

REALIZAÇÃO DE UM SISTEMA ORIENTADO A OBJETOS PARA O ENSINO DE REDES DE COMPUTADORES

Guy BARROSO, Luís ARAÚJO JR, Régis SIMÃO, Mauro OLIVEIRA

**LAR - Laboratório Multiinstitucional de Redes
e Sistemas distribuídos
UECE/ETFCE/UFC**

Av. Paranjana, 1700 - Campus do Itaperi
Fortaleza/CE - CEP: 60.740-000
e-mail: DF14@C53000.PETROBRAS.ANRJ.BR

SUMÁRIO

MinhocO é uma nova versão de um sistema destinado ao ensino de redes de computadores, cuja concepção e implementação são baseadas na orientação a objetos. Este sistema faz uso de técnicas de prototipagem, sendo constituído de uma rede local implementada completamente em software ("zero slot LAN") e de um método de utilização.

Neste artigo descrevemos as características do sistema Minhoco, sua arquitetura, e as consequências que a implementação desta versão numa linguagem orientada a objetos trouxe, tanto para a interface homem-máquina do sistema, quanto para os momentos pedagógicos do método proposto.

1. INTRODUÇÃO

O objetivo deste artigo é mostrar a evolução de um sistema de ensino de redes, baseado em computador, e a contribuição trazida pela abordagem de sua nova versão orientada a objetos tanto na sua concepção como na sua implementação em C++. Este sistema, denominado MinhocO, foi projetado para uso conjunto com um método

de ensino. O processo ensino/aprendizagem baseado neste método é realizado através da demonstração de conceitos¹ (onde a funcionalidade de cada componente do sistema é justificada) e pela disponibilidade do MinhocO como sistema de desenvolvimento.

As características adicionais que a programação orientada a objetos traz para o método de ensino proposto para o sistema MinhocO são apresentadas neste artigo. São também identificados aspectos que vão ao encontro do Método, na forma de vantagens e potencialidades.

Na seção 2 identificamos o ambiente no qual o MinhocO - Método e Arquitetura - está inserido: descrevemos sua evolução, comentamos sua fundamentação pedagógica e detalhamos suas características.

A seção 3 apresenta a nova interface do sistema MinhocO implementado, enfatizando a importância da interface com o usuário de um sistema de ensino baseado em computador, através de algumas teorias e diretrizes práticas para o projeto da interface homem-máquina em sistemas interativos. Critérios relativos à qualidade do software orientado a objetos, que influenciam o Método, são considerados. As características principais do ambiente são gradativamente delineadas. Esta seção apresenta ainda exemplos de algumas atividades possíveis com o sistema apresentado, a fim de ilustrar os momentos pedagógicos do Método no contexto da abordagem orientada a objetos.

Em conclusão é destacada a relevância do sistema como um produto já validado, sua potencialidade enquanto ambiente de pesquisa, bem como seu caráter multidisciplinar.

2. O CONTEXTO

2.1. HISTÓRICO

A rede Minhoca [OLIV87, COEL89] foi a primeira implementação de uma filosofia pedagógica na qual se inscreve o sistema MinhocO. A partir de então foram desenvolvidas várias versões [OLIV91]. O sistema teve sua arquitetura revista dentro de uma concepção orientada a objetos [OLIV92a, ALME93] cujo protótipo [OLIV93b] foi desenvolvido no Laboratório MASI da Universidade Paris 6, na França.

¹ A informação veiculada de forma diretiva que caracteriza o ensino tradicional, e substituída, neste método, pela detecção de sua necessidade por parte do aluno.

A versão orientada a objetos tratada neste artigo acrescenta vantagens ao Método MinhocO, mantendo, no entanto, toda a filosofia e estrutura de base das versões anteriores: uma rede local completamente implementada em software.

2.2. FUNDAMENTAÇÃO PEDAGÓGICA

A psicologia cognitiva [PIAG75] estuda, através do método experimental, as atividades do indivíduo relacionadas ao seu conhecimento do mundo (percepção, representação figurativa, compreensão e produção da linguagem, etc.). Dentro desta abordagem o sujeito forma seus conhecimentos numa interação permanente entre o objeto e suas representações mentais. O fator fundamental do desenvolvimento cognitivo é a "tentação" de restaurar o equilíbrio perdido por esta interação. O conhecimento passa assim de um estado de desequilíbrio a outro, por fases transitórias durante as quais os conhecimentos anteriores são postos em dúvida. Se este momento de desequilíbrio for superado, acontece aí uma reorganização de conhecimentos, durante a qual novas aquisições são integradas ao saber antigo [ROGA87].

No comportamentalismo ("behaviorism"), teoria que se contrapõe a abordagem cognitiva, o motor do processo de aprendizagem é o "reinforcement", ou seja, o ato de recompensar o comportamento do indivíduo [SKIN68]. Assim, dentro de circunstâncias idênticas, uma ação que produz um resultado desejável será repetida por um indivíduo (ou animal) desde que o resultado seja também desejável e, ao contrário, uma consequência nefasta de uma ação o conduzirá a evitar esta ação. Numa visão mais comportamentalista, a atividade de aprendizagem consiste em programar o ambiente do estudante passo a passo.

Ao observarmos estas duas teorias de aprendizagem (o comportamentalismo e o cognitivismo) quando relacionadas ao ensino por computador, constatamos que, se por um lado o comportamentalismo é um processo simples e eficaz, por outro lado ele tende a considerar o computador como uma "máquina de ensinar" que conduz o aluno num esquema pré-estabelecido. Já a abordagem cognitiva conduz a uma visão do computador como um "terreno de aprendizagem" onde as possibilidades de experimentações as mais complexas, aceleram e viabilizam a aquisição do saber, permitindo ao aluno explorar o seu potencial cognitivo. Esta noção de "terreno de aprendizagem", apesar de atraente, é um conceito difícil de formalizar e por em prática [OLIV93a].

