

UMA EXPERIÊNCIA EM IMPLEMENTAÇÃO E TESTES
DE PROTOCOLO PARA REDES DE COMPUTADORES

PROJETO CEPINNE - PROTOCOLO DE TRANSPORTE

MIGUEL MAURÍCIO ISONI
UNIVERSIDADE FEDERAL DO PIAUÍ
N.P.D. - CAMPUS UNIVERSITÁRIO
64.000 - TERESINA - PI

JOSÉ ALBOS RODRIGUES
JACQUES P. SAUVÉ
JOSÉ ANTÃO B. MOURA (*)
UNIVERSIDADE FEDERAL DA PARAÍBA
DEPARTAMENTO DE SISTEMAS E COMPUTAÇÃO
GRUPO DE REDES DE COMPUTADORES
58.100 - CAMPINA GRANDE - PB

RESUMO

Este artigo trata da experiência da UFPb em implementar a camada de transporte (camada 4) para sua integração à rede do projeto CEPINNE (Centro Piloto de Serviços de Teleinformática para Aplicação em Ciência e Tecnologia na Região Norte e Nordeste).

(*) - Departamento de Engenharia Elétrica

1 - INTRODUÇÃO

Aspectos importantes no projeto de uma rede de computadores são o desenvolvimento e a implementação de seus protocolos de comunicação. Para reduzir a complexidade desses aspectos, a Organização Internacional de Padronização (ISO) desenvolveu o Modelo de Referência para Interconexão de Sistemas Abertos (RM/OSI) [1]. O RM/OSI estrutura os diversos protocolos da rede em sete camadas hierárquicas. Cada camada reúne funções bem definidas comuns a uma determinada etapa de funcionamento e presta serviços à camada imediatamente superior a partir dos serviços da camada imediatamente inferior.

A camada mais baixa, camada (1), define as características mecânicas, elétricas, funcionais e de procedimentos dos meios físicos usados para transmissão de bits. A camada de enlace de dados, camada (2), tem como funções: detectar, e possivelmente corrigir, erros na camada (1); e fornecer à camada (3) a capacidade de controlar o chaveamento de circuitos. A camada da rede, camada (3), provê os meios para estabelecer, manter e terminar conexões de rede, e meios funcionais e de procedimento para troca de informação - através de conexões - entre duas entidades da camada (4). A camada de transporte, camada (4), tem como propósito fornecer o serviço de transferência transparente de dados (fim-a-fim) entre entidades da camada (5) utilizando conexões-de-rede. A camada da sessão, camada (5), tem o propósito de organizar e sincronizar o diálogo, e gerenciar a troca de dados entre entidades da camada (6). A camada da apresentação, camada (6), fornece serviços que podem ser selecionados pela camada (7) e que gerenciam a entrada, troca, apresentação e controle de dados estruturados. A camada de aplicação, camada (7), tem como propósito servir de "janela" entre usuários comunicantes através da qual ocorre toda troca de informações significativa para esses usuários.

Este artigo trata da experiência da UFPb em implementar a camada de transporte (camada 4) para sua integração à rede do projeto CEPINNE [2]. O projeto CEPINNE visa interligar os recursos computacionais das Universidades do Norte e Nordeste. A implantação do protocolo de transporte da rede CEPINNE ficou a cargo inicialmente, das Universidades Federais da Paraíba e Pernambuco. Subsequentemente, o mesmo protocolo será implementado nas outras Universidades.

2 - A REDE CEPINNE E O SEU PROTOCOLO DE TRANSPORTE

Numa primeira fase, a rede CEPINNE constituiu-se de dois nós comutadores de Pacotes REXPAC (Rede Experimental por Comutação de Pacotes) desenvolvidos pelo CPqD/TELEBRÁS [4] (vide Figura 1). Um desses nós está localizado em Belém - interligando os equipamentos Terminais de Dados (ETDs) das Universidades UFPa e UFPe - e o outro em Recife - interligando UFPb, UFPe e UFBA. A velocidade entre nós é de 2.400 bps e, a velocidade entre nó e ETD é de 1.200 bps (adotada em caráter inicial).

