

CHAO LUNG WEN



**AMBIENTE
COMPUTACIONAL
DE APOIO À
PRÁTICA CLÍNICA**

Tese apresentada à Faculdade de Medicina
da Universidade de São Paulo para
obtenção de título de Doutor em Medicina

Área de concentração: Patologia

Orientador: Prof. Dr. Paulo Sérgio Panse Silveira

**SÃO PAULO
2000**

À minha esposa Rosangela Suetugo Chao e às minhas filhas, Maira Lie Chao e Mariana Mie Chao, a essência da magia que é a razão da minha energia, persistência e luta.

Aos meus pais Sr. Chao Chih Chun e Sra. Chao Cheng Tao, pessoas que sempre foram exemplos de coragem, amor, determinação, retidão, e perseverança.

Aos meus irmãos Chao Lung Chih e Chao Su Yi, pessoas que representaram, para mim, a união nos momentos importantes.

Aos meus sogros Sr. Kazuyuki Suetugo e Sra. Sumiko Suetugo, pessoas importantes no conjunto que cerca minha vida.

Ao meu cunhado, Paulo Suetugo, pessoa que participou comigo em vários momentos importantes que superamos com determinação.

Ao Prof. Dr. György M. Böhm, mais que um professor, um amigo com quem interagi tantos anos e com quem participei de lutas que me trouxeram cada vez mais experiência e amadurecimento e, sem dúvida, um professor no sentido profundo da palavra.

Ao Prof. Dr. Paulo Sérgio Panse Silveira, um amigo de década e meia e meu orientador, um homem de valor, e sua esposa Miriam Lemos, pessoas com grande senso de amizade. A história já mostrou que muitos homens de valor nem sempre foram reconhecidos na sua época.

Meus sinceros agradecimentos

Ao Prof. Dr. Raymundo Soares de Azevedo Neto e à sua esposa Leila Strazza de Azevedo, exemplos de pessoas com uma forma toda especial de ser e incentivar, mesmo sem estarem presentes.

Ao Prof. Dr. Jayme de Oliveira Filho, um amigo que sempre compartilhou entusiasticamente de todos os projetos, e às suas sugestões ao Tot-Win.

À Marcela Rocha de Oliveira, uma pessoa convicta, delicada e sensível, que sempre compartilhou entusiasticamente de várias idéias, e como fruto disto, nasceram vários trabalhos importantes.

À minha equipe da Intec System, Eglén Mari Akinaga, Marcelo Minoru Onoda e Cristiane Rosolem, pessoas sem as quais uma grande parte dos meus trabalhos não seria possível.

Ao Prof. Dr. Sebastião Almeida Prado Sampaio, uma pessoa de visão, entusiasta e o motivador do meu primeiro desenvolvimento em informatização de consultórios.

Ao Prof. Dr. Eduardo Massad, que sempre propiciou um ambiente de trabalho favorável, facilitador de minha jornada.

À minha equipe da Telemedicina, Adalberto Paulo Filho, Carlos Martins, Ethel Shuñá Queiroz, Luciano Gonçalves Correia, Marcos Akito Tikami, Sandra Rodrigues Gomes e Sílvia Regina Caporrino, um grupo que é exemplo de trabalho com alegria e entusiasmo.

À Galderma Brasil, que em 96 e 97, acreditou e participou comigo e com Prof. Dr. Jayme de Oliveira Filho, em vários projetos de informática na Dermatologia.

Existem várias pessoas a quem eu gostaria de dedicar especial atenção pela importância que representaram:

Dr. Antonio Fernandes Ventura

Dr. Antonio Rahme Amaro

Dr. Cássio Martins Villaça Neto

Celso Luiz da Silva

Prof. Dr. Gregório Santiago Montes

Prof. Dr. Flair Carrilho

Dr. Guilherme Del Fiol

Dr. Henrique Shiguekiyo Kikuta

Prof. Dr. Paulo Campos Carneiro
Dr. Ricardo Silveira de Paula
Dra. Salete Regina Pessanha de Paula
Prof. Dr. Salvador José de Toledo Arruda Amato
Yann Marche
Dr. Wu Tu Hsing

A todos os usuários do Tot-Win, à Disciplina de Informática Médica e os outros que participaram de várias formas da continuidade do aprimoramento do sistema ou que contribuíram para a realização deste trabalho.

Meus agradecimentos especiais a esta Faculdade e seus Professores, responsáveis pela minha formação.

Pensamentos ...

Um rio não teme obstáculos,
Só o engrandece...



No momento certo,
Ele os supera...

A força de um rio vem de várias fontes,
Enquanto caminha,
Novas energias juntam-se a ele,
Tornando-o cada vez mais forte.

Ao superar um obstáculo,
O rio fornece vida e sustento,
A todos que estão ao redor dele...

Cabe a todos os seres humanos reconhecerem a
sua própria essência...



Índice Analítico

INTRODUÇÃO.....	4
Breve histórico dos computadores.....	4
Evolução do mundo computacional	12
<i>Internet e World Wide Web</i>	13
Aspectos Gerais.....	13
Um breve histórico	14
Infraestrutura de telecomunicação no Brasil	15
Aspectos sobre Engenharia de <i>Software</i>	16
Aspectos Gerais.....	16
Evolução dos <i>Softwares</i>	17
Aspectos da <i>Interface</i> Homem-Computador	20
Aspectos sobre comportamento humano no uso de um sistema computacional	21
Modelos de Projeto de <i>Interface</i>	21
Prototipagem.....	22
Formas de apresentação de um <i>software</i>	23
Problemas no desenvolvimento de <i>software</i>	23
Algumas considerações em relação aos <i>softwares</i>	24
Revisões de <i>Software</i>	26
Impacto do custo de defeitos de <i>Software</i>	27
Confiabilidade de <i>Software</i>	26
Manutenção de <i>Software</i>	27
Efeitos decorrentes da manutenção	27
Domínio da Informação	28
A <i>interface</i> e o usuário	28
Informatização de consultórios e clínicas no Brasil	28
Evolução dos usos da informática na área médica	32
O <i>Tot-Clin</i>	34
O <i>Tot-Win</i>	35
OBJETIVOS.....	38
MATERIAIS E MÉTODOS.....	39
Equipamentos.....	39
Desenvolvimento.....	39
Estrutura para testes técnicos, composta por	39
Linguagem de desenvolvimento e <i>softwares</i> de apoio	40
Escolha da Linguagem de desenvolvimento e ambiente operacional	40
Recursos Humanos.....	45
Método.....	45
Fases para administração e desenvolvimento de um <i>software</i>	45
Fases de desenvolvimento do <i>Tot-Win</i>	46
Aspectos de Iconografia	50
Imagens digitais - formato de arquivo e modelagem dos bancos de dados.....	51
Modelagem de dados – Unidade Celular de Informação.....	52
Estruturação dos módulos do sistema	53
Bancos de dados de apoio e padronização de termos médicos	54
Desenvolvimento de aspectos <i>Web</i>	54
RESULTADOS.....	55
Desenvolvimento de um programa para automação de clínicas e consultórios	56
Desenvolvimento de recursos especiais e funções específicas do programa	56
Aspectos de Iconografia	58
Exibição de Informações	58
Recursos para finalidades assistenciais	60
Ficha clínica.....	60
Banco de imagens	65
Banco de seqüências de vídeos.....	65
Banco de orientação médica.....	65
Banco de medicamentos	65

