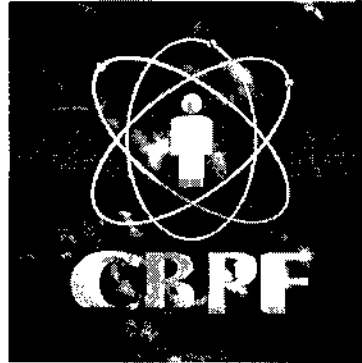


WORKSHOP ON INTERNET



<http://www.cbpf.br>

CENTRO BRASILEIRO DE PESQUISAS FÍSICAS - CBPF

UNISYS

WORLDWIDE INFORMATION SERVICES

THE WORLD WIDE WEB

Márcio PORTES DE ALBUQUERQUE
Nilton ALVES Jr.

DE 21 À 25.AGO.1995



Márcio P. de Albuquerque
mpa@cat.cbpf.br

Nilton Alves Jr.
naj@cat.cbpf.br

RESUMO

Este trabalho surgiu do convite feito pela UNISYS do Brasil para que os autores coordenassem um dos dias da semana do *Workshop on INTERNET*. O tema/título proposto, *The World Wide Web*, tem como objetivo principal discutir os tópicos mais relevantes deste que é o serviço mais atual, promissor e abrangente da INTERNET.

Na primeira parte deste texto apresentamos um histórico das comunicações em rede, alguns conceitos básicos da filosofia cliente/servidor e falamos sobre as técnicas utilizadas na manipulação de imagens. Na segunda parte são descritos com alguma profundidade os detalhes da linguagem HTML e do padrão CGI de comunicação, entre servidores de informação e aplicações escritas em linguagem C. A instalação e configuração de um servidor WWW sobre uma plataforma UNIX completam este trabalho.

São citadas ainda, algumas características principais dos navegadores MOSAIC e NETSCAPE, e do sistema gerenciador de base de dados bibliográficos do CBPF, SINFOCX. Ao longo do texto, inúmeros endereços HTTP relevantes em assuntos variados são citados.

ÍNDICE

1. INTRODUÇÃO	5
2. A INTERNET	6
3. FILOSOFIA CLIENTE / SERVIDOR	8
4. A LINGUAGEM DE MARCAÇÃO - HTML	10
4.1. CRIANDO DOCUMENTOS HTML	11
4.2. O DOCUMENTO HTML MÍNIMO	11
5. OS SOFTWARES MOSAIC E NETSCAPE	12
6. A MANIPULAÇÃO DE IMAGENS	13
7. DETALHES DA LINGUAGEM HTML	15
7.1. MARCADORES BÁSICOS	15
7.2. LIGAÇÃO COM OUTROS DOCUMENTOS	16
7.3. MARCADORES ADICIONAIS	18
7.4. IMAGENS	21
7.5. FORMULÁRIOS	22
8. COMMON GATEWAY INTERFACE (CGI)	23
9. SINFOCX/CBPF	24
10. O SERVIDOR WWW	25
10.1. O HTTP DEAMON	25
10.2. ARQUIVO DE CONFIGURAÇÃO	25
10.3. PARA MAIORES DETALHES	27
ANEXO A (<i>Links</i> diversos)	29
ANEXO B (O programa post-query.c)	33
ANEXO C (O arquivo de configuração httpd.conf)	35
REFERÊNCIAS	37
PAGINA PRINCIPAL UNISYS	39

1. INTRODUÇÃO

A computação moderna aliou ao seu modelo inicial, onde os computadores eram máquinas exclusivamente dedicadas à realização de cálculos complexos, o conceito de automatização da informação. Podemos observar esta automatização no nosso dia a dia através de diversos sistemas que permitem a manipulação da informação, como por exemplo os bancos de dados, os sistemas distribuídos, os sistemas baseados na filosofia Cliente/Servidor, o correio-eletrônico e num breve futuro na realização de teleconferências. O fácil manuseio de informações dado pela informática promove um salto quantitativo e qualitativo nos métodos de trabalho de instituições de diferentes fins. Esta filosofia vem permitindo cada vez mais que a automação seja realizada por parte de profissionais em áreas distantes da informática. As instituições de pesquisas são o ponto de partida para novos projetos e novas idéias do ponto de vista tecnológico, criando sistemas que interagem cada vez mais numa escala mundial.

Recentemente, ao lado de outros serviços disponíveis na Internet, como o WAIS, o Gopher, o Archie, o Netfind, etc, nós vimos aparecer o World Wide Web, ou o W3 (ou em português a “teia mundial de comunicação digital”). O World Wide Web é um universo de diferentes informações acessíveis via rede de computadores (no caso a Internet). Filosoficamente poderíamos compará-lo à um sistema de acesso à uma base de conhecimento humana, que é compartilhada por todos independentemente de sua localidade geográfica. Este sistema começou nos laboratórios do Centro Europeu de Pesquisas Nucleares, o CERN, em Genebra (Suíça) e está se tornando cada vez mais popular devido as suas potencialidade de distribuição, procura e acesso à diferentes tipos de informações.

No seu conteúdo estão definidos de um lado o software de acesso e de outro, uma série de protocolos e convenções que definem a comunicação entre as diferentes máquinas. O W3 usa as técnicas de hipertexto e multimídia para promover e facilitar o uso da comunicação para qualquer um dos seus integrantes, seja na busca ou no fornecimento de informações. Não podemos deixar de mencionar que prover informações, que sejam acessíveis via W3, exige aliar a técnica à um projeto artístico que simbolizem os serviços oferecidos pelo sistema servidor, paralelamente à uma organização coerente dos diversos *links* oferecidos. Para terminar esta introdução devemos ressaltar que a melhor maneira de aprender ou de consultar documentos WWW é dentro da própria rede. Uma infinidade de documentos estão atualmente disponíveis e podem ser consultados *on-line* sob a forma de hipertextos. Ao longo deste trabalho citamos alguns links principais que podem servir como ponto de partida para consultas de diversos assuntos na INTERNET.

2. A INTERNET

A INTERNET é uma rede de redes que interliga computadores de diferentes plataformas e indivíduos em diferentes países. Ela nasceu efetivamente em 1982 com o aparecimento do protocolo de comunicação entre redes. O termo atual amplamente utilizado para simbolizar esta rede planetária é de *Information SuperHighway*. Ela implementa o protocolo de comunicação TCP/IP (Transmission Control Protocol / Internet Protocol) que permitir trafegar diferentes suportes para a informação como sons, imagens, textos, filmes, etc. A INTERNET interliga as redes universitárias, comerciais, militares, científicas, etc, conectando computadores do mundo inteiro. Essas redes são conectadas entre si por recursos que variam desde linhas telefônicas de discagem comum até linhas privadas dedicadas e de alta velocidade, satélites, ligações por microondas e por fibra ótica. A INTERNET é formada por mais de 4 mil redes de computadores, interligando mais de 2 milhões de equipamentos em mais de 100 países. Estima-se hoje (1995) que quase 40 milhões de pessoas usem a Internet.

Quando alguém se conecta na INTERNET, o seu computador se torna uma extensão do que parece ser um enorme sistema de comunicação, onde pela primeira vez na história humana, contrariamente aos meios de comunicações modernos, um indivíduo pode participar ativamente da rede, isto é, ele pode prover a rede de informações. O computador passa a estar ligado à milhares de outros computadores, conectados entre si por centenas de redes. As informações são passadas entre estes computadores conforme sejam requisitadas. Cada uma das redes que compõe a INTERNET é administrada, mantida e paga por instituições educativas e por outras organizações que compõem as diversas redes. Não existe nenhum "administrador" central da Internet. No entanto, ao longo dos anos, alguns procedimentos e atitudes se estabeleceram, formando um conjunto de "boas maneiras" para os usuários da INTERNET.

A lista a seguir descreve a história das comunicações em rede ao longo das últimas 4 décadas:

- 1957 - A partir do lançamento do Sputnik pela URSS, os EUA formam a ARPA -Advanced Research Program Agency.
- 1962 - Paul Baran, descreve redes de chaveamentos de pacotes na revista americana: "On Distributed Communication Networks".
- 1969 - O Departamento de Defesa Americano encomenda a ARPANET, para desenvolver especialmente pesquisas em redes de computadores.
- 1971 - Ray Tomlinson, da BBN, inventa um programa de *E-mail* para rede distribuídas, na Inglaterra.
* A rede contém neste instante *23 Hosts*.
- 1973 - ARPANET se liga a Inglaterra e a Noruega.

