FMA 215

Aula 15: Uso do X11 em Rede

Prof. Jorge L. deLyra

09 de Setembro de 2002

1 Alguns Conceitos Relevantes

O sistema X11 é um sistema de janelas que permite tanto o uso local de janelas em uma máquina quanto o uso de janelas de forma distribuída através da rede. A rede de que se trata aqui é, de forma geral, a Internet, mas nesta aula vamos estar usando o sistema no âmbito de uma rede local de alta velocidade, como é o caso aqui do Instituto de Física.

Vamos nos referir aqui, com frequência, ao conceito de "desktop". Para nós um desktop será um conjunto de janelas, gerenciado por um programa administrador de janelas (window manager), no âmbito do sistema X11. É o tipo de estrutura que normalmente aparece no monitor de uma máquina quando a usamos através de um sistema gráfico de janelas.

Nesta aula vamos assumir que o aluno já conheça o uso básico do sistema de janelas X11 para o caso de janelas locais em uma máquina. Assumimos, por exemplo, que os conceitos de foco e de mudança de posição e tamanho de janelas sejam bem conhecidos e as técnicas correspondentes dominadas.

Vamos fazer uma pequena exploração das operações básicas de abertura de janelas remotas através da rede e, em seguida, tentar ilustrar as possibilidades de uso de versões extendidas do sistema X11 para apoio a atividades cooperativas de trabalho interativo em equipe através da rede.

Idealmente este tipo de aplicação deveria incluir também a parte de transmissão de voz e som através da rede. Entretanto, ainda estamos em um estágio um pouco primitivo, tanto no que diz respeito à disponibilidade do hardware adequado quanto à disponibilidade de software. Os novos drivers de placas de som e a nova arquitetura de som do Linux, que são apropriados para este tipo de uso, já foram incluídos na versão experimental 2.5 do kernel, mas ainda não houve um release estável deste subsistema. Assim, estaremos nos limitando ao compartilhamento de janelas e aplicativos, bem como à comunicação através da palavra escrita.

Além disso, como nesta aula estamos interessados apenas em ilustrar as possibilidades, vamos usar diretamente as contas de usuário dos alunos, em caráter temporário, para os nossos exercícios cooperativos. Para o trabalho cooperativo de grupos de várias pessoas a melhor alternativa é a definição no sistema de contas especiais de grupo, com nomes apropriados, definindo-se desta forma grupos aos quais os usuários pertençam. Desta forma o grupo tem a sua própria área de disco, acessível a todos os membros, que é "território neutro", não sendo a conta pessoal de nenhum dos usuários em particular.

2 Tarefas

Este conjunto de tarefas está dividido em duas partes. A primeira parte deve ser feita por cada aluno individualmente. Para realizar a segunda parte é necessário trabalhar em pares. Para a maior parte das tarefas desta parte é conveniente, mas não necessário, que os dois usuários estejam utilizando os terminais da sala Pró-Aluno.

É bom lembrar que as máquinas da Pró-Aluno estão em uma rede fechada e que não são acessíveis diretamente da Internet. Entretanto, o servidor `socrates` faz parte das duas redes, de forma que, se o desktop virtual VNC for aberto nele, então este desktop será acessível tanto de dentro quanto de fora da rede da sala, permitindo assim que as tarefas sejam realizadas por um aluno dentro da sala e outro em alguma máquina da rede do Instituto, por exemplo.

2.1 Parte Individual

1. Abra uma sessão X11 em uma máquina da sala Pró-Aluno. Verifique os valores das variáveis de environment `HOST` e `DISPLAY`. Rode o programa `xclock` e, se tudo correr bem e ele entrar, elimine-o em seguida. Desta forma testamos que as janelas estão sendo abertas normalmente em seu sistema local.

