

Terminais Inteligentes: alternativa estratégica para otimização de recursos computacionais

Reinaldo J. Moreira , Hebert Luiz A. Costa, Jailton C. Junior, Gicele M. T. de Souza, Ulisses A. Leitão

Faculdades Integradas de Caratinga, Instituto Doctum – Educação e Tecnologia
Pç. Cesário Alvim, 110 – 35.300-036 Caratinga – MG - Brasil

{petrok, bicudo}@spep.com.br, {hebert, gicele, tecnologia}@doctum.com.br

***Abstract.** This paper describes the Smart Clients Approach as a strategic solution in order to optimize computational resources. The Smart Clients are similar to the Thin Clients but, on the other hand, they have a reasonable local processing and RAM memory capabilities. We argue that, in this case, it is a resource wasting to use a fat server to do what the station can do. A new instalation script was developed to carry out a sistematic use of local applications, making possible a full control of load balacing within the local network.*

***Resumo.** Este artigo descreve a abordagem de Terminais Inteligentes como solução estratégica visando a otimização de recursos computacionais. Terminais Inteligentes são similares a Thin Clients. Entretanto, possuem razoável capacidade de processamento e de memória RAM locais. Neste caso, é um desperdício de recursos utilizar o servidor para realizar tarefas que podem rodar nas estações. Um novo script de instalação foi desenvolvido para realizar o uso sistemático de aplicações locais permitindo o pleno controle de balanceamento de carga na rede local.*

1. Introdução

O termo Terminais Inteligentes (em inglês *Smart Clients*) se refere a estações de trabalho com supressão de hardware, mas com poder de processamento local. São estações tipo *thin clients*, sem HD, que executam tarefas como uma estação desktop padrão. Entretanto, ao contrário das estações *thin clients* tradicionais, os Terminais Inteligentes têm capacidade de processamento local, o que permite ao projetista do sistema computacional a utilização de técnicas de balanceamento de carga entre o servidor e a estação como estratégia de otimização de recursos computacionais.

Tais estações se apresentam como ideais para a criação de diversos ambientes computacionais, em especial para laboratórios de inclusão digital, pois agregam duas características essenciais: economia e flexibilidade. A economia é conseguida com a simplificação do hardware. A flexibilidade vem do uso de processamento local, tornando a estação capaz de executar diversas tarefas sem a sobrecarga da rede ou do servidor, além de facilitar enormemente a administração do sistema.

O presente artigo apresenta a solução de Terminais Inteligentes desenvolvida dentro do projeto LibertasBR¹ e tem como objetivo implementar uma solução de boot remoto com balanceamento de carga entre o servidor e as estações remotas. Concebido como uma personalização da distribuição GNU/Linux-Debian voltada para o setor público, o LibertasBR, desenvolvido pela UFMG, Faculdades Doctum e Prodabel, permite a instalação e configuração de Terminais Inteligentes de forma simples e prática. Em uma de suas utilizações mais relevantes, o LibertasBR está

1 www.libertasbr.dcc.ufmg.br

viabilizando a implantação de noventa e quatro Telecentros de Inclusão Digital em comunidades carentes do norte e nordeste de Minas, projeto Cidadão.NET.

2. Abordagens tradicionais

Em ambientes, corporativos ou educacionais, de uso intensivo de recursos computacionais sem, entretanto, necessidade de alto poder de processamento, soluções do tipo *Network Computer* tem-se tornado gradativamente predominantes. Duas são as estratégias de abordagem comumente apresentadas neste caso.

2.1 Soluções baseadas em Terminais X

A solução mais popular e que tem sido responsável por um crescente interesse na utilização do Linux em diversos setores, é a solução baseada em terminais X. Diversos projetos de software livre são dedicados a implementar soluções que fazem amplo uso das capacidades intrínsecas de protocolo de acesso remoto à interface gráfica de usuário Xwindow, veja [Leung(2004) , Brin(2004)]. Todos estes projetos podem ser caracterizados pela implementação de *thin clients*, em que todos os aplicativos são configurados para rodarem em servidores remotos. Talvez o projeto mais ativo com esta abordagem seja o LTSP – Linux Terminal Server Project (<http://www.ltsp.org>). Uma introdução ao projeto estará sendo apresentada no atual Fórum [McQuillan, 2004]. Entretanto, projetos alternativos estão em atividade: o Projeto Netstation, (<http://netstation.sourceforge.net>); o projeto Thinstation (<http://thinstation.sourceforge.net>), o projeto Diet-PC (<http://diet-pc.sourceforge.net>), o DRBL -Diskless Remote Boot in Linux, [Blake 2004], dentre outros. Utilizando soluções de boot via rede, como p.ex. a solução do projeto Netboot (<http://netboot.sourceforge.net>), Etherboot (<http://sourceforge.net/projects/etherboot>) ou, ainda, implementando soluções próprias, todos estes projetos possibilitam a utilização de estações sem disco de pouca memória RAM, deixando todos os aplicativos rodando no servidor.

