

# *Desenvolvendo Sistemas na Internet*

*Marcos Villas\**

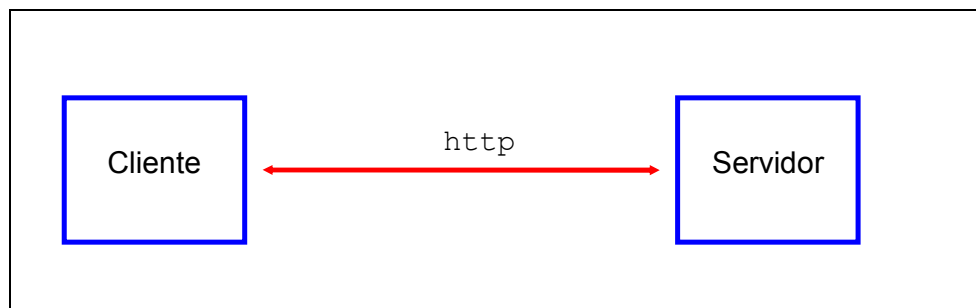
Há pouco tempo, sistemas eram implementados por grandes programas, acionados por comandos em lote ou através de terminais com baixa (ou nenhuma) capacidade gráfica. Com o surgimento dos micros e sua interligação em rede, aparece um novo paradigma: a arquitetura cliente/servidor. Paralelamente, a tecnologia de rede TCP/IP sai do seu habitat originalmente acadêmico e conquista o mundo, através da Internet.

Este artigo descreve e exemplifica a tecnologia existente na Internet para desenvolvimento de sistemas, em especial, a tecnologia relacionada ao WWW. Serão descritos os componentes da sua arquitetura cliente/servidor e apresentadas características de dois ambientes operacionais para o desenvolvimento de sistemas: Unix e Windows NT. Finalizando, mostraremos algumas dicas peculiares a sistemas desenvolvidos para a Internet.

Todos os recursos apresentados podem ser usados para sistemas na Internet. Aplicações internas à uma empresa (intranet) também podem ser desenvolvidas usando a mesma tecnologia, bastando para isto que esta empresa tenha uma rede TCP/IP.

## *Sistemas em Rede TCP/IP*

O modelo mais usado atualmente em sistemas de uma rede TCP/IP é o de cliente/servidor, com interface gráfica e multimídia, sem sessão, implementado através da tecnologia existente no WWW (World Wide Web): o cliente WWW (*browser*), o protocolo HTTP e o servidor WWW. [figura 1]



*Figura 1: Elementos básicos do WWW*

Normalmente cliente e servidor estão em computadores diferentes, ambos conectados à mesma rede TCP/IP, como por exemplo a Internet.

---

\* *Marcos Villas (villas@rsi.com.br) é diretor da RSI - Redes e Sistemas de Informação (http://www.rsi.com.br) e professor do Departamento de Informática da PUC-Rio.*

## *O Protocolo HTTP*

O HTTP (Hypertext Transfer Protocol) é o protocolo de comunicação entre cliente e servidor no WWW. Toda informação transmitida entre cliente e servidor possui um cabeçalho. É neste cabeçalho que estão os comandos (métodos) que indicam as ações a serem tomadas pelo servidor ou pelo cliente.

A situação mais comum de utilização do WWW é o cliente “pedir páginas de informação” a um servidor e, a partir de “âncoras” (links), recuperar outras páginas. Estas páginas de informação são codificadas em HTML (Hypertext Markup Language). Elas residem no servidor e são transferidas para o cliente a seu pedido, através do HTTP. Uma vez no cliente, os comandos desta linguagem são interpretados e seu resultado apresentado ao usuário.

Neste cenário, ao pedir uma página a um servidor, o cliente lhe envia um cabeçalho com o método GET indicando a página ser recuperada. O servidor responde com um cabeçalho indicando o tipo da informação (por exemplo, um arquivo HTML) e a página em si. Em outras situações outros métodos são usados. Por exemplo, ao enviar um formulário do cliente para o servidor, o método normalmente utilizado é o POST.

O Netscape, e mais recentemente outros *browsers*, permitem a manipulação de *cookies*, que são informações armazenadas pelo servidor no cliente. Através de cabeçalhos específicos é possível armazenar no cliente qualquer informação, como a última página visitada e a data desta visita. Por definição, não é possível para um servidor recuperar informações armazenadas pelos outros servidores.