- 1974** - A Tese de Doutorado de Robert Metcalfe estabelece o padrão Ethernet.
- Vinton Cerf e Bob Kahn detalham nesta mesma época o protocolo de comunicação entre redes, o TCP (Transmission Control Protocol).
* A rede alcança **62 Hosts**.
- 1982** - O TCP/IP se torna o conjunto de protocolos da ARPANET (nesta época são feitas as primeiras referências à Internet - uma interconexão de redes).
- A Europa cria a EUNet (European Unix Network).
- 1983** - a Universidade de Wisconsin desenvolve o servidor de nomes.
- A Universidade de Berkley lança o Unix 4.2, incorporando o TCP/IP.
* A Internet conta com **500 Hosts**.
- 1984** - O DNS (domaine name server) é lançado.
- O Japão estabelece a JUNET (Japan Unix Network).
- 1886** - O NSFNET ("National Science Foundation Network - EUA") é criado com uma velocidade na espinha dorsal de 56Kbps.
* A Internet alcança **5.000 Hosts**.
- 1987** * A Internet alcança **20.000 Hosts**.
- 1989** - O backbone da NSFNET é de 1.544Mbps
* A Internet tem **100.000 Hosts**.
- 1990** - A ARPANET é extinta.
- 1991** - A companhia Thinking Machine lança o WAIS - Wide Area Information System.
- A Universidade do Minnesota lança o Gopher.
* A Internet alcança **617.000 Hosts**.
- 1992** - O CERN lança o World Wide Web.
- O Backbone da NSFNET passa para 44.736Mbps.
- 1993** - A NSF cria a InterNIC.
- o congresso americano aprova a lei de infra-estrutura para a informação.
- as empresas e a mídia descobrem a INTERNET.
* **2.000.000 Hosts** fazem parte da Internet.
- 1994** - O Mosaic domina a Internet; o WWW e o Gopher proliferam.
- os shoppings centers chegam a Internet.
* mais de **3.000.000 Hosts** fazem parte da Internet.
- 1995** - a NSF descomissiona a NSFNET e transfere os fundos para a vBNS (very High Speed Backbone Service); os fornecedores de rede assumem a Internet.
-

Abaixo são listados alguns dos diferentes países que fazem parte da Internet, com o número de nós presente em cada país, (A lista foi fornecida por um servidor netfind: netfind.if.usp.br, e os dados são de 1992). Ela pode vir a ser útil quando queremos localizar a origem de um determinado servidor.

3340 edu (U.S. Educational Institutions)	26 il (Israel)
968 com (U.S. Commercial Institutions)	25 za (South Africa)
836 de (Germany)	24 pl (Poland)
603 ca (Canada)	24 mx (Mexico)
378 au (Australia)	21 kr (Republic of Korea)
360 uk (United Kingdom)	20 ie (Ireland)
350 jp (Japan)	20 gr (Greece)
263 se (Sweden)	18 is (Iceland)
248 gov (U.S. Gov. Institutions)	13 pt (Portugal)
191 mil (U.S. Military Institutions)	13 hk (Hong Kong)
184 fr (France)	13 cs (Czech and Slovak Republic)
181 no (Norway)	11 us (United States)
168 nl (Netherlands)	9 cl (Chile)
163 net (Inst. Named by Network Connection)	7 ee (Estonia)
129 it (Italy)	6 sg (Singapore)
122 org (Non-profit Institutions)	4 hu (Hungary)
85 fi (Finland)	2 pr (Puerto Rico)
80 dk (Denmark)	2 lu (Luxembourg)
68 at (Austria)	1 yu (Yugoslavia)
63 ch (Switzerland)	1 ve (Venezuela)
58 es (Spain)	1 tn (Tunisia)
43 br (Brazil)	1 th (Thailand)
42 nz (New Zealand)	1 int (International)
36 tw (Taiwan)	1 arpa (Old ARPANET names)
33 be (Belgium)	1 ar (Argentina)
	1 aq (Antarctica)

3. FILOSOFIA CLIENTE / SERVIDOR

O surgimento da Arquitetura Cliente/Servidor (ACS) advém da mistura de duas outras tecnologias: *hosts* e redes de microcomputadores PC.

Nas décadas 60 e 70 a informatização das empresas apoiava-se principalmente na computação de grande porte ou *host*, onde os dados e aplicações estão armazenados em uma máquina central. A principal vantagem vem da própria centralização que faz com que todos os processos e transações aconteçam em um só ponto. Isto traz facilidades óbvias no controle do acesso aos dados, aplicações e de administração. A principal desvantagem está curiosamente na própria centralização. O sistema sendo centralizado cria um único ponto de falibilidade, o que compromete a disponibilidade do sistema como um todo. (ex.: agência bancária com o sistema fora do ar). Uma outra desvantagem é relativa ao fato do desenvolvimento de aplicativos ser centralizado e controlado pela equipe do CPD. Além disto o custo operacional, aquisição, desenvolvimento e manutenção era muito elevado e restrito a poucas empresas do mercado.

A partir da década de 80, o rumo da informatização das empresas foi alterado. Os usuários passaram a ter computadores pessoais ao lado de seus terminais, para serviços rápidos e não disponíveis no grande porte, principalmente editoração de textos. Porém a verdadeira revolução aconteceu quando os vários usuários de PCs conectaram suas máquinas através de redes e passaram a compartilhar recursos e periféricos. As vantagens aqui são

exatamente as desvantagens da outra arquitetura: os dados e aplicações estão espalhados pela rede (empresa). A interface dos aplicativos são de melhor qualidade e freqüentemente gráfica e por isso aumentam a produtividade pela facilidade de utilização. A rede passou a disponibilizar a todos os usuários de maneira diferenciada, impressoras, linhas de comunicação, serviços de fax, softwares, etc. E por fim, como os computadores eram autônomos e independentes, a falha de um deles era um fato isolado e sem comprometimento para os demais. A principal desvantagem dessa arquitetura está na dificuldade de administração física e lógica.

Estava então desenhado o panorama: as empresas tinham de gerenciar dois tipos distintos de arquitetura, cada uma com suas vantagens e desvantagens, e além disso a integração entre elas era de difícil desenvolvimento. Numa visão simplificada, a computação Cliente/Servidor baseia-se na distribuição dos vários componentes de um sistema ou aplicação. Essa distribuição se dá obedecendo os critérios: Clientes são solicitantes, consumidores de recursos, enquanto que os Servidores são os fornecedores destes recursos.

Idealmente, o hardware da ACS deve permitir a conexão dos mais diversos tipos de computadores, de um simples microcomputador até o *mainframe*, através de todo tipo de topologia de rede e utilizando qualquer tipo de protocolo de comunicação. Uma das principais características do hardware da ACS é a escalabilidade, ou seja, a habilidade de se crescer na arquitetura à medida exata que o negócio exija. Em outras palavras, na ACS, os investimentos são feitos de forma modular e gradual, em total oposição a arquitetura centralizada.

O software Cliente/Servidor é aquele que garante que qualquer aplicação possa utilizar os recursos disponibilizados na rede. É indispensável que haja uma forma bem definida de comunicação entre a aplicação que fornece recursos à rede (aplicação servidora) e a aplicação que os consome (aplicação cliente); caso contrário corre-se o risco de disponibilizar uma série enorme de recursos na rede e poucas aplicações poderem tirar proveito deles.

Esta arquitetura se aplica à empresas que precisam passar pelos processos *downsizing* ou *upsizing*, ou seja qualquer processo e/ou qualquer empresa que esteja modificando o seu sistema de computação. O caráter corporativo da ACS implica em uma grande facilidade de se chegar ao que hoje se chama *rightsizing*, que nada mais é que *downsizing* + *upsizing* + bom senso, [Sybase - 93].

A utilização de sistemas abertos torna o ambiente Cliente/Servidor mais adequado às realidades do mercado pois existe uma variedade maior de equipamentos e ambientes operacionais, o que aumenta a competição e assim proporcionando uma potencial redução de custos. O ambiente Internet é um sistema aberto que predominantemente utiliza o protocolo TCP/IP, que é o protocolo de comunicação padrão do sistema operacional UNIX. Este fato implica em uma facilidade imediata de utilização do UNIX para acessar navegadores Web. O WWW é um sistema que funciona segundo a ACS. De um lado existem as estações que agem

como servidoras de arquivos HTML, fornecendo informações ou acesso a base de dados, do outro existem as estações clientes que rodam softwares como o Mosaic e o Netscape e que solicitam serviços aos servidores.

4. A LINGUAGEM DE MARCAÇÃO - HTML

Abaixo citamos alguns anacronismos muito comuns no assunto abordado neste texto:

- WWW** - World Wide Web, abreviadamente Web ou W3;
- SGML** - Standart Generalized Markup Language é um padrão para linguagem de marcação (etiquetas).
- DTD** - Document Type Definition é uma linguagem de marcação específica, escrita usando SGML.
- HTML** - HyperText Markup Language, HTML é um DTD em SGML. Podemos dizer que a HTML é um conjunto de estilos (indicados pelos marcadores ou etiquetas) que definem diversos componentes de um documento para WWW. Esta linguagem foi inventada por Tim Berners-Lee enquanto trabalhava no CERN.
- URL** - Uniform Resource Locator é basicamente um endereço Internet com sintaxe própria para cada um dos diversos casos de aplicação (arquivos, Gophers, News, HTTP, Parcial, etc...)

HTML é um tipo de documentos com marcadores que está de acordo com o padrão ISO8879, "The Standart Generalized Markup language (SGML)". HTML é uma linguagem que suporta uma maneira simples e objetiva de estruturar a codificação de um documento com um mínimo de acréscimo de informação ao texto propriamente dito. SGML é uma maneira padrão de descrever o que uma linguagem de marcação deve oferecer (ver referências [HTML]).

