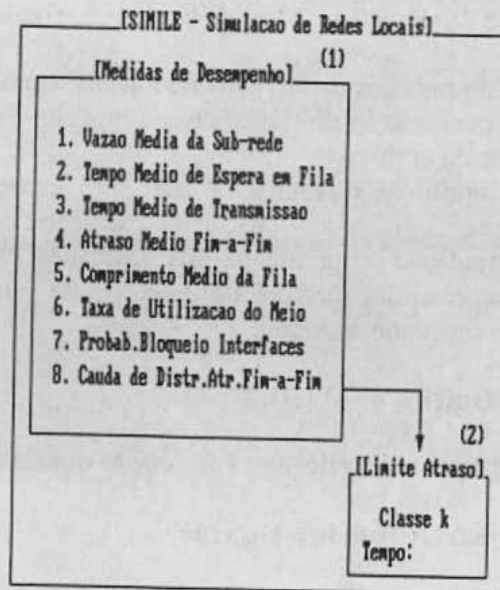


Fig. 3.4 Relacao das Medidas de Desempenho

Na janela (1) da fig. 3.2 é solicitado o número de classes de pacotes (n) previstas no modelo (voz, dados, imagens, etc.). Em seguida, são solicitadas informações para cada tipo de classe de pacote. Assim, será aberta a janela (2) (k representa o tipo de classe, $k \leq n$). Nessa janela deverá ser informado o número de interfaces pertencentes à classe k . A seguir será aberta a janela (3) contendo opções referentes às informações da classe k . Com a escolha da opção "Liberacao Pacotes", por exemplo, será aberta a janela (4), que contém as opções "Não Exaustiva", "Exaustiva" e "Limitada". Sendo escolhida a opção "Limitada", aparecerá a janela (5), onde deverá ser informado o número de pacotes que cada interface da classe k poderá transmitir quando ela detém o meio de transmissão.

Após a entrada das informações de todas as classes de pacotes, será aberta a janela (1) da fig. 3.3 com os tipos de protocolos previstos no SIMILE, que são: Passagem de Ficha em Anel, Passagem de Ficha em Barra e Barra com CSMA-CD.

Os protocolos Passagem de Ficha em Anel e Passagem de Ficha em Barra, conforme foram definidos na seção 2, funcionam de forma semelhante e suas entradas são iguais. Estas entradas são: nível de prioridade para cada classe e limite máximo de tempo de transmissão de pacotes para cada nível de prioridade, conforme fig. 3.3. Para o protocolo CSMA-CD são solicitados o número de tentativas de retransmissão, o tempo de propagação do sinal de transmissão (*slot time*) e o *jam*, mostrados na fig. 3.3.

A partir do momento em que as classes e o protocolo já estejam definidos, será mostrada uma janela com as opções de Medidas de Desempenho previstas pelo SIMILE, conforme mostra a fig. 3.4, para que sejam escolhidas aquelas desejadas pelo usuário.

Em seguida, na janela (1) da fig. 3.5, é solicitado o tipo de Término da Simulação, que pode ser pelo "Tempo de Simulação" ou por "Total de Pacotes Transmitidos".

O usuário poderá executar o simulador várias vezes, neste caso, ele deverá informar o número de execuções desejadas, as sementes para o processo de chegada, as sementes para o processo de serviço e o nível de confiança, conforme apresentados na fig. 3.5

E, para finalizar a entrada dos dados, é aberta a janela (1) da fig. 3.6 onde o usuário é questionado se deseja gravar as informações do seu modelo. Se ele responder que sim, então é solicitado o nome do arquivo.

O usuário tem a opção de continuar ou não com a execução do programa de simulação. Se o usuário confirmar a continuação da execução do programa, então é dado início a execução da simulação propriamente dita e, quando esta termina, é feita uma análise dos resultados, apenas das medidas de desempenho solicitadas pelo usuário. O resultado dessa análise é mostrado no vídeo.

4 Projeto Orientado a Objetos do SIMILE

Essa seção apresenta aspectos mais relevantes do projeto orientado a objetos do SIMILE.