A conexão ETD-nó será feita segundo a interface X.25 do CCITT [5], possivelmente implementada através de um conversor de protocolos CX.25 desenvolvido pelo CPqD da TELEBRÁS, o qual permite até 4 linhas assíncronas. Serão utilizadas as facilidades de comunicação de dados oferecidas pela EMBRATEL.

Quanto ao protocolo de transporte, (ver ref. 3) a implementação já executável, nos ETDs da UFPb e da UFPe constituiu-se do seguinte:

- 1) A interface com a camada superior (possivelmente aplicação) - Esta é composta de primitivas de serviços agrupadas em três categorias: estabelecimento e encerramento de conexão; transferência de dados usando conexão e transferência de dados sem

usar conexões;

- 2) A interface com o conversor CX.25 - Nesta, as primitivas são classificadas em dois grupos: gerência de circuito virtual e transmissão e recepção de informação;
- 3) O protocolo de transporte propriamente dito através da troca das suas "unidades de dados" as quais classificam-se em: estabelecimento e encerramento de conexão e transferência de dados.

3 - METODOLOGIA DE IMPLEMENTAÇÃO ADOTADA

Similarmente à UFPe, a implantação do protocolo de transporte de rede CEPINNE na UFPb ficou a cargo de uma equipe multidisciplinar, composta por três professores e três alunos dos Cursos de Pós-Graduação em Sistemas e Computação e Engenharia Elétrica. A composição da equipe visou atender, principalmente, aos seguintes objetivos: desenvolver recursos humanos especializados em redes de computadores; familiarizar o pessoal da equipe com o uso de recursos de Teleinformática e introduzir uma cultura de processamento distribuído e desenvolvimento de software de comunicação no âmbito da UFPb.

O projeto global da implementação do protocolo de transporte na UFPb, teve como ponto de partida, um estudo aprofundado dos documentos técnicos fornecidos pela EMBRATEL, sobre o Projeto Técnico da Rede CEPINNE [6] [7], e da especificação do protocolo de transporte fornecida pela PUC/RJ. Este estudo culminou num levantamento de dúvidas e discussões no âmbito da equipe, e em conjunto com a equipe da UFPe, (isto é, e o seu Grupo de Redes de Computadores e Sistemas Distribuídos - REDIS).

A natureza das dúvidas e a complexidade do projeto global (veja seção 4.2), levaram à divisão do esforço total

de implementação em quatro segmentos (vide ilustração na Figura 2):

- Implementação da Estação de Transporte (ET);
- Implementação da Interface da ET com o Conversor CX.25;
- Implementação da Interface da ET com a Camada Superior;
- Implementação de um ambiente de procedimentos de testes.

4 - IMPLEMENTAÇÃO

4.1 - Ambiente de Implementação

A implementação do protocolo de transporte para a rede CEPINNE foi realizada num PDP-11/34, o qual será a máquina hospedeiro da UFPb (inicialmente) sob o ambiente de programação UNIX. A linguagem de programação C foi adotada devido a sua excelente interface com o sistema operacional UNIX. Excetuando-se a definição de dutos ("pipes") permanentes, a implementação do protocolo de transporte não requereu quaisquer modificações no sistema operacional. (*)

4.2 - Implementação da Estação de Transporte

A implementação da ET considerou três pontos básicos:

- 1 - A ET é usada por mais de um processo usuário. Há portanto compartilhamento de seus recursos;

(*) - A versão 7 do UNIX roda na UFPb e normalmente não possui dutos permanentes: UNIX system III e System V têm esta facilidade.

- 2 - Devido ao seu tamanho, a implementação da ET dentro de cada processo usuário não é admissível. Também por questões de tamanho, a ET também não deve ser implementada como parte do sistema operacional, numa máquina como PDP-11/34;
- 3 - A ET não pode atender a interrupção das linhas assíncronas do CX.25 pois não faz parte do núcleo do sistema operacional. Ela também não pode ler as linhas diretamente pois (no UNIX), ficaria bloqueada até chegar alguma informação na linha sendo lida; isto impediria que a ET atendesse a outros pedidos de serviços.