Um documento HTML tem a forma de um texto plano com vários marcadores. Estes marcadores são delimitados pelos sinais de menor < e maior > e os nomes dos marcadores expressam a estrutura do documento. Podemos citar como exemplo, a existência de marcadores que envolvem os títulos (<H1>Isto É Um Cabeçalho no Nível 1</H1>), o título do documento (<TITLE>O Título</TITLE>), listas (para uma lista ordenada) e assim por diante.

Apesar dos esforços para que os marcadores envolvam somente formatação, alguns autores pedem a inclusão de marcadores de definição de apresentação do tipo texto, onde a palavra 'texto' aparece com uma fonte em negrito. Alguns interpretadores não consideram este tipo de marcador.

A linguagem HTML está oficialmente sob o controle de um IETF (*Internet Engineering Task Force*) Working Group. A primeira versão, HTML1.0 é basicamente a HTML dos clientes WWW, mas a capacidade de mostrar imagens somente ficou disponível em junho de 93. A versão em uso desde dezembro de 94 foi fechada como sendo HTML2.0.

Os principais conversores de texto em formatos variados para HTML foram compilados por Rich Brandwein e estão disponíveis no seguinte endereço URL:

<http://info.cern.ch/hypertext/WWW/Tools/Filters.html>.

4.1. Criando documentos HTML

Os documentos HTML são escritos no formato texto convencional (ASCII) e podem ser criados em qualquer editor. Alguns navegadores de WWW (WWW para X-Windows, CERN para NeXT, etc) possuem um editor HTML bem rudimentar em um ambiente WYSIWYG (*What You See Is What You Get*). Alguns editores facilitam a criação de páginas HTML, como o HTML ASSISTANT, HTML Editor ou o Editor W3e. Este último pode ser encontrado no NCE/UFRJ (Núcleo de Computação Eletrônica) e pode ser obtido via "FTP anonymous" em <http://www.nce.ufrj.br>. Outros endereços relativos a conversão e editoração podem ser encontrados no seguinte endereço URL:

<http://www.vvw.ac.nz/who/Nathan.Torkington/ideas/www-html.html>

Você pode visualizar um documento sendo criado com o Mosaic ou o Netscape. No caso do Mosaic, abra a opção Open Local a partir do menu File. Após editar o arquivo HTML, salve as mudanças. Retorne ao Mosaic e recarregue o documento (*RELOAD*). As mudanças aparecerão diretamente na tela.

4.2. O documento HTML mínimo

Abaixo está listado um exemplo bem simples de um documento HTML, criado com qualquer editor de texto ASCII.

```
<TITLE>Exemplo simples de HTML</TITLE>
<H1>Este é um subtítulo de nível 1</H1>
 Bem vindo ao mundo do HTML
 Este é um parágrafo.<P>
 E este é um segundo.<P>
```

A HTML utiliza os marcadores para informar ao navegador Web como mostrar o texto (informação). O exemplo acima usou:

* o marcador <TITLE> e o seu correspondente </TITLE>, que especifica o título do documento

- * o marcador de subtítulo <H1> e seu correspondente </H1>
- * o marcador de separador de parágrafos <P>

Como mencionado anteriormente, os marcadores HTML são delimitados pelos símbolos “menor que” (<) e “maior que” (>), e no interior está presente o nome do marcador. Os marcadores são usados aos pares, i.e., <H1> e </H1>. Os marcadores de finalização são iguais aos marcadores de início, exceto por possuírem o símbolo ‘/’ dentro do marcador. No exemplo, o marcador <H1> informa ao navegador Web para iniciar a formatação de um subtítulo de nível 1; o marcador </H1> informa ao navegador que a formatação acabou.

A única exceção à regra dos pares é o marcador de parágrafo <P>, só sendo necessária a marcação inicial. Devemos ressaltar que nem todos os navegadores WWW aceitam todos os marcadores definidos no padrão HTML. Se um determinado navegador não aceita um certo marcador, ele simplesmente o ignora.

É importante ressaltar que o HTML não faz distinção entre letras maiúsculas e minúsculas, assim sendo <TITLE>, <title> e <TITIE> representam o mesmo marcador.

5. OS SOFTWARES MOSAIC E NETSCAPE

Ambos os softwares são os meios de navegação pelo “World Wide Web”. Eles são na realidade sistemas de hipermídia projetados para procurar e recolher informações disponíveis na Internet. Eles apresentam uma interface unificada que é capaz de reconhecer vários protocolos, formatos de dados e arquivos usados na Internet. Ambos implementam a filosofia cliente/servidor, onde a parte do servidor corresponde a um computador que se encontra em qualquer parte da Internet (configurado como servidor WWW), e a parte cliente é implementada tanto pelo Mosaic quanto pelo Netscape. O Mosaic foi desenvolvido pela *National Center for Supercomputing Administration* (NCSA da Universidade de Illinois nos EUA - <http://www.ncsa.uiuc.edu/>), e é um software de domínio público e pode ser conseguido diretamente através de um ftp anonymous na NCSA. O Netscape é um software privado, porém é distribuído gratuitamente pela NETSCAPE COMMUNICATIONS CORPORATION (<http://www.mcom.com/>) com autonomia para 90 dias.

Ambos os softwares permitem ao usuário configurar os parâmetros de formatação presentes num documento HTML e elaborar uma lista de ponteiros para páginas em outros servidores personalizados (através dos *Bookmarks*). Eles exigem evidentemente que o computador esteja conectado em rede, falando o protocolo TCP/IP. O computador rodando o software cliente exige uma interface gráfica com o usuário e devido as transferências de imagens, que ocupam geralmente uma grande parte do tempo de transferência, eles requerem

um computador cliente com placa gráfica de alta resolução¹. Devemos ressaltar que a velocidade destes softwares pode se tornar relativamente lenta, pois os atuais *links*, presentes no hipertexto, são ainda lentos em alguns horários, principalmente os internacionais. A REDE-RIO tem feito um enorme esforço para efetuar melhorias nos *links* internos e externos, porém a demanda crescente de utilização é ainda maior que as melhorias atuais.

O Netscape apresenta uma vantagem prática importante sobre o Mosaic, pois ele permite a manipulação do documento HTML recebido enquanto ele se encarrega de receber as imagens que estão no seu interior. A Figura 1 apresenta o layout de ambos os softwares. Na figura a URL especificada aponta para a *Home Page* do CBPF, através do endereço: <http://www.cbpf.br/>. Numa página HTML aparecem freqüentemente algumas palavras ou frases sublinhadas. Estas representam *hiperlinks*, isto é, uma vez selecionadas com auxílio do *mouse* elas requisitarão uma nova conexão com um servidor em qualquer lugar na Internet. Estas conexões podem ser de vários tipos, e serão comentadas posteriormente.

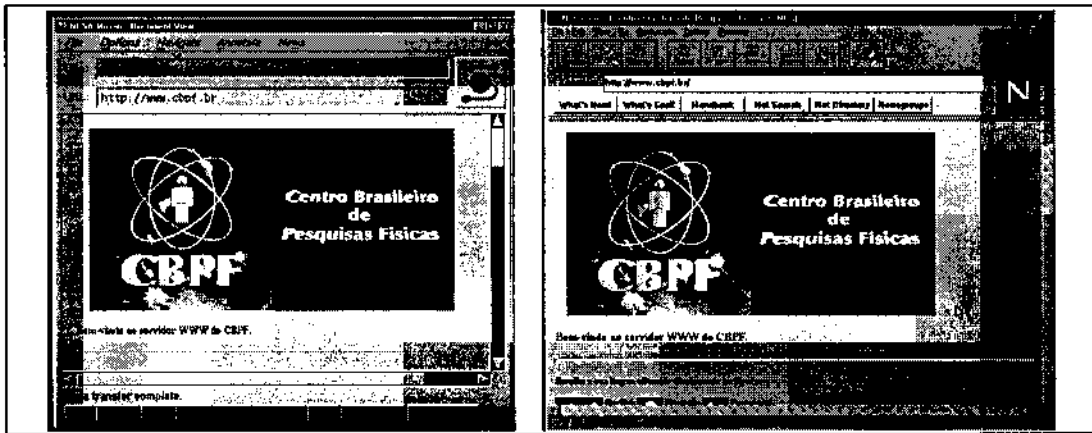


Figura 1 - Representação dos softwares Mosaic e Netscape. Ambos são softwares que permitem a navegação pelo World Wide Web. São sistemas de hipermídia projetadas para recolher e procurar informações na Internet.

6. A MANIPULAÇÃO DE IMAGENS

Este item é dedicado à parte artística na construção de um serviço W3, pois devemos freqüentemente manipular imagens que estejam em perfeito acordo com os serviços e *links* oferecidos. Desta forma pretendemos dar alguns conceitos básicos que possam servir como pontos introdutórios quando da criação de uma imagem.