4.1 Introdução ao Projeto Orientado a Objetos

O projeto orientado a objetos, como outras metodologias de projeto orientado a informação, cria uma representação do domínio do problema do mundo real e mapeia este problema em um domínio de solução que é o software [9].

O projeto de um sistema de software orientado a objetos é construído em torno de um conjunto de classes¹ que caracterizam o comportamento dos objetos do mundo real relativos ao sistema. Em geral, uma classe é a descrição de um conjunto de objetos que possuem os mesmos atributos e comportamentos e, um objeto é uma instância específica de uma classe com comportamento bem definido. Os objetos de cada classe são manipulados através do envio de mensagens. Estas mensagens representam o conjunto de ações que são tomadas sobre o conjunto de objetos.

O primeiro passo, e o mais importante, no projeto orientado a objetos é a identificação das classes de objetos que compõem o sistema do mundo real. O segundo é a definição das mensagens que estabelecem as interfaces dos objetos [8].

4.2 Classes de Objetos

A partir dos modelos descritos na seção 2 foram identificadas as classes de objetos do simulador. As principais classes são sucintamente descritas a seguir:

- 1) Controle: O objeto desta classe controla todo o processo de simulação (inicialização, processamento e apresentação dos resultados).
- 2) Protocolo: É uma classe virtual que engloba os aspectos gerais dos protocolos de acesso ao meio, conforme padrão IEEE 802.
- 3) PassagemFicha: As operações desta classe definem o controle da simulação quando os protocolos de acesso ao meio são Passagem de Ficha em Anel e Passagem de Ficha em Barra.
- 4) Csmacd: As operações desta classe definem o controle da simulação quando o protocolo de acesso ao meio é em Barra com CSMA-CD.
- 5) Evento: Contém informações sobre cada evento.
- 6) ListaEventos: As operações desta classe permitem o gerenciamento dos eventos que ocorrem durante a simulação. Contém apontadores para os eventos.
- 7) Ficha Livre: Define a ficha livre que é passada de interface a interface nos protocolos Passagem de Ficha em Anel e Passagem de Ficha em Barra.
- 8) Classe: Contém informações gerais sobre cada classe de pacotes (que pode ser: voz, dados, imagens, etc.).
- 9) ListaClasses: Contém apontadores para as classes.
- 10) Interface: Contém informações sobre cada interface.
- 11) ListaInterfaces: Contém apontadores para as interfaces.
- 12) Pacote: Contém informações sobre cada pacote.
- 13) FilaPacotes: As operações desta classe permitem o controle da entrada e da saída de pacotes nas filas das interfaces. Contém apontadores para os pacotes.
- 14) MeioTrans: Define o meio de transmissão.
- 15) Relógio: Define o tempo simulado.

1. O termo classe ou classe de objetos usado nesta seção representa um conjunto de objetos com as mesmas características. O termo classe de pacotes diz respeito aos tipos de pacotes (voz, dados, imagens, etc.) que são especificados nos modelos de RLCs.

- 16) AnelLógico: Controla a ordem de passagem de ficha entre as interfaces, quando o protocolo sendo simulado é do tipo Passagem de Ficha em Barra. Contém apontadores para as interfaces.
- 17) Função: Operações desta classe permitem a geração de amostras de intervalos de tempo segundo funções de distribuição de probabilidade definidas no SIMILE.
- 18) TWindow: Esta é uma classe definida pela Zortech. Ela controla a abertura, edição e fechamento de janelas.
- 19) Menu: Esta é uma classe definida pela Zortech. Ela controla a escolha de opções em um menu.

4.3 Modelo Cliente-Servidor

O projeto de um sistema orientado a objetos não é realizado apenas em termos dos objetos, mas também dos serviços que eles fornecem a outros objetos. O modelo cliente-servidor dá uma visão geral do relacionamento entre os objetos. Nesse modelo cada objeto interage com os outros objetos através de mensagens que passam ou buscam informações, solicitam aos objetos a implementação de um procedimento, etc. [10].

