

VISÃO DE UM SISTEMA DE TERMINAL VIRTUAL

Oswaldo Vernet de Souza Pires

Núcleo de Computação Eletrônica - UFRJ

Caixa Postal 2324

20001 - Rio de Janeiro - RJ

Telefone: (021) 290-3212

SUMÁRIO

Este artigo procura elucidar o Serviço de Terminal Virtual (STV), que opera na Camada de Aplicação da Arquitetura RM-OSI. Inicialmente, discutimos a motivação para a existência do STV; identificamos as classes de serviços, introduzimos o Modelo Abstrato e as Primitivas de Serviços disponíveis ao usuário. Concluímos com um breve relato da implementação que estamos realizando deste serviço no NCE/UFRJ.

ABSTRACT

This paper tries to describe the Virtual Terminal Service (VTS), which operates in the Application Layer of RM-OSI Architecture. We initially discuss the motivation for the VTS; we identify Classes of Services, introducing the Abstract Model and the Service Primitives available to the user. We conclude with a brief report about the implementation of those services we are currently doing at NCE/UFRJ.

1. INTRODUÇÃO.

É indiscutível atualmente a importância que assumem os terminais de vídeo nas mais diversas aplicações computacionais, sendo raros os sistemas de computadores que não possuam um ou mais terminais conectados a seus equipamentos. Evidentemente, as técnicas e conceitos envolvidos no projeto de terminais evoluem juntamente com o surgimento de novas abordagens na concepção de sistemas de computação. Com a ênfase que se dá atualmente à interconexão de equipamentos em redes, surge naturalmente a necessidade de estender o escopo das aplicações de terminais de maneira que eles sejam capazes de controlar e acessar recursos disponíveis através de uma rede de computadores.

Esta extensão não é, infelizmente, imediata e se depara com alguns problemas de incompatibilidade. Dentre eles, o que mais nos chama a atenção é a diversidade de tipos de terminais, que diferem tanto no aspecto funcional (funções suportadas) quanto nos protocolos de comunicação (TTY, BSC, Poll-Select...). Outras diferenças advêm das naturezas diversas das aplicações, dando origem a terminais orientados a "scrolling", páginas ou telas previamente parametrizadas para situações específicas.

Visando resolver esta adversidade, surgem algumas propostas:

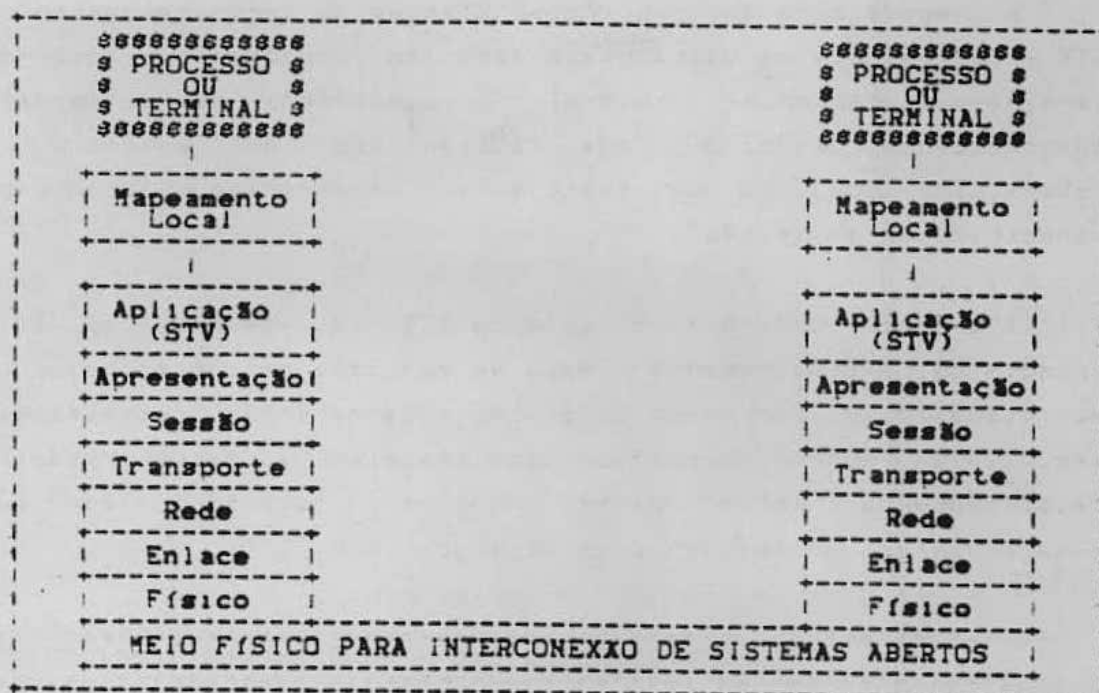
- (a) para cada tipo de terminal que se queira conectar a determinada máquina, reescreva-se a parte do sistema operacional responsável por manipular terminais ("driver" ou "handler" de terminais), acrescentando procedimentos para lidar com o recém-instalado equipamento;
- (b) usar um processador "front-end" que incorpore o "handler" de terminais, não onerando o sistema operacional. Isto, entretanto, não se aplica a terminais não-endereçáveis ("burros").