A partir desta reflexão ressaltamos dois aspectos que influenciam o Método MinhocO: o papel da motivação no processo de aprendizagem de redes de computadores (nem sempre presente no ensino tradicional) e a importância do trabalho cooperativo como um forte componente cognitivo neste processo. O fato do sistema MinhocO ser baseado numa rede local real (e não numa simulação monousuária) adiciona potencialidades importantes para um trabalho cooperativo.

2.3. PRINCÍPIOS E CARACTERÍSTICAS DO MÉTODO

O método utilizado no sistema MinhocO é composto por quatro momentos pedagógicos (Sensibilização, Análise, Síntese e Avaliação). Estes momentos pedagógicos respeitam os seguintes princípios:

MOTIVADOR: A pedagogia proposta no Método está centrada no aluno (i.e., favorece a abordagem cognitiva).

MODULAR: Permite a análise e a construção estruturada de um sistema de rede, i.e., cada problema particular (e possivelmente suas soluções) é analisado à medida que se constrói uma rede real (MinhocO não é uma simulação).

COOPERATIVO: Propõe a utilização tanto pelo professor (para demonstrar os assuntos de uma maneira prática), quanto pelos alunos (para pôr em prática as experiências propostas ou para desenvolverem suas próprias idéias) mas sobretudo permitindo a resolução coletiva dos problemas.

SIMPLES: Fácil instalação do Sistema e utilização do Método, além de fazer uso de uma interconexão simples e de máquinas bastante difundidas.

Durante o desenrolar dos momentos pedagógicos que agrupam as atividades do Método proposto na utilização do MinhocO, a postura dos agentes professor e aluno são invertidas, ou seja, elas evoluem no sentido de que o aluno passa de uma participação passiva a ativa, enquanto o oposto ocorre com o professor [OLIV93a].

A seguir descrevemos, de forma sumária, cada um desses momentos pedagógicos:

SENSIBILIZAÇÃO: Aprendizagem por demonstração.

A idéia neste momento pedagógico é motivar os alunos através de demonstrações práticas, permitindo aos mesmos uma visão macro sobre os temas abordados.

ANÁLISE: Aprendizagem através de um esforço de abstração.

Este momento pedagógico objetiva identificar problemas pontuais e buscar possíveis soluções clássicas ou não para estes problemas. Para tanto, o sistema dispõe de vários protótipos com níveis de complexidade gradativos, que acumulam funções (no sentido OSI) em direção a uma rede de computadores completa. O aluno tanto pode analisar estes protótipos, como também recompilar parte deles (testando assim suas dúvidas), de acordo com a orientação do professor.

SÍNTESE: Aprendizagem pela realização concreta.

Neste momento, MinhocO é disponível como um sistema de desenvolvimento onde o aluno poderá implementar experimentos de laboratório (protocolos, interfaces, algoritmos, etc).

AVALIAÇÃO: Aprendizagem pela associação ao mundo real.

Espera-se neste momento que o aluno seja capaz de extrapolar os conceitos aprendidos dentro do contexto mais restrito que é o sistema MinhocO, associando-os a produtos e normas existentes. Esta extrapolação do "mundo MinhocO" deve fornecer ao aluno parâmetros que lhe permitem avaliar características de protocolos e serviços proprietários ou normalizados.

2.4. A ARQUITETURA DO SISTEMA MINHOCO

A técnica de resolução de problema por prototipagem apresenta dois aspectos no contexto do Método MinhocO. O primeiro é o uso da programação como um mecanismo para a resolução de problemas, que se verifica no terceiro momento pedagógico do Método (Síntese). O segundo aspecto se manifesta durante toda a utilização do sistema: o conceito de protótipo é usado como uma bancada de testes permitindo a realização de soluções intermediárias que conduzirão a uma solução satisfatória.

A arquitetura proposta para o sistema MinhocO é constituída de um Gerador de Protótipos MinhocO (GPM), de um módulo de objetos de

base (MOB), e de um conjunto de protótipos para serem utilizados de acordo com o método proposto (Fig. 1).

