

TÉCNICAS DE PROGRAMAÇÃO DE *HOME-PAGES* E TUTORIAIS MULTIMÍDIA EM HTML¹

PAULO CÉSAR DA COSTA PINHEIRO

Departamento de Engenharia Mecânica da UFMG

Av. Antônio Carlos 6627; 31270-901 Belo Horizonte, MG

Fone: (31) 3499-5451; Fax: (31) 3443-3783

e-mail: pinheiro@dedalus.lcc.ufmg.br pinheiro@demec.ufmg.br

RESUMO - O uso de sistemas multimídia tem se tornado a cada dia mais comum no ensino, permitindo a automatização das atividades repetitivas do ensino: estudos dirigidos, instrução programada, aplicação e correção de provas (fechadas), apresentação de imagens etc. Uma ferramenta muito interessante de se criar tutoriais multimídia é a linguagem HTML (Hyper Text Markup Language), utilizada para desenvolver as *home-pages* da Internet. O HTML é uma linguagem interpretada de descrição de páginas que possui um conjunto completo de padrões para estruturação de dados em multimídia, incluindo gráficos, navegação hipertexto, som, vídeo etc. O HTML é simples e de fácil implementação. Os tutoriais desenvolvidos em HTML podem ser utilizados em qualquer computador e sistema operacional (multiplataforma), e distribuídos diretamente pela Internet. Devido a estas características, o HTML deve se tornar o padrão mundial de multimídia. Este trabalho apresenta a metodologia do desenvolvimento de *home-pages* e tutoriais multimídia para o ensino de engenharia, usando como ferramenta o HTML.

1. INTRODUÇÃO

A multimídia combina uma interface amigável com a habilidade do computador de controlar diversos periféricos, aumentando a interação usuário/software. Ela tornou-se popular a partir de 1987 graças ao Macintosh e ao Software HyperCard.

O hipertexto é a forma mais comum de multimídia, e tem sido amplamente utilizado na construção de tutoriais para o ensino de engenharia auxiliado por computador (CAE). O hipertexto cria uma árvore de conhecimento, possibilitando a navegação amigável dentro de um banco de dados, de modo a encontrar as informações que se deseja. Uma ferramenta muito interessante de se criar documentos hipertextos com recursos multimídia é a linguagem HTML (Hyper Text Markup Language), utilizada para desenvolver *home-pages* na Internet.

A HTML foi criada em 1990 por Tim Berners-Lee do CERN (Centre Européen pour la Recherche Nucleaire, Suíça). A simplicidade e facilidade de implementação das *home-pages* em HTML, e o desenvolvimento do *browser* (software gráfico para a manipulação destas informações) Mosaic, em 1992, pelo NCSA (Centro Nacional de Aplicações em Supercomputação) ajudou muito na popularização da WWW. Estava assim criada a WWW (Web), serviço mais utilizado na Internet e, sem dúvida, o responsável por sua difusão comercial.

O HTML é uma linguagem interpretada de descrição de páginas que possui um conjunto de padrões para estruturação de dados em multimídia, incluindo navegação hipertexto, botões, tabelas,

¹ PINHEIRO Paulo César da Costa. Técnicas de Programação de Home-pages e Tutoriais Multimídia em HTML. In: XXVI CONGRESSO BRASILEIRO DE ENSINO DE ENGENHARIA (COBENGE-98), 18-21 Outubro 1998, São Paulo, São Paulo, Anais... São Paulo: ABENGE, Associação Brasileira de Ensino de Engenharia, 1998, A7v. v.5, p.2303-2322, CD-ROM.

gráficos, som, vídeo etc. Ela é utilizada para compartilhar informações, publicar e transferir trabalhos científicos através da Internet, de uma forma simples e transparente para o usuário, interligando documentos através de vínculos hipertexto. O HTML consiste basicamente de um texto normal com *tags* (funções, atributos) delimitando porções de texto selecionadas, que ensinam ao *browser* formatá-lo de forma particular, inserir figuras, desenhar linhas, abrir outras páginas etc. Um formulário HTML é tanto um documento ativo quanto um aplicativo centrado em documento.

A grande vantagem de desenvolver tutoriais em HTML é que podem ser utilizados em qualquer computador e sistema operacional (multiplataforma), e distribuídos diretamente pela Internet. O único software necessário para se visualizar um aplicativo HTML *browser*. O *browser* em um computador PC (ou Unix ou Macintosh), pode receber a mesma página igualmente de um servidor PC, Unix ou Macintosh, em qualquer lugar do mundo. Como o aplicativo fica disponível na Internet, os usuários podem acessar a versão mais atualizada deste aplicativo. Devido a estas características, o HTML deve se tornar o padrão mundial de multimídia. Assim, como a implementação de um tutorial em HTML é fácil, torna-se possível o desenvolvimento de tutoriais de alto nível para ensino de engenharia.