Uma das maiores dificuldades em se trabalhar com imagens é a falta de padronização visando facilitar a troca de dados em diferentes máquinas. Apesar de uma padronização definitiva ainda não ter sido elaborada, o crescente mercado da multimídia impôs uma

¹ - Alguns sistemas computacionais permitem a consulta ao WWW através de uma tela texto, do tipo VT100. Para isto deve-se rodar um software cliente chamado Lynx.

necessidade de padronização pelos fabricantes. Desta forma, cada fabricante propôs uma pré-padronização, aguardando que novas técnicas poderosas de processamento de imagens possam definir aquele padrão que sobressairá como padrão definitivo para manipulação de imagens. Este problema é encontrado por exemplo na diversidade de formatos de estocagem de imagens. Atualmente nós encontramos uma grande variedade de formatos, onde cada um implementa uma técnica ou uma codificação diferente, promovendo o ganho em espaço ou qualidade. De uma forma geral alguns formatos de imagens parecem sobreviver dentro dos serviços WWW. Alguns deles são: o formato GIF (Graphic Interchange Format, da empresa americana CompuServe), o TIFF (ou TIF no caso dos PC's, que quer dizer Tag Image File Format, da empresa americana Aldus) e o JPG do grupo de especialistas em fotografia. No caso de filmes o mais difundido atualmente é o MPEG, definido pelos especialistas da "Motion Pictures Group".

Devemos ressaltar que a característica principal de uma imagem digital é que cada pixel representa uma intensidade luminosa (luminância), captadas pelos elementos sensíveis de uma camera ou de um scanner. No caso de imagens coloridas a luminância esta dividida nas três cores básicas, o vermelho, o verde e o azul (RGB). Uma imagem monocromática, de qualidade padrão, geralmente codifica a luminância em 256 níveis de cinza, isto é em 8 bits. A imagem também é codificada espacialmente, através da determinação do número de pixels por polegada (dpi's). Especificamente no caso de imagens coloridas nós podemos encontrar freqüentemente dois casos: as imagens de 256 cores ou aquelas em cores reais (*true color*). Em 256 cores, cada pixel da imagem digital aponta para uma tabela de 256 posições onde cada posição define as três componentes RGB definindo assim a cor do pixel (estas imagens são geralmente conhecidas como imagens indexadas). Esta técnica limita a qualidade da imagem, pois o número total de cores é limitado. Apesar deste limite, muitos casos apresentam resultados satisfatórios, e por isto este tipo de codificação é amplamente utilizada. Nas imagens em cores reais ("true color") cada pixel mapeia 3 bytes correspondentes a cor em questão. É evidente que neste caso teremos uma gama imensa de possibilidades de cores, mas em contrapartida o arquivo final é extremamente maior, e assim devemos também possuir uma placa gráfica que implemente o sistema de *true color*, o que não é o caso da maioria das placas gráficas SVGA modernas. Podemos em alguns casos incluir um quarto campo ao lado das cores básicas que indica o grau de transparência do pixel, permitindo a realização de montagens de uma imagem com alguma outra, [Lindley - 91].

Um outro ponto importante que foge ao escopo deste documento são as técnicas de compressão de imagens. Porém devemos mencionar alguns pontos principais neste assunto, pois alguns formatos que trabalhamos na geração de imagens podem implementar estas técnicas. O formato TIF é o codificador mais aberto em termos de estocagem de imagem, nele podem ser definidos várias formas de compressão de imagens, algumas sem perdas de informação, outras com perdas². Em geral no momento em que salvamos a imagem somos questionados sobre o tipo de compressão que queremos usar. O formato JPG permite que

² - Devido a versatilidade deste formato, em termos de opção de evolução, ele tem grandes chances de se tornar o padrão de codificação de imagens.

estabelecamos um grau de compressão em função de uma perda ponderada em resolução e qualidade da imagem. Já o formato GIF da CompuServe não permite a manipulação da imagem no momento da gravação, para que ganhemos em espaço de estocagem. De uma forma geral estes três formatos são muito bem aceitos no mundo WWW, por todos os softwares de leitura, seja o Netscape ou o Mosaic.

7. DETALHES DA LINGUAGEM HTML

Esta é a tradução³ do guia para a criação de documentos em HTML, a linguagem de marcação (markup language) usada convencionalmente pelo WWW (World Wide Web). O guia original pode ser encontrado em:

<http://www.ncsa.uiuc.edu/demoweb/html-primer.html>

7.1. Marcadores Básicos

Title: Cada documento HTML deve ter um título. Um título aparece geralmente separado do documento e é usado para identificar o documento em outros contextos (por exemplo: busca no WAIS). Nas versões do NCSA-Mosaic para o X-Windows e MS-Windows o campo *Document Title* fica no topo da tela abaixo dos menus, ou na barra título das janelas.

Subtítulos: o HTML tem 6 níveis de subtítulos, numerados de 1 até 6; com o 1 sendo o mais proeminente. Os subtítulos são mostrados com um tipo de letra maior e mais escuro (negrito) que o texto normal. O primeiro subtítulo em cada documento deverá ser marcado como <H1>. A sintaxe da marcação dos subtítulos é:

<H_y>Texto do subtítulo </H_y>

Onde y é um número de 1 a 6 especificando o nível do subtítulo.

Parágrafos: diferente dos documentos da maioria dos processadores de textos, os CRs no HTML não são significantes. O texto sempre ficará formatado em qualquer ponto do arquivo e espaços múltiplos serão convertidos em um único espaço. Porém, existem algumas exceções: o espaço entre um <P> ou um marcador <H_y> por exemplo, será ignorado.

Você deverá separar os parágrafos com <P>. O navegador vai ignorar quaisquer identações ou linhas em branco no texto original. O HTML se baseia exclusivamente nos marcadores para fazer a formatação das instruções, e sem marcadores <P> o documento se torna um enorme parágrafo. A exceção é o texto marcado como "pre-formatado" que será

³ Parte desta tradução foi feita pelo NCE: <http://www.nce.ufjf.br/~cracky/html-1.html>

explicado posteriormente. Por exemplo, as sentenças a seguir produzem uma saída idêntica ao primeiro exemplo mostrado aqui:

```
<TITLE>Exemplo simples de HTML</TITLE>
```

```
<H1>Este é o subtítulo 1</H1>Bem vindo ao mundo do HTML. Isto
é um parágrafo.<P>E este é um segundo.<P>
```

Entretanto, para preservar a legibilidade dos arquivos HTML, os subtítulos deverão ficar em linhas separadas, e os parágrafos deverão vir separados por linhas em branco, mais o marcador <P>. No HTML+, o sucessor do HTML ainda em desenvolvimento, o marcador <P> se torna um *container* de texto, assim como o texto de um subtítulo está contido dentro do <H1> ... </H1>:

```
<P>
Isto é um parágrafo no HTML+.
</P>
```

A diferença é que o marcador de término </P> pode ser omitido, isto é, se um navegador vê um marcador <P>, ele sabe que deve haver um </P> implícito no fim do parágrafo. Em outras palavras, no HTML+, o marcador <P> é um marcador de início de parágrafo.

A grande vantagem é que você pode especificar opções de formatação para um determinado parágrafo. Por exemplo, no HTML+, você pode centralizar um parágrafo escrevendo:

```
<P ALIGN=CENTER>
```

Este marcador define é um parágrafo centralizado. É HTML+, portanto você ainda não pode fazer isto. Esta mudança não causará efeito em nenhum documento que você escreva agora, e continuarão com a mesma aparência vista nos navegadores HTML+.

7.2. *Ligação com Outros Documentos*

Esta grande característica da HTML vem de sua habilidade em ligar regiões de texto e imagens à outros documentos. O navegador destaca esta região com outras cores e/ou sublinhado para indicar que são *links* de hipertextos, geralmente abreviados como *hyperlinks* ou simplesmente *links*. O marcador para referências de hipertexto é o <A>, que espera por uma âncora. Para incluir uma âncora em seu documento:

- 1 - Inicia a âncora com <A . Atenção: ponha um espaço após o A.
- 2 - Especifique o documento que será apontado escrevendo o parâmetro HREF="NomeDoArquivo" seguido de um símbolo de "maior que".

3 - Escreva o texto que servirá como link do hipertexto no documento corrente.

4 - Ponha o marcador para terminar a âncora: ``.

Veja um exemplo:

```
<A HREF="MaineStats.html">Maine</A>
```

Esta entrada transforma a palavra "Maine" no *hiperlink* para o documento `MaineStats.html`, que está no mesmo diretório que o primeiro documento. Você pode ligar documentos em outros diretórios especificando o *path* completo do documento corrente para o documento a ser ligado. Por exemplo, um *link* para um arquivo `NJStats.html` localizado no sub-diretório `AtlanticStates` ficaria assim:

```
<A HREF="AtlanticStates/NJStats.html">New Jersey</A>
```

Isto é chamado de *link* relativo. Você também pode usar o *path* completo do arquivo que deseja. Note, porém que os *paths* devem ser escritos na sintaxe padrão do UNIX.