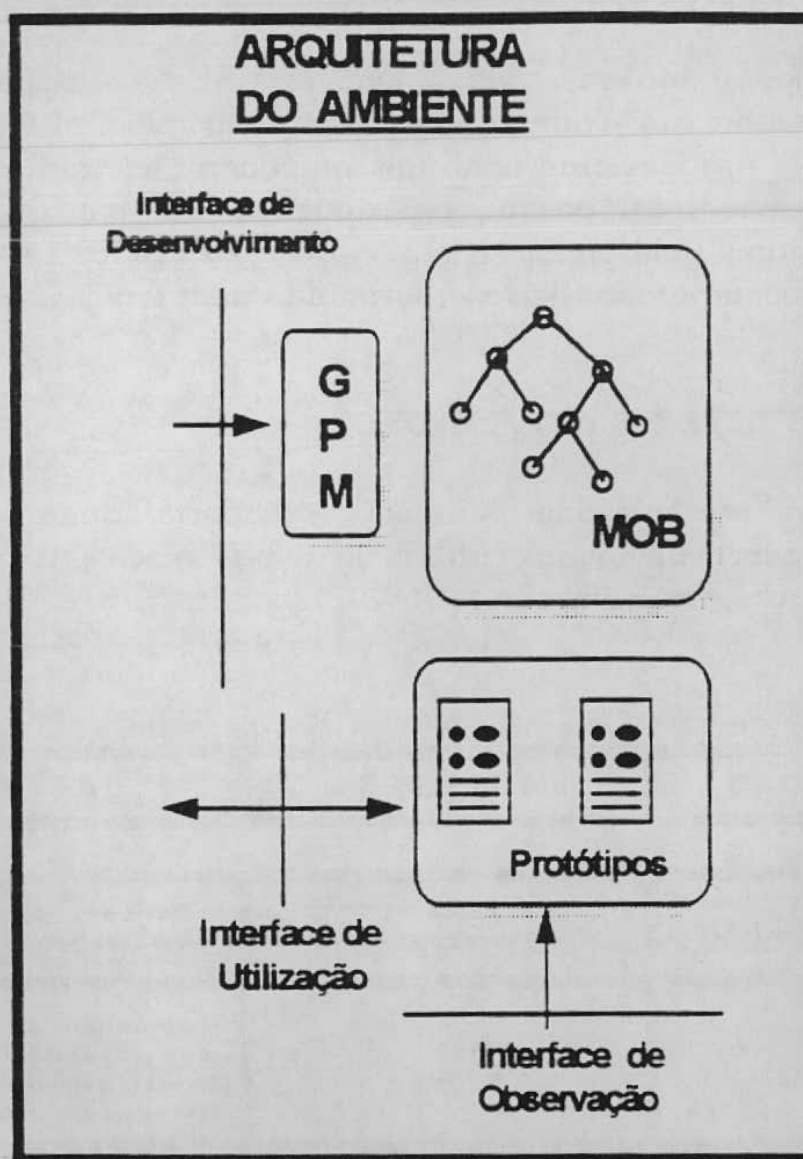


Figura 1: Arquitetura do Ambiente MinhocO

O módulo GPM [LALA93] é uma ferramenta que permite a criação e configuração (instanciação e supressão) de protótipos de rede. O professor pode desenvolver com facilidade suas próprias idéias (por exemplo, o modelo conceitual de um protótipo que não esteja disponível). O aluno pode testar mais facilmente suas realizações,

substituindo-as por classes já implementadas, ou ainda adicionando sub-classes ao sistema.

O MOB é um repositório das classes que são instanciadas pelo GPM de modo a gerar os protótipos a serem utilizados no sistema.

Os protótipos, implementados em C++, são executados em microcomputadores PC-compatíveis (sistema operacional DOS²), numa rede local sem praticamente nenhum componente eletrônico ("zero slot LAN"). Esta rede local possui, basicamente, um par trançado com quatro conectores (padrão EIA-RS232c) que são ligados na porta serial de cada microcomputador da rede, formando uma topologia em barra.

2.5 AGENTES E ATIVIDADES

A arquitetura do sistema MinhocO comporta quatro atividades (concepção, desenvolvimento, utilização e observação) e três agentes (professor, aluno e especialista³) [OLIV93b].

ATIVIDADES	AGENTES	AÇÕES
CONCEPÇÃO	Especialista Professor	Concepção dos MBOs Modelagem dos MBOs
DESENVOLVIMENTO	Professor Aluno	Modelagem de aplicações Modelagem de protocolos Modelagem de protótipos Geração de protótipos
UTILIZAÇÃO	Professor Aluno	Modelagem, geração e utilização de projetos Configuração de projetos
OBSERVAÇÃO	Aluno	Observação dos protótipos

Tabela 1: Ações dos agentes (geração e configuração de protótipos)

²Uma versão do sistema MinhocO para o ambiente Windows é também disponível.

³ O especialista enquanto conhecedor das idiossincrasias do sistema ("hardware" e sistema operacional), implementa as classes do mesmo, concebendo assim, junto com o professor, o repositório de classes MOB.

3. VERSÃO MINHOCO 1.0

Nesta seção descrevemos a nova interface homem-máquina implementada no sistema MinhocO, as vantagens que a abordagem orientada a objetos fornece ao método proposto e apresentamos um exemplo de uso do sistema em função dos momentos pedagógicos do referido método.

3.1. A INTERFACE HOMEM-MÁQUINA

A interação homem-máquina é um processo extremamente cognitivo [COUT88]. Mesmo as atividades mais rotineiras, tais como a edição de textos, envolvem a resolução de problemas e requerem a formulação de seqüências de comandos que precisam ser passadas ao computador. Para realizar estas tarefas os projetistas de sistemas interativos precisam explorar idéias da psicologia cognitiva e fatores humanos, que são classificadas em três categorias: modelos teóricos (e.g. "Model of Human Processor") [CARD83], diretrizes práticas (e.g. "Command Language Grammar") [MORA81], e estratégias de testes [SHNE87].

Por enquadrar-se como uma atividade cooperativa que faz uso de uma LAN (rede Minhoca), a interface do sistema MinhocO reveste-se de uma característica peculiar: ela não envolve apenas a interação individual de cada usuário com cada um dos computadores da rede Minhoca (fig.2), como acontece nos sistemas CAL ("Computer Aided Learning"). Ela é antes, um ambiente coletivo e cooperativo (no sentido CSCW) devido os experimentos (agente aluno) e demonstrações (agente professor) envolverem sempre mais de um posto de trabalho.