2. A PROGRAMAÇÃO HTML

O tutorial em HTML pode ser programado a partir de qualquer editor de textos (ASCII) [1, 2, 3, 4, 9, 11], e interpretado e visualizado em qualquer *browser*. A linguagem HTML possui mais de sessenta comandos (*tags*) destinados a formatar a aparência da página (*tags*) com seus elementos (textos e imagens), e criar ligações (*links*) com outras páginas da rede.

Apesar da programação HTML não ser complicada, existem softwares especializados (Web Editors), que facilitam a implementação das *home-pages*, sem ser necessário o conhecimento da linguagem HTML. Estes softwares permitem criar *home-pages* de aspecto profissional, usando técnicas WYSIWYG (What You See Is What You Get). Entre os mais conhecidos estão o PageMill, HotDog, FrontPage, e HoTMetaL. Assim, procedimentos relativamente complexos (tabelas, formulários e links) são implementados facilmente com a ajuda de janelas e menus amigáveis, além de outras características que permitem criar rapidamente páginas Web com ótima apresentação. Eles também possuem tutoriais, exemplos de páginas e guias de referência de HTML. Estas ferramentas tornam fácil o desenvolvimento de uma *home-page*, a qualquer um, com qualquer nível de habilidade. *Wizards* e WYSIWYG estão rapidamente substituindo a programação manual de HTML.

Entretanto, estes softwares WYSIWYG além de gerar muitas linhas de código supérfluas, limitam as possibilidades da programação, podendo tornar frustrante a sua utilização. Sendo assim, é melhor utilizar um software que trabalhe ao nível das *tags* HTML, para controlar melhor os efeitos desejados.

Como cada *browser* tem uma interpretação diferente das *tags*, não existe um verdadeiro editor HTML Wysiwyg. No entanto, a maioria dos editores se aproximam bastante da interpretação das *tags* dos dois *browsers* mais populares: Netscape e Explorer. Pronta a página, recomenda-se visualizar a página no *browser* usual para ver o resultado final.

Claris HomePage	http://www.claris.com
FrontPage	http://www.microsoft.com
Home Site	http://www.allaire.com
HotDog	http://www.sausage.com
HotMetal	http://www.sq.com
Netscape Composer	http://www.netscape.com
W3e	http://www.nce.ufjf.br/~cracky/w3e

A sintaxe de uma *tag* é a seguinte: <TAG>texto</TAG>. Onde <TAG> inicia o comando, </TAG> termina o comando e "texto" é o que será exibido pelo *browser*. Os comandos podem ser escritos tanto em maiúsculo quanto em minúsculo. Alguns desses comandos são do tipo liga-desliga: a primeira ocorrência do comando ativa o seu efeito e a segunda </TAG> o fecha. Se esquecida, o comando continuará ativo. Em outros comandos, somente a primeira parte é necessária, como no caso do comando <P>, usado para terminar um parágrafo.

A estrutura mínima de um programa HTML é composta dos seguintes comandos:

Comando HTML	Operação
<html>	Inicializa o programa
<head>	Início dos comandos do título da página
<title> Texto para Título </title>	Título da página
</head>	Fim dos comandos do título da página
<body>	Início da formatação da página
{ Corpo do Programa em HTML }	
</body>	Fim da formatação da página
</html>	Finaliza o programa

2.1 - Comandos de Estilo

Os Comandos de Estilo permitem formatar a página. Porém, as limitações do HTML não permitem a especificação dos tipos de letras ou estilos. Os únicos atributos de texto que se pode utilizar são tamanho da fonte, negrito, itálico etc, sendo possível sobrepor dois ou mais comandos.

 NEGRITO 	 	 1 a 7;default = 3
<I> <i>ITÁLICO</i> </I>	<i> </i>	<H1> TAMANHO MÁXIMO </H1>
<U> <u>SUBLINHADO</u> </U>		<H2> TAMANHO GRANDE </H2>
<I><U> <i><u>TUDO JUNTO</u></i> </I></U>		<H3> TAMANHO MÉDIO </H3>
 Parágrafo (sem pular linha)		<H5> TAMANHO PEQUENO </H5>
<P> Parágrafo (pula 1 linha)		<H6> TAMANHO MÍNIMO </H6>

É comum o uso de alguns espaços em branco no início dos parágrafos, mas os *browser* ignoram todos os espaços adicionais nos arquivos HTML. Para forçar a existência de espaços pode-se utilizar o comando " " (non-breaking space) ou a *tag* <TAB INDENT=number>, onde o atributo INDENT especifica o número de espaços as serem reservados.

2.2 - Comandos de Ligação

Os *links* são as ligações de palavras-chave com outras páginas, documentos, arquivos ou pontos dentro da mesma página. Pode-se fazer ligações com imagens ou arquivos de som, vídeos etc. Para criar o recurso de hipertexto (*hyperlinks*, palavras-chaves ou imagens que, ao ser clicadas, abrem outros documentos ou arquivos da rede), existem comandos externos (que ligam a uma outra página), e internos (que ligam a um determinado ponto dentro da mesma página).