Para que os dois primeiros pontos sejam atendidos, a ET foi desenvolvida como um processo servidor ("daemon"). Com esta decisão, e juntamente com o ponto 3, dois problemas devem ser resolvidos:

- a) Como fazer a comunicação entre usuários e o processo servidor (ET) e;
- b) Como este servidor tem acesso ao conversor CX.25.

A solução adotada para o problema em a) é a seguinte. A comunicação entre processos no UNIX (Versão 7) se faz através de arquivos. O uso de arquivos do tipo duto (característica do UNIX) simplifica a comunicação, quando comparado com o uso de arquivos normais. Para poderem se comunicar com dutos, dois processos devem ter sido criados por um ancestral comum que também deve criar os dutos de comunicação. Isso não é o caso aqui pois a ET é um processo servidor inicializado independentemente de qualquer outro processo. Nossa solução foi a de usar uma característica não padrão do UNIX (Versão 7) mas disponível no nosso ambiente: dutos permanentes.

Com esta solução e a fim de desvincular os vários processos usuários, cada um desses processos cria e mantém

seu próprio conjunto particular de dutos com a ET. Há, po
rém, um duto geral de entrada para a ET, conhecido por todos
os processos usuários através do qual eles passam o nome de
seus dutos permanentes e a sua identificação antes de pode
rem se comunicar com a ET (através de seus dutos particulaa
res). Os dutos no conjunto particular são classificados em:

- 1 - Duto normal - Do usuário para a ET, no qual são
passadas as primitivas de serviços;
- 2 - Duto urgente - Do usuário para a ET, através do
qual são passados pedidos de interrupções (dados
urgentes);
- 3 - Duto de controle - Da ET para o processo usuário
no qual a ET envia sinalização e/ou respostas às
chamadas de primitivas de serviço.
- 4 - Duto de dados - Da ET para o processo usuário,
no qual são passadas as mensagens, interrupções
e telegramas (após serem indicadas como "válidi
das" ou presentes pelo código de sinalização a
próprio, no duto de controle). Esse duto é uti
lizado também para passar as informações de identi
ficação do processo usuário remoto quando do
estabelecimento de uma conexão.

Para solucionar o problema b) (comunicação com con
versor CX.25) definimos quatro processos independentes (um
para cada linha assíncrona), encarregados de ler informações
nas linhas e enviá-las para a ET através de dutos.

A figura 3 ilustra a estrutura de comunicações da ET
quando existem dois processos usuários ativos.

A filosofia que adotamos para o desenvolvimento da
ET não seguiu um método específico de programação modular,
mas foram observados alguns conceitos mínimos para que alcanç
ássemos um grau satisfatório e conveniente de modularizaç
ão. Satisfatório, porque conseguimos desenvolver e desmembr
ar todo o problema satisfazendo à definição da ET. Conve

niente, pois utilizando a linguagem C, conseguimos construir os programas estruturadamente dentro do nosso ambiente de programação alcançando funcionalidade, portabilidade e legibilidade.

Certas estatísticas sobre a implementação final podem ser apreciadas na figura 4.

4.3 - Teste de Implementação

A ET foi implementada em três etapas: estabelecimento de conexão; encerramento de conexão e transferência de dados. Tal divisão facilitou não somente a implementação propriamente dita como também a aplicação de testes a cada etapa já implementada.

A inexistência de uma ET já testada, e sem um conversor CX.25 disponível, levou a decisão de submeter ET a testes utilizando a arquitetura da figura 5. Para esta arquitetura foi desenvolvido um módulo de software que interage com a ET tanto pelo lado do processo usuário como pelo lado da interface inferior.