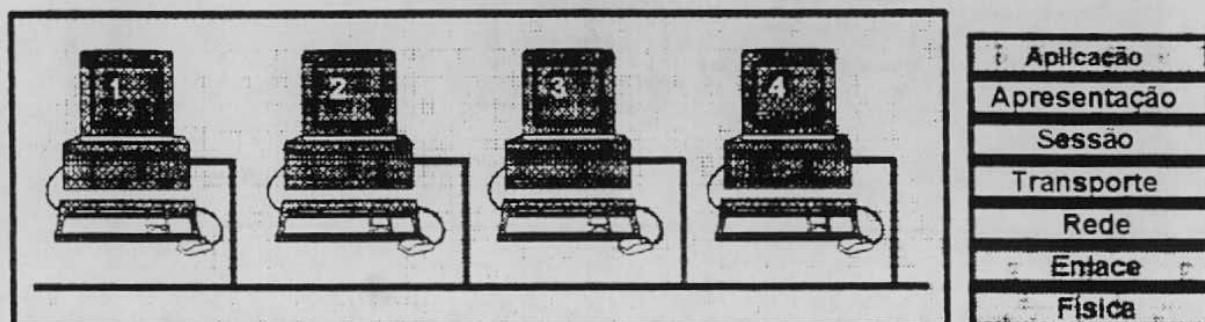


Figura 2: Estrutura física da Rede Minhoca

Dada esta complexidade adicional comentada anteriormente, no projeto da interface MinhocO vem-se observando a adequação de sua característica mutiposto a alguns princípios ergonômicos clássicos tais como: consistência, concisão, carga cognitiva, interação dirigida pelo usuário, flexibilidade, estruturação em diálogo, e predição de erros [SCAP87, SHNE87]. O resultado deste estudo consiste num trabalho em andamento no LAR, e que foge ao escopo deste artigo.

A versão atual da interface homem-máquina do MinhocO (fig.3) fez uso de bibliotecas orientadas a objetos, apresentando as vantagens intrínsecas ao uso de ferramentas especializadas: boa portabilidade, extensibilidade e flexibilidade de utilização. Estas características, juntamente com a inclusão de critérios ergonômicos adequados, constituem uma importante contribuição desta implementação para o Método MinhocO, uma vez que todos os protótipos desta versão são disponíveis de uma forma integrada, ao contrário das versões anteriores onde os protótipos eram autônomos (carregados individualmente).

A evolução futura do sistema trará ainda mais importância para esse aspecto de interface homem-máquina, quando for incorporado um sistema de ajuda inteligente ("Intelligent Help"). O objetivo é que este sistema de ajuda permita a identificação do agente e forneça para ele ajudas contextuais [LIMA93].

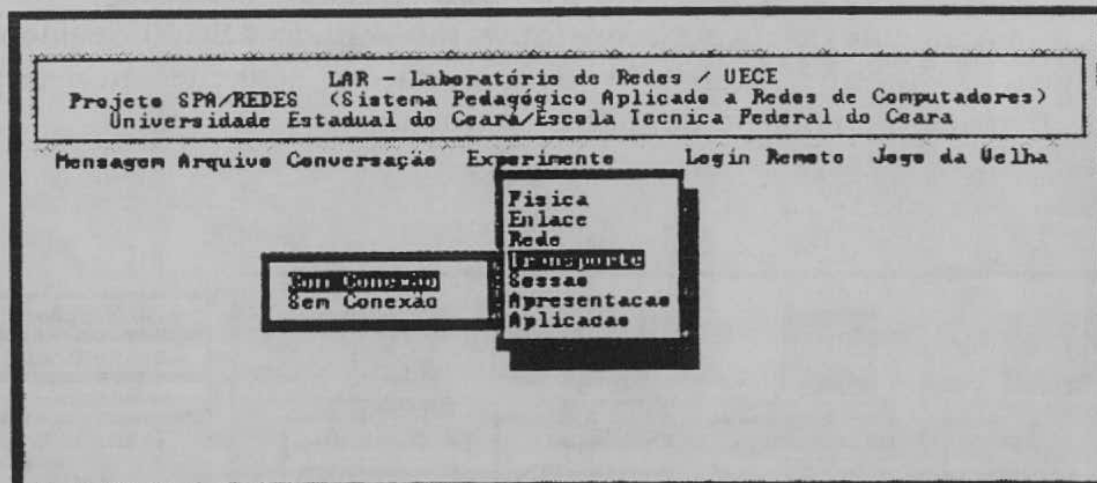


Figura 3: Interface da atual versão do MinhocO

3.2. EXEMPLOS DE ATIVIDADES DO MINHOCO

Para ilustrar, de maneira pragmática, os quatro momentos pedagógicos do Método Minhoco, tomemos um exemplo relacionado com o protocolo de acesso ao meio (Fig.2).

MOMENTO DE SENSIBILIZAÇÃO:

O agente professor neste momento explica ao agente aluno o conceito de protocolo de acesso ao meio (sub-camada MAC do nível de enlace), motivando-o aos momentos seguintes do Método.

1. Configuração da rede Minhoco com o protocolo CSMA:

As estações 1, 2, e 3 enviam simultaneamente diferentes arquivos-texto (cada arquivo constituído por quadros iguais) para a estação número 4 que, por sua vez, apresenta no monitor de vídeo os quadros recebidos. O agente aluno percebe que os quadros são recebidos de forma aleatória (política de acesso probabilística).

2. Configuração da rede Minhoco com o protocolo TOKEN BUS:

Repetindo-se o experimento anterior, agora com o protocolo TOKEN BUS, o aluno percebe que os quadros são recebidos em sequência (política de acesso determinística).

Além desta demonstração, o agente professor pode apresentar aos alunos a modelagem orientada a objeto destes protocolos no contexto do sistema (definição das classes, árvores de herança e conteúdo, diagrama Entidade/Relacionamento, etc).

MOMENTO DE ANÁLISE:

Neste momento, os agentes professor e aluno analisam o funcionamento e implementação dos protocolos de acesso ao meio.

Os experimentos devem ser refeitos pelos alunos, sendo a implementação das classes disponíveis aos mesmos. Alterações feitas pelo agente aluno nestas implementações, gerando novos protótipos, são previstas no Método Minhoco.