O comando para *links* externos é: Texto do *link* onde a expressão "endereço" é o local onde se encontra o arquivo, ou a página, a ser acessado. Esse endereço pode ser o nome do arquivo, se ele se encontra no mesmo diretório da página, ou um endereço na Internet (URL). No caso de uma URL, deve ser definido o protocolo HTTP. O "Texto do *link*" é o texto exibido pelo *link*, a ser clicado, e que aparecerá com cor diferente e sublinhado na tela, de modo a ser identificado como um *hyperlink*.

Os *links* internos são úteis na criação de índices ou sumários. Quando ativados, levam o *browser* a um ponto determinado do documento, acessando rapidamente partes de documentos extensos. Para tanto, é necessário definir o *label*, que são marcas pelas quais o *browser* identifica a posição no texto.

O comando para *links* internos é: ` Texto`
onde "#" indica que aquele endereço é um *label*, e não um *link* externo. O *label* deve ser posicionado no início da seção do documento a ser exibida, para onde foi direcionado o *browser*, através do comando:

` Texto opcional `

2.3 - Figuras e Tabelas

Os formatos *.GIF (Graphics Interchange Format) e o *.JPG (Joint Photographic Expert Group) são os padrões de arquivo de imagens da Internet. Eles comprimem a informação gráfica (reduzem o tamanho do arquivo) e o transformam em código binário, que pode ser enviado pela Internet. O formato JPG é recomendado para fotos em geral, imagens com tons degradê (exceto as com fundo transparente), colagens e texturas complexas. O formato GIF é recomendado para desenhos, imagens com poucas cores, fundo transparente e animações (versão 89a). JPG só pode ser salvo com 24 bits (milhões de cores), e GIF só em 8 bits ou menos. Em caso de GIF, para se determinar a melhor taxa de compressão, deve-se diminuir progressivamente o número de cores (bits) até que a arte comece a se deteriorar.

Para apresentar uma figura utiliza-se o seguinte comando:

``

onde "texto_alternativo" é um texto a ser apresentado no caso da figura não ser carregada; "nome_figura", o nome e localização (caso não esteja no mesmo diretório) da figura, e "GIF" a extensão da figura (JPG, GIF, BMP).

Apesar das imagens serem responsáveis pelo visual da *home-page*, seus arquivos são grandes, aumentando consideravelmente o tempo de *download* pela internet. Assim, é importante que as imagens adicionem um real significado ao documento HTML, para que seja justificado seu uso. Para diminuir o tempo de conexão, deve-se procurar reduzir o tamanho do arquivo da imagem, reduzindo as dimensões (pixels) e o número de cores em um software gráfico. Como sugestão, usar imagens menores que 50k, e quanto menores, melhor. As imagens na Web devem ser pequenas (máximo 640x480), para evitar a necessidade de rolar ou redimensionar o tamanho da janela do *browser*.

Se existirem muitas imagens a serem apresentadas, em vez de colocar todas as imagens na *home-page*, deixá-las como arquivos externos ligados por *hiperlinks*, de modo a serem carregadas somente sob demanda do usuário, ou dividir a *home-page* em várias páginas interligadas. Várias imagens pequenas carregam mais rápido do que uma grande. Muitos *browsers* fazem o "cache", arquivando as imagens no computador, de modo que se forem usadas em várias páginas, elas serão lidas diretamente do HD, em vez de serem carregadas através da Internet.

Para evitar que a tela fique em branco por muito tempo, recomenda-se também declarar o tamanho de todas as imagens, para que o *browser* apresente imediatamente o texto formatado em volta do espaço reservado para a figura, antes de se carregar as imagens:

`<IMAGE SRC="nome_figura.GIF WIDTH=xxx HEIGHT=yyy >`

Para as imagens *.GIF deve-se usar só as cores da palheta de 216 cores da internet (web safe colors), pois poderão ser vistas sem distorções em qualquer equipamento. Em GIF transparentes, trabalhar com cor de fundo próxima ou igual à cor da página onde a GIF será colocada. Para exportar

GIF transparente, deve-se primeiro reduzir para modo *indexed* e depois selecionar a cor a se tornar "transparente". Ao alterar para modo *indexed*, deve-se escolher o menor número de cores possível (começando com 4), e aumentando em múltiplos de 4. Alguns programas gráficos oferecem opção "*no dithering*" na conversão para GIF. Isto reduz o ruído da imagem sobre um fundo de cor sólida.

Como o HTML é uma linguagem limitada, as tabelas são muito utilizadas como recurso para posicionar as imagens e texto dentro da página. As tabelas são compostas de linhas, e cada linha pode conter várias células, formando uma tabela com várias colunas. Dentro das células se coloca o texto, imagens ou outras tabelas. As *tags* de definição de tabelas são:

<TABLE> e </TABLE>:	início e fim de uma tabela
<TR> e </TR>:	início e fim de uma linha
<TD> e </TD>:	início e fim de uma célula

Apesar de não ser obrigatório o uso das *tags* de fechamento dos elementos da tabela (<TR>, </TD>), elas são úteis para a visualização da programação.