As figuras 4.1 e 4.2 representam o modelo cliente-servidor do SIMILE. Essas figuras usam as seguintes convenções:

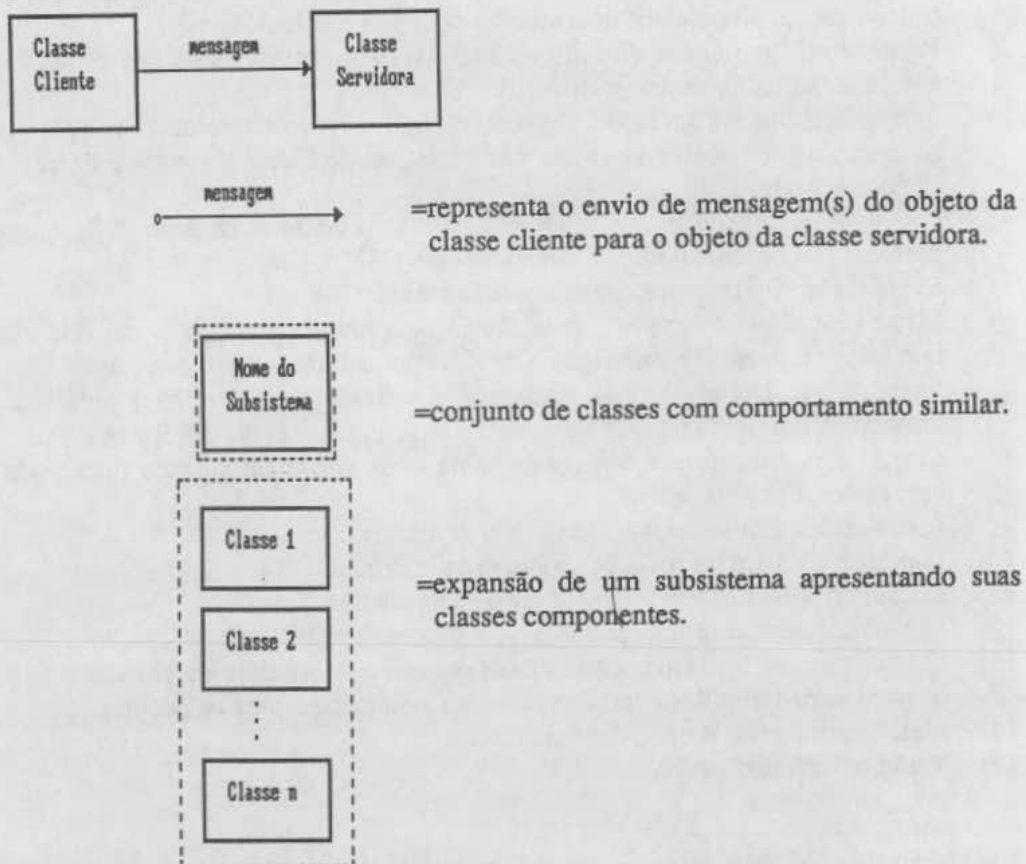
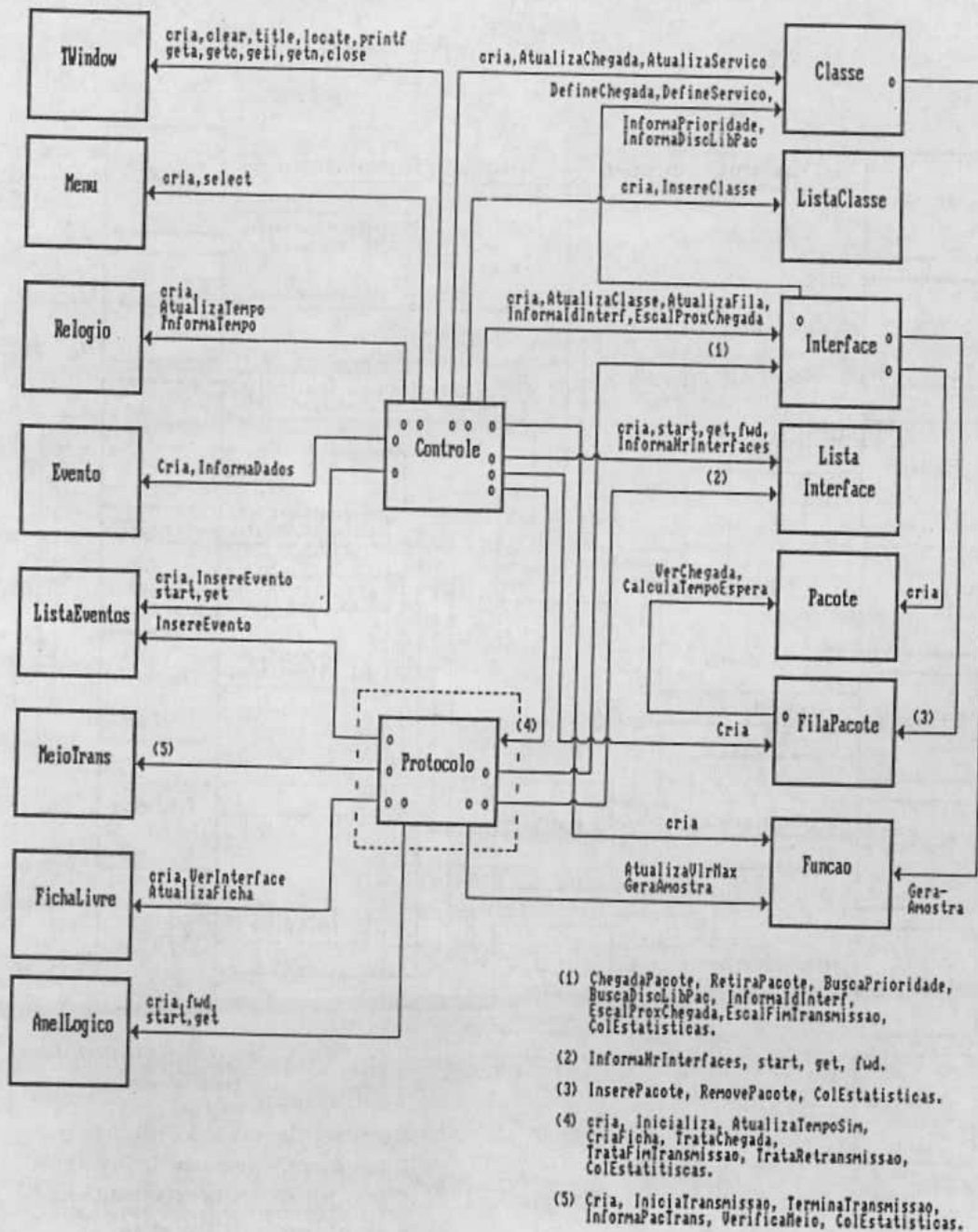