As vantagens de se ter uma concepção e implementação orientadas a objetos se pronunciam. Os aspectos de legibilidade provindos da modularidade, interfaces explícitas, visibilidade parcial, presentes no

sistema MinhocO, facilitam sobremaneira a concentração do aluno no tema sendo abordado no momento. A inteligibilidade das ações implementadas pelo código-fonte dos protocolos aumentam ainda a produtividade no processo de aprendizado.

MOMENTO DE SÍNTESE:

O objetivo neste momento é permitir ao agente aluno, a concepção de um protocolo (ou parte dele) de acesso ao meio, ou ainda de funcionalidades correlatas.

Consideremos a Figura 4, onde uma rede possui o protocolo CSMA e a outra o protocolo TOKEN BUS, analisados no momento anterior⁴. O agente professor propõe ao aluno a resolução do seguinte problema: permitir que dois usuários, pertencentes às duas redes, se comuniquem.

Neste momento, ao invés dos aspectos relativos à legibilidade e inteligibilidade do software orientado a objetos (Momento de Análise), são os critérios relativos à qualidade de software que se sobressaem como facilitadores do processo de desenvolvimento realizado pelo aluno: extensibilidade, reusabilidade, modificabilidade, verificabilidade, facilidade de uso.

Além disso, o aluno tem em mãos todo o potencial de uma linguagem orientada a objetos e suas técnicas de programação específicas, tais como: encapsulamento, proteção, composição modular, herança, e polimorfismo). Estas técnicas, agindo como facilitadoras e incrementadoras da produtividade do desenvolvimento, fazem com que o aluno dedique a maior parte do tempo preocupado com o problema sendo solucionado, ao invés de se deter em detalhes de implementações.

MOMENTO DE AVALIAÇÃO

A intenção maior deste momento é associar toda a fundamentação em protocolos de acesso ao meio, ensinada nos momentos anteriores no escopo da sistema MinhocO, a produtos existentes no mercado.

O professor propõe ao aluno avaliar a adequação dos protocolos das redes Ethernet e TOKEN RING a diversas aplicações (controle de

⁴ Na verdade este momento refere-se a um outro possível experimento, no caso tendo como objetivo o ensino de protocolos inter-redes.

processos, aplicações comerciais, etc), em função de critérios variados, tais como desempenho, confiabilidade, custo, etc.

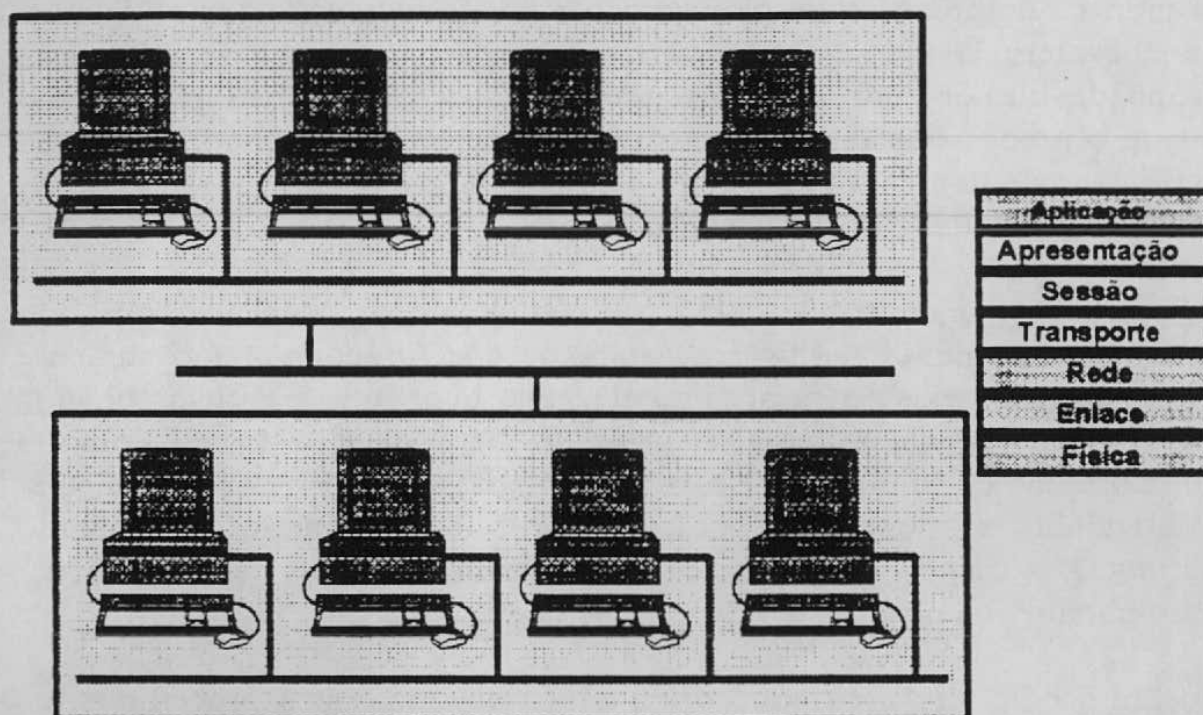


Figura 4: Protocolo interredes

3.3. VANTAGENS DA VERSÃO ORIENTADA A OBJETOS

A orientação a objetos, usada como um conjunto de técnicas para o projeto e implementação de software, torna-se uma poderosa aliada para que seja alcançado o principal objetivo da engenharia de software: fornecer subsídios para se produzir software com qualidade a custos mínimos. Dentro dos critérios usados para descrever a qualidade de um software, como fatores que favorecem diretamente o grau de aprendizagem e utilização do Método MinhocO, citamos a extensibilidade, reusabilidade, verificabilidade, facilidade de uso e legibilidade. A seguir comentamos a forma através da qual estes critérios foram alcançados no MinhocO 1.0 e como eles influenciam positivamente no método pedagógico. Abordamos, também, algumas das técnicas da programação orientada a objetos que foram usadas no projeto e implementação do MinhocO e facilitaram a obtenção destes

critérios de qualidade, tais como: polimorfismo, herança e visibilidade parcial (encapsulamento e proteção).