Estas propostas talvez viabilizassem a conexão de variados tipos de terminais, mas certamente não generalizariam suas funções para acesso ou controle de informações através da rede. Para isto, duas propostas são lançadas: o Sistema de Terminal Paramétrico e o Sistema de Terminal Virtual.

O Sistema de Terminal Paramétrico visa, conforme o nome o sugere, parametrizar as diferenças entre os terminais, definindo um conjunto de parâmetros específicos para cada tipo de terminal. Estes são conectados a um equipamento concentrador que fica encarregado de interfaceá-los com a rede. Este equipamento, em geral a própria máquina hospedeira, atuará como um conversor de protocolos.

O Serviço de Terminal Paramétrico é elucidado pelo PAD ("Packet Assembler/Disassembler"), que deve ser usado em redes cujas interfaces sejam definidas pela Recomendação X.25. Este serviço possui deficiências: já que as aplicações computacionais envolvendo terminais enquadram-se geralmente na camada 7 da Arquitetura RM-OSI (Camada de Aplicação), detalhes do serviço para conexão de terminais não devem depender da organização das camadas inferiores. Além disso, a solução proposta pelo PAD, além de restringir-se à parametrização dos terminais existentes na época em que foi concebida, funciona somente em redes com interface X.25.

Já o Sistema de Terminal Virtual (STV) aborda o problema introduzindo um modelo abstrato de comunicação que possibilita o mapeamento das funções mais comumente encontradas em terminais (endereçamento de cursor, mudança de atributos na tela, etc) em uma estrutura de dados cujas características são negociadas entre os parceiros. Este modelo é usado na definição de um conjunto de primitivas de serviços. A figura a seguir ilustra como o STV, oferecido na Camada de Aplicação do Modelo de Referência para Interconexão de Sistemas Abertos (RM-OSI), pode ser usado na comunicação entre dois sistemas abertos:



Talvez a aplicação mais comum para o STV seja um processo executando em um sistema aberto gerando telas para um terminal em outro sistema aberto. Entretanto, outras possibilidades não são descartadas: dois terminais comunicando-se ou até mesmo dois processos, sem a existência de terminal. Tudo depende do mapeamento que se realiza localmente, que não é objeto de padronização da ISO.

II. CLASSES DE SERVIÇOS.

O Serviço de Terminal Virtual reside na Camada de Aplicação da Arquitetura RM-OSI. Como os demais Serviços oferecidos nas diversas camadas desta arquitetura, o STV resume-se em uma comunicação horizontal entre duas entidades de aplicação cooperantes em um mesmo nível (o sétimo, neste caso). Estas entidades que constituem o STV utilizam-se dos serviços oferecidos pela Camada de Apresentação e pela Sub-Camada Comum de Aplicação (CASE KERNEL).

A necessidade de identificar Classes de Serviços dentro do STV é proveniente da vasta quantidade de funções e tipos de terminais, tornando inviável a concepção de um modelo absolutamente genérico. Cada Classe vem de encontro a determinado conjunto de facilidades necessárias a algum tipo específico de aplicação.