CID (Código Internacional de Doenças).....	65
SNOMed (Systematized NOMenclature of Medicine).....	66
Doenças e síndromes	66
Crítérios TNM de estadiamento tumoral.....	66
Interação medicamentosa	66
Memento.....	67
Emissão de laudos.....	67
Recursos para finalidades administrativas e financeiras	67
Recursos para finalidades científicas e didáticas	69
Aspectos de segurança de acesso	72
Elaboração de banco de dados de apoio (banco de imagens, orientações médicas, etc.)....	73
Desenvolvimento de uma área de apoio ao usuário do <i>Tot-Win</i> através da <i>Web</i>	73
DISCUSSÃO.....	75
Alguns fatores relacionados com o insucesso da informatização	75
Informatização eficiente.....	78
Fatores técnicos.....	79
Fatores humanos envolvidos.....	80
Fatores organizacionais envolvidos	81
Desvantagens dos sistemas computadorizados	81
Qualidade da infra-estrutura física, rede local e estabelecimento de processos de contingência.....	82
<i>Hardware</i>	82
Infraestrutura ambiental.....	83
Rede local.....	84
Processo de contingência em caso de falha do servidor de rede	84
Contingência de cópia de segurança de dados.....	85
Comportamento evolutivo de um <i>software</i>	86
Aspectos relacionados com qualidade de um <i>software</i>	88
Desenvolvimento do <i>Tot-Clin</i> e <i>Tot-Win</i>	89
<i>Internet</i> como instrumento de apoio ao paciente e para pós-consulta	94
Modelagem de dados usando conceitos de <i>Data Warehouse</i>	95
Desenvolvimento baseado em objetivo.....	100
Iconografia e <i>Interface</i> HCI.....	102
Aspectos de estruturação de telas.....	103
Aspectos sobre a prescrição computadorizada de medicamentos	105
Módulos para prescrição de medicamentos no <i>Tot-Clin</i> e no <i>Tot-Win</i>	108
Aspectos importantes sobre uso de padrões de codificação	109
Capacidade de intercâmbio de dados.....	110
Confiabilidade de sistema computacional	112
Estratégia de cópia de segurança de dados	112
Validação jurídica dos dados eletrônicos	116
Manutenção de <i>Software</i>	119
Aspectos sobre obsolescência de <i>software</i>	122
Atualização do <i>Tot-Clin</i> para processamento do ano com 4 dígitos	122
Importância do envolvimento de um profissional de informática médica.....	123
CONCLUSÃO	126
GLOSSÁRIO.....	130
REFERÊNCIAS BIBLIOGRÁFICAS	149
ÍNDICE REMISSIVO	154
ANEXO.....	159

Introdução

Breve histórico dos computadores

É inegável que a sociedade moderna se rendeu à sedução de uma série de pequenos equipamentos, como microcomputadores em diferentes formas e tamanhos (de mesa, *notebooks*, *palm tops*, *hand helds*, além de projetos para versões de computadores que poderão ser vestidos), relógios com inúmeros recursos, aparelhos portáteis de som e televisão, telefones fixos e celulares, etc., que, a partir da segunda metade do século XX, estão mudando drasticamente as formas de trabalho e armazenamento de informações, sem nenhum precedente histórico. Nem sempre os aparelhos foram tão pequenos: no início do desenvolvimento da computação eletrônica (50 anos atrás), os equipamentos eram enormes e consumiam grandes quantidades de válvulas, e que, analogamente aos formidáveis tiranossauros que desapareceram por causa da seleção natural, foram selecionados e otimizados, tornando-se tão pequenos que funcionam a pilha e cabem no bolso.

O ser humano já sonhava com máquinas que o pudessem auxiliar a executar cálculos há séculos. Exceto por alguns instrumentos como o ábaco e a régua de cálculo, uma das primeiras calculadoras registradas pela história foi construída em 1642 por Blaise Pascal, e por isto foi batizada de Pascalina. Este invento era capaz de realizar as operações matemáticas de adição e subtração, mas o fato mais importante era de que tinha capacidade de memorização, isto é, armazenamento dos resultados (figura 1).

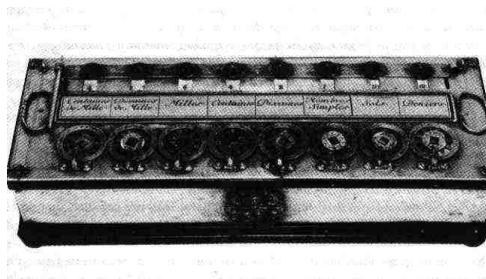


imagem obtida de The MacTutor History of Mathematics Archive^[75].

Figura 1: A Pascalina, calculadora projetada por Blaise Pascal em 1642 com capacidade de soma, subtração, e armazenamento dos resultados^[75].

Gottfried Leibnitz, em 1671, aperfeiçoou a Pascalina, que passou a fazer as quatro operações matemáticas básicas e era capaz de extrair a raiz quadrada. Influenciado por velhas fontes chinesas, este filósofo e matemático alemão encontrou e iniciou o desenvolvimento formal da aritmética binária, que, posteriormente, tornou-se a base para os computadores eletromecânicos do século XX (figura 2) ^[6].