A necessidade de reusabilidade do código fonte do MinhocO é um fator primordial no processo de planejamento e modularização das atividades de laboratório. Pode-se definir vários níveis de complexidade para cada grupo de alunos (por exemplo, reusar classes existentes para então iniciar o processo aprendizagem num estágio mais avançado), ou ainda definir momentos estanques e de complexidades gradativas (quando se reusaria o resultado obtido anteriormente).

O polimorfismo contribui com a reusabilidade de software tornando possível implementar softwares genéricos que funcionam não somente com os objetos existentes atualmente, mas também para novos objetos que porventura sejam adicionados. A herança é a forma como o polimorfismo de inclusão é tratado em linguagens orientadas a objetos. A atividade de desenvolvimento, descrita anteriormente no método MinhocO, é calcada muitas vezes em procedimentos tais como imitação, refinamento ou combinação de classes já existentes.

No MinhocO, com o enfoque orientado a objetos, estes procedimentos são tomados como essenciais, ao contrário das versões anteriores do sistema que utilizavam técnicas de projeto que os negligenciam. A herança de classe é um mecanismo adequado para atingi-los. O emprego da herança permite a extensão, especialização, ou mesmo combinação de classes já existentes para criar novas classes. As propriedades (variáveis, métodos, assertivas) podem ser preservadas ou sobrepostas por novas propriedades. Assim, é possível também adicionar no sistema MinhocO novas propriedades àquelas herdadas, na classe que está sendo criada (subclasse). Esta forma de relacionamento, por herança, é usada pelo aluno no momento de Síntese, durante o processo de desenvolvimento ou extensão da funcionalidade de alguns dos protótipos.

Vimos, assim, como a extensibilidade (i.e., a facilidade com a qual o software pode ser adaptado a alterações na sua especificação) favorece o processo de realização de soluções intermediárias no Momento de Síntese, quando os agentes incrementam a funcionalidade do seu desenvolvimento em direção a um protótipo satisfatório. Os princípios que norteiam o grau de extensibilidade, a saber: simplicidade do projeto e descentralização (autonomia) dos seus componentes, são seguidos pelo sistema MinhocO, onde cada camada do modelo OSI/ISO é implementada como uma classe (com exceção da camada de enlace que comporta duas classes: LLC e MAC).

Quanto à legibilidade modular, ela permite que, quando muito, o observador precise examinar poucas classes para compreender a funcionalidade da rede e suas interfaces com as possíveis aplicações (Momento de Análise). A verificabilidade e garantia da correção das atividades de programação ficam também simplificadas, visto que podem ser feitas pela simples substituição dos objetos instanciados no protótipo em desenvolvimento, no Momento de Síntese.

A visibilidade parcial (encapsulamento e proteção) das classes e sub-classes que representam cada camada da rede, possibilitam, através dos seus métodos, tanto o exame dos serviços oferecidos (ocasião em que a classe é observada no seu ponto de vista externo pelas demais classes que fazem acesso aos seus serviços), quanto o exame dos protocolos existentes em cada camada (ocasião em que a classe é observada no seu ponto de vista interno, caracterizado pela implementação dos serviços oferecidos pela classe). No MinhocO isto se revela particularmente útil, e pode ser ilustrado pelo seguinte exemplo: para a implementação de uma sub-classe (Momento de Síntese) na sub-camada LLC, não interessa se o acesso ao meio (sub-camada MAC) é modelado por uma sub-classe CSMA/CD ou CSMA/CA.

4. CONCLUSÃO

A razão de ser do sistema MinhocO é oferecer aos agentes (aluno e professor), facilidades para a manipulação de protótipos disponíveis no sistema, devido a sua concepção orientada a objetos (OO), ou ainda, o desenvolvimento de um novo sistema (conjunto de protótipos) a partir de um esquema conceitual próprio do agente.

A implementação do MinhocO numa linguagem OO (C++) permite a concepção de uma interface mais rica do ponto de vista cognitivo. No entanto, apresenta-se como mais relevante o fato das ferramentas de programação OO (visibilidade parcial, polimorfismo, herança, etc), importantes no trato da qualidade do software (extensibilidade, reusabilidade, facilidade de uso, etc), serem completamente compatíveis com a proposta pedagógica do Método MinhocO.

Com a instanciação de objetos de classes distintas, representando os protocolos e serviços de uma rede, podem ser simulados vários protótipos de redes com o MinhocO (simulações de erros podem também ser realizadas) a partir de um sistema integrado, ao contrário das versões anteriores, onde esta atividade exigia a manipulação de diversos protótipos (arquivos executáveis isolados), dificultando a

aplicação do método pedagógico proposto para o uso do sistema. Esta é uma diferença importante entre o sistema MinhocO e os seus antecessores.

MinhocO é uma ferramenta didática já operacional. Versões anteriores do sistema, implementadas em Pascal e em C (possuindo a mesma filosofia pedagógica), têm sido usadas em vários cursos ministrados desde 1989 em empresas e universidades, tanto no Brasil quanto na França. O sistema se apresenta também como um tema de pesquisa multidisciplinar (pedagogia [OLIV93a], engenharia de software [OLIV93b] e tutores inteligentes [LIMA92, OLIV93c]).