2. Elimine a variável `DISPLAY` executando o comando `unsetenv DISPLAY`. Verifique que teve sucesso tentando ecoar na tela o valor da variável. Tente rodar de novo o programa `xclock`, o que acontece desta vez? Identifique o nome da sua máquina local e rode mais uma vez o `xclock`, desta vez da forma

   ```
   xclock -display <hostname>:0.0
   ```

   onde você deve substituir o nome (hostname) da máquina local. Se o `xclock` entrar, elimine-o antes de continuar.

3. Use agora o `setenv` para restituir a variável `DISPLAY` com o valor adequado, usando

   ```
   setenv DISPLAY <hostname>:0.0
   ```

   Em seguida rode o programa `xclock` sem opções. O que acontece desta vez? Se o `xclock` entrar, elimine-o antes de continuar.

4. Abra um segundo `xterm` em sua máquina e dentro dele faça um login remoto no servidor `socrates`. Para ajudar a identificar o terminal que está no `socrates`, você pode usar o comando com a opção `-title`, da forma

   ```
   xterm -title socrates &
   ```

   Uma vez no `socrates`, verifique os valores das variáveis `HOST` e `DISPLAY`. Tente rodar um `xclock` e verifique o que acontece. Se o `xclock` entrar, elimine-o antes de continuar.

5. Elimine a variável `DISPLAY` em sua seção do `socrates` com `unsetenv DISPLAY`. Verifique que teve sucesso tentando ecoar na tela o valor da variável. Tente rodar de novo o programa `xclock` e veja o que acontece. Rode mais uma vez o `xclock`, desta vez da forma

   ```
   xclock -display <hostname>:0.0
   ```

   onde você deve substituir o nome (hostname) da máquina local. O que acontece desta vez? Se o `xclock` entrar, elimine-o antes de continuar.

6. Vá para o outro terminal, que está com a sua sessão na máquina local, e autorize o servidor X11 que está controlando o seu monitor a receber conexões que partam do servidor `socrates`, usando o comando

   ```
   xhost +socrates
   ```

A partir deste momento o sistema `socrates` está autorizado a abrir janelas na tela X11 do seu sistema local. Volte ao terminal da sessão no `socrates` e tente de novo executar o comando que havia tentado lá anteriormente. Se o `xclock` entrar, elimine-o antes de continuar.

7. Volte ao terminal de sua máquina local e desautorize quaisquer conexões remotas usando o comando `xhost -`. Em seguida tente de novo abrir o `xclock` a partir do `socrates`. Depois de ver o que acontece, volte a autorizar conexões do `socrates` em sua máquina local. Elimine todos os processos `xclock` que estiverem rodando antes de continuar.

8. Vá ao terminal do `socrates` e use o `setenv` para restituir a variável `DISPLAY` com o valor adequado, usando

   ```
   setenv DISPLAY <hostname>:0.0
   ```

   Em seguida rode o programa `xclock` sem opções. O que acontece desta vez? Se o `xclock` entrar, elimine-o antes de continuar.

2.2 Parte Cooperativa

Para esta parte das tarefas, ache um colega para fazê-las com você. Os dois precisam estar em sessões de X11 em sistemas na rede e ter acesso ao servidor `socrates`. Os sistema locais utilizados devem ter boa conectividade com a rede onde está o servidor `socrates`, para tornar viáveis na prática estes exercícios. Quaisquer máquinas na rede local do Instituto ou em qualquer ponto da rede da USP satisfazem a este quesito.

Máquinas residenciais dificilmente serão satisfatórias para a execução destas tarefas, a não ser que os participantes tenham em casa um dos serviços de internet rápida que estão disponíveis hoje em dia. Mesmo que este seja o caso, é necessário também que os sistemas residenciais e os sistemas de conexão com a Internet estejam bem instalados e configurados. Os sistemas precisam ter todo o software necessário instalado, enquanto a rede precisa estar permitindo o tráfego dos protocolos necessários.