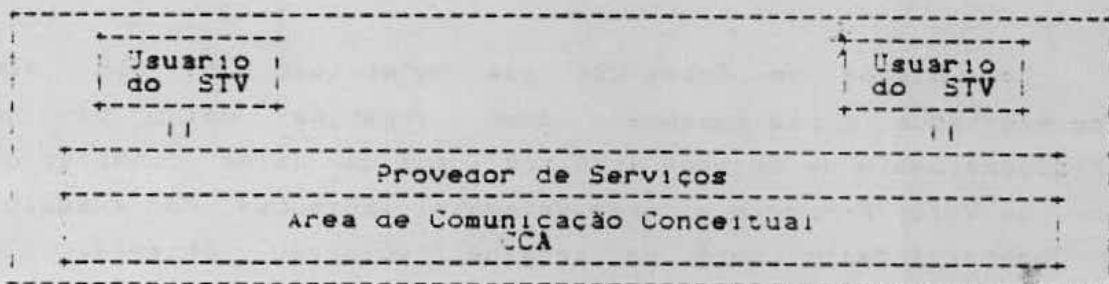
A classificação dos serviços do STV foi baseada em dois princípios. Primeiramente levou-se em conta o tipo de objeto a ser visualizado, por exemplo, pontos, caracteres, fac-símiles, etc... O outro critério diz respeito a estruturação e relacionamento entre estes objetos, podendo eles ser independentes, organizar-se em arranjos, etc...

	NENHUMA ESTRUTURACAO	AGRUPAMENTO HIERARQUICO
CARACTERE	CLASSE BASICA	CLASSE FORMULARIO
BITE	CLASSE IMAGEM	
EL. GRAFICO		CLASSE GRAFICA
MISTO DOS ANTERIORES		CLASSE MISTA

Para aplicações orientadas a caracteres temos disponíveis os serviços da CLASSE BASICA, onde nenhuma estruturação da imagem visualizada existe, ou da CLASSE FORMULARIO, onde os objetos a serem visualizados agrupam-se hierarquicamente (campos, sub-campos, títulos, etc...). Em se tratando de aplicações gráficas, a hierarquização dos objetos é natural (ponto, figura, quadro, etc...), não havendo o correspondente não-estruturado. Neste artigo tratamos especificamente da CLASSE BASICA.

III. O MODELO ABSTRATO.

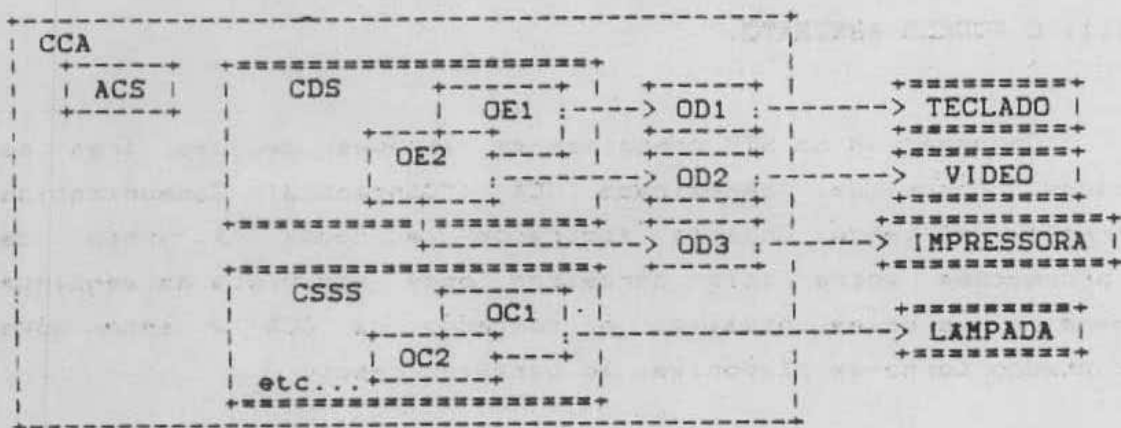
Os usuarios do STV comunicam-se através de uma área de dados conceitual denominada CCA ("Conceptual Communication Area"), contendo Objetos Abstratos de Dados. A troca de informações entre dois parceiros pode ser vista da seguinte maneira: um deles atualiza o conteúdo da CCA e este novo conteúdo torna-se disponível ao parceiro remoto.