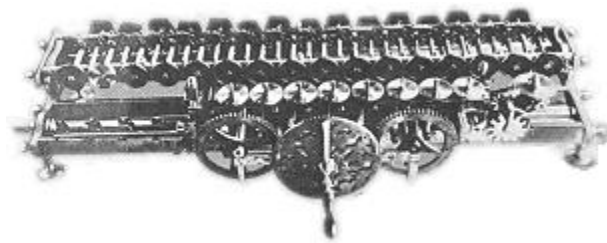


imagem obtida de Ricky's Home Page ^[64].

Figura 2: A calculadora universal, com capacidade de soma, subtração, multiplicação, divisão e raiz quadrada foi um aperfeiçoamento da máquina de Pascal feito pelo matemático alemão Gottfried Wilhelm von Leibnitz, em 1672.

Em 1801, Joseph Marie Jacquard, retomou a idéia dos cartões perfurados, porém utilizando agora folhas de cartolina dotadas de furos, que comandavam o primeiro tear totalmente automático (figura 3).

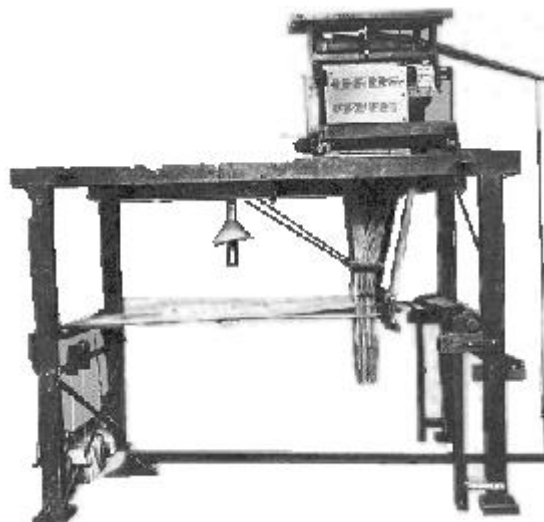


imagem obtida de Ricky's Home Page ^[64].

Figura 3: Tear de Joseph Marie Jackquard.

Charles Babbage, baseado no modelo do tear mecânico de Jacquard, projetou a Máquina de Diferenças em 1822 e a Máquina Analítica em 1833, que podem ser consideradas como os primeiros computadores do mundo,

capazes de memorizar padrões através do uso de cartões perfurados (figuras 4 e 5), isto é, teriam sido, mas a construção de ambas nunca foi possível com a tecnologia disponível na época.

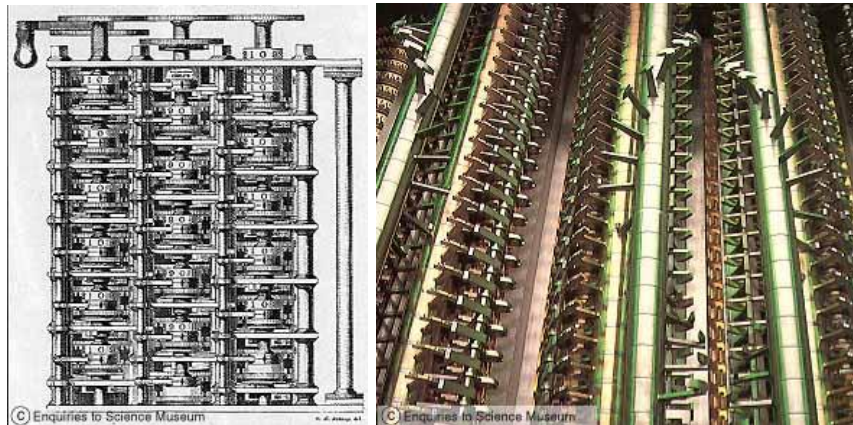


imagem obtida de The Science Museum London ^[76]

Figura 4: Aspectos da Máquina de Diferenças. À esquerda: parte construída em 1832. À direita: aspecto do mecanismo da Máquina de Diferenças nº 2, montada em 1991 para o *Science Museum* de Londres.

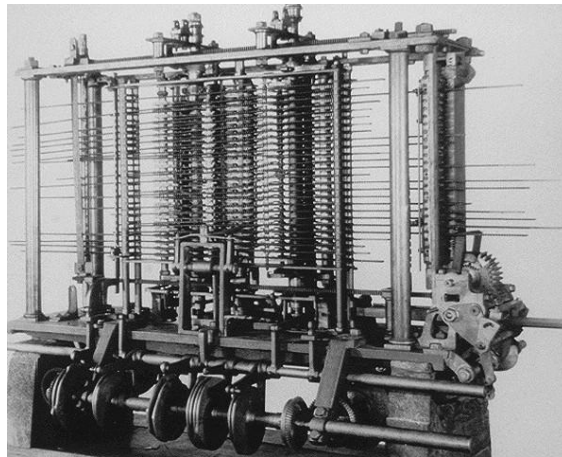


imagem obtida de NSF SUCCEED Engineering Visual Database ^[58]

Figura 5: Máquina Analítica projetada por Charles Babbage.

A assistente de Babbage, Augusta Ada King, Countess of Lovelace (filha do poeta Lord Byron), uma brilhante matemática, criou uma nova forma de interação com as invenções de Babbage usando cartões perfurados. Ela é freqüentemente lembrada como a primeira programadora de computador em decorrência desta descoberta.

Em 1890, Hermann Hollerith, idealizou uma máquina também baseada em cartões perfurados, porém a sua máquina era fundamentalmente um

tabulador, um contador de informações com capacidade para executar algumas operações aritméticas simples (figura 6). As máquinas de cartão perfurado de Hollerith permitiram que o censo americano de 1890 fosse processado em 4 semanas, o que antes demorava 7 anos. Mais tarde, Hollerith fundou sua própria empresa que, posteriormente transformou-se na gigantesca *IBM*.



imagem obtida de Ricky's Home Page^[64].

Figura 6: Máquina de Hermann Hollerith (1890).

Em 1906 foi inventada a válvula por Lee de Forest. O aparecimento do computador, como concebemos atualmente, ocorreu com o surgimento da eletrônica.

Em 1941, Conrad Zuse construiu a primeira calculadora elétrica do mundo, controlada por programa. De forma independente, três anos mais tarde, nos Estados Unidos, Howard Aiken desenvolveu o *Mark I*, também apelidado de o “Monstro de Cambridge”.