No protocolo HTTP não existe o conceito de sessão, necessário a muitos sistemas (ex.: comércio eletrônico, *home banking*). Existem várias formas de simular uma sessão em HTTP, e todas elas passam pela criação, pelo servidor, de uma identificação da sessão. Esta informação precisa ser recuperada a cada solicitação do cliente. As formas tradicionais são o uso de campos escondidos (*hidden*) em formulários, de *cookies* e de informações adicionais no caminho (*path*) que indica o programa a ser acionado no servidor.

## *Programas no Servidor*

Muitos dos sistemas implementados na Internet possuem programas no computador servidor, que são ativados pelo servidor WWW. [figura 2]

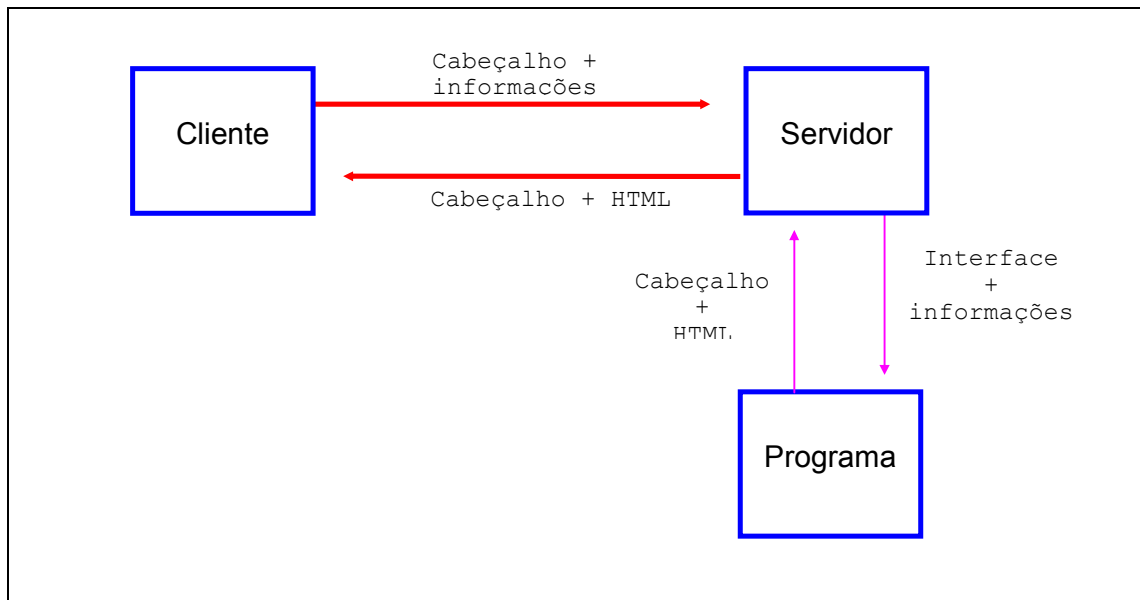


Figura 2: Comunicação entre Servidor e Programa

Várias informações são passadas do servidor WWW para estes programas. Estas informações podem ser divididas em dois grupos: (1) informações de aplicação, passadas pelo cliente, por exemplo através de um formulário eletrônico, que dizem respeito à aplicação e (2) informações do protocolo sobre a comunicação entre o cliente e o servidor.

Um formulário eletrônico é definido pelo “comando” (*tag*) <FORM>. Ele permite implementar a “entrada de dados” de um sistema na Internet. No formulário, os campos podem ser definidos para aceitar texto (uma linha, várias linhas, uma senha), itens selecionáveis (*check boxes*, lista de valores), itens selecionáveis mutuamente exclusivos (*radio buttons*), imagens e até mesmo arquivos. Cada campo tem um nome, e são os pares “nome = valor” que são enviados do cliente para o servidor, quando o formulário é submetido. O programa no servidor que receberá estas informações é definido através de uma URL (ex.: <http://www.bvrj.com.br/desafio/ranking.dll>).

Como exemplo de informações do protocolo temos o número IP do cliente, o método usado, o browser usado pelo cliente e a identificação do usuário (caso seja um acesso identificado).

O programa no computador servidor recebe estas informações segundo uma interface. A mais comum delas chama-se CGI (Common Gateway Interface). Programas que usam esta interface são denominados programas CGI. Outros tipos de interface são implementados por APIs para comunicação com o servidor: WSAPI para o WebSite (O’Reilly), NSAPI para o Netscape e ISAPI para o IIS (Microsoft). As APIs permitem estender a funcionalidade dos servidores, permitindo customizar o processamento de *logs* de acesso, filtros para tipos especiais de documentos etc.

Estes programas podem ser codificados em qualquer linguagem, porém devem ser compatíveis com o ambiente operacional do servidor. Em UNIX é comum termos programas desenvolvidos em PERL e C; em Windows NT, Visual Basic e Delphi.