Vamos abordar quatro aspectos relevantes de *link*:

1. **Links relativos versus Paths completos:** Em geral, você deve usar os *links* relativos pois: i) você escreve menos; ii) é mais fácil mover um grupo de documentos de um lugar para outro, porque os *paths* relativos continuarão válidos. Entretanto, devemos usar os *paths* completos quando ligamos documentos que não estão diretamente relacionados. Por exemplo, considere um grupo de documentos que formam um manual. Os *links* dentro do grupo devem ser relativos. Os *links* para outros documentos, talvez uma referência para um software, deverão ser escritos com os *paths* completos. Com isto, se você move o manual para um diretório diferente, nenhum dos *links* precisarão ser atualizados.

2. **Uniform Resource Locator (URL):** o WWW usa os URLs para especificar a localização dos arquivos em outros servidores. Um URL inclui o tipo do recurso que está sendo acessado, i.e., um Gopher, WAIS, etc, o endereço do servidor e a localização do arquivo. A sintaxe é: "TIPO://host.domain[:port]/path/nomedoarquivo", onde TIPO é um destes :

file:	um arquivo no seu sistema local, ou em um servidor FTP-anonymous
http:	um arquivo no servidor WWW
gopher:	um arquivo no servidor Gopher
wais:	um arquivo no servidor WAIS
news:	um grupo USENET-newsgroup
telnet:	uma conexão com um serviço do tipo TELNET

O número da porta em [:port] pode ser omitido. A menos que a variável *port* venha especificada, desconsidere este parâmetro na linha de comando.

A marcação definida na linha abaixo, fará o documento "NCSA's Beginner's Guide to HTML" um *hyperlink* para o seu documento:

```
<A HREF="http://www.ncsa.uiuc.edu/General/Internet/WWW/HTMLPrimer.html">
  NCSA's Beginner's Guide to HTML</A>
```

3. *Links* para seções específicas em outros documentos: âncoras podem ser usadas para navegarmos para uma seção particular de um documento. Suponhamos que desejássemos definir um *link* entre o documento A e uma seção particular do documento B (documentoB.html). Primeiro devemos definir uma âncora no documento B:

```
Here's <A NAME="MyReference">some text</A>
```

O segundo passo é criar o *link* no documento A, incluindo além do nome do arquivo, o nome da âncora separados pelo símbolo #.

```
This is my <A HREF="documentB.html#MyReference">link</A> to document B
```

Assim sendo, ao se efetuar o *click* na palavra "link" no documento A, o usuário irá direto para o documento B, onde se encontram as palavras "some text" no documento B, referenciado pelo Name MyReference.

3. *Links* para Seções Específicas no Documento Corrente: a técnica é exatamente a mesma, apenas o nome do arquivo é omitido. Para fazer um *link* para âncora "MyReference" de dentro do mesmo arquivo (documento B), usamos:

```
This is <A HREF="#MyReference">Referencia link</a> from within Document B
```

7.3. Marcadores Adicionais

As duas seções anteriores, Marcadores Básicos e Ligação com Outros Documentos, são suficientes para produzir documentos HTML simples. Para documentos mais complexos a linguagem HTML possui outros marcadores mais complexos, dos quais passaremos a descrever alguns.

- **Listas:** HTML suporta listas numeradas, listas não numeradas e definições de listas. Os exemplos abaixo falam por si só.

<code></code>	<code></code>
<code>maçãs</code>	<code>maçãs</code>
<code> bananas</code>	<code> bananas</code>
<code>uvas</code>	<code>uvas</code>
<code></code>	<code></code>

Teremos como resultados, respectivamente:

• maçãs	1. maçãs
• bananas	2. bananas
• uvas	3. uvas

Existe ainda a possibilidade de precisarmos de listas compostas, bastando para isto utilizarmos o marcador `` ou `` após `` e assim por diante.

Listas de Definição usualmente consiste de um termo alternativo (DT) e uma definição (DA). Navegadores WWW, usualmente formatam a definição em uma nova linha. Vamos a um exemplo:

```

<DL>
<DT>CBPF
<DD>CBPF, Centro Brasileiro de Pesquisas Físicas é um instituto do
CNPq, com atuação em pesquisa básica em Física, com atuação
principalmente nas áreas de Estado Sólido, Física Nuclear, Física de
Partículas, Teria de Campos e Biofísica.
<DT>CAT
<DD>CAT, Coordenação de Atividades Técnicas é um setor do CBPF que
presta serviços de apoio, desenvolvimento e projetos nas áreas de Química,
Eletrônica, Informática, Mecânica e Vidraria Fina.
<DT>CDI
<DD>CDI, Coordenação de Documentação e Informação Científica é um
setor do CBPF que presta serviços de apoio e consulta a dados bibliográficos
de uma maneira geral.
</DL>

```

Como resultado, teremos:

CBPF

CBPF, Centro Brasileiro de Pesquisas Físicas é um instituto do CNPq, com atuação em pesquisa básica em Física, com atuação principalmente nas áreas de Estado Sólido, Física Nuclear, Física de Partículas, Teria de Campos e Biofísica.

CAT

CAT, Coordenação de Atividades Técnicas é um setor do CBPF que presta serviços de apoio, desenvolvimento e projetos nas áreas de Química, Eletrônica, Informática, Mecânica e Vidraria Fina

CDI

CDI, Coordenação de Documentação e Informação Científica é um setor do CBPF que presta serviços de apoio e consulta a dados bibliográficos de uma maneira geral.

• **Texto Preformatado:** utilizando o marcador composto `<PRE> ... </PRE>`, fazemos o interpretador HTML considerar fontes com largura fixa, espaços, novas linhas e tabulação. Múltiplos espaços serão considerados, assim como linhas em branco e caracteres de tabulação. É importante ressaltar que alguns caracteres que possuem significado especial na HTML, tais como `<` `>` e `&` são obtidos através das seqüências `<`, `>` e `&`, respectivamente. Existe uma lista destes caracteres no seguinte endereço URL:

<http://www.ncsa.uiuc.edu/General/Internet/WWW/HTMLPrimer.html>

• **Endereços:** o marcador `<ADRESS>` é geralmente usado para especificar o autor do documento e uma maneira de obter contato com ele, normalmente seu endereço *email*. Observe que este marcador não é um ponteiro para um endereço URL, apenas um formato típico para visualização: o *email* assume a forma pequena, negrito e itálica.

```
<ADRESS>
Meu endereço é naj@cat.cbpf.br
</ADRESS>
```

• **Formatação de Caracteres:** a linguagem HTML suporta a formatação de palavras ou frases em vários estilos. Existem dois tipos de estilos: o lógico e o físico. A diferença entre eles é muito sutil e pode ser vista considerando que o estilo lógico formata o texto de acordo com o seu significado, enquanto que o estilo físico define uma aparência específica para o texto. Por exemplo: a SGML define o primeiro nível de cabeçalho somente como um primeiro nível de cabeçalho, porém não especifica se este nível deve ser relativo a fonte TIMES em negrito e de 24 pontos centrado no alto da página. É possível posteriormente modificar para fonte HELVÉTICA de 20 pontos justificado a esquerda. Uma vez mudado o estilo lógico, todas as marcações no documento mudarão.

Alguns estilos lógicos: `<CITE>` para títulos de documentos, normalmente em itálico; `<CODE>` para representar códigos de computadores; `` para enfatizar, normalmente em negrito; ...

Alguns estilos físicos: `` texto em negrito, `<I>` texto em itálico, `<TT>` texto com fonte de largura fixa.

Como em quase todos marcadores, o efeito atua no texto interno aos marcadores inicial e final:

`Este texto será mostrado em negrito.`

`<I>Este outro em itálico.</I>`

O resultado fica :

Este texto será mostrado em negrito

Este outro em itálico

- **Fim de Linhas:** o marcador `
` força o fim de linha sem espaços extras entre a linha e a posterior.

- **Traço Horizontal:** o marcador `<HR>` produz um traço horizontal com comprimento variável automático, de acordo com o tamanho da janela de exibição do documento.

7.4. Imagens

Os navegadores WWW atuais suportam a exibição de imagens, entre outros formatos GIF, TIFF, JPEG ou XBM, e textos simultaneamente. Apesar de ser mais agradável e muitas vezes intuitivo, a utilização de imagens toma tempo de processamento e aumenta o tempo de demora de exibição do documento. Assim sendo, é recomendável a utilização criteriosa de imagens nas páginas HTML, até mesmo porque a velocidade de transmissão em nosso país é muito baixa (*link* RNP/REDE RIO 256Kbits/s) se comparada com a de outros países. A inclusão de imagens no documento é semelhante a utilização da âncora HREF. Abaixo está a sintaxe geral:

``

onde *alinhamento*, se definido, pode assumir os valores BOTTOM, TOP ou MIDDLE e configura o alinhamento do texto com a imagem. O parâmetro *imagem_URL* especifica o nome da imagem que pode conter qualquer endereço URL no seu conteúdo. A figura 2 exemplifica o uso de imagens com a linguagem HTML. Os comandos utilizados são os seguintes :

`<a img align=top src="gif/English.gif"> English`

`<a img align=top src="gif/Francais.gif"> Français`

`<a img align=top src="gif/Brasil.gif"> Português`



Figura 2 - Utilização de imagens e texto na linguagem HTML.