As máquinas destinadas à guerra vieram logo em seguida. O *Colossus*, como parte de um projeto de Inglaterra, continha 1.500 válvulas. O *ENIAC*, concluído em 1946 pela Universidade de Pensilvânia, foi encomendado pelo exército em 1943, como fruto do projeto *Manhattan*, em plena II Guerra Mundial, destinado a fazer cálculos balísticos. Tinha 18.000 válvulas, 1.500 relés, e ocupava uma sala de 9 por 30 metros (figura 7). Podia fazer 500 multiplicações por segundo, porém só armazenava vinte números de 10 dígitos. Em ambos, as válvulas tinham que ser constantemente substituídas porque queimavam após poucos minutos de trabalho.

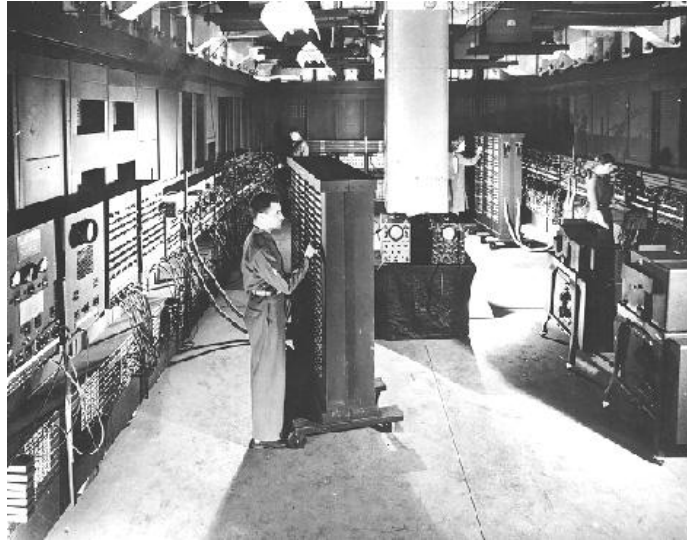


Figura 7: Vista geral do *ENIAC* em 1946. A máquina ocupava uma sala de 9 x 30 metros. Os controles aparecem à esquerda e uma pequena parte do dispositivo de saída está à direita ^[32].

Estes modelos levaram ao aparecimento das primeiras máquinas fabricadas pela *IBM*, *UNIVAC* e outras grandes empresas do ramo da época.

O transistor, inventado em 1948 por William Shockley, veio a substituir com maior eficiência as válvulas e foi incorporado aos computadores de segunda geração a partir de 1953.

Em 1962 foram usados pela primeira vez discos magnéticos para armazenamento de informações, no computador *Atlas*, em substituição às imensas unidades de fita magnética. Os discos utilizados nesta época ainda eram fisicamente muito diferentes do que conhecemos atualmente.

A Corrida Espacial da década de 60 pode ser considerada como um marco histórico no desenvolvimento computacional. O governo Americano investiu bilhões de dólares em pesquisas e a criação de um pólo de pesquisas avançadas, a *ARPA (Advanced Research Projects Agency)*, que promoveu o surgimento de vários conceitos o quais foram empregados e que refletem direta e indiretamente muito do que há no mundo da informática hoje.

Em 1963 começou a produção dos primeiros circuitos integrados (*chips*), que permitiram a junção de vários transistores em uma só pastilha de silício, o que foi um dos grandes passos, senão o maior na evolução da informática

até hoje. Neste momento a eletrônica deu um salto quantitativo e qualitativo e os computadores construídos com estes *chips* formaram a Terceira Geração. Com isto os grandes computadores da década de 50 puderam ser reduzidos a um tamanho tão pequeno que ocupariam apenas uma fração do espaço ocupado por um dos *chips* atuais. Isto, de fato, chegou a ser feito por um grupo de estudantes do Departamento de Engenharia Elétrica da Universidade de Pensilvânia, em 1996, comemorando os 50 anos da ativação do *ENIAC*. Recriaram a sua arquitetura básica e circuitos o mais fielmente possível, mas utilizando a tecnologia CMOS de hoje: o resultado do projeto “ENIAC-on-a-Chip” foi acomodar o computador em um *chip* de 7,44 mm x 5,29 mm x 0,5 μm (figura 8).

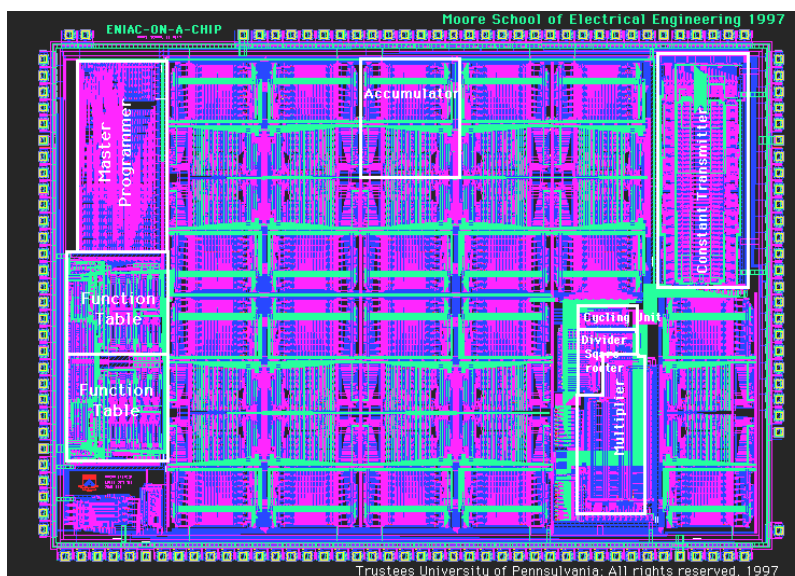


Figura 8: Imagem do *chip* construído por um grupo de estudantes do Departamento de Engenharia Elétrica da Universidade da Pensilvânia, em um projeto denominado de "ENIAC-on-a-Chip", sob supervisão de Spiegel e Ketterer, em comemoração aos 50 anos de ativação do *ENIAC*. Utilizou a tecnologia CMOS e acomodou todo o *ENIAC* em um *chip* de 7,44 mm x 5,29 mm x 0,5 μm ^[80].

Em 1968, Gordon Moore e Robert Noyce fundaram a *Intel*.