Este módulo interativo permite que o operador, a partir do terminal, exerça o papel de um usuário (ativando primitivas de serviço) ou de outra ET (recebendo e/ou enviando unidades de dados do protocolo fragmentos). Dessa forma toda a inteligência de teste fica com o operador. É ele quem verifica o comportamento da ET através do exame dos fragmentos ou códigos de retorno que ela emite quando da chamada de primitivas ou da chegada de fragmentos.

A geração de cenários de teste é feita antecipadamente para evitar dificuldades resultantes do uso do testador (observando o comportamento da ET) em

paralelo com outras atividades (por exemplo, a geração de cenários). Os cenários escolhidos para submeter à ET devem ser bem objetivos, uma vez que é impraticável e enfadonho aplicar todas as combinações de cenários possíveis. Nossa bateria de testes incluiu cerca de 200 cenários de teste.

5 - OBSERVAÇÕES FINAIS

Como adiantamos, a fim de possibilitar uma interpretação precisa e sem ambigüidades do protocolo de transporte e agilizar sua implementação, decidimos por identificar inicialmente uma série de pontos que foram discutidos no seio da equipe da UFPb e em entendimentos com a equipe da UFPe.

Os pontos levantados foram classificados em duas categorias:

- Dúvidas de abrangência global, que foram discutidas e dirimidas ente a UFPb e UFPe;
- Dúvidas referentes a decisões locais, levantadas apenas para promover uma discussão mais ampla sobre a implementação do protocolo no seu ETD.

Como a especificação do protocolo de transporte utilizou-se de uma técnica informal (linguagem natural), muitas imprecisões e ambigüidades foram detectadas. Uma nova especificação revisada mas ainda informal que reflete os pontos levantados e discutidos foi então produzida pelas UFPb e UFPe. A especificação final sô foi produzida à luz de nossa experiência de implementação do protocolo. Não precisamos enfatizar que isto dificultou o trabalho de implementação e teste e que este procedimento será evitado em implementações futuras de protocolos.

Com relação aos testes, a principal observação é a de que há uma tendência natural de se querer automatizar a

geração e aplicação de cenários de testes (que são muito te_udiosas). Isso implica, entretanto, no desenvolvimento de um testador que substituísse uma ET e um elemento humano. O de_u desenvolvimento de um testador com as características dese_uja das requer um estudo bem elaborado e sistemático, devendo ser abordado oportunamente.

Para adiantar os trabalhos de testes, desenvolveu-se um módulo de software para ligação direta de duas ETs idênti_ucas evitando assim o uso do conversor CX.25. O software de_u desenvolvido simula a gerência de circuito virtual. Fez-se tam_ubém uma especificação dos serviços do conversor CX.25 usando uma técnica formal para evitar interpretação dúbias surgidas do exame da documentação fornecida, a qual emprega uma lin_uguagem natural.

Concluindo, o andamento do projeto de implementação do protocolo de transporte da rede CEPINNE na UFPb exigi_uu uma afinidade entre o Gerente de Projeto e a equipe encarre_ugada do desenvolvimento, pois foi através dele que um sub_u-grupo particular de trabalho expunha o progresso de suas a_utividades e se inteirava das necessidades dos demais; neces_usidades estas a serem atendidas pelo sub-grupo particular. Esta estratégia possibilitou uma integração efetiva dos sub_u-grupos levando a uma evolução harmoniosa do projeto de tal forma que todos estavam a par de detalhes do trabalho de ca_uda sub-grupo e tinham ainda uma visão do projeto como um to_udo. Tentava-se evitar assim as consequências danosas de even_utuais deserções dos membros da equipe e que felizmente não aconteceu. O fato de familiarizar todos os membros da equipe com detalhes do trabalho de cada sub-grupo também teve uma motivação didática de formação de recursos humanos.

Em termos didáticos acadêmicos, o projeto resultará em três teses de mestrado na área de redes de computadores.