A CCA é constituída pelos seguintes componentes:

- (a) **CDS** - ("Conceptual Data Store"), contendo um ou dois **Objetos de Exposição**, dependendo do modo de operação do STV;
- (b) **CSSS** - ("Control, Signalling and Status Store"), contendo **Objetos de Controle**;
- (c) **ACS** - ("Access Control Store"), que registra o estado corrente de distribuição dos "Tokens";

A figura a seguir ilustra um exemplo de mapeamento dos **Objetos Abstratos de Dados** nos dispositivos físicos, via os **Objetos de Dispositivo** (também chamados **Dispositivos Virtuais**).



Os Objetos de Exposição que constituem o CDS são implementados, fisicamente, como arranjos uni, bi ou tridimensionais de Objetos Atômicos, cada um deles constituído por um Valor Primário e, possivelmente, atributos. Por exemplo, o Objeto Atômico para um terminal colorido orientado a caracteres seria constituído por um caractere (valor primário) e a cor correspondente (atributo).

Os Objetos de Controle que compõem o CSSS permitem aos usuários tratar eventos não mapeáveis nos Objetos de Exposição, geralmente associados a características específicas dos Dispositivos Físicos. O conteúdo de um Objeto de Controle pode ser uma variável inteira, lógica, um caractere, ou uma cadeia de bites.

Os Objetos de Dispositivo ou Dispositivos Virtuais permitem o mapeamento das informações contidas no CDS nos Dispositivos Físicos.

Os Objetos Abstratos de Dados são negociados no momento do estabelecimento da conexão entre os parceiros, fazendo parte do Ambiente de Terminal Virtual ("VTE - Virtual Terminal Environment"), melhor detalhado na seção V.

IV. MODOS DE OPERAÇÃO E DIREITOS DE ACESSO.

As normas [1] e [2] definem dois modos de operação para o STV, a saber: modo Assíncrono e modo Síncrono.

No primeiro caso, o CDS é constituído por dois Objetos de Exposição: um deles só pode ser atualizado pelo usuário iniciador da conexão e o outro pelo usuário respondente. A comunicação se dá na forma de dois monólogos em direções opostas.

No modo Síncrono, o CDS é constituído por um único Objeto de Exposição. Para atualizá-lo, um usuário deve possuir o Direito de Acesso ao objeto, que é atribuído a um deles no momento da conexão e transferível mediante solicitação durante o diálogo.

No modo Assíncrono existem dois Direitos de Acesso não transmissíveis: WACI ("Write-Access Connection Initiator"), atribuído ao usuário que iniciou a conexão e WACA ("Write-Access Connection Acceptor"), atribuído ao respondente. A cada Objeto de Exposição está associado um Direito de Acesso, garantindo que o iniciador só poderá atualizar um deles e o respondente, o outro. Já no modo Síncrono, existe um único tipo de Direito de Acesso, denominado WAVAR ("Write-Access Variable"), cuja posse inicial é combinada em tempo de conexão e pode ser transferido de um usuário para outro durante o diálogo.

Os Objetos de Controle podem ou não estar sujeitos aos Direitos de Acesso. Se estiverem, a semântica da atualização se dá de modo análogo aos Objetos de Exposição, e o objeto é dito Controlado. Caso contrário, o objeto pode ser atualizado por qualquer dos usuários, sendo de responsabilidade mútua o protocolo de acesso; são ditos, neste caso, Não-Controlados.

Além dos Direitos de Acesso, Objetos de Controle podem possuir um atributo denominado "trigger". No modo Síncrono, o

"trigger" assegura a transferência do WAVAR ao parceiro sempre que uma atualização for realizada sobre aquele objeto; no modo Assíncrono, o "trigger" causa a remessa de todas as atualizações realizadas sobre objetos controlados pelo mesmo Direito de Acesso ao parceiro. A existência do "trigger" só tem sentido para Objetos Controlados.