1. Cada um dos dois deve abrir uma sessão no X11 de um sistema local. A primeira providência é estabelecer um sistema de comunicação entre vocês. Para isto, use o talk: um dos dois descubra quais são o username e o hostname da máquina do outro e digite num terminal

   ```
   talk <username>@<hostname>
   ```

O outro deve responder seguindo as instruções que aparecem na tela. Com isto, aparecerão no terminal duas janelas, cada uma mostrando o que digita um dos dois parceiros da conversação. Mantenham esta janela aberta e reservada para esta finalidade.

2. Cada um dos dois parceiros deve abrir a sua sessão X11 para conexões a partir da máquina do outro, usando o comando

   ```
   xhost +<hostname-do-outro>
   ```

   Façam isto, usando o `talk` para se comunicar e passar as informações necessárias. Cada um dos dois deve testar o sucesso da operação abrindo momentaneamente um

   ```
   xclock -display <hostname-do-outro>
   ```

   na tela do outro e confirmando com o colega, através do `talk`, que está tudo funcionando corretamente.

3. Cada um dos dois deve usar os programas `xloadimage`, `xdvi`, `gv`, etc, para mostrar ao outro uma ou mais imagens e textos. Em todos os casos deve ser usada a opção `-display <hostname-do-outro>` e o colega deve confirmar o sucesso da operação usando o `talk`.

4. O mesmo pode ser feito com aplicativos interativos tais como o `emacs`, mas é necessário cuidado, responsabilidade e respeito mútuo para fazer este tipo de coisa, pois o usuário que abre uma janela deste tipo na tela do outro está permitindo a este outro que escreva em sua conta.

   O mesmo é verdade para qualquer programa que permita que se escreva um arquivo, tais como o `xmgr`. No caso do `emacs` e de outros aplicativos que permitam a abertura de uma shell, trata-se de uma abertura completa da sua conta ao seu parceiro. Naturalmente, estamos assumindo que haja completa confiança e respeito entre os dois parceiros, que irão agir de forma responsável.

   Mantendo estes fatos em mente, experimentem um pouco com este tipo de coisa, usando os muitos aplicativos X11 existentes nos sistemas.

5. Como vocês deve ter percebido, esta forma de cooperação através da rede é um tanto incompleta, devido ao fato de que os dois parceiros não podem estar vendo o mesmo desktop simultaneamente. Para corrigir isto, vamos usar o sistema VNC, que é uma extensão do X11 normal.

A primeira coisa a fazer é um dos dois parceiros inicializar um desktop virtual ao qual ambos poderão se conectar ao mesmo tempo. Valerão aqui as mesmas observações sobre confiança e responsabilidade que foram feitas anteriormente. Um dos dois parceiros deve entrar no servidor `socrates` usando o comando `telnet` e, uma vez lá, iniciar um desktop virtual usando o comando

```
vncserver -alwaysshared -depth 16 -geometry 800x600
```

onde a opção `-depth` define a profundidade de cor do desktop que é criado e a opção `-alwaysshared` fará com que o servidor aceite conexões adicionais sem quebrar as conexões existentes, independentemente de se usar ou não a opção `-shared` no cliente, como veremos adiante. O uso do `telnet` em vez do `ssh` para esta função específica é recomendado para evitar problemas devidos ao fato de que o VNC interage mal com os sistemas de autorização e forwarding do `ssh`.

Caso o usuário nunca tenha feito este tipo de coisa antes, o servidor solicitará a definição de uma password de acesso ao desktop. O usuário deve inventar uma password, que pode ser longa mas não deve conter caracteres especiais. Não use a sua password de login no sistema, invente algo bem diferente. O servidor irá informar na tela qual foi o número de display que foi associado a este desktop virtual. O usuário deve marcar este número em algum lugar.

6. O usuário que inicializou o desktop deve passar para o seu parceiro o número de display e a password relativos ao desktop. Caso ele já tenha uma password anterior que não deseje revelar, pode mudar a password para um valor temporário, usando o comando `vncpasswd`, restituindo a password original da mesma maneira ao final dos exercícios.