FIGURA 1

CONFIGURAÇÃO INICIAL DA REDE CEPINNE

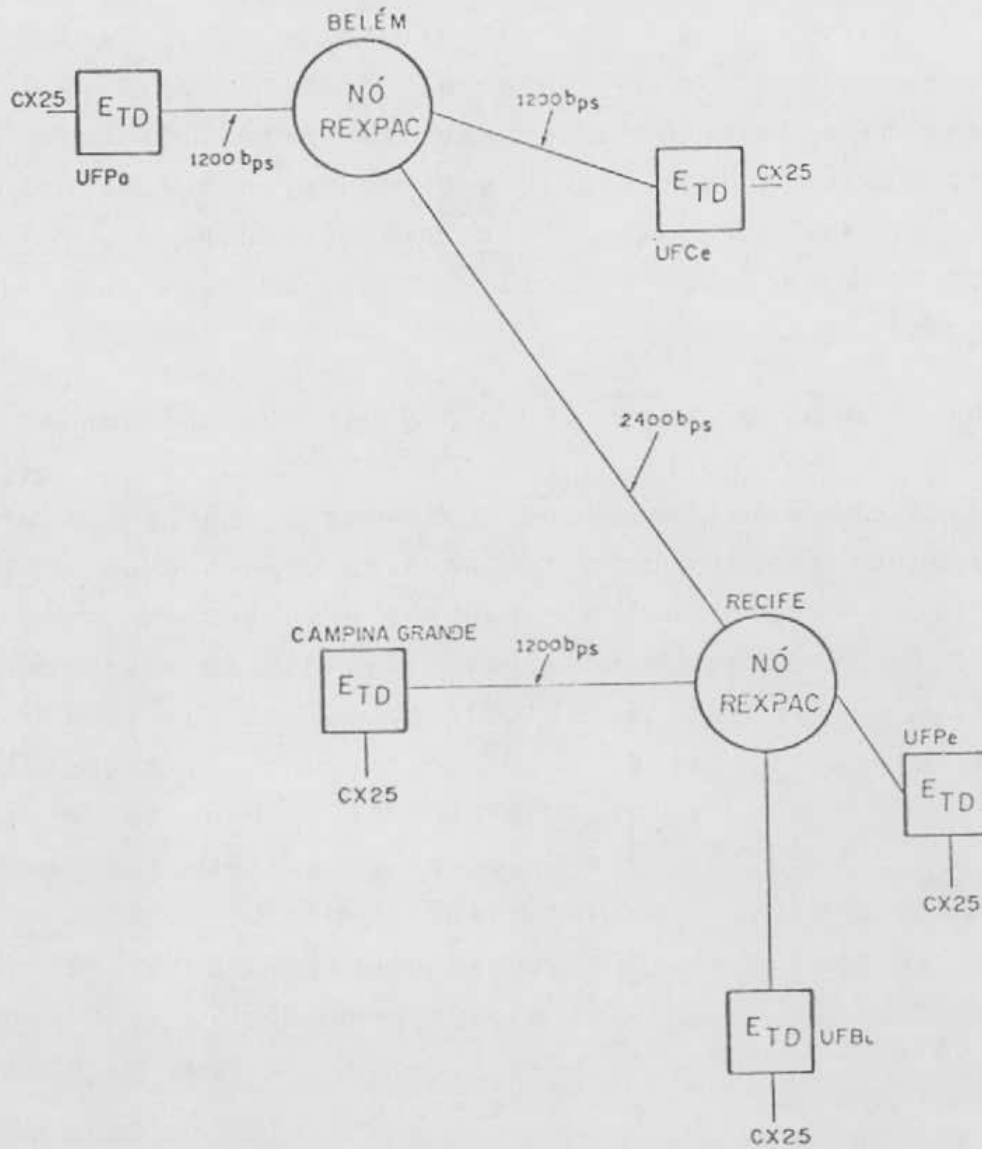


FIGURA 2

OS SEGMENTOS DE IMPLANTAÇÃO
DA ET E SEUS SERVIÇOS NA
UFPb

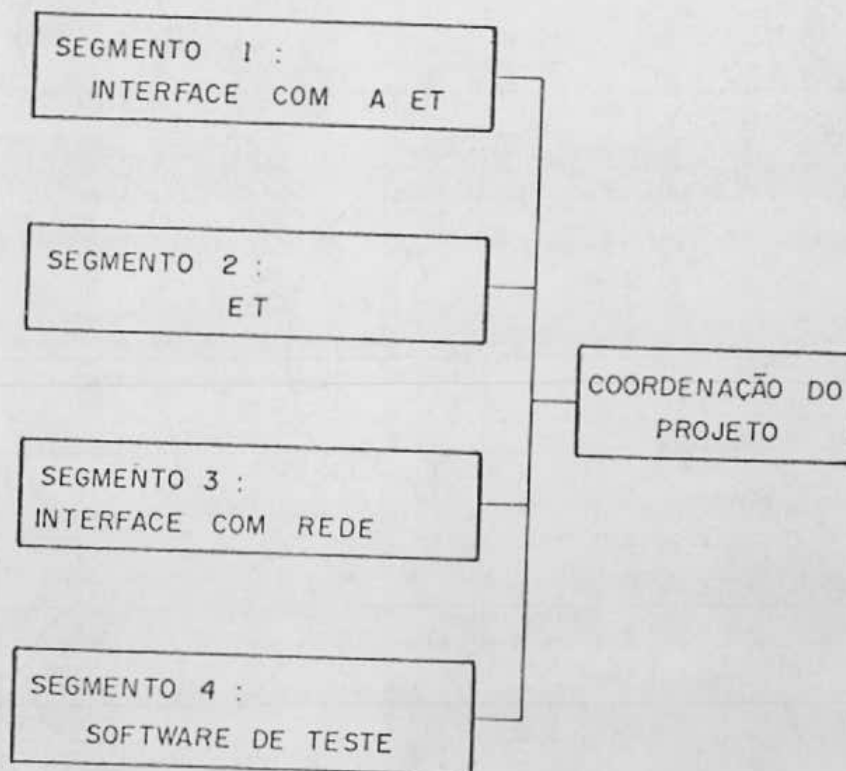


FIGURA 3

VISÃO GERAL DA IMPLEMENTAÇÃO DA ET
NO UNIX

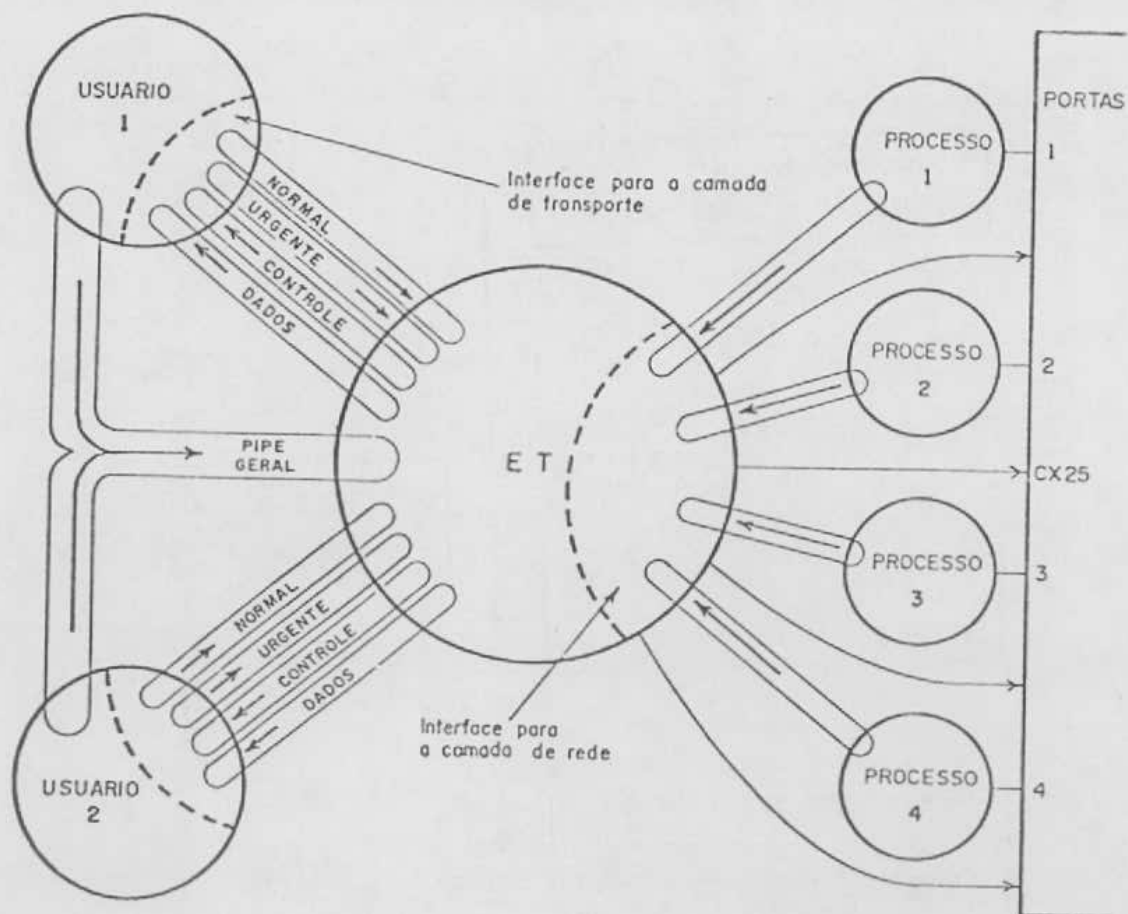


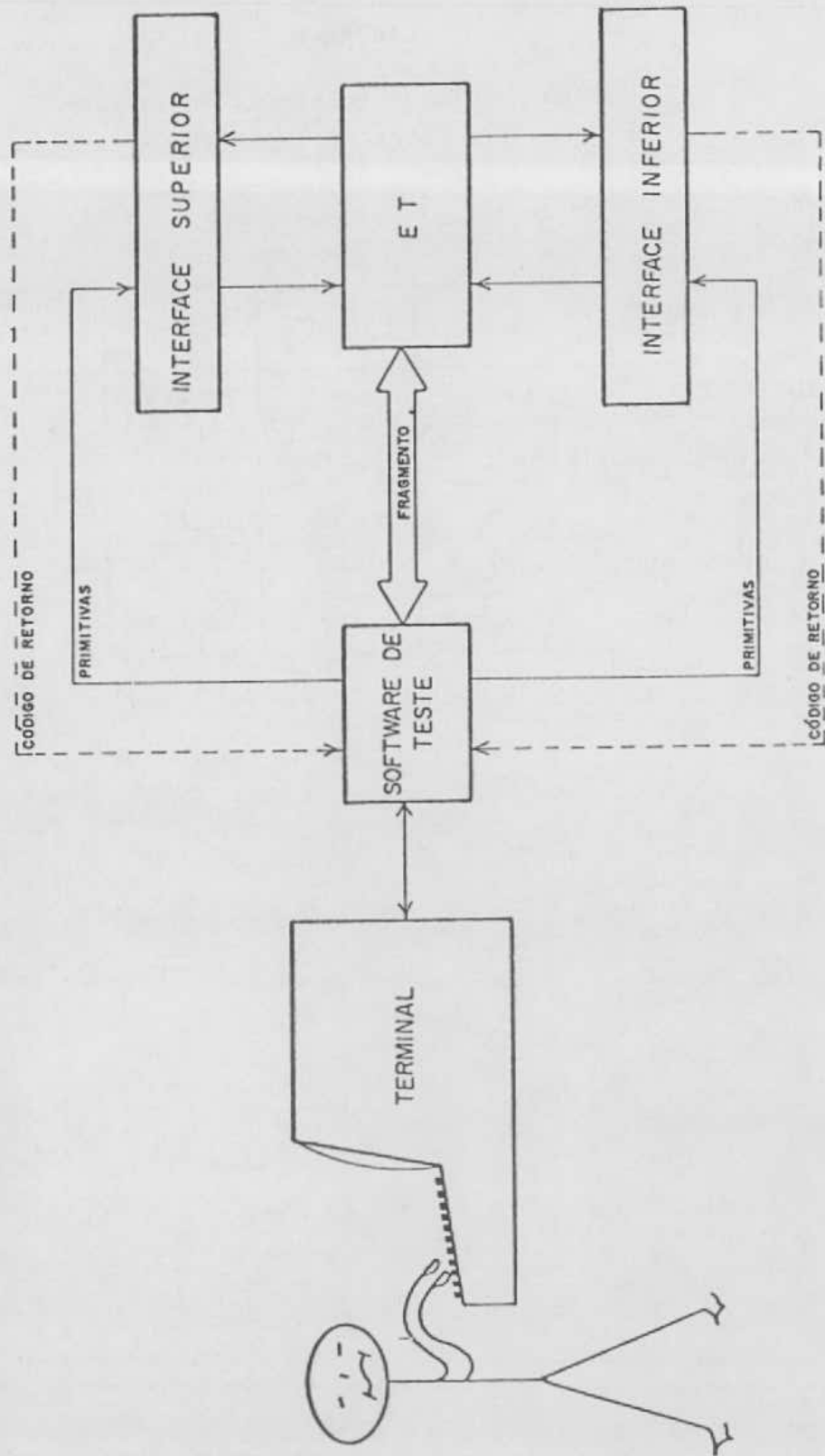
FIGURA 4

TAMANHO DA IMPLEMENTAÇÃO NO UNIX
E A FORÇA DE TRABALHO