### *Programas no Cliente*

Os *browsers* evoluíram ao ponto de permitirem a execução de programas no próprio cliente. Estes programas são codificados em linguagens interpretadas, executadas por seus respectivos interpretadores. Existem dois modelos básicos:

- *Código fonte embutido*: pequenos programas escritos em JavaScript (Netscape) ou VB Script (Microsoft) são colocados em uma página HTML através do *tag* <SCRIPT>. No *tag* é indicado qual o interpretador a ser usado.
- *Código objeto não embutido*: programas escritos em Java (Sun) ou segundo o ActiveX (Microsoft) têm seu código objeto interpretado por “máquinas virtuais”, que garantem a segurança do acesso. Estes programas são carregados através do *tag* <APPLET>.

Programas no cliente são usados em muitos casos como “entrada de dados” para outros programas no servidor. Eles estendem a funcionalidade dos formulários eletrônicos do HTML, permitindo críticas mais elaboradas e interface interativa diferenciada. Outra característica destes programas no cliente é a de poder se comunicar com outros servidores, via TCP/IP, que não o servidor WWW. Por exemplo, um programa no cliente pode se comunicar com um servidor de banco de dados para atualizar dados.

### *Ambientes Operacionais*

O TCP/IP é um protocolo para rede de comunicação de dados que teve a sua origem em ambientes operacionais UNIX. Este é o ambiente mais flexível, testado e usado para desenvolvimento de aplicações Internet. O servidor mais popular no momento é o Apache (sucessor do NCSA), pois além de ser muito bom e ter o código-fonte em C disponível, é gratuito. Em UNIX, o protocolo entre servidor e programa normalmente é o CGI, com as informações disponibilizadas para o programa ou via variáveis de ambiente ou via entrada padrão (*standard input*).

Em ambiente Windows NT os principais servidores são o WebSite, Netscape e IIS. O IIS, que era um sistema em separado (EMWAC), passou a fazer parte do Windows NT. Entre o servidor e os programas podem ser usados os protocolos WinCGI (as informações são colocadas em um arquivo do tipo INI, com sessões e pares de variável = valor) ou APIs próprias.

## *Melhorando a Aplicação*

Um sistema desenvolvido especialmente para a Internet tem algumas peculiaridades não encontradas em outros sistemas:

- Embora, do ponto de vista de quem projeta a interface de um sistema, a navegação pareça ser hierarquizada e/ou dirigida através das âncoras existentes em cada página, qualquer página e/ou programa pode ser referenciada imediatamente, bastando para isto indicar a sua URL (seu endereço). Ou seja, um usuário pode ou não seguir uma seqüência pré-definida de páginas.
- Todo acesso a páginas e/ou programas WWW é registrado em um *log*, mantido pelo servidor WWW. Através deste *log* é possível não só contabilizar os acessos realizados e/ou preparar estatísticas de utilização (exs: que páginas são mais acessadas; qual o horário de maior utilização), mas também identificar uma seqüência de acessos, de forma a determinar que páginas são realmente acessadas em uma “sessão de utilização”, muito embora tecnicamente não exista este conceito no WWW.
- Programas desenvolvidos para a Internet podem e devem ser integrados a outros sistemas e/ou serviços Internet. Por exemplo, um programa que aceita pedidos de produtos via Internet pode enviá-los diretamente a uma base de dados, ao mesmo tempo que envia um *email* ao responsável pelo acompanhamento do pedido.

### *Exemplo: Desafio da Bolsa de Valores do Rio de Janeiro*

O Desafio da Bolsa ([www.bvtj.com.br/desafio](http://www.bvtj.com.br/desafio)) é um sistema de simulação de investimentos no mercado de ações que permite ao seu usuário vivenciar o dia a dia deste mercado num ambiente real onde, exceto a decisão de comprar ou vender, que é do participante, todos os demais dados e eventos (empresas, papéis, cotações, direitos atribuídos às ações) são reais. Existem “ciclos” de simulação, que duram alguns meses.

O computador onde está o Desafio usa Windows NT 4.0, com servidor IIS 2.0. Seus programas foram desenvolvidos com Visual Basic 4.0 com protocolo WinCGI. Disponíveis aos participantes existem programas para cadastramento, operações, consulta a carteira de ações e consulta ao ranking de desempenho dos participantes. Outros programas, usados apenas pela BVRJ, fazem atualização diária dos dados reais e geram estatísticas sobre os participantes de um ciclo, entre outros.

Estatísticas de todo o *site* da BVRJ, incluindo o Desafio, são produzidas a partir de *logs* mensais por programas PERL.