{páginas on-the-fly}

ALERTA: No IE 4.0 algumas páginas com os programas (arquivos .TXT) aparecem todas "emboladas", sem separação de linhas. Páginas com extensão .TXT e comandos HTML são executadas como .HTML! (Parabéns, Microsoft!) No Navigator 4.0 isto aparece OK, mas exemplos de tags HTML aparecem erradas. O curso é melhor visto no Navigator 3.0! Isto é a evolução dos browsers, nesta guerra para ver quem ganha mais dinheiro...

Este não é propriamente um curso de PERL mas um curso de CGI com alguma coisa de PERL.

Muitos dos conceitos de CGI aqui apresentados servirão nos cursos de Java.

Na realidade PERL não é nossa linguagem predileta. Nossa escolha hoje em dia é pelo Java.Mas, se seu provedor de espaço não lhe permite disparar aplicações servidoras em Java, mas apenas em PERL, você deve aprender um pouco desta linguagem para poder fazer suas aplicações.

Vamos usar no curso o arquivo CGI-LIB que não é muito do gosto de puristas em PERL, mas funciona... Na última aula ensinaremos como trabalhar sem o acréscimo deste arquivo.

1.1 UMA CONVERSA

Na Internet (no TCP/IP) temos sempre uma conversa entre uma aplicação cliente (um browser geralmente) e uma aplicação servidora (noutra máquina).Existem vários tipos de aplicações servidoras.

A aplicação servidora mais comum é aquela que responde a um pedido assim, do cliente:

http://domain-name/paginaX.html

Automaticamente tres coisas são feitas:

1. É enviada ao cliente uma primeira linha assim:
   "Content-type:text/html"
2. Depois (esse o grande mistério da Internet) é enviada UMA LINHA EM BRANCO!!! O interessante é que se essa MISTERIOSA LINHA EM BRANCO não for enviada nada funciona.
3. Finalmente o servidor que habita o tal domain-name procura em suas entranhas um arquivo chamado paginaX.html e envia cada uma das linhas para o cliente.

Esses arquivos com extensão HTML têm linhas mais ou menos assim (você já deve estar cansado de saber isso):

```
&ltHTML>&ltHEAD>

&ltTITLE&gtHello!</TITLE>

</HEAD>&ltBODY>

HELLO PERL!

</BODY></HTML>
```

A aplicação cliente recebe cada uma dessas linhas. A primeira (chamada:header) ele não projeta. Nem a MISTERIOSA LINHA EM BRANCO. Mas essa tal MISTERIOSA LINHA EM BRANCO serve para dizer que, a partir dela, as próximas linhas deverão ser interpretadas e projetadas na tela.

1.2 OUTRA CONVERSA

Para um computador poder ser um servidor que faz essa mágica daí de cima não é preciso muita coisa. Ele tem que estar rodando uma aplicação feita segundo um conjunto de normas (um protocolo) chamado HTTP. Se essa aplicação HTTP tiver mais um conjunto de atividades definidas pelo protocolo CGI, essa máquina pode fazer uma mágica melhor ainda:

Quando o cliente mandar para ela o string:

http://domain-name/programaX.exe

ela executa o tal programaX.

Legal,não?

Agora,só para contrariar, eu posso querer que esse programaX faça uma coisa semelhante ao caso anterior: mande para o cliente uma página-da-Internet em que vai estar escrito:HELLO PERL!

Vamos imaginar que o CGI tem um espaço mágico em que,tudo que é escrito nele é enviado para o cliente que disparou o programa.

Então fica fácil. É só o programaX escrever/printar nesse lugar:

1. O header:

   "Content-type:text/html"

2. Depois "escrever" a MISTERIOSA LINHA EM BRANCO. O que significa que depois da linha anterior se dá DOIS "retorno de carro" ou "`a linha".

3. Finalmente é printada cada uma das linhas:

   &ltHTML>&ltHEAD>

   &ltTITLE&gtHello!</TITLE>

   </HEAD>&ltBODY>

   HELLO PERL!

   </BODY></HTML>

Na realidade não precisamos dar um "retorno de carro" depois de cada linha porque no HTML todo o arquivo poderia ser uma linha só! Essa divisão em "linhas" pode ser feita apenas para ajudar o entendimento de quem lê o arquivo fonte.