Para facilitar o uso do MinhocO, em acordo com o Método proposto, um guia usuário do sistema [OLIV92b] é disponível. Em anexo é apresentado um exemplo do experimento 2.4 ("A Simplex electronic mail - Teacher Notes").

Finalmente, vale destacar dois tratamentos dispensados atualmente ao sistema apresentado neste artigo: como um produto disponível ao ensino de redes de computadores e como um ambiente de pesquisa dentro um esforço que está sendo desenvolvido para a criação do LAR (laboratório multi-institucional de redes e sistemas distribuídos, no estado do Ceará) no contexto do PROTEM/CNPq.

AGRADECIMENTO

Agradecemos ao aluno Henrique Almeida e aos demais alunos participantes do Projeto SPA/REDES da UECE, Custódio Albuquerque, Maciste Saldanha, Marcelo Quinderé e Marcos Paulo, pelo interesse e trabalho dedicado ao sistema Minhoca. Também agradecemos a PETROBRÁS, que tem apoiado o referido projeto (contrato ASFOR 260.2.009/93).

BIBLIOGRAFIA

- [ALME93] Almeida H., Lalane S., Oliveira A.M.B.: *Modelagem do Sistema MinhocO: Um Simulador Orientado a Objeto para o Ensino de Redes de Computadores*. XI Simpósio Brasileiro de Redes de Computadores. Campinas, São Paulo. 1993.
- [CARD83] S.Card, T.Moran, A.Newell: *The Psychology of Human-Computer Interaction*, ISBN 0-89859-243-7, Lawrence Erlbaum Associates, 1983.
- [COEL89] Coelho Filho J.M., Silva Jr J.C.T., Vieira M.H.B., Oliveira A.M.B.: *Minhoca plus, uma rede local para fins didáticos*. VII Simpósio Brasileiro de Redes de Computadores. Porto Alegre. 1989.
- [COUT88] J.Coutaz, L.Bass: *Ergonomics and Software Principles for the Construction of Interactive Software*; Rapport ENSIMAG RR 732-I; septembre 1988.
- [LALA93] Lalanne S.: *Generateur d'un Gestionnaire de Base d'Informations de Gestion. Mémoire en Informatique Fondamentale en vue des Applications*. Conservatoire National des Arts et Métiers. Paris. 1993.
- [LIMA92] Lima J.C.M.: SINTONIA: *Vers un système intelligent pour la construction des diagnostics dans les tuteurs d'informatiques*. Thèse de Doctorat, Université Pierre et Marie Curie - Paris VI. 1992.
- [LIMA93] Lima J.C.M., Oliveira A.M.B., Paget M.M., Claude J.P.: *A CAL to teach computer networks with contextual helps*. Conference on Computer Aided Engineering Education '93 - IEEE. Bucharest. 1993.
- [MORA81] T.Moran: *The Command Language Grammar: a representation for the user interface of interactive computer systems*; International Journal of Man-Machine Studies, 15, 1981, 3-50.
- [NORM86] D.A.Norman, S.W.Draper: *User Centered System Design*; Lawrence Erlbaum Associates, 1986.
- [OLIV87] Oliveira A.M.B., Soares L.F.G.: *Um sistema de computação interredes para aplicação em um ambiente educacional*. XIV SEMISH. Salvador. 1987. Dissertação de Mestrado. Departamento de Engenharia Elétrica da PUC/RJ. Rio de Janeiro. 1987.
- [OLIV91] Oliveira A.M.B., Peyrin J.P.: *Minhonix, a distributed system for teaching*. IX Simpósio Brasileiro de Redes de Computadores. Florianópolis. 1991.
- [OLIV92a] Oliveira A.M.B., Claude J-P: *Un Environnement Orienté Objet pour l'Enseignement des Réseaux Informatiques*. Rapport Interne - Equipe CCR (Contrôle du Comportement des Réseau) - Université Pierre et Marie Curie. France. Juillet 1992.
- [OLIV92b] Oliveira A.M.B., Saleh M.: *MinhocO Users Guide*. Publication interne du groupe CCR (Contrôle du Comportement des Réseau) du Laboratoire MASI - Université Pierre et Marie Curie. France. 1992.
- [OLIV93a] Oliveira A.M.B.: *Spécification d'une Méthode et Réalisation d'un Environnement pour l'Etude des Resaux Informatiques et des Systèmes Répartis*. These de Doctorat de l'Université Paris 6. Octobre 1993.
- [OLIV93b] Oliveira A.M.B., Lalanne S., Claude J-P.: *A Prototype Generator for a Network Teaching System*. XIX Seminário Integrado de Software e Hardware. Florianópolis. 1993.
- [OLIV93c] Oliveira A.M.B., Paget M-M., Lima J.C.M.: *Towards an intelligent system in the teaching of computer networks*. CAL 93, York, UK. 1993.
- [PIAG75] Piaget J.: *Psychologie et Pédagogie*. Presses Universitaires de France. 1975.
- [ROGA87] Rogalski J.: *Quelques elements de théorie Piagetienne et didactiques des mathématiques*. IREM, Université Pierre et Marie Curie - Paris VII. 1987.
- [SCAP87] D.L.Scapin: *Guide ergonomique de conception des interfaces homme-machine*; Rapport INRIA 77, octobre, 1987.
- [SHNE87] B.Shneiderman: *Designing the User Interface: Strategies for Effective Human-Computer Interaction*, Addison Wesley Publishing Company, 1987.
- [SKIN68] Skinner B.F.: *The Technology of Teaching*. Prentice-Hall. New Jersey. 1968.