Outros centros de pesquisas que surgiram na década de 60 também tiveram grande papel, como o MIT (*Massachusetts Institute of Technology*) e o SRI (*Stanford Research Institute*) que trabalhavam em pesquisas sobre a interação homem-máquina e cujos conceitos foram fundamentais para a evolução da micro-informática tal qual conhecemos hoje, com propostas de

novos dispositivos como o *mouse* e interligação de computadores entre si para a troca de informações - a rede *Ethernet*.

O PARC (*Palo Alto Research Center*), criado pela *Xerox* em 1970, também teve papel importante. Neste centro foi desenvolvida toda a base da comunicação visual através de uma *interface* gráfica, usando componentes em forma de *ícones*, janelas e a estrutura WYSIWYG (*What You See Is What You Get - O que você vê é o que você obtém*), que mostra na tela seu trabalho exatamente da maneira como sairá impresso. A *Xerox*, apesar de possuir o domínio da tecnologia desde a década de 1970, não se interessou pelos computadores pessoais - os microcomputadores. Todas as grandes empresas estavam interessadas no mercado dos computadores de grande porte, para serem implantados em empresas. Não tinham a visão de pessoas usando computadores em suas casas.

O aparecimento dos microcomputadores ocorreu por volta de 1972, com a invenção dos microprocessadores. O primeiro foi lançado em 1971 pela *Intel*, o 4004, cujo *chip* tinha apenas 2.300 transistores e velocidade de processamento de 108 KHz. Em abril de 1972, a *Intel* lançou microprocessador 8008, o primeiro microprocessador de 8 *bits*. O surgimento dos microprocessadores foi decorrente do domínio da tecnologia LSI (*Large Scale Integration*) que permitia a junção de vários circuitos integrados em um só.

Por volta de 1976, surgem nos EUA os lançamentos do PET (*Personal Eletronic Transactor*), *Apple II* e do *TRS-80*, os primeiros microcomputadores pessoais. Em 1977, o microcomputador *Apple II* é um sucesso; o valor médio do equipamento era de US\$ 1.195,00 para uma configuração com 16 *quilobytes* de memória RAM, sem monitor. Na ocasião, em meio doméstico, era comum utilizar o aparelho de televisão como dispositivo de trabalho com o microcomputador. Em 1980, a *Apple II* já tinha 50% do mercado de computadores pessoais^[9].

Em 1978, a *Oracle* desenvolve o primeiro banco de dados relacional.

Em outubro de 1979, a *Software Arts* lançou o primeiro programa de planilha eletrônica, *Visicalc*, e imediatamente foi um sucesso. O número de cópias mensais comercializadas cresceu de 500 para 12.000 entre 1979 e 1981.

Em 1979 a *Apple* começou a desenvolver outro microcomputador, o *Lisa*, baseado nos conceitos que Steve Jobs tinha visto em sua visita ao PARC: a *interface* gráfica. Através de uma *interface* gráfica, o microcomputador tornava-se mais amigável. Funções antes disponíveis somente através de comandos complicados e de difícil memorização podia ser apresentado através de símbolos (ou *ícones*) na tela do computador.

Em 1981, a *IBM* lança o seu *PC* (Personal Computer) associado ao sistema operacional *MS-DOS*, que se transformou pouco tempo depois em sinônimo de “microcomputador profissional”. Atualmente, os sucessores deste modelo são os microcomputadores mais difundidos e utilizados mundialmente.

Em 1984, a *Apple* introduz o *Macintosh*, utilizando um microprocessador de 32 *bits* da *Motorola* (64000), com 8 MHz de *clock*, incorporando o conceito de *interface* gráfica para a comunicação com o usuário. Posteriormente a *Microsoft* adota a *interface* gráfica, lançando o *Windows*.

Diversos outros fabricantes de *software* criaram seus próprios ambientes gráficos, tais como o *DESQ (Quaterdeck)* - que foi re-desenvolvido e lançado depois como *DESQView* - o *VisiOn (VisiCorp)*, o *TopView (IBM)* e o *GEM (Digital Research)*. Destes, o que foi mais utilizado como suporte a diversos programas que utilizavam recursos gráficos, pelo suporte ao WYSIWYG, foi o *GEM*.

Em novembro de 1985, a *Microsoft* lança o *Windows 1.0* e em 1990, lança o *Windows 3.0*, que se torna a primeira *interface* gráfica para *PC* a ser amplamente utilizada como base para o desenvolvimento de outros sistemas.

Em 1992, a *Microsoft* lança o *Windows 3.1*, corrigindo os erros da versão anterior. As vendas ultrapassaram 1 milhão de cópias.

Em 1993, a *Intel* lança o microprocessador *Pentium*, com 3,3 milhões de transistores e começa a ser distribuído o *Mosaic*, o primeiro *browser* para a *Web*. Em 1994 é lançada a primeira versão do *Netscape Navigator*.

Em agosto de 1995 é lançado o *Windows 95*, e a *Netscape* lança a versão 2.0 do *Navigator* com recursos para troca de e-mails.

Em 1997, a *Intel* anuncia o lançamento do *Pentium II*, com 7,5 milhões de transistores.

Em julho de 1997, a *Microsoft* lança o *Windows 98*. Em 17 de fevereiro de 2000, a *Microsoft* lança o *Windows 2000*.

Existe uma grande quantidade de textos sobre a história dos computadores, disponíveis na própria *Web* ^[9, 17, 18, 26, 48, 49, 57, 65, 77, 78, 81].

Evolução do mundo computacional

Nestas duas últimas décadas a informática tem evoluído com velocidade sem paralelo na história. Recursos computacionais que há 10 anos atrás eram restritos a equipamentos de alto custo, hoje estão totalmente disponíveis para ambientes de microinformática de uso rotineiro e a custos acessíveis.

São exemplos deste progresso: discos de armazenamento de dados da ordem de dezenas de *gigabytes* e memórias RAM da ordem de centenas de *megabytes*, microprocessadores e correspondentes velocidades de processamento mais elevadas (figura 9 e tabela 1), monitores e placas de vídeo com maior definição e número de cores, entre outros. Os microcomputadores atuais têm capacidade de processamento comparáveis aos minicomputadores e *mainframes* de pequeno porte da década de 1980.