V. SERVIÇOS E PRIMITIVAS.

A CLASSE BÁSICA do STV oferece os seguintes serviços:

(a) Estabelecimento de Conexão.

Um usuário, denominado iniciador, requer ao STV o estabelecimento da conexão com algum parceiro remoto. Neste momento, o iniciador menciona o nome de um "profile", que é uma proposta de configuração para o Ambiente de Terminal Virtual (VTE), já mencionado. Esta proposta ("profile") é examinada pelo parceiro, que poderá ou não aceitá-la. Caso ambos estejam de acordo, o parceiro é denominado respondente e a conexão se diz estabelecida. Caso contrário, o iniciador é notificado da recusa. Além do "profile", o iniciador deve mencionar que Direitos de Acesso estarão sendo usados (WAVAR ou WACI+WACA) e, no modo Assíncrono (WACI+WACA), quem vencerá as colisões que eventualmente surjam entre primitivas ("collision-winner").

Primitivas: VT-ASSOCIATE (Request, Indication, Response, Confirm).

(b) Término de Conexão.

Um dos usuários solicita ao STV o término da conexão. Existem duas maneiras de fazê-lo: solicitação educada ou abrupta. No primeiro caso, o parceiro pode recusar, e a conexão não será desfeita. No segundo caso, a conexão é desfeita, com

possível perda de informações. Há ainda a possibilidade de o STV informar aos usuários o aborto de uma conexão.

Primitivas: VT-RELEASE (Request, Indication, Response, Confirm);
 VT-U-ABORT (Request, Indication);
 VT-P-ABORT (Indication).

(c) Negociação.

Através dos Serviços de Negociação, os usuários podem seleccionar, modificar ou substituir o Ambiente inicial fixado em tempo de estabelecimento de conexão mediante o "profile" mencionado pelo iniciador. Este Ambiente pode ser alterado durante a vida da conexão de duas formas: em uma única interação (SIN - "Single-Interaction Negotiation") ou sob forma de um diálogo onde valores são oferecidos, aceitos e rejeitados para os diversos parâmetros que compõem o Ambiente (MIN - "Multiple-Interaction Negotiation"). Dependendo do tipo de negociação suportada, a CLASSE BÁSICA do STV subdivide-se em três subconjuntos: VT-A (apenas a negociação em tempo de conexão), VT-B (VT-A + SIN) ou VT-C (VT-B + MIN).

Primitivas: VT-SWITCH-PROFILE (Request, Indication, Response, Confirm).
 VT-START-NEG (Request, Indication, Response, Confirm);
 VT-END-NEG (Request, Indication, Response, Confirm);
 VT-NEG-OFFER (Request, Indication);
 VT-NEG-INVITE (Request, Indication);
 VT-NEG-ACCEPT (Request, Indication);
 VT-NEG-REJECT (Request, Indication).

(d) Transferência de Dados.

Os usuários atualizam a CCA através dos Serviços de Transferência de Dados, que se resume em sequências de itens

alterando Objetos de Exposição ou Controle.

Primitivas: VT-DATA (Request, Indication);
VT-UNCONTROLLED-DATA (Request, Indication).

(e) **Remessa.**

Por conveniência do Provedor de Serviços (constituído pelas duas máquinas abstratas que implementam o protocolo de Terminal Virtual), os itens que contêm atualizações provenientes de um usuário sobre os Objetos de Exposição ou Controle podem ser armazenados em uma fila e remetidos ao parceiro somente em determinados momentos. Estes instantes de remessa podem ser implícitos (quando iniciada negociação, pedido de término ordenado ou quando da atualização de Objetos de Controle com "trigger" no modo Assíncrono) ou explícitos, através do Serviço de Remessa. O usuário que provoca a remessa pode solicitar uma notificação do parceiro de que os dados foram resimmente transferidos.

Primitivas: VT-DELIVER (Request, Indication);
VT-ACK-RECEIPT (Request, Indication).