7.5. Formulários

Definimos formulários como sendo páginas HTML, que possuam campos onde o usuário tem acesso obrigatório ou não, para entrar com algum tipo de dado. A parte principal destes formulários é relativa aos campos de entrada, ou em linguagem de programação, janelas de extração. O marcador utilizado é o `<FORM>...</FORM>`. Um exemplo típico envolve além deste marcador, vários atributos que especificam ação, método, tipo de entrada, valor inicial, tamanho da janela, tamanho máximo da entrada, etc. Vamos analisar o exemplo abaixo:

```
<FORM METHOD="POST" ACTION="http://www.cbpf.br/cgi-bin/sinfocx">
<IMG SRC="gif/blueball.gif">Autor: <INPUT TYPE="text" SIZE="40" NAME="por">
<P>
<IMG SRC="gif/blueball.gif">Título: <INPUT TYPE="text" SIZE="40" NAME="Nome">
<P>
<HR>
<INPUT TYPE="submit" VALUE=" Inicia Procura ">
</FORM>
```

Observe no marcador inicial a definição do método POST para submeter a saída do formulário ao servidor de requisições. Apesar deste ser o método mais indicado para os sistemas servidores, existe um outro chamado GET. A diferença está no fato deste último enviar ao servidor as informações como parte da URL, enquanto que aquele utilizado no exemplo envia as informações obtidas no formulário numa parte destinada ao envio de dados, como se fossem variáveis.

O parâmetro ACTION define o endereço URL que irá utilizar as informações obtidas no formulário. Se omitido, o URL do documento atual passará a ser utilizado.

É recomendável, a título de estética, que se utilize o marcador de linha, `<HR>`, antes e depois de `<FORM>...</FORM>` para destacar a área do formulário do resto da página HTML. Entre os marcadores inicial e final do formulário pode haver qualquer outro marcador a exceção de outro formulário. Especificamente falaremos do marcador INPUT, porém existe ainda os marcadores SELECT e TEXTAREA.

O marcador INPUT é utilizado para especificar um único elemento de entrada de informação por vez e portanto não possui marcador final. Os atributos mais freqüentes são:

TYPE - podendo ser *text* que é o padrão, *password* onde os caracteres são representados por asterísticos, *checkbox* com duas opções ON ou OFF, *submit* que envia a informação para o arquivo URL no servidor remoto, ...

NAME - é um nome simbólico que não é exibido na página HTML. É utilizado para controle interno da requisição ao servidor.

SIZE - é o tamanho físico da janela de extração em caracteres. O padrão é 20.

Para obter mais informação sobre este tema, bem como as diversidades de aplicações, visite a NCSA no endereço URL abaixo:

<http://www.ncsa.uiuc.edu/SDG/Software/Mosaic/Docs/fill-out-forms/overview.html>

8. COMMON GATEWAY INTERFACE (CGI)

A Common Gateway Interface (CGI) é um padrão para interfacear aplicações externas com servidores de informação, tais como HTTP e Servidores Web (ver referência [CGI]). Um documento HTML é estático, o que significa que ele existe em estado constante, ou seja, é apenas um arquivo texto que não muda. Um programa CGI, ao contrário, é executado em tempo real, de tal forma que sua saída é uma informação dinâmica.

Vamos considerar que alguém precise tornar acessível uma base de dados UNIX através do WWW, para que pessoas distantes possam utiliza-la. Basicamente é necessário criar um programa CGI que o *daemon Web* execute e transmita a informação ao servidor da base de dados e receba de volta ao cliente a resposta à sua requisição. Isto é um exemplo de um *gateway*, e é também onde CGI 1.1 teve sua origem.

Quando utilizamos o termo *gateway*, estamos nos referindo a programas que guiam as requisições de informação e retornam ou geram documentos. No caso de programas CGI, o servidor pode fornecer informações que não estejam na forma visual para o cliente, tal como um banco de dados SQL, e atua como uma ponte entre o cliente e o servidor, produzindo algum elemento que pode ser utilizado pelo usuário.

O exemplo da base de dados é bem simples apesar de quase sempre ser difícil a sua implementação. Na realidade não existe limites para um programa CGI, porém deve ser sempre lembrado que o programa não deve tomar muito tempo de processo pois o usuário estará esperando pela resposta a sua requisição.

Os programas CGI podem ser escritos em qualquer linguagem como por exemplo: C/C++, PERL, TCL, The Bourne Shell, The C Shell, FORTRAN, VISUAL BASIC, AppleScript, Para aqueles que querem utilizar a linguagem C, existem alguns programas fornecidos pelo CERN ou pela NCSA, que permitem a comunicação com o servidor. Um destes programas se chama `post-query.c` e a sua listagem se encontra no anexo B. Este programa para ser utilizado deve ser compilado ao lado do `util.c`, também disponíveis no CERN ou na NCSA.

9. SINFOCX/CBPF

Em 92 começou-se a desenvolver um sistema gerenciador das bases de dados da Coordenação de Documentação e Informação Científica do CBPF, baseado na arquitetura cliente/servidor (Netware SQL386 - bibliotecas de funções XQL e CXL e Linguagem C) para ser acessado através dos sistemas operacionais DOS e de rede NOVELL. Este sistema foi implementado e sofisticado enquanto o inventário de toda biblioteca (80.000 volumes) era realizado, tornando confiável todas as bases de dados envolvidas (livros, jornais, revistas, periódicos, monografias e usuários). Em maio de 95 este sistema tornou-se público e com uma média de 30 acessos/dia.

O segundo passo na informatização daquelas bases de dados seria oferecer acesso externo através da rede Internet. Assim sendo, estamos desenvolvendo o Sistema de INFORMAÇÃO Científica em ambiente UNIX (SINFOCX) que é um sistema gerenciador de bases de dados totalmente desenvolvido pela Coordenação de Atividades Técnicas (CAT/CBPF). Atualmente já implantados o acesso a base de dados de livros e monografias (18.000) e o acesso as bases de jornais, revistas e periódicos (62.000). Futuramente estará oferecendo serviços de consultas também às bases de *pre-prints*, conferências e seminários, COMUT, instituições de pesquisas nacionais e internacionais, etc.

O SINFOCX pode ser acessado através da *home page* (HP) do CBPF ou diretamente pelo endereço URL:

<http://www.cbpf.br/CDI/biblin.html>.

É um sistema totalmente desenvolvido em linguagem C, e cujas bases de dados são arquivos ASCII gerados pelo sistema SINFOC. Os resultados iniciais revelaram a necessidade de uma máquina dedicada para estes serviços no futuro, quando houver múltiplos acessos simultâneos. Este fato é esperado a medida que o servidor WWW/CBPF começar a ser mais divulgado, pois a biblioteca é considerada a mais completa da América Latina na área de Física. Está em fase de estudos de viabilidade técnica e financeira, a implantação de um

servidor de bases de dados comercial (hardware e software) para uso e gerenciamento otimizado da tecnologia cliente/servidor.

Na construção da página HTML de entrada e saída do SINFOCX, utilizamos os marcadores, bem como os conceitos de formulários e CGI discutidos nas seções anteriores.

10.O SERVIDOR WWW

Uma estação deve ser escolhida como servidora e configurada para responder às solicitações de clientes WWW. O protocolo de transferência de hipertextos HTTP (*Hypertext Transfer Protocol*) é o formato usado pelos servidores WWW desde de 1990. O HTTP é um protocolo ao nível de Aplicação na camada OSI com características necessárias para implementar uma rápida troca de arquivos entre as máquinas cliente e servidora. Uma outra característica do protocolo HTTP é que ele é um gerenciador que permite a transferência de informações independentemente do seu tipo de suporte (texto, som, imagens, filmes, etc), ver referência [HTTP]. A configuração de uma máquina servidora WWW implica em duas fases: i) na definição de alguns parâmetros básicos em um arquivo de regras e ii) na execução de um programa (httpd) que ficará residente e que responderá as solicitações dos clientes. É sobre este programa httpd e ao seu arquivo de configuração que nós nos interessaremos nos parágrafos seguintes.

10.1. O http daemon

O httpd é o software residente que responde as solicitações de estações servidoras. Para efetuarmos a sua instalação devemos configurar alguns parâmetros básicos dentro de arquivos que são carregados automaticamente no momento em que o httpd é executado⁴. Antes de falarmos um pouco sobre a configuração devemos mencionar que existem dois “construtores” de softwares httpd, o NCSA e o CERN. As versões atuais são bastante compatíveis entre elas, porém os números de versão são independentes, e devemos ficar atentos no momento em que estamos acessando os documentos de configuração seja no CERN ou no NCSA. Os códigos são disponíveis em ambas localidades (<http://www.w3.org/> ou <http://www.ncsa.uiuc.edu/>) para diversas plataformas nos formatos binário e fonte (ver referência [HTTPD]).