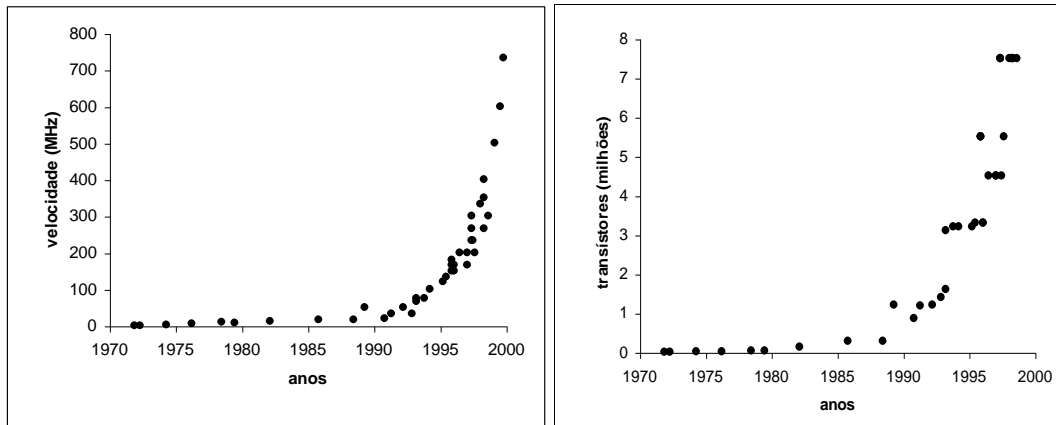


Figura 9: Gráficos demonstrativos da evolução dos processadores quanto à velocidade e número de transistores.

Tabela 1: Número de transistores e velocidade de alguns processadores produzidos pela *Intel*, exemplificando a evolução tecnológica que ocorreu entre 1971 e 1999.

Ano	Processador	Transistores	Velocidade (<i>clock</i>) (MHz)
Nov/71	4004	2.300	0,108
Abr/72	8008	3.500	0,2
Abr/74	8080	6.000	2
Jun/79	8088	29.000	4,77
Fev/82	80286	134.000	12
Out/85	386 DX	275.000	16
Abr/89	486 DX	1.200.000	50
Mar/93	<i>Pentium</i>	3.100.000	66
Nov/95	<i>Pentium Pro</i>	5.500.000	180
Mai/97	<i>Pentium II</i>	7.500.000	233
Fev/99	<i>Pentium III</i>	7.500.000	500

Dados obtidos da *Intel Corporation*^[36]

Em 1965, Gordon Moore, um dos fundadores da *Intel*, previu que a densidade de transistores dos microprocessadores seria duplicada a cada 2 anos, o que ficou conhecido como "Lei de Moore".

Internet e World Wide Web

Aspectos Gerais

A *Internet* é uma revolução e uma nova forma da humanidade de se relacionar com a informação. Ela rompe todas as barreiras físicas, e permite que uma pessoa em qualquer lugar da terra tenha acesso às informações que, praticamente, abrangem todas as áreas do conhecimento humano. O maior problema atual da *Internet* não é verificar se existe alguma área que

não esteja representada, mas é selecionar as informações de qualidade e que sejam confiáveis no mundo *Web*.

Um breve histórico

A *Internet* teve suas origens com a *ARPANET* (*Advanced Research Project Agency* – Departamento de Defesa Americana, 1969), na época de Guerra Fria, quando era constituída pela interligação de 4 computadores. Nesta ocasião já foram implementados alguns dos recursos atualmente utilizados de forma ampla, como o e-mail^[70]. Mas a *Internet* somente adquiriu expressão após a invenção da *World Wide Web* (WWW) que é uma forma de transmissão de informações com recursos de multimídia, através da estrutura física da *Internet*. A *Web* começou em março de 1989, com Tim Berners-Lee do *European Laboratory for Particle Physics* (mais conhecido como *CERN*) quando ele propôs um novo conjunto de protocolos para um sistema de distribuição de informações da *Internet* ^[66]. Neste momento surgiu o protocolo da WWW que foi rapidamente adotado por outras organizações, e foi constituído um consórcio de organizações, chamado de *3W Consortium* (liderado pelo *MIT*, *CERN* e pelo *INRA*), que uniu seus recursos para prosseguir com o desenvolvimento de padrões WWW. O *NCSA* (*National Center for Supercomputing Applications*) assumiu o projeto para o desenvolvimento de um aplicativo gráfico de fácil utilização que iria estimular o desenvolvimento comercial e o suporte à WWW, lançando em 1993 o primeiro *browser* (*Mosaic*) nas plataformas *Unix*, *Macintosh* e *Microsoft Windows*. Em 1994, o *Netscape* (1994) lançou o *Netscape Navigator*. A partir de então, com a liberação de rede para o mercado comercial, a *Internet* experimentou uma explosão, tornando-se um grande meio de informação disponível mundialmente.

Até 1990, a *Internet* comemorava o fato de que tinha 100.000 usuários, todos universitários e pesquisadores. Atualmente, somente no Brasil, estima-se que haja aproximadamente 5 milhões de usuários e as previsões são de que se alcance a marca de 8 milhões até o final de 2000.

Infraestrutura de telecomunicação no Brasil

O Brasil sempre teve problemas consideráveis tanto em relação à disponibilidade quanto em relação à qualidade das linhas telefônicas. Este cenário vem se modificando nestes últimos dois anos em decorrência de investimentos na modernização da infraestrutura de telecomunicação nacional, resultando no fornecimento de linhas digitais e linhas de banda larga, como as disponibilizadas pelas tecnologias ISDN e ADSL. As TVs a cabo também se tornaram uma alternativa para acesso rápido à *Internet*. Até o segundo semestre de 1999 só estava disponível o sistema unidirecional. A partir do primeiro semestre de 2000 vários bairros de São Paulo já estão começando a ter serviço bidirecional.

A tecnologia ADSL (*Asymmetric Digital Subscriber Line*), permite velocidade de transferência de dados até 140 vezes superiores aos oferecidos pelos *Modems* analógicos atuais, através de uma linha telefônica normal. Permite que se fique conectado permanentemente na *Internet* com um custo fixo mensal, sem pagamento de impulsos telefônicos. Esta particularidade facilita o uso no dia a dia, pois a *Internet* está sempre disponível, sem necessidade de fazer uma nova conexão toda vez que se deseja usá-la. Além disto, a linha de voz continua disponível para uso normal mesmo durante a navegação na *Web*, porque através da divisão da linha telefônica em duas faixas de frequências, reserva-se a faixa abaixo de 4 KHz, para a voz, e acima desta frequência, para tráfego de dados. A palavra “assimétrica” da sigla ADSL aparece porque existe uma reserva maior de largura de banda para recepção do que para envio de dados; as velocidades de recepção podem variar de 256 *quilobytes* até 8 *megabytes*. O custo de uma linha ADSL, incluindo taxas do provedor de acesso, gira em torno de 130 reais/mês.