2.4 - Recursos Multimídia

A multimídia é a combinação de texto, imagens, som, vídeo, comunicações etc, que puderam ser unidos por meio do computador. Um *browser* pode apresentar textos, arquivos HTML, imagens (GIF, JPEG, etc), áudio (*.WAV, *.MP3), vídeo (*.AVI, *.MPG) etc. Para a manipulação dos diversos recursos multimídia junto com o arquivo HTML, existem bibliotecas que estendem o conjunto de tipos de arquivos (áudio, vídeo, animações) que o *browser* pode apresentar: os *Plug-In*.

Plug-In são bibliotecas softwares (*library*), geralmente desenvolvida em C ou C++, funcionando como parte do código do software do *browser*, desenvolvidos especificamente para se comunicar com o *browser*, estendendo suas características, dando-lhe a habilidade de apresentar diferentes tipos de mensagens de áudio e vídeo. O *plug-in* é compactado como uma DLL no Windows ou uma biblioteca compartilhada (*.so) no Unix, e é carregado na memória juntamente com o *browser*. Se um arquivo é recebido com um tipo MIME associado ao *plug-ins*, o *browser* chama funções na biblioteca *plug-in* compartilhada, apresentando (ou tocando) o arquivo dentro da Janela do *browser*. Diferentemente dos *Applets* Java, que por razões de segurança são limitados, um *plug-in* pode acessar arquivos no disco, abrir conexões na rede, rodar outros programas e enviar comandos ao sistema operacional.

Deve-se assegurar que o conteúdo do texto e os recursos multimídia estejam integrados. O uso dos recursos multimídias visam a suportar os objetivos contidos no texto, e devem ser utilizados com parcimônia, para não distrair e tomar o tempo do usuário. O uso excessivo ou desnecessário da multimídia torna-se confuso e incômodo. O uso de som e vídeo reduzem o desempenho do sistema e aumentam o tamanho do arquivo.

Os arquivos de som são menores que os arquivos de vídeo, sendo os formatos Wave (*.WAV) e *.MP3 os preferidos. A música ou voz podem criar um interessante ambiente de software, e sinais sonoros podem receber o usuário e conduzir a navegação no tutorial. As animações podem simular o software a ensinar uma operação específica. Diagramas (desenhos) animados podem mostrar processos do mundo real de modo efetivo. Geralmente é melhor converter as animações para os formatos *.GIF ou *.AVI. As imagens de vídeo são utilizadas para apresentar cenas do mundo real, mas deve-se evitar que o vídeo distraia a atenção do usuário do foco principal do tutorial. Locutores podem ser utilizados para comunicar uma mensagem pessoal ao usuário. Os formatos preferidos de vídeo são o *.AVI e *.MPG.

2.5 - Forms

Vários outros comandos que permitem a apresentação de cores, formulários, gráficos sensitivos, *applets*, sons, vídeo etc, fogem do escopo deste trabalho. O conhecimento mais profundo dos recursos do HTML podem ser obtidos em vários tutoriais disponíveis *on-line*, com todos os comandos da linguagem [1, 3, 4, 11]. Contudo, somente com os comandos aqui apresentados, é possível desenvolver um aplicativo em HTML.

Devido a suas limitações, o HTML não permite o uso de recursos tais como preencher o espaço em branco, múltipla escolha, falso/verdadeiro, pergunta/resposta, questões abertas e planilhas de cálculo, sendo necessário um software que processe questões aplicadas durante o tutorial. Este processamento pode ser realizado em linguagem C ou Java, utilizando-se a CGI (Common Gateway Interface), de modo a realizar o intercâmbio de informações dinâmicas através da Web. A página de questões é aberta como um FORM. O FORM submete os dados de entrada do usuário ao programa CGI, que acessam um banco de dados com as questões, e se encarrega do processamento desses dados. Desta forma, pode-se efetuar a avaliação do aprendizado do aluno. O aluno preenche um formulário na *home-page*, e as respostas são submetidas a um programa que as avaliará, retornando ao aluno a nota de sua avaliação *on-line* [6]. Os *scripts* CGI são difíceis de serem escritos e requerem um especialista em programação.

- * Preparador de formulário: <http://www.response-o-matic.com/>
- * Programas CGI: <http://www.ferca.net/esgratis/cgis.html>; <http://worldwidemart.com/scripts/>
- * Servidor para CGIs: <http://www.cgi-free.com/>

O Java é uma linguagem de programação orientada a objetos (simplificação da linguagem C⁺⁺), desenvolvida pela Sun Micro Systems que adiciona recursos gráficos, de animação, de sons, formulários interativos, cálculos etc à HTML. A linguagem Java, em vez de interpretar a linguagem, um programa compilado (*applet*) é transferido para ser executado no cliente, conseguindo-se assim uma maior velocidade de execução. Isto permite o processamento no cliente. Isto torna possível aplicações como planilhas de cálculo ou validação de dados, além de animações com qualidade muito superior à do HTML.