NOME DO SEGMENTO	LINHAS	BYTES	HOMENS / HORA
1-INTERFACE COM A ET	1.622	31.373	120
2-SOFTWARE DE TESTES	650	1.257	50
3-ET	5.235	119.117	402
4-INTERFACE COM REDE	520	1.100	40
→ TOTAL	8.127	152.847	612

FIGURA 5

AMBIENTE DE TESTES



6 - REFERÊNCIAS

- 1 - ISO; "OPEN SYSTEM INTERCONNECTION - BASIC REFERENCE MODEL (ISO DRAFT PROPOSAL 7498) - ISO TC 97/SC 16", ACM - SIGCOM COMPUTER COMMUNICATION REVIEW, Vol. II, Nº 2, Abril 1981.
- 2 - J.P.SAUVÉ, J.A.B. MOURA; "REDE CEPINNE", Datanews Ano IX, Nº 231, Junho 1984.
- 3 - PUC, "ESPECIFICAÇÃO DE UM PROTOCOLO DE TRANSPORTE PARA REDES QUE UTILIZAM CIRCUITOS VIRTUAIS", Departamento de Informática, Dezembro 1982.
- 4 - TELEBRÁS, "ESPECIFICAÇÃO DA REDE E DA CENTRAL DE COMUTAÇÃO REXPAC", CPqD/TELEBRÁS, PROJETO REXPAC, Julho 1983.
- 5 - CCITT; "INTERFACE BETWEEN DATA TERMINAL EQUIPAMENTO (DTE) AND DATA CIRCUIT EQUIPAMENT (DCE) FOR TERMINALS OPERATING IN THE PACKET MODE ON PUBLIC DATA NETWORKS", YELLOW BOOK Vol. VIII, Fascículo VIII.2, Recomendação X.25, Gênêve, Suíça, Novembro 1980 (pp.100 - 190).
- 6 - EMBRATEL; "PROJETO TÉCNICO DE REDE CEPINNE", Segmento de Implantação da Rede, Vol. I.
- 7 - EMBRATEL; "PROJETO TÉCNICO DA REDE CEPINNE", Segmento de Implantação da Rede, Vol. II.