Uma conexão do tipo ISDN baseia-se na tecnologia de uma linha telefônica especial, totalmente digital, denominada DVI (*Digital Voice and Image*). É composta por 2 canais independentes, pelos quais pode-se trafegar voz, dados e imagens. Permite conexão à *Internet* a partir de 64 Kpbs, podendo chegar a 128 Kpbs, caso as duas linhas estejam sendo

utilizadas simultaneamente. Esta tecnologia também permite acessar a *Internet* e falar ao telefone simultaneamente. Os custos envolvidos no uso desta tecnologia são o custo de impulso da ligação e assinatura com o provedor.

A TV a cabo é também uma alternativa para acesso rápido à *Internet*. Existem dois tipos no mercado:

- Unidirecional para *downstream*; neste modelo é necessário o uso de linha telefônica para o envio das requisições, sendo o *downstream* feito de forma mais rápida pelo sistema de TV a cabo.
- Bidirecional; nesta forma é dispensável o uso de telefone, pois tanto as requisições como os *downstreams* são feitas através do sistema de TV a cabo.

As tecnologias de ADSL e TV a cabo são soluções com boa relação custo/benefício para usuários que utilizam a *Web* com muita frequência e queiram estar permanentemente conectados, sem sacrificar uma linha telefônica. Atualmente, abril de 2000, esta tecnologia está implementada somente em algumas regiões, embora a perspectiva seja de crescimento muito rápido no decorrer de 2000 e 2001.

Aspectos sobre Engenharia de *Software*

A engenharia de *software* já é uma tecnologia altamente desenvolvida e existe uma ampla literatura disponível para leitura. Alguns pontos relevantes para o desenvolvimento de um *software* são sumarizados a seguir.

Aspectos Gerais

O principal desafio nas primeiras décadas da era do computador foi desenvolver equipamentos que reduzissem os custos de processamento e armazenamento de dados. No entanto, a microeletrônica avançou e, principalmente a partir da década de 1980, esta meta começou a ser alcançada. O poder de processamento dos computadores de grande porte (*mainframe*) da década de 1980 estão disponíveis hoje sobre uma escrivaninha. O melhor desempenho do *hardware*, diminuição do tamanho e

custos menores ofereceram ambiente para o aparecimento de computadores mais sofisticados. Atualmente, o problema é diferente: o principal desafio, desta década de 1990, foi melhorar a qualidade e reduzir os custos de soluções baseadas em computador, implementadas com o uso de *software*.

Evolução dos *Softwares*

O período compreendido entre 1950 e meados de 1960 pode ser chamado de a **primeira era dos sistemas computadorizados**. Caracterizou-se por mudanças freqüentes no *hardware*, sendo que o desenvolvimento do *software* era considerado por muitos, como uma atividade secundária. Os desenvolvimentos eram feitos sem uma administração mais profissional, como planejamento de prazos e custos. Os *softwares* eram projetados sob medida, específicos para cada aplicação, com distribuição relativamente limitada. Geralmente eram desenvolvidos para uso próprio (próprio programador ou própria organização) e, na maioria dos casos, os sistemas eram executados em ambiente *batch* (lote), com talvez raras exceções, como o desenvolvimento do primeiro sistema de reservas da *American Airlines* e os sistemas de defesa americana, que eram *on-line*. Devido à limitação, os *hardwares* eram dedicados à execução de apenas uma aplicação em cada momento. Não tinham recursos para executar multitarefa.

A **segunda era da evolução dos sistemas computadorizados** ocorre entre meados da década de 1960 e o final da década de 1970. A multiprogramação e os sistemas multiusuários introduziram novos conceitos relacionados com a interação homem-máquina. Os *softwares* caracteristicamente eram utilizados para gerenciar em tempo real, com coleta, análise e transformação de dados, de várias fontes, simultaneamente. A evolução do armazenamento *on-line* levou ao surgimento dos sistemas de gerenciamento de banco de dados. O *software* passou a ser visto como um produto, não mais considerado como um simples apêndice do *hardware*. Este cenário possibilitou o aparecimento das “*software houses*”. Os programas passaram a ser desenvolvidos visando a distribuição mais ampla, contando-se centenas a milhares de cópias.

A **terceira era da evolução dos sistemas computadorizados** inicia-se em meados da década de 1970 e continua até fins da década de 1980, e é caracterizado pelos sistemas distribuídos (múltiplos computadores, cada um executando funções concomitantemente e comunicando-se um com outro), e foram os responsáveis pelo aumento da complexidade dos sistemas computacionais. Esta era é caracterizada pelas redes globais e locais, comunicações digitais de banda larga, crescente aumento de demanda por acessos *on-line* de dados, uso generalizado de microprocessadores, computadores pessoais e estações de trabalho de mesa. O computador pessoal foi o fator que estimulou o aparecimento e crescimento de muitas empresas de *software*. O *hardware* se torna num produto comum, com características e capacidades muito semelhantes entre eles, enquanto que o *software* se torna fator que os diferencia um do outro. Enquanto a taxa de crescimento das vendas dos computadores pessoais começou a se estabilizar a partir de meados de 1980, as vendas de *software* continuaram a crescer. As pessoas gastaram mais dinheiro com *software* do que com *hardware* ^[63].

A **quarta era da evolução dos sistemas computadorizados** compreende o período de meados de 1980 até hoje. É caracterizada pelos poderosos sistemas de mesa, sistemas especialistas, rede neurais artificiais, processamento paralelo e tecnologias orientadas a objeto substituindo as abordagens convencionais de desenvolvimento.

Talvez há 20 anos atrás menos de 1% do público fosse capaz de definir o que era um “programa de computador”. Mas mesmo hoje, abril de 2000, talvez muitas pessoas, incluindo muitos profissionais, ainda não entendam realmente o que é um *software*, e que merece algumas considerações:

- Um *software* é um elemento lógico, e não físico, o que lhe confere características bem diferentes do *hardware*.
- Um *software* é projetado ou desenvolvido por processo de engenharia, mas não é manufaturado no sentido convencional.