(f) **Transferência de Direito de Acesso.**

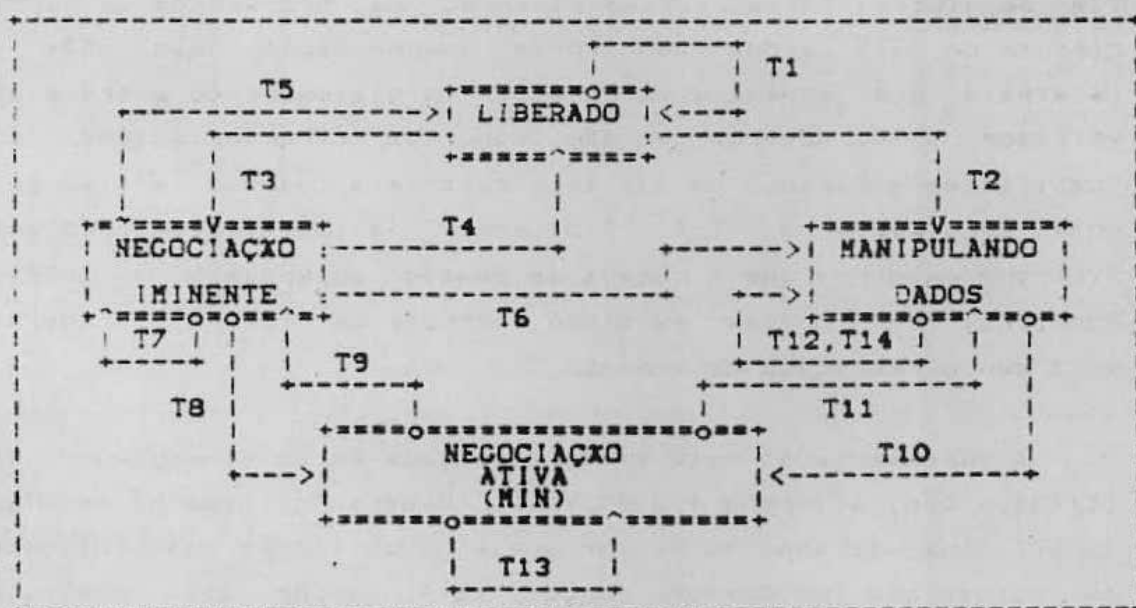
No modo Síncrono, o WAVAR pode ser atribuído ao parceiro quando um usuário atualiza um Objeto de Controle com "trigger" ou pode ser solicitado mediante o Serviço de Transferência de Direito de Acesso, caso se queira produzir alguma alteração sobre um objeto abstrato e não se possua o WAVAR.

Primitivas: VT-GIVE-TOKENS (Request, Indication);
VT-REQUEST-TOKENS (Request, Indication).

VI. DIAGRAMA DE ESTADOS.

Examinando macroscopicamente a vida de uma conexão de Terminal Virtual, podemos distinguir quatro estados em que o Provedor de Serviços pode-se situar em determinado instante:

LIBERADO	Não existe conexão nesta fase.
MANIPULANDO DADOS	A informação é efetivamente trocada entre os usuários.
NEGOCIANDO AMBIENTE	Negociação em múltiplas interações.
NEGOCIAÇÃO IMINENTE	Não há transferência ou negociação, pois o ambiente não foi especificado completamente.



- T1: VT-ASSOCIATE malogrado.
 T2: VT-ASSOCIATE com sucesso e ambiente completo.
 T3: VT-ASSOCIATE com sucesso e ambiente incompleto.
 T4: VT-RELEASE com sucesso.
 T5: VT-U-ABORT ou VT-P-ABORT.

- T6: VT-SWITCH-PROFILE com sucesso.
- T7: VT-RELEASE, VT-SWITCH-PROFILE ou VT-START-NEG malogrados.
- T8: VT-START-NEG com sucesso.
- T9: VT-END-NEG com alerta.
- T10: VT-START-NEG com sucesso.
- T11: VT-END-NEG com sucesso.
- T12: VT-SWITCH-PROFILE.
- T13: VT-END-NEG malogrado.
- T14: VT-START-NEG ou VT-RELEASE malogrados.