Fig. 4.3 Notação do Modelo Cliente-Servidor



4.1 Modelo Cliente-Servidor

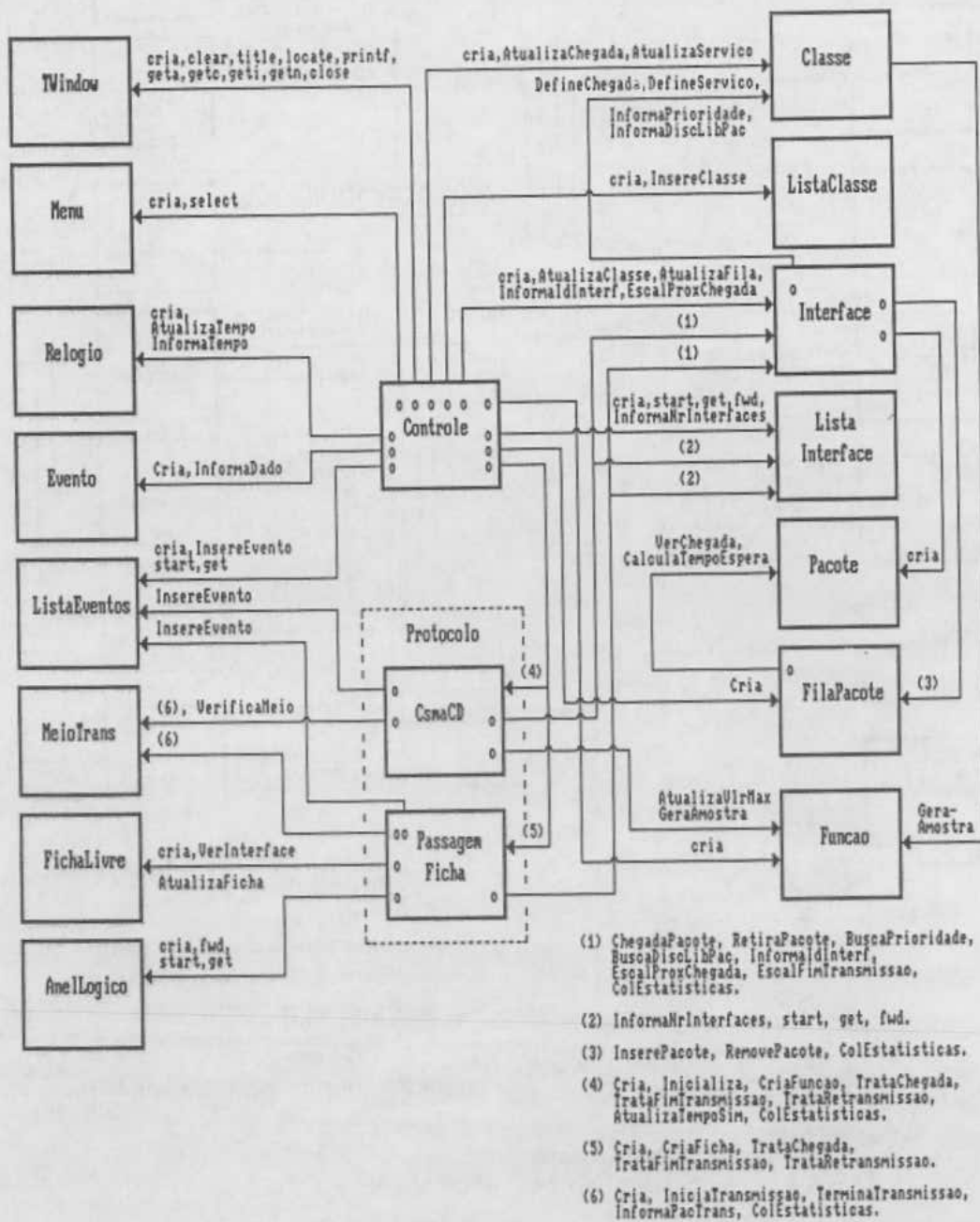


Fig. 4.2 Modelo Cliente-Servidor

Na fig. 4.1 o subsistema "Protocolo" agrega as classes que definem os protocolos de acesso simulados pelo SIMILE, no caso, PassagemFicha e CdmaCD.

A fig. 4.2 apresenta o modelo cliente-servidor da fig. 4.1 com expansão do subsistema "Protocolo", mostrando suas classes componentes e seus relacionamentos específicos com os outros objetos do sistema. A classe Protocolo é uma classe virtual que é herdada pelas classes PassagemFicha e CdmaCD, ela gerencia algumas funções que são independentes do tipo de protocolo simulado. Pode-se observar que os objetos das classes PassagemFicha e CdmaCD são independentes, isto é, objetos da classe PassagemFicha não se comunicam com objetos da classe CdmaCD e ambos se relacionam com os demais objetos do sistema. Nota-se, também, que as mensagens enviadas pelos objetos das classes PassagemFicha e CdmaCD são, na grande maioria, as mesmas, havendo assim, uma reutilização de software.

Com exceção das classes "TWindow", "Menu", "FichaLivre" e "Anel Lógico", todas as classes apresentadas nas figuras 4.1 e 4.2 podem constar em projetos de simuladores de redes locais por serem independentes do tipo de protocolo de acesso ao meio utilizado. As classes "TWindow" e "Menu" são necessárias para a interface do sistema. A classe "FichaLivre" é somente utilizada na simulação do Protocolo com Passagem de Ficha e a classe "Anel Lógico" também é somente utilizada quando o protocolo simulado é do tipo Passagem de Ficha em Barra.

Com a estrutura modular apresentada no projeto orientado a objetos do SIMILE, a implementação de um novo protocolo poderá ser feita através da inclusão de uma nova classe de objetos ao protótipo do SIMILE e da definição dos relacionamentos entre os novos objetos e os já existentes, sem modificar a estrutura básica do projeto. Além do mais, as classes já implementadas poderão ser, também, utilizadas pelo novo protocolo, caracterizando aqui a reutilização de software destacada neste artigo.