   Feita a transmissão destas informações através do `talk`, cada um dos dois usuários pode se conectar ao desktop virtual através do comando

   ```
   xvncviewer -shared socrates:<numero-de-display>
   ```

   onde deve-se substituir o número de display fornecido pelo servidor. Cada um dos dois usuários terá de digitar a password do desktop. Caso o servidor tenha sido iniciado *sem* a opção `-alwaysshared`, é importante não esquecer a opção `-shared`, ou o segundo usuário que entrar causará a queda automática da conexão do primeiro.

7. No monitor de cada uma das duas máquinas locais será aberta uma janela de X11 contendo o desktop virtual! Trata-se de uma janela de um desktop (o seu desktop local) contendo um outro desktop, que está rodando no servidor. Este desktop virtual tem o seu próprio cursor gráfico, que é capturado pelo mouse quando este entra na janela. De agora em diante, é importante manter as ações dos dois usuários coordenadas através do canal de comunicação estabelecido pelo `talk`.

   Observem que o controle do mouse e do teclado passa livre e dinamicamente de um usuário para o outro, com base na atividade destes dispositivos. Para que um dos dois usuários possa fazer alguma coisa no desktop sem ser perturbado, é necessário que o outro mantenha o seu mouse fora do desktop e coloque o foco em alguma outra janela do seu desktop local, por exemplo na janela onde está rodando o programa `talk`.

   Experimentem um pouco com a mecânica deste processo. Observe que tudo que um usuário estiver fazendo é agora visível também para o outro, de forma dinâmica.

8. Dependendo do conteúdo dos arquivos de configuração da conta do usuário que iniciou o desktop, partindo do arquivo `.vncrc`, é possível que já existam algumas janelas no desktop virtual. Eliminem todas as janelas já existentes no desktop, caso haja alguma, dando `exit` nos terminais ou eliminando editores e outros possíveis aplicativos com os comandos apropriados de cada um.

9. Vamos agora abrir terminais dentro do desktop a partir de fora, um de cada um dos dois usuários. Primeiramente o usuário que inicializou o desktop deve abrir o seu terminal e, dentro dele, dar o comando `xhost` apropriado para permitir conexões a partir da máquina de seu parceiro. Note que o desktop VNC é de fato um servidor X11 como qualquer outro, a menos de não estar controlando um monitor diretamente, de forma que todas as operações relativas a ele devem se processar como de hábito com um servidor X11. Como já fizemos antes, devemos executar neste caso

   ```
   xhost +<hostname-do-outro>
   ```

   Para abrir os terminais basta executar, em algum outro terminal de seus desktops locais, o comando

```
xterm -display socrates:<numero-de-display>
```

onde deve ser substituído o número de display do desktop. É conveniente usar também uma ou mais das opções `-rv`, `-fn 7x14bold` e `-geometry 80x24`. A fonte discriminada aqui é particularmente recomendada, dadas as dimensões um pouco reduzidas do desktop que foi criado.

10. Observe que cada um dos dois terminais está rodando na conta do respectivo usuário, com autorização de escrita. Se quiserem, vocês podem agora iniciar uma sessão de `talk` dentro de um destes terminais, eliminando a necessidade de ter uma sessão de `talk` externa. Com ainda maior simplicidade, basta cada um abrir um editor `emacs` em modo "scratch" e escrever dentro, pois ambos verão o que cada um escrever. Para abrir um `emacs` dentro do desktop pode-se usar

    ```
    emacs -display socrates:<numero-de-display>
    ```

    tanto na janela externa quanto na janela que está rodando dentro do desktop. Aqui também deve ser útil usar algumas das opções adicionais mencionadas anteriormente. Experimentem algumas destas alternativas de forma a estabelecer um canal de comunicação mais conveniente.

11. Utilizem o editor `emacs` (na realidade, também pode ser qualquer outro editor) para fazer uma edição de texto da qual ambos possam participar, se revezando nos trabalhos de digitar e de monitorar o que o outro está fazendo. Para este tipo de coisa fica particularmente claro quão útil seria um canal de voz pela rede. Utilizem outros aplicativos para mostrar de forma compartilhada diversas figuras e gráficos.