Curso JavaScript	http://www.elogica.com.br/bbs/curso/
Cursos, exemplos, notícias	http://sunsite.unc.edu/java.faq
Java home page	http://java.sun.com
Tutorial Java. André Cesta	http://www.dcc.unicamp.br/~cmrubira/aacesta/
Utilitários Java	http://www.blackdown.org/~kbs/index.html

2.6 - Equações Matemáticas e Símbolos

Apesar do HTML ter sido desenvolvido visando a divulgação de documentos científicos, até a versão 3.2 do HTML (atual) ainda não foi implementado o conjunto de *tags* (MathML, Mathematical Markup Language) que possibilite a apresentação de letras gregas, símbolos e equações matemáticas. Em 7 de Abril de 1998 o W3C (World Wide Web Consortium) finalmente recomendou o MathML (<http://www.w3.org/math/>), compatível com o XML (Extensible Markup Language), padrão proposto para suceder o HTML. O MathML define um conjunto de elementos e atributos XML (*tags*) utilizados para descrever a notação matemática, e apresentá-las num *layout* bi-dimensional (atributos de apresentação), e o outro para descrever o real significado das notações (atributos de conteúdo).

Apesar de ser factível a programação manual do MathML, ela é muito trabalhosa e deve ser realizada através de um software assistente. Os editores de equações WebEQ e MathType 4.0, são

capazes de gerar o código MathML a ser utilizado nas *home-pages* com matemática.

Entretanto, os *browsers* comerciais existentes no mercado ainda não reconhecem as *tags* MathML, que talvez sejam implementadas no Netscape 5.0. Poucos *browsers* são atualmente capazes de interpretar o MathML: Amaya (IBM-PC) (<http://www.w3.org/Amaya/>), Arena (Linux, Unix/X11) (<http://www.yggdrasil.com/Products/Arena/>). Outra solução é usar o *applet* Java WebEQ (<http://www.webeq.com/webeq/>), que ensina ao *browser* a interpretar as *tags* MathML (e WebText também) e apresentar as equações matemáticas correspondentes. Todavia o WebEQ não é totalmente compatível (ainda) com o MathML, possuindo algumas omissões e adições.

Hoje, a maioria das *home-pages* que incluem símbolos e equações matemáticas, são criadas adicionando *links* às imagens *.GIF destes símbolos e equações. Isto é realizado utilizando o editor de equações do processador de textos (WordPerfect e Word), e salvando o documento com as equações, no formato HTML. Durante este processo, as equações são convertidas em gráficos *bitmap* (*.GIF) e "linkados" ao documento HTML. Apesar desta técnica parecer "marreta", ela funciona bem para as equações criadas com o editor de equações. O mesmo deve ser feito com todas as letras gregas e símbolos especiais que aparecem no texto, pois se forem criados com as fontes especiais, não serão apresentados pelo *browser*.

Pode-se também criar as equações no tratamento de texto, e realizar a captura da imagem da tela por um software tipo Paint Shop Pro, cortá-la e salvá-la com um arquivo GIF. Deve-se também utilizar um software para transformar a imagem na versão GIF89, que possui transparência, de modo que a equação possa se sobrepor ao fundo escolhido para a página. Porém, as equações apresentadas desta maneira são impressas com baixa resolução, não se adaptam às fontes escolhidas pelo programador, além de aumentar o tempo de carregamento destas páginas.

Os *browser* Netscape e Explorer 3.0 (ou superior) introduziram a *tag* permite que qualquer fonte do sistema DO USUÁRIO possa ser interpretada pelo *browser*. Assim, especificando a fonte *symbol* (que é disponível em vários sistemas), pode-se apresentar letras gregas, símbolos e equações matemáticas simples. Os atributos COLOR e SIZE podem também ser utilizados através do atributo FACE. Caso não seja encontrada a fonte especificada, será utilizada a fonte *default*. Exemplo:

$y = \langle \text{font face} = \text{"symbol"} \rangle 5 \int \langle \text{font face} = \text{"symbol"} \rangle x^2 \sin \theta \langle \text{font face} = \text{"symbol"} \rangle dx$

$$y = 5 \int x^2 \sin \theta dx$$

O uso desta técnica possui alguns inconvenientes: a *tag* tem que ser introduzida manualmente, a fonte utilizada deve ser de uso comum, e podem existir problemas de interpretação (ou impressão) nos diferentes sistemas (Dos, Unix, Macintosh). Como parte dos símbolos matemáticos pertencem ao conjunto de caracteres *ascii* estendidos, para acessá-los é necessário usar "*alt+código_ascii*;" (digitado no teclado numérico) ou "&#código_ascii;".