ANEXO

TEACHER NOTES

EXPERIMENT 2.4: A simplex electronic mail - Data-Link LAYER

- **Purpose:**

This experiment is designed to illustrate an application to be developed by the students in the Minhoca System, using the Minhoca Interface on the level 2.

The main topics illustrated are:

- Services
- Interfaces

- **Scenario:**

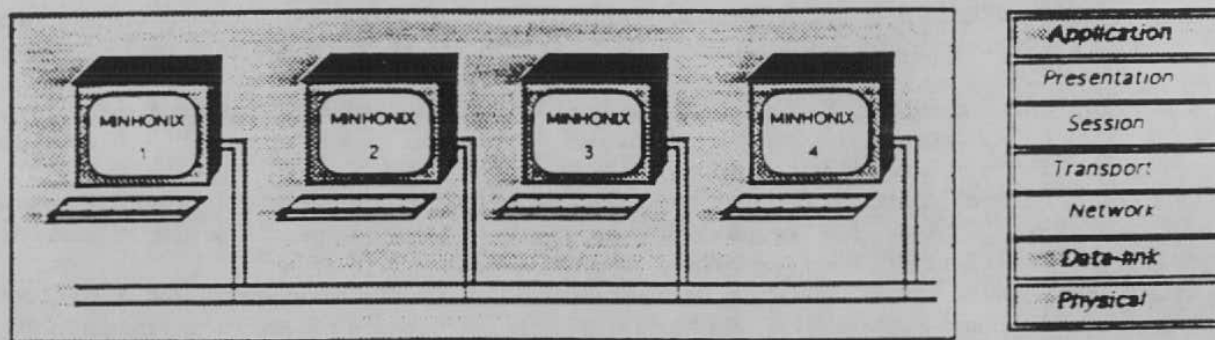


Figure 2.4

This experiment requires in the first phase only one IBM-PC microcomputer for each student (or group) to develop its application. Afterwards, this application must be tested in the classical configuration of the system, shown above.

- **Pedagogical moment: Synthesis**

It is supposed that the students know the protocol concept and a bit about computer network architecture. At this stage, the fact that students understand or not the Minhoca protocols have not a big importance. The teacher must explain this context to the students before asking them to perform this task (simple electronic mail).

• Activity

1. Initialisation

1.1 The teacher must install the Minhoca Cable 1



The teacher must connect the 4 machines with the Minhoca nodes (figure 1.3)



The teacher can look the Chapter IX (How to install Minhoca)

1.2 The teacher asks the users to load the Minhoca LAN-4



Each user must load the **mnet23** prototype (Mnet1 directory, MnetFR-1 diskette) on his station.



The screen should show a presentation picture of the system

1.3 Each user must configure the station address and the transmission rate.



The users must type RETURN key.



The screen should show the following menu:

NUMBER STATION (1, 2, 3 or 4):

Tx RATE (110, 600, 2400, 9600 or 19200):



The user must type a address number (1 through 4, as show in the figure 1.4) and 9600 bps.



The prototype is loaded in the interruption vectors. The screen will show the system prompt (came back to the operating system): A>



It's very important to explain the students that process above, according the course level.

1.4. After having presented Minhoca LAN to the users the teacher can propose the students to read the comments about the system

2. Simplex electronic mail

2.1 The teacher asks the students to load the Simplex Electronic Mail Application



Each user must load the **mnet24** prototype



The screen should show the following menu:

<i>DELIVER MAIL</i>	<i><D></i>	<i>(Send one electronic mail)</i>
<i>MAIL BOX</i>	<i></i>	<i>(Receive a electronic mail)</i>
<i>EXIT</i>	<i><E></i>	<i>(Return to the operating system)</i>



It's very important to explain here the relationship between the **mnet14a** (network protocols) and **mnet14b** (application) prototypes.

2.2 The teacher asks the user 1 to send a mail to the user 2, 3 and 4, sequentially.



They must choose the **DELIVER MAIL** option.

2.3 The teacher asks the users 2, 3 and 4 to look their mail box



The users 2, 3 and 4 must type the **MAIL BOX** option



The mail sent by the user 1 must be there.

2.4 The teacher asks the user 4 to send a broadcast mail



The user 4 must choose the **DELIVER MAIL** option and type **ZERO** as destin station

2.5 The users 1, 2 and 3 must look their mail box



The users 1, 2 and 3 must type the **MAIL BOX** option



The mail sent by the user 4 must be there.

2.6 Repeat the step 2.4 and 2.5 but with the station 2 logically disconnected



Before the user 4 to sent his mail the user 2 must type the option **EXIT** and the option **D** (Desenable)



The screen of the station 2 will show a system prompt and it doesn't receive the mail sent by the user 4.

2.7 Repeat the step 2.6 but now with the the station 2 logically activeted (outside of the Minhoca):



Before the user 4 to sent his mail, the user 2 must type the option **EXIT** and the option **E** (Enable)



The screen of the station 2 will show a system prompt and it will receive the mail sent by the user 4.

- 2.8. Repeat the step 2.4 and 2.5 but with the station 2 Physically disconnected



Before the user 4 sent his mail, the user 2 must disconnect the Minhoca Node in his station.



The same remark of the step 2.6

3. Implementation

- 3.1 The teacher can ask the students to implement the Minhoca LAN-4 interface in C language



The teacher must give (and explain) to the students the Minhoca LAN interface that was implemented in Pascal or C Language.



The aim is to familiarize the students with the interface concept and with the Minhoca interface itself.

- 3.2 The teacher can ask the student to implement an Electronic Mail Application, equivalent to the **mnet24** prototype.



If a student doesn't achieve this task in the normal time, the teacher can help him showing the **mnet24** implementation (ANNEX).



The works implemented by the student must be tested with the **mnet23** prototype.