VII. CONSIDERAÇÕES SOBRE IMPLEMENTAÇÃO E CONCLUSÕES.

No NCE/UFRJ, estamos implementando o subconjunto VT-A da CLASSE BÁSICA do STV, suportando apenas o modo Assíncrono de operação. A opção por implementarmos este subconjunto deve-se aos seguintes fatos: primeiramente, os protocolos de outras classes do STV ainda não foram padronizados pela ISO; os terminais que pensamos em conectar inicialmente ao sistema são voltados a caracteres e não suportam características que justifiquem a mudança do ambiente durante a conexão, motivo pelo qual escolhemos o VT-A; finalmente, a implementação do modo Síncrono exigiria que a Camada de Sessão suportasse a Unidade Funcional Half-Duplex, exigindo controle de "Tokens", o que não está sendo realizado no momento.

A implementação está sendo realizada em um computador VAX 11/780, com a versão 4.5 do VMS. O usuário utiliza os serviços do STV link-editando ao seu programa de aplicação uma biblioteca de rotinas que implementam funções mais comuns dos terminais. Estas rotinas mantêm diálogo com o STV através de Caixas Postais ("Mailboxes"), enviando ao sistema mensagens que excitam a máquina de estados.

Nesta primeira implementação, não estamos considerando a

Camada de Apresentação nem a Sub-Camada Comum de Aplicação (CASE KERNEL). As primitivas do STV são mapeadas diretamente em primitivas da Camada de Sessão, conforme [6].

No estágio atual de implementação, estamos depurando a máquina de estados, juntamente com a Camada de Sessão, especificada em [8]. Os próximos passos seriam: tratar as atualizações dos Objetos de Exposição e Controle, implementar o processo de mapeamento local e construir a biblioteca de rotinas do usuário.

Para a simulação do diálogo entre dois usuários do STV, promovemos a duplicação dos processos que compõem o sistema: 1 processo para o STV, 1 processo para a Camada de Sessão e 1 processo para simular a Camada de Transporte, ainda não especificada: ficamos ao todo com 7 processos: 5 simulando STV+Sessão+Transporte+Sessão+STV e 2 simulando os usuários.

A comunicação entre camadas também é realizada com Caixas Postais. Entretanto, para evitar que a mensagem inicial do usuário seja reescrita em cada Caixa Postal no trânsito pelas camadas da rede, mantemos um "heap" em memória compartilhada ("Global Section" no VMS) para onde a mensagem é copiada uma só vez e liberada quando conveniente; ponteiros para a mensagem são efetivamente escritos nas Caixas Postais.

Temos constatado ao longo das simulações que a abundância de processos aliada a operações de entrada/saída em Caixas Postais são catastróficos em termos de desempenho no VMS. A solução a médio prazo, para uma versão mais refinada, seria projetar um sistema de comunicação totalmente em memória compartilhada, evitando o uso de Caixas Postais.

VIII. BIBLIOGRAFIA

- [1] ISO/DIS 9040 - "Information Processing Systems - Open Systems Interconnection - Virtual Terminal Services - Basic Class".
- [2] ISO/DIS 9041 - "Information Processing Systems - Open Systems Interconnection - Virtual Terminal Protocol - Basic Class".
- [3] ECMA-87 - "Generic Virtual Terminal Model and Service Description".
- [4] LOWE, H. - "OSI Virtual Terminal Service".
Proc. of IEEE, Vol 71, no. 12, Dec/83.
- [5] VERNET, O. - "Concepção de um Sistema de Terminal Virtual".
Relatório Técnico, NCE 01/88.
Jan/88, NCE/UFRJ.
- [6] PIRMEZ, L. - "Estudo dos Serviços de Sessão".
Relatório Técnico, NCE 00787.
Maio/87, NCE/UFRJ.
- [7] VERNET, O. - "Especificação do Sistema de Terminal Virtual".
Relatório Técnico, ainda não publicado.
NCE/UFRJ.
- [8] PIRMEZ, L. - "Especificação da Camada de Sessão".
Relatório Técnico, ainda não publicado.
NCE/UFRJ.