1.3 COMO CRIAR UM PROGRAMA QUE FAZ ESSA BOBAGEM EM PERL

É claro que uma página-da-Internet com a frase: HELLO PERL! pode ser feita de maneira mais simples usando HTML. Mas para você começar a entender como se faz "páginas on-the-fly" vamos

ver E FAZER essa página através de um programa na linguagem PERL. Sugerimos que você abra um arquivo em branco com um editor ASCII (NotePad, por exemplo). Depois chame a página (veja logo aí embaixo) que contém o programa e usando COPY-AND-PASTE a copie nesse arquivo. Você vai poder depois passá-lo para seu servidor UNIX (não como você faz com uma página HTML - vamos falar disto daqui a pouco). Além disso pode colocar o programa sobre a janela de seu browser para acompanhar a explicação passo a passo que vamos dar.

Sempre que você for para uma página-da-Internet com programa, para voltar ao curso use BACK. Vamos experimentar? Clique então aqui para ver e copiar o programa

1.4 OLHA AÍ NA JANELA

A primeira linha que você deve estar vendo aí na janela do seu NotePad (você fez o que dissemos,não?) serve para dizer aonde está o interpretador PERL na máquina de seu provedor de servidor. É assim que se designa um PATH em UNIX.

Acontece que esse PATH é de onde o pessoal da HIWAY (nosso provedor de espaço - "web hosting") coloca o interpretador PERL. Você tem que ver com o seu onde é que está o dito cujo (se ele não tiver isso instalado peça para instalar!).

É bom comentar que a linguagem PERL é interpretada e não compilada como o C++, o DELPHI etc. Por isso a performance de programas PERL é baixa. Ele só deve ser usado em coisas simples. Se você é do BRADESCO ou do ITAÚ use C++ ou Assembler (é ruim, eihn?...)

Na segunda linha além do que é printado temos os dois "retorno de carro" que provocam a "impressão" da MISTERIOSA LINHA EM BRANCO. Isso é feito com dois "scapes":

\n\n

NUNCA SE ESQUEÇA DESSE DETALHE! SENÃO NADA FUNCIONA. E você fica se descabelando achando que seu programa tem algum erro esquisito. Apenas você se esqueceu dessa bobagem...

O resto do programa é óbvio. Repare que no fim de cada linha de PERL você tem que colocar ponto-e-vírgula. Se você está acostumado com Visual Basic, vai esquecer e o programa vai dar erro. Aliás, fazer debuging de programa CGI/PERL é meio complicado porque você não tem uma máquina UNIX. Existe PERL para WINDOWS mas na realidade é para DOS e roda de maneira não gráfica. Assim, analise bem seu programa antes de mandar para o servidor. Se der erro você tem que tentar corrigir e mandar de novo. Enquanto isso o relógio está rodando e grana entrando no bolso do provedor, da telefônica etc..

Nada melhor para terminar esse capítulo do que rodar o programa que colocamos no nosso servidor. Espero que você depois de vê-lo (não é grande coisa) tente colocar o seu no seu servidor e executá-lo.

1.5 COMO TRANSPORTAR

Atenção: você tem que ver com seu provedor de "web hosting" qual o diretório onde você pode colocar programas PERL. NÃO É O MESMO ONDE VOCÊ COLOCA AS PÁGINAS HTML. Geralmente ele se chama: CGI-BIN. No caso da HIWAY ele se chama CGI-LOCAL.

Estes provedores gratuítos tipo: Geocities, Internetclub etc. normalmente não tem este recurso...

Para transferir o arquivo você usa o modo ASCII e não BINÁRIO como faz com as páginas HTML e figuras. Na HIWAY isso é muito fácil pois existe um programa, o WS-FTP (muito comum, por sinal) com interface visual em que apenas se clica na opção de transmissão como BINÁRIO. Peça

a seu provedor para explicar direitinho como se faz no seu caso. Se ele não explicar, mude para a HIWAY (não ganho comissão - juro!).

É costume colocar como extensão do arquivo de programa:pl. Mas tem gente que usa:cgi.Tanto faz. Vamos chamá-lo de prog1.pl.

Você tem que executar o programa colocando:
http://seu-domain-name/cgi-local/prog1.pl

Se a pasta for:CGI-BIN use isso,claro.Você pode referenciá-lo de uma página com &ltA HREF etc.

Boa sorte e tente rodar esse programa.