10.2. Arquivo de configuração

O httpd carrega um arquivo de configuração chamado httpd.conf. Este arquivo de configuração define como o httpd vai traduzir uma requisição de uma informação em um documento. Dentro deste arquivo encontramos uma série de definições e de opções. Cada

⁴- Tais modificações só são realizadas com a autoridades do supervisor do sistema (*user root*), pois estamos reconfigurando uma parte da estação.

linha no seu interior que não inicie com o caracter '#' é uma linha de definição de um parâmetro para o servidor. A definição é feita editando diretamente o arquivo `httpd.conf` através de editores ASCII. No Anexo C você poderá encontrar uma configuração exemplo de um servidor WWW. A execução do `httpd` será da seguinte maneira :

```
httpd -r /my/config/directory/httpd.conf
```

O arquivo de configuração é especificado através do *flag* `(-r)`⁵. Antes de executarmos o `httpd` precisamos definir um diretório de trabalho, isto é, um local onde todos os arquivos `.html` do servidor estarão presentes. Suponhamos para fins ilustrativos que este seja o diretório `/Public/Web`. Devemos desta forma adicionar a seguinte linha ao arquivo `httpd.conf`:

```
Pass /* /Public/Web
```

Suponhamos para melhor compreensão que o nosso *host* tenha o seguinte nome: `www.myhost.br` (endereço IP: `200.32.33.127`). O arquivo default utilizado pelo servidor rodando o *daemon* do CERN no momento de uma solicitação é o `"Welcome.html"`. Desta forma quando algum cliente solicita a URL `http://www.myhost.br` o servidor enviará o arquivo `Welcome.html`, que se encontra no diretório `/Public/Web` como resposta, pois nenhum arquivo em especial foi solicitado. Este arquivo `"Welcome.html"` é comumente chamado de *HomePage*.

O `httpd` pode rodar em dois modos : em *"Stand-alone"* ou de acordo com *"Internet daemon"* (`inetd`). No modo *Stand-alone* o `httpd` é executado no momento de *boot* da estação de trabalho servidora. Ele espera por uma chamada de um cliente e se auto duplica para responde-la. O CERN recomenda que o servidor rode no modo *Stand-alone*. Se no arquivo de configuração `httpd.conf` a variável `ServerType` não estiver definida, o servidor entrará automaticamente no modo *"StandAlone"*.

Outro parâmetro à definir dentro do arquivo `httpd.conf` é o nome do usuário que será responsável pelo funcionamento do `httpd`. Este parâmetro é importante no caso de envio de mensagens ou de execução de scripts ou programas, através da interface CGI. De uma forma geral recomendamos à criação de um usuário *webmaster*, fazendo com que o servidor responda pelo endereço eletrônico `webmaster@www.myhost.br`.

Uma opção durante o uso do *daemon* `httpd` é na criação de arquivos *Logging*, para arquivar possíveis erros ou os acesso ao servidor. Uma vez configurado este serviço, os dados de acesso e de erros são guardados em arquivos, permitindo contabilizar em seguida o acesso

⁵ - Se este nome não estiver presente no momento do lançamento ele será substituído por um nome *default* correspondente à `"etc/httpd.conf"`.

aos documentos disponíveis. Os arquivos são definidos pela linha *AcessLog* e *ErrorLog*. O diretório root também é definido no arquivo *httpd.conf* através da linha *ServerRoot*. Os scripts e os programas executáveis são chamados através da Interface CGI, descritas anteriormente. Todos os arquivos executáveis devem estar dentro do diretório definido na linha *Exec*, como por exemplo:

```
Exec /* /Public/Web/cgi-bin
```

10.3. Para maiores detalhes

Para ver maiores detalhes sobre a configuração do servidor e as suas diversas opções de funcionamento veja os próprios arquivos disponíveis no “*hiperlink*” do CERN, no endereço : “<http://www.w3.org/hypertext/WWW/Daemon/User/CommandLine.html>”. Uma opção que pode vir a ser útil no caso de problemas de configuração é o *flag* (-v) ou (-vv) que indicam modo de *debug*, fazendo com que o *httpd* entre em modo “*verbose*”.

Para terminar não podemos deixar de mencionar que o nome do servidor depende exclusivamente do *Domain Name System* (DNS) no qual esta definida a rede onde trabalhamos. De qualquer forma uma vez o *httpd* instalado a máquina passa a ser visível através do seu endereço IP (no nosso exemplo é 200.32.33.127). Recomendamos fortemente a utilização de nomes intuitivos do tipo: *www.instituto.br* pois eles facilitam o encontro da estação servidora por qualquer pessoa que não conheça a sua localidade. Em seguida recomendamos que você cadastre o seu servidor junto a organizações nacionais e internacionais (RNP, Embratel, CERN, NCSA, etc).

ANEXO A

Links diversos

LINKS NACIONAIS

<p>"http://www.cr-sp.rnp.br/" São Paulo, SP</p> <p>"http://www.unicamp.br/Welcome.html"</p> <p>"http://www.unicamp.br/Welcome.html"</p> <p>"http://www.unicamp.br/Welcome.html"</p> <p>"http://www.dcc.unicamp.br/"</p> <p>"http://dca.fee.unicamp.br"</p> <p>"http://www.fee.unicamp.br"</p> <p>"http://www.iqm.unicamp.br/iqm.html"</p> <p>"http://www.dcc.unicamp.br/cats/"</p> <p>"http://www.ftpt.br"</p> <p>"http://www.cnptia.embrapa.br"</p> <p>"http://www.iisis.org.br/"</p> <p>"http://ecc.ftpt.br"</p> <p>"http://www.cr-sp.rnp.br/"</p> <p>"http://www.cr-sp.rnp.br/"</p> <p>"http://www.futuro.usp.br"</p> <p>"http://www.ime.usp.br"</p> <p>"http://www.if.usp.br"</p> <p>"http://www.lsi.usp.br"</p> <p>"http://www.futuro.usp.br"</p> <p>"http://www.uniemp.br/"</p> <p>"http://www.inpe.br"</p> <p>"http://www.icmsc.sc.usp.br"</p>	<p>Campinas.</p> <p>Unicamp - Universidade Estadual de Campinas.</p> <p>CCuec - Centro de Computação Unicamp.</p> <p>Departamento de Ciência da Computação da Unicamp.</p> <p>Departamento de Engenharia - Faculdade de Engenharia Elétrica.</p> <p>Fee - Faculdade de Engenharia Elétrica.</p> <p>IQ - Instituto de Química.</p> <p>Cats - Programa de Capacitação Avançada em Tecnologias de Software.</p> <p>FTPT - Fundação Tropical de Pesquisas e Tecnologia "André Tosello".</p> <p>CNPtia/Embrapa - Centro Nacional de Pesquisa Tecnológica em Informática para a Agricultura, Empresa Brasileira de Pesquisa Agropecuária.</p> <p>Iisis - Instituto Internacional de Integração de Sistemas.</p> <p>ECC - Escola Comunitária de Campinas.
São Paulo (capital).</p> <p>RNP/SP - Rede Nacional de Pesquisa</p> <p>Usp - Universidade de São o Paulo.</p> <p>Ime - Instituto de Matemática e Estatística.</p> <p>IF - Instituto de Física (USP)</p> <p>LSI - Laboratório de Sistemas Integráveis.</p> <p>Futuro - Projeto Escola do Futuro.</p> <p>Uniemp - Fórum Permanente das Relações das Universidade/Empresa.</p> <p>Inpe - Instituto Nacional de Pesquisas Espaciais / São José dos Campos.</p> <p>ICMSC/Usp - Instituto de Ciências Matemáticas de São Carlos.</p>
<p>"http://www.rnp.br/" Rio de Janeiro, RJ</p> <p>"http://www.rnp.br/"</p> <p>"http://www.inf.puc-rio.br"</p> <p>"http://guarani.cos.ufrj.br/HOME.html"</p> <p>"http://alpha.lncc.br/"</p> <p>"http://www.lafex.cbpf.br/"</p> <p>"http://www.cbpf.br"</p> <p>"http://obsn.on.br/"</p> <p>"http://www.impa.br/"</p> <p>"http://ax.ibase.br/"</p> <p>"http://gemini.mz.cef.gov.br/"</p>	<p>RNP-RJ - Rede Nacional de Pesquisa.</p> <p>Puc-Rio/DI - Dept. de Informática Pont. Univ.Católica do Riode Janeiro.</p> <p>UFRJ/Coppe - Coord. dos Programas de Pós-Graduação em Engenharia Universidade Federal do Rio de Janeiro.</p> <p>LNCC - Laboratório Nacional de Computação Científica.</p> <p>Lafex - Lab. de Cosmologia e Física Experimental de Altas Energias.</p> <p>CBPF - Centro Brasileiro de Pesquisas Físicas.</p> <p>OBSN - Observatório Nacional.</p> <p>Impa - Instituto de Matemática Pura e Aplicada.</p> <p>Ibase/Alternex - Instituto Brasileiro de Análises Sociais e económicas.</p> <p>Cef - Caixa Econômica Federal.</p>
<p>"http://dcc.ufmg.br/" Belo Horizonte, MG.</p>	

"http://dcc.ufmg.br/"
 "http://dcc.ufmg.br/"
 "http://www.cpdee.ufmg.br/"

UFMG - Universidade Federal de Minas Gerais.
 DCC - Departamento de Ciência da Computação.
 CPDEE - Centro de Pesquisa e Desenvolvimento
 em Engenharia Elétrica.