Como exemplo podem-se citar as redes ópticas, capazes de suportar aplicações heterogêneas (dados, voz, imagens, etc.) [5]. A simulação dessas redes poderá ser realizada pelo SIMILE apenas com a inclusão de classes de objetos que definem as características específicas de funcionamento dos protocolos de acesso ao meio dessas redes.

No que diz respeito à interface do SIMILE, o novo protocolo deverá ser incluído na janela (1) da fig. 3.3. Também outras janelas poderão ser definidas para a entrada das informações específicas desse protocolo, caso necessário.

A implementação do SIMILE foi feita na linguagem de programação C++ [13], utilizando os recursos dessa linguagem e as facilidades oferecidas pela programação orientada a objetos.

5 Conclusão

O SIMILE apresenta uma interface bastante simples e uma estrutura modular e reutilizável. Ele foi projetado para avaliar o desempenho de Redes Locais com Protocolos de Acesso ao Meio, conforme Padrão IEEE 802 e encontra-se, atualmente, em fase de validação.

As características de modularidade, extensibilidade e reutilização de software apresentadas na Abordagem Orientada a Objetos e utilizadas no desenvolvimento do SIMILE facilitam, aos projetistas de Redes Locais, o estudo de novos protocolos de

acesso ao meio, uma vez que a inclusão de um novo protocolo ao protótipo implica, apenas, no acréscimo de classes de objetos e na definição de relacionamentos entre os novos objetos com os já existentes, podendo o novo protocolo reutilizar as classes de objetos já implementadas no SIMILE.

Referências Bibliográficas

- [1] Booch, G. Object Oriented Design. The Benjamin/Cummings Publishing Company, Inc., 1991.
- [2] Brasileiro, M.A.G., Moura, J.A.B., Cabral, M.I.C. Avaliação de Desempenho da Integração Voz e Dados em Redes Locais com Passagem de Ficha. *7o. Simpósio Brasileiro de Telecomunicações*. Florianópolis, 1989.
- [3] Cabral, M.I.C., Brasileiro, M.A.G. e Silva, H.M. SAVAD - Uma Ferramenta para Avaliar o Desempenho de Sistemas Distribuídos. *Simpósio Franco-Brasileiro em Sistemas Distribuídos*. Florianópolis, 1989.
- [4] Doyle, R.J. Object-Oriented Simulation Programming. *SCS Multiconference on Object-Oriented Simulation*. San Diego, Califórnia, 1990.
- [5] Galdino, J.F., Giozza, W.F. Redes Locais de Altíssima Velocidade. *9o. Simpósio Brasileiro de Telecomunicações*. São Paulo, 1991.
- [6] IEEE. Project 802. Local Area Network. IEEE Computer Society. IEEE 802.3 - CSMA/CD, IEEE 80.4 - Token Bus, IEEE 802.5 - Token Ring, IEEE 802.2 - LLC, 1985.
- [7] Moura, J.A.B., Sauv e, J.P., Giozza, W.F. e Ara ujo, J.F.M. Redes Locais de Computadores (protocolos de alto n vel e avalia  o de desempenho). Editora McGraw-Hill, S o Paulo, 1986.
- [8] Pinson, L.J., Wiener, R.S. An Introduction to Object-Oriented Programming and Smalltalk. Addison-Wesley Publishing Company, 1988.
- [9] Pressman, R.S. Software Engineering. Second Edition, McGraw-Hill International Editions, 1987.
- [10] Sellers, B.H., Edwards, J.M.. The Object-Oriented Systems Life Cycle. *Communication of the ACM*. Vol. 33, No. 9, 1990.
- [11] Soares, L.F. Modelagem e Simula  o Discreta de Sistemas. VII Escola de Computa  o, S o Paulo, 1990.
- [12] Soares, L.F. Redes Locais. Editora Campus, 1990.
- [13] Stroustrup, B. The C++ Programming Language. Reading. Mass: Addison-Wesley, 1986.
- [14] Tarouco, L.M.R. Redes de Computadores (Locais e de Longa Dist ncia). McGraw-Hill. 1986.