Algumas *tags* auxiliam a formatação das equações:

_{texto}Subscrito

^{texto}Superscrito

<PRE WIDTH=80>texto</PRE> Texto pre-formatado

A *tag* <PRE> define um texto com fonte de espaçamento fixo, que será apresentado tal qual composto pelo autor: com as quebras de linhas e espaços em branco. Não é necessário utilizar a *tag*
 para indicar a quebra de linha. O atributo WIDTH (opcional) especifica o número máximo de

caracteres por linha.

Os métodos das imagens GIF e da *tag* são os mais facilmente realizáveis, pois as páginas são compatíveis a todos os sistemas, de modo a atingir uma maior audiência; não são necessários de grandes conhecimentos específicos para serem criados; permitem buscas no texto; e os arquivos são pequenos de modo a facilitar o *download*. Entretanto as possibilidades de layout são limitadas, a programação é trabalhosa e diminuem a interatividade.

Os documentos escritos em código Tex e LaTeX podem ser convertidos automaticamente em documentos HTML, com a conversão das letras gregas, símbolos e equações matemáticas em imagens GIF pelos softwares LaTeX2HTML (<http://www.msi.umn.edu/software/latex2html/>) e HTMX (<http://math.uic.edu/~fields/htmx/>). As equações escritas código Tex e LaTeX também podem ser interpretadas pelos *browsers* comerciais através do *applet* HotEqn (<http://www.esr.ruhr-uni-bochum.de/VCLab/software/HotEqn/HotEqn.html>) e do *plug-in* IBM TechExplorer (<http://www.software.ibm.com/enetwork/techexplorer/>).

Os arquivos *.PDF (Portable Document Format) e *.PS (PostScript) são outra forma de apresentar a informação matemática através da Web. O formato PDF é um formato *PostScript*, otimizado para o transporte na Web. Outra característica importante é que o PDF é um protocolo de texto, que permite a busca como qualquer outro arquivo texto. Esta é uma solução universal, pois PostScript é o padrão para impressão, e praticamente todas as plataformas são suportadas. O *plug-in* Acrobat Reader (<http://www.adobe.com>) permite que o *browser* apresente *on-line* os documentos PDF. Os arquivos PDF possuem grande qualidade de impressão, total controle sobre o layout do documento, interatividade, e fácil edição. Porém os custos dos softwares de edição são altos, a necessidade do *plug-in* diminui a audiência, os tempos de *download* são muito maiores e diminui a interatividade (busca e interface) com os outros conteúdos da Web.

2.7 - Mailing Lists e GuestBooks

As *Mailing Lists* (listas de discussão) reúnem pessoas sobre a discussão de um tema específico, através de mensagens distribuídas por *e-mail*. Qualquer mensagem enviada ao *e-mail* da lista, é enviada a todos os seus participantes. Para participar é necessário assiná-la, através do envio de uma mensagem para o endereço da lista. As listas de discussão permitem criar uma comunidade cibernética de discussão sobre o assunto da *home-page*. Normalmente, os provedores de acesso possuem gerenciadores de *Mailing Lists*. Caso isto não seja disponível, pode-se usar os seguintes servidores (gratuitos):

MakeList: <http://www.makelist.com>
ListBoot: <http://www.listboot.com>
OneList: <http://www.onelist.com>
EGroups: <http://www.eGroups.com>

Os *GuestBooks* (Livro de visitas) permitem ao programador um conhecimento do perfil dos usuários de sua página. Alguns seguintes servidores (gratuitos) de *GuestBook* possuem a habilidade de capturar informações dos visitantes e manter uma base de dados destas entradas: e-mail, URL, localidade e comentários:

Dreambook: <http://www.dreambook.com>
FreeGuestBooks: <http://www.freeguestbooks.com>
GuestWorld: <http://www.GuestWorld.com>
LPage: <http://www.Lpage.com>

A home-page pode também contar com outros recursos, tais como contadores de acesso (<http://www.fastcounter.com/> e <http://www.mnetweb.com/pagectrs.shtml>) e análise estatística da utilização (<http://www.hitbox.com/wc/world.html> <http://www.showstat.com> e <http://www.freestats.com/>).

3. O DESENVOLVIMENTO DE *HOME-PAGES* E TUTORIAIS EM HTML

Como se pode notar, a programação HTML é muito simples, fácil e pode ser realizada por qualquer um. Para que as páginas construídas em HTML sejam compreendidas pelos vários *browsers*, nos diversos sistemas operacionais, a HTML é uma linguagem de programação simples, não possuindo muitos recursos. Assim, o desenvolvimento de *home-pages* exige boa dose de criatividade para compensar as limitações da linguagem.

Um *site* técnico na internet deve ser enxuto, com pouca firula e muita informação. A criação destas *home-pages* é relativamente fácil de ser realizada. Não estamos propondo a criação de uma *home-page* profissional, cheio de firulas, sons etc, mas a apresentação do conhecimento de uma forma lógica e interativa e dinâmica. Deve-se planejar previamente o que se pode fazer com o *site*, e começar com uma *home-page* básica, de modo a disponibilizar imediatamente os conhecimentos e informações que se deseja difundir. Com o tempo se adiciona os recursos sofisticados à *home-page*. A grande diferença fundamental entre Internet e os meios de comunicação tradicionais é a interatividade e rapidez de atualização da informação.