"http://www.inf.ufsc.br/" Florianópolis, SC
 "http://www.inf.ufsc.br/"
 "http://www.inf.ufsc.br/"
 "http://edugraf.ufsc.br/"

UFSC - Universidade Federal de Santa Catarina.
 Inf - Departamento de Informática.
 Edugraf - Laboratório de Software Educacional.

"http://tucano.inf.ufrgs.br/homepage.html" RS
 "http://tucano.inf.ufrgs.br/homepage.html"
 "http://www.cesup.ufrgs.br/"
 "http://tucano.inf.ufrgs.br/homepage.html"
 "http://www.mat.ufrgs.br/mat.html"
 "http://www.ufsm.br/"

Porto Alegre.
 Cesup - Centro de Supercomputação.
 UFRGS/Inf - Instituto de Informática.
 Mat - Instituto de Matemática.
 UFSM - Universidade Federal de Santa
 Maria/Santa Maria, RS.

"http://www.cr-df.rnp.br/home.html" Brasília, DF
 "http://www.cr-df.rnp.br/home.html"
 "http://antares.ibict.br/"

 "http://asparagin.cenargen.embrapa.br/"
 "http://www.eme.br/"

RNP/BSB - Rede Nacional de Pesquisa.
 Ibict/Antares - Rede de Serviços de Informação
 em Ciência e Tecnologia Instituto Brasileiro
 de Informação em Ciência e Tecnologia.
 BBRC - Brazilian Bioinformatics Resource
 Center- Embrapa.
 Estado-Maior do Exército - Ministério do
 Exército.

LINKS INTERNACIONAIS

Sistemas e Hardwares :

"http://www.ibm.com"
 "http://www.austin.ibm.com/pspinfo"
 "http://www.sun.com"
 "http://www.sgi.com"
 "http://www.dec.com"
 "http://www.hp.com"
 "http://www.nec.com"
 "http://www.mips.com"
 "http://www.compaq.com"
 "http://www.us.dell.com"
 "http://www.intel.com"
 "http://www.motorola.com"
 "http://www.apple.com"
 "http://power.globalnews.com"

IBM - International Business Machines."
 IBM - Software.
 Sun - Sun Microsystems."
 SGI - Silicon Graphics Inc."
 DEC - Digital Electronic Computers.
 HP - Hewlett-Packard.
 NEC - Nippon Electronic Computer.
 MIPS - MIPS Technologies, Inc.
 Compaq - Compaq Computer
 Dell - Dell Computer Corp.
 Intel - Intel Corp.
 Motorola - Motorola Corp.
 Apple - Apple Corp.
 Power PC - Home Page.

Softwares (Sistemas, Banco de Dados, Rede, ...)

"http://www.microsoft.com"
 "http://www.lotus.com"
 "http://www.mcom.com"
 "http://www.novell.com"
 "http://www.oracle.com"

Microsoft - Microsoft Corp
 Lotus - Development Corp.
 Nestscape Communications Corp.
 Novell Inc.
 Oracle Corp.

"http://www.sybase.com"	Sybase Inc - SQL Server.
"http://www.taligent.com"	Taligent Inc.
"http://www.borland.com"	Borland .
"http://www.wri.com"	Wolfram Research - MATHEMATICA.
"http://www.maplesoft.com"	Maple - Waterloo Maple Software.
"http://www.ncsa.uiuc.edu"	NCSA - National Center for Supercomputing Administration.
"http://ds.internic.net"	InterNIC - Directory & Database Services (USA).
"http://www.directory.net"	Open Market - 2500 links comerciais e governamentais (USA).
"http://www.uwm.edu/Mirror/inet.services.html"	Conexões especiais na Internet - (USA).

ANEXO B

Programa post-query.c (CGI)

```

cat1[mpa]% more post-query.c

#include <stdio.h>
#include <stdlib.h>

#define MAX_ENTRIES 10000

typedef struct {
    char *name;
    char *val;
} entry;

char *makeword(char *line, char stop);
char *fmakeword(FILE *f, char stop, int *len);
char x2c(char *what);
void unescape_url(char *url);
void plustospace(char *str);

main(int argc, char *argv[]) {
    entry entries[MAX_ENTRIES];
    register int x,m=0;
    int cl;

    printf("Content-type: text/html%c%c",10,10);

    if(strcmp(getenv("REQUEST_METHOD"),"POST")) {
        printf("This script should be referenced with a METHOD of POST.\n");
        printf("If you don't understand this, see this ");
        printf("<A HREF='http://www.ncsa.uiuc.edu/SDG/Software/Mosaic/Docs/fill-out-forms/overview.html'>forms overview</A>.%c",10);
        exit(1);
    }
    if(strcmp(getenv("CONTENT_TYPE"),"application/x-www-form-urlencoded")) {
        printf("This script can only be used to decode form results. \n");
        exit(1);
    }
    cl = atoi(getenv("CONTENT_LENGTH"));

    for(x=0;cl && (!feof(stdin));x++) {
        m=x;
        entries[x].val = fmakeword(stdin,'&",&c);
        plustospace(entries[x].val);
        unescape_url(entries[x].val);
        entries[x].name = makeword(entries[x].val,'=');
    }

    printf("<H1>Query Results</H1>");
    printf("You submitted the following name/value pairs:<p>%c",10);
    printf("<ul>%c",10);

    for(x=0; x <= m; x++)
        printf("<li> <code>%s = %s</code>%c",entries[x].name,
            entries[x].val,10);
    printf("</ul>%c",10);
}

```

Obs : Para efetuar este programa use o util.c, que pode ser encontrado em "<http://www.ncsa.uiuc.edu/SDG/Software/Mosaic/Docs/fill-out-forms/>. Use o compilador cc ou o gnu gcc.

ANEXO C

Arquivo de Configuração de um servidor Web

httpd.conf

```

#
#       Sample configuration file for cern_httpd for running it
#       as a normal HTTP server.
#
# See:
#       <http://info.cern.ch/hypertext/WWW/Daemon/User/Config/Overview.html>
#
# for more information.
#
# Written by:
#       Ari Luotonen April 1994 <luotonen@dxcern.cern.ch>
#
#
#       Set this to point to the directory where you unpacked this
#       distribution, or wherever you want httpd to have its "home"
#
ServerRoot      /where/ever/server_root

#
#       The default port for HTTP is 80; if you are not root you have
#       to use a port above 1024; good defaults are 8000, 8001, 8080
#
Port            80

#
#       General setup; on some systems, like HP, nobody is defined so
#       that setuid() fails; in those cases use a different user id.
#
Userid          nobody
Groupid         nogroup

#
#       Logging; if you want logging uncomment these lines and specify
#       locations for your access and error logs
#
# AccessLog      /where/ever/httpd-log
# ErrorLog       /where/ever/httpd-errors
LogFormat       Common
LogTime         LocalTime

#
#       User-supported directories under ~/public_html
#
UserDir         public_html

#
#       Scripts; URLs starting with /cgi-bin/ will be understood as
#       script calls in the directory /your/script/directory
#
Exec           /cgi-bin/* /Public/Web/cgi-bin/*

#
#       URL translation rules; if your documents are under /local/Web
#       then this single rule does the job:
#
Pass           /*      /Public/Web/*

```

REFERÊNCIAS

- [CGI] - "*Common Gateway Interface*" - CERN
<http://www.w3.org/hypertext/WWW/CGI/Overview.html>
- [Lindley - 91] "*Practical Image Processing In C*" - C.A.Lindley - J.Wiley & Sons, Inc - 1991.
- [HTML] - "*Como criar páginas HTML*" - UFRJ/NCE <http://www.nce.ufrj.br/~cracky/tutorial/>
- [HTML] - "*HyperText Markup Language (HTML)*" - CERN
<http://www.w3.org/hypertext/WWW/MarkUp/MarkUp.html>
- [HTTP] - "*Basic HTTP*" - WWW - CERN Organisation.
<http://www.w3.org/hypertext/WWW/Protocols/HTTP/HTTP2.html>
- [HTTP] - "*Hypertext Transfer Protocol*" - CERN
<http://www.w3.org/hypertext/WWW/Protocols/Overview.html>
- [HTTPD] - "*CERN httpd 3.0 Guide*" - CERN -
<http://www.w3.org/hypertext/WWW/Daemon/User/Guide.html>
- [HTTPD] - "*CERN httpd Administration*" - CERN -
<http://www.w3.org/hypertext/WWW/Daemon/User/Admin.html>
- [HTTPD] - "*Installing W3C httpd*" - CERN -
<http://www.w3.org/hypertext/WWW/Daemon/User/Installation/Installation.html>
- [Sybase - 93] - "*Descobrimos a filosofia Cliente Servidor*" - Relatório Técnico 1993
 Tecnologia Cliente/Servidor Ltda. São Paulo / SP.