12. Tentem rodar alguns screen savers dentro do desktop para ver se tudo funciona como se espera. Tentem, por exemplo, o comando

    ```
    xlock -nolock -mode penrose
    ```

    Experimentem também com outros modos do screen saver. Tentem também jogar algum jogo se cartas, como o "free cell" ou algo parecido. Será que vocês conseguem cooperar num jogo deste tipo através deste sistema?

13. Experimentem tentar mudar o gerenciador de janelas dentro do desktop. Será necessário achar e eliminar o processo correspondente ao `fvwm`

que já está rodando e em seguida iniciar algum outro gerenciador de janelas no background, tal como, por exemplo, o `afterstep`, o `olwm`, `icewm` ou algum outro. Depois de experimentar um pouco, talvez seja melhor voltar para o `fvwm`, que deve estar mais bem configurado para este tipo de desktop.

14. Uma das características importantes do sistema VNC é que trata-se de um sistema de desktops persistentes, ou seja, o desktop permanece em operação enquanto o servidor onde ele está rodando não cair ou for rebootado, mesmo se as conexões com os clientes forem interrompidas por qualquer motivo. Ao voltar, os clientes encontrarão o desktop exatamente na situação em que o deixaram. Experimentem quebrar suas conexões com o servidor. Isto pode ser feito de diversas formas: matando-se o processo do `xvncviewer` com um `[Ctrl]-C`, usando-se os botões da barra superior da janela onde está o desktop ou usando-se o comando `kill`. Depois, voltem a estabelecer as conexões e verifiquem que o desktop está exatamente como o deixaram.

3 Problemas e Dicas

1. É possível usar o `xvncviewer` em modo "read-only", através da opção `-viewonly`. Claramente, esta é uma forma interessante de organizar aulas através da rede, seja envolvendo máquinas remotas, seja envolvendo máquinas em uma sala de aula informatizada. Tentem organizar um grupo no qual uma pessoa faz uma apresentação que vai ser assistida pelas outras. Vamos descobrir com que número máximo de participantes ainda é prático fazer este sistema funcionar.

2. Para fazer uma coisa mais aventurosa, com resultados que talvez não saibamos prever com tanta facilidade, que tal tentar abrir um programa `xvncviewer` dentro de outro que já esteja rodando? A forma mais imediata de se fazer isto é definir dois desktops, digamos um de tamanho `800x600` e outro de tamanho `640x480`, abrir um `xvncviewer` com o desktop maior e depois outro dentro deste com o desktop menor. Mas o que acontece se abrirmos os dois processos `xvncviewer` com o *mesmo* desktop? Experimente! E depois descubra como sair desta!

4 Uma Lista de Comandos Relevantes

`fvwm` : um administrador de janelas (window manager) altamente configurável pelo usuário; existem duas versões muito parecidas, chamadas `fvwm1` e `fvwm2`, uma mais antiga e outra mais recente, cujos arquivos de configuração adotam sintaxes um pouco diferentes, de forma que as configurações de uma podem não funcionar muito bem com a outra; em geral aquela versão que é o default do sistema se chamará simplesmente `fvwm`.

`icewm` : um administrador de janelas (window manager).

`olwm` : um administrador de janelas (window manager).

`vncpasswd` : muda a password de um desktop VNC.

`vncserver` : inicializa um desktop virtual VNC.

`xhost` : permite o controle de acesso de outros hosts a um servidor X11.

`xloadimage` : mostra uma imagem dentro de uma janela ou no fundo da tela do sistema X11.

`xlock` : funciona como "screen saver" ou como "screen lock".

`xmgr` : um aplicativo para a plotagem de gráficos e para a manipulação numérica de dados.

`xvncviewer` : o cliente VNC, abre uma janela contendo uma imagem de um desktop virtual.