2.2 Solução baseada em servidores VNC

Uma solução menos tradicional foi recentemente relatada por Yu e Cimafranca [Yu et al., 2004]. A solução proposta consiste no desenvolvimento de uma distribuição Linux reduzida, com a implementação do sistema VNC – Virtual Network Computing (veja <http://www.csd.uwo.ca/staff/magi/doc/vnc/index.html>). Baseado no protocolo RFB – Remote Framebuffer, o VNC é, em essência, um sistema de terminal remoto, o qual propicia o acesso interativo ao ambiente de desktop de qualquer máquina na rede local ou pela Internet. Além disto, tanto os servidores como os visualizadores VNC estão portados para diferentes arquiteturas e sistemas operacionais. Assim o VNC permite a implementação de um sistema computacional multiplataforma inteiramente transparente. Rodando sob protocolo TCP-IP, o VNC pode ainda ser visualizado em máquina virtual java e mesmo browsers tradicionais. O VNC se comporta, para uma aplicação Linux, como se fosse o próprio servidor X, podendo, entretanto, ser visualizado remotamente, por uma máquina de arquitetura e sistema operacional diferentes, desde que esteja rodando o aplicativo de visualização do VNC adequado. Esta tecnologia é a base de soluções de integração de sistemas operacionais por metaframe, como a proposta proprietária recentemente ofertada no mercado pela Syst International (<http://www.metasys.com.br>).

3. Metodologia

Existe uma desvantagem comum às estratégias de solução apresentadas acima. Ambas soluções são ancoradas na utilização de *thin clients* como Terminais Burros. Em muitos casos, entretanto, temos estações de trabalho com poder de processamento e disponibilidade de memória locais razoavelmente

bons. Nestes casos é um enorme desperdício utilizar estas estações apenas como terminais burros², rodando apenas o Kernel do Linux, a biblioteca SVGA e o servidor X. Assim, a estratégia utilizada pelo presente trabalho foi a de adaptar o script DRBL, disponível para as distribuições Debian e Red Hat, visando:

- (a) Simplificar o script, retirando os itens relacionados à configuração de cluster para processamento paralelo, já que esta funcionalidade não estava entre as especificações do projeto;
- (b) Criação de uma interface de instalação mais amigável, utilizando Dialog e Xdialog;
- (c) Implementação de opção para utilização do LDAP como alternativa ao NIS.

4. A implementação de Terminais Inteligentes

A estrutura Cliente/Servidor é a base para a implementação de Terminais Inteligentes empreendida. O Servidores provêm os serviços DHCP, TFTP, NFS e LDAP ou NIS. O DHCP fornece aos clientes um endereço IP para a sua identificação na rede. O TFTP disponibiliza a imagem do Kernel bootável, possibilitando que as estações clientes inicializem o sistema operacional. O NFS exporta o sistema de arquivos e o NIS (LDAP) fornece os serviços de autenticação de usuários. Com a disponibilização desses serviços conseguimos um servidor que atenda um sistema computacional robusto e eficiente.

Através dos serviços disponibilizados pelo servidor, cada estação cliente adquire sua própria configuração mantida no servidor para leitura e escrita. O Servidor disponibiliza um sistema GNU/Linux básico, usando apenas alguns arquivos compartilhados, incluindo as pastas /bin, /sbin, /usr e /lib.

O sistema passou a ser denominado RBL, Remote Boot LibertasBR. Sua instalação é feita através de telas simples e intuitivas, onde o usuário pode fazer toda configuração, seguindo apenas um manual simples e prático. Foi desenvolvido um *live CD* como base de instalação, como a que foi feita pelo sistema Pequinzilla [Blake, 2004], onde as imagens das estações e do servidor são criadas automaticamente, com uma configuração personalizada para o LibertasBR. Os scripts do RBL tem por finalidade montar os seguintes componentes:

RBL Server

Servidor provê os serviços de DHCP, TFTP, NFS e NIS (LDAP). Os clientes, através do endereço IP, carregam o kernel e suas configurações através da rede. Durante o procedimento de boot, o cliente monta o RBL Client Template usando o serviço de NFS e o NIS (LDAP) para a autenticação e manipulação das informações dos usuários.