Os tutoriais são compostos de "páginas", de qualquer tamanho, interligadas por *hiperlinks* (botões), que podem ser manipulados pelas teclas de direção ou *mouse*. Cada página pode conter texto, figuras, fotos, sons etc. Como as páginas são ligadas umas às outras pelo hipertexto, o conhecimento é apresentado de forma não linear, o mesmo modo de aprender e de pensar da maioria das pessoas.

Para que o tutorial hipertexto cumpra seus objetivos, ele deve possuir uma estrutura lógica bem organizada, de modo que o usuário possa dirigir a navegação de acordo com suas associações pessoais. Encher de *links* inúteis e desnecessários torna o tutorial confuso e de difícil utilização. Por outro lado, se os *links* hipertexto são usados criteriosamente, eles aumentam a organização da informação, tornando-a mais acessível aos usuários. Eles podem criar novas inter-relações lógicas no conhecimento, sem apagar as já existentes.

Deve-se fazer com que o *site* seja dinâmico, interessante e rápido de usar. Defina o melhor possível as necessidades de apresentação, não usando características mais complexas do que as necessárias. Características desnecessárias e não utilizadas diminuem o desempenho do *site*, seu aprendizado e produtividade, além de forçar o uso de configurações mais caras que as necessárias. Evitar colocar muitas imagens, pois a velocidade de acesso é baixa. Tentar sempre dividir a página ao máximo possível, colocando índice no começo (através de *links* internos) ou mesmo, dividindo-a em várias páginas pequenas. O *site* deve possuir um mapa de navegação e fácil acesso à informação. O texto deve possuir um bom contraste com as cores do fundo.

Uma atenção especial deve ser dada à interação, navegação avaliação e registro de respostas, tipos de arquivos (gráficos, sons e texto). Evitar usar características exclusivas do Netscape ou do Explorer, para não perder de imediato a possibilidade de ser visitado por 50% dos Internautas. Como cada *browser* tem uma interpretação diferente das *tags*, pronta a página, recomenda-se visualizar a página no *browser* usual para ver o resultado final.

O conteúdo da home-page deve informar adequadamente, ser completo e atualizado, de modo a cumprir seus objetivos e prestar um serviço real. Deve-se procurar prever as dúvidas dos usuários e estabelecer mecanismos de interação (e-mail, guestbook) de modo a conhecer sua opinião, a fim de poder melhorar continuamente o conteúdo e apresentação do *site*. Colocar sempre a data de atualização

e o *e-mail* do responsável no fim da página.

Um dos maiores problemas *Webmaster* é encontrar os *links* inválidos (broken links) na *home-page*. Como a Web muda constantemente, a verificação e atualização dos *links* pode ser trabalhosa, mesmo em pequenos *sites*. O NetMechanic (<http://www.netmech.com>) e o WebSiteGarage (<http://www.websitegarage.com>) possuem um serviço gratuito de verificação e validação de *home-pages*. Um robot analisa todo o *site*, procurando *tags* HTML inválidas, testando, verificando e validando cada *link*, e gerando um relatório de todos os links inválidos encontrados e o status de cada *link* testado.

4. HOSPEDANDO E DIVULGANDO A HOME-PAGE

Caso não possua um provedor próprio, existem vários provedores de acesso à Internet que disponibilizam espaço e divulgam a *home-page* de graça. De modo geral, estes provedores proíbem conteúdo ofensivo, referências publicitárias e patrocinadores.

321Website (10Mb) <http://www.321website.com/>
Abordo Informática (0,5Mb) <http://www.abordo.com.br>
Angelfire (5Mb) <http://www.angelfire.com/>
Chez (10Mb) <http://www.chez.com>
Crosswinds (Ilimitado): <http://www.crosswinds.net>
CyNetCity (12Mb) <http://www.cynetcity.com/>
Easy Space (25Mb) <http://www.easypost.com/freeap.html>
Escalix (5Mb) <http://www.escalix.com/freepage/>
Focus-Asia (2Mb) <http://www.focus-asia.com/>
FortuneCity (20Mb) <http://www2.fortunecity.com/downtown/index.cgi>
Free Nation (7Mb) <http://www.freenation.com/> <http://www.cybercity.hko.net/>
Free Yellow Pages (12Mb) <http://www.freeyellow.com/>
Geocities (11Mb) <http://www.geocities.com/join/freehp.html>
HyperMart (10Mb) <http://www.hypermart.net/>
Infoseek (7MB) (http://pages.infoseek.com/webchat3.so?homepage_center.x=1)
Intelcities (3Mb) <http://www.intelcities.com/>
Intermega (30Mb) <http://www.intermega.com.br>
Internet Club (15Mb) <http://www.internet-club.com/>
Megapolis (10Mb) <http://www.megapolis.net/>
Mygale: (10Mb) <http://www.mygale.org>
NetTax (3Mb) <http://www.nettaxi.com/>
ProHosting (10Mb) <http://free.prohosting.com/>
Rampage (100Mb) <http://www.neocarf.com/front2.html>
Royalty Studios (50Mb) <http://www.royaltystudios.com/freeweb/enter.html>
Space Ports (10Mb) <http://www.spaceports.com/>
Starmédia Órbita BR (5Mb): <http://www.starmedia.com>
Talk City (12Mb) <http://www.talkcity.com/>
Telepolis (2Mb): <http://www2.telepolis.com/>
Terravista (2 Mb) <http://www.terravista.pt>
The Globe (5Mb): <http://www.theglobe.com/>
Towne Square 2000 (10 páginas) <http://townsquare.usr.com/>
Tripod (11Mb) <http://www.tripod.com/>
Uncut Productions Web Services (2Mb) <http://www.upws.com/>
Web Jump (25Mb) <http://www.webjump.com/>
Xoom (11Mb) <http://xoom.com/home/> <http://www.xoom.com>