RBL Client

Cada estação da rede RBL tem sua própria configuração, mantida em seus diretórios específicos /etc, /root e /var, armazenada no servidor para leitura e escrita de seus clientes.

RBL Client Template

Nada mais é que um sistema Debian básico criado através do comando *debootstrap*, usando apenas alguns arquivos compartilhados, incluindo as pastas /bin, /sbin, /usr e /lib. Ele cria o template dos clientes copiando toda árvore de diretórios do servidor e utiliza o comando *chroot* para instalar os pacotes necessários para os clientes. Como o servidor e os clientes não têm a mesma árvore e configurações do sistema Debian, os computadores não necessitam ter a mesma configuração de hardware.

² Tal fato foi reconhecido pelos desenvolvedores do LTSP. Assim, um dos pontos que mereceu maior atenção dos desenvolvedores do LTSP na elaboração de sua versão 4.0 foi exatamente o de tornar mais fácil a instalação de aplicativos locais. Entretanto, as opções da estrutura de sistemas de arquivos propostas pelo projeto tornam esta tarefa desnecessariamente complexa.

5. Análise de Resultados

Uma das principais características apresentadas é a flexibilidade oferecida pelo RBL para a criação e administração de ambientes computacionais. Assim, pode-se conceber desde ambientes onde todas as aplicações sejam executadas nas estações, até sistemas onde todas as aplicações rodam remotamente no servidor. Ambientes com balanceamento de carga, com algumas aplicações rodando no servidor e outras nas estações, são configurados pela simples instalação de aplicativos em diretórios compartilhados ou não, respectivamente.

Aplicações exclusivamente locais são instaladas e configuradas em diretórios específicos das estações, como /etc, /root, /var e, em especial, /usr/local. Aplicações remotas são executadas a partir de diretórios comuns a todas as estações, como é o caso do /opt, /bin, /sbin, /usr/bin. Um exemplo pode ser dado com a utilização de dois aplicativos muito comuns: Mozilla e OpenOffice. O primeiro deles pode ser executado remotamente a partir do diretório /usr/sbin, comum a todas as estações. O segundo pode ser instalado localmente no diretório /usr/local/openoffice, pois se trata de um aplicativo pesado, podendo aumentar o tráfego da rede, prejudicando a performance do sistema.

6. Conclusão

A estratégia de terminais inteligentes traz diversas vantagens para o planejamento de um ambiente computacional, em especial para atender a atual demanda de instalação de Telecentros e laboratórios educacionais. O script RBL é mais uma alternativa em Software Livre possibilitando a criação de ambientes computacionais baratos, funcionais e eficientes, capazes de atender com qualidade as necessidades dos mais diversos segmentos.

7. Referências

- McQuillan, Jim, (2004) "Introduction to the Linux Terminal Server Project (LTSP)", In: V Fórum Internacional do Software Livre, palestra agendada para 3/6/2004. Veja também <http://www.ltsp.org>
- Yu, William and Cimafranca, Dominique (2004) "Desktop Guerrilla Tactics: a Portable Thin Client Approach", <http://www.linuxjournal.com/article.php?sid=7109>, Abril.
- Leung, C. T., (2004), "A computer Lab with no Windows, part I", <http://www.linuxjournal.com/article.php?sid=7418>. Fevereiro.
- Leung, C. T., (2004), "A computer Lab with no Windows, part II", <http://www.linuxjournal.com/article.php?sid=7419>. Fevereiro.
- Brin, Leon (2004) "Linux-based X Terminals with XDMCP" <http://www.linuxjournal.com/article.php?sid=6713>. Maio.
- Blake, kuo-Lien Huang, Steven Shiau and H. T. Wang, "Diskless Remote Boot Linux", (2004) <http://drbl.nchc.gov.tw/>

Agradecimentos

Trabalho de desenvolvimento tecnológico realizado em colaboração com a equipe LibertasBR, sob a coordenação dos profs. Wagner Meira, Dorgival Neto e Renato Ferreira, DCC-UFMG. Suporte financeiro: Instituto Doctum - Educação e Tecnologia e CNPq - Conselho Nacional de Pesquisa.