Uma vez implantado o *site*, ele deve ser amplamente divulgado. Além de enviar um e-mail para os conhecidos que possuem interesse em comum, recomenda-se cadastrá-la nos mecanismos de busca mais famosos. Alguns servidores permitem um sistema de cadastramento automático da *home-page* nos principais mecanismos de busca nacionais (Achei, Cadê, Onde Ir, RadarUol e Surf) e internacionais (Altavista, Excite, Lycos e Web Crawler):

Register-It: <http://www.register-it.com>

Submit It: <http://www.submit-it.com>

Vitrine: <http://www.casio.com.br/vitrine> (Nacionais e internacionais)

5. CONCLUSÕES

A implementação de uma *home-page* ou um tutorial multimídia em HTML é uma tarefa relativamente simples. A linguagem HTML é simples e de fácil aprendizado e implementação, diferentemente das linguagens de alto nível tais como C++ ou Pascal. Com um software de tratamento de texto usual e o conhecimento do HTML é possível desenvolver um tutorial multimídia. O tutorial resultante permite a apresentação de uma grande quantidade de informações, de uma forma mais atraente e amigável do que nos livros texto e apostilas, devido à interatividade e à facilidade com que se pode navegar pelo documento.

Os tutoriais multimídia apresentam seu conteúdo de maneira lógica e direta. Entretanto, por mais completo que seja o aplicativo multimídia, ele é incapaz de inferir, aprender, se auto-aperfeiçoar e solucionar todas as dúvidas, erros e enganos dos alunos. O programa pode detectar que o aluno errou, mas não sabe qual o erro, nem porque o aluno errou. Assim, continua imprescindível a presença do professor no processo de aprendizagem. O computador não é uma panacéia para os problemas do ensino, ao contrário, pode acentuar os erros já existentes no modelo pedagógico atual, um vez que o professor perde o controle do processo de aprendizagem, se não acompanhar, por outros modos, o desenvolvimento do ensino.

REFERÊNCIAS

- [1] CASTRO Maria Alice Soares. Tutorial HTML, <http://www.icmsc.sc.usp.br/manuals/>
- [2] HTML Guide. <http://coolzine.com/html>
- [3] MOREIRA José de Albuquerque. Guia de Iniciação ao HTML. <http://www.mat.unb.br/~moreira/cartilha.html>
- [4] NCSA. A Beginner's Guide to HTML. <http://www.ncsa.uiuc.edu/General/Internet/WWW/HTMLPrimer.html>; <ftp://pub@ncsa.uiuc.edu>
- [5] <news:comp.infosystems.www.authoring.html>
- [6] PEREIRA Luiz Gustavo Anflor. Tutorial sobre a Construção de Questões. <http://www.pop-rs.rnp.br/ovni/questao/questao.htm>
- [7] PINHEIRO Paulo César da Costa. Desenvolvimento de um Tutorial Hipertexto em HTML. In: XXV CONGRESSO BRASILEIRO DE ENSINO DE ENGENHARIA (COBENGE-97), 12-15 Outubro 1997, Salvador, BA, *Anais...* Salvador: ABENGE, Associação Brasileira de Ensino de Engenharia, 1997, v.3/4, p.1171-1184.
- [8] RAMALHO José Antônio. Monte Sua Home Page e Caia na Rede. Informática Exame, v.11, n.122, p.94-96, 98, Maio 1996. <http://www2.uol.com.br/info/arquivo/ie122/tutorial.html>
- [9] ROCHA Helder L.S. Como Criar sua Home Page. <http://www.dsc.ufpb.br/~helder/html-ref/tutorial/>
- [10] SHIKLI Peter. Designing Winning Web Sites for Engineers. Machine Design, v.69, n.21, 6 Nov 1997, p.30,32,35,36,38,40.
- [11] YALE. Yale Web Style Guide. <http://info.med.yale.edu/caim/manual>