- Os aspectos determinantes de qualidade durante a manufatura de um *hardware* é diferente dos aspectos da produção de um *software*. Problemas que não existem ou que seriam facilmente corrigidos na manufatura de um *software* podem causar conseqüências graves na produção de um *hardware*. Um exemplo de problema que não existe na produção de *software* e que pode afetar na produção de *hardware* é a sala limpa na produção dos discos rígidos. Um outro exemplo é uma peça de uma linha de montagem, cujo equivalente em um *software* são os vários módulos e rotinas que, ao apresentar algum problema, pode causar menos distúrbio para ser substituída do que no caso de um *hardware*.
- Os custos do *software* estão principalmente concentrados na fase da engenharia.
- *Software* não se “desgasta”. O *software* não é sensível aos problemas ambientais que desgastam os equipamentos, mas ele pode “deteriorar”. Isto será discutido mais adiante.
- Quando um componente de *hardware* se desgasta, ele pode ser substituído por outro de reposição. No *software* não existe peça de reposição. Toda falha indica erro de projeto ou de codificação. Assim, o processo de manutenção de um *software* é consideravelmente mais complexo.
- Diferentemente dos projetos de *hardware* que podem contar com uma grande quantidade de componentes digitais disponíveis para a construção do *hardware*, os projetos de *software* não dispõem destas facilidades, pois salvo raras exceções, não existe disponibilidade de componentes de *software* que possam ser utilizados para montagem de novos *softwares*. Embora exista literatura abordando sobre a reusabilidade de *software*^[79], implementações bem sucedidas deste conceito somente foram obtidas recentemente.

Aspectos da *Interface* Homem-Computador

Existem vários estudos que tentam compreender a forma de como o ser humano percebe o mundo através de seu sistema sensorial, mas a nossa intenção neste texto é concentrar nossa atenção apenas sobre como, qualitativamente, o ser humano interage com o computador. Através desta abordagem pode-se estabelecer os fundamentos para a elaboração das *interfaces* que possibilitem ao usuário de um sistema computacional receber as informações, guardá-las na memória (humana) e “processá-las”, usando os raciocínios indutivo e dedutivo ^[55]. Principalmente sentidos visual, tátil e auditivo são considerados na *interface* ser humano-computador.

Apesar da tendência atual de cada vez mais os projetos de *Human Computer Interface* (HCI) se basearem na comunicação gráfica, grande parte das informações ainda são apresentadas na forma de texto. A leitura, o processo de extrair informações de textos através da decodificação dos padrões visuais e recuperação do significado das palavras ou frases, é uma atividade inerente à maioria das *interfaces*. O tamanho do texto, tipo de fonte, extensão da linha de texto, letras maiúsculas, localização e cor, todos afetam a facilidade com que ocorre a extração de informações.

À medida que as informações são extraídas da *interface*, elas são armazenadas para posterior recuperação e uso. Além das informações, o usuário precisa lembrar dos comandos, seqüências de operação, alternativas de trabalho e outros recursos disponíveis do sistema, que nem sempre estão evidentes na *interface* (teclas de atalho, janelas e menus secundários, etc.). Todas essas informações são armazenadas na memória humana – um sistema extremamente complexo que, segundo se acredita atualmente, é composto de memória de curto prazo (*Short-Term Memory* – STM) e memória de longo prazo (*Long-Term Memory* – LTM) ^[42]. O *input* sensorial (visual, auditivo, tátil) é colocado num “*buffer*” e depois armazenado em STM, de onde pode imediatamente ser usado. O tamanho do “*buffer*” e a extensão de tempo durante o qual o uso pode ocorrer são limitados. O conhecimento é mantido em LTM e forma a base de nossa resposta aprendida quando uma HCI é usada. Tanto as informações semânticas como

as sintáticas são armazenadas em LTM. Caso a especificação de uma *interface* entre o ser humano e o computador não atenda adequadamente às características da STM e/ou LTM, o desempenho do elemento humano pode ter menor rendimento.

Aspectos sobre comportamento humano no uso de um sistema computacional

As pessoas são diferentes umas das outras, e isto pode interferir como elas utilizam sistemas baseados em computador, que extrapolam o simples aspecto dos elementos básicos da percepção humana; fatores como nível de habilidade individual, variações de personalidade e distinções comportamentais também interferem no processo. Uma mesma *interface*, usada por duas pessoas com a mesma educação e formação cultural, mas com personalidades completamente diferentes, poderia parecer adequada para uma pessoa e inadequada para outra.

Deve-se considerar que cada usuário de computador tem uma personalidade particular. Na maioria dos casos, a personalidade de um indivíduo está intimamente ligada ao seu estilo cognitivo. A HCI deve ser projetada para acomodar diferenças de personalidade ou, alternativamente, é projetada para acomodar uma personalidade “típica” de uma categoria de usuários.

Em maior ou menor grau, a proliferação de sistemas baseados em computador fez eclodir em algumas pessoas um fenômeno de medo irracional em relação a produtos e sistemas de alta tecnologia. À medida que o nosso conhecimento em relação ao elemento humano é ampliado e a HCI é projetada de modo a acomodar cada vez mais as necessidades humanas em relação aos sistemas baseados em computador, é possível que o nível de “terror tecnológico” seja reduzido significativamente^[43].

Modelos de Projeto de *Interface*

Todo projeto deve iniciar-se com um entendimento sobre os usuários finais aos quais o sistema se destina, incluindo características como a idade, sexo, capacidades físicas, educação, “bagagem” cultural ou técnica,

motivação, metas e personalidade. Os usuários podem ser classificados como:

- *Usuários principiantes*: sem conhecimento da mecânica do uso da *interface* do sistema e pouco conhecimento semântico (compreensão das funções que são executadas, do significado da entrada e saída, e dos objetivos dos sistema) da aplicação ou uso do computador em geral.
- *Usuários instruídos e intermitentes*: razoável conhecimento semântico da aplicação, mas relativamente pouca lembrança de informações da mecânica de uso da *interface*.
- *Usuários instruídos e freqüentes*: bom conhecimento semântico e da mecânica de uso da *interface*, indivíduos capazes de procurar atalhos e modos abreviados de interação.

Prototipagem

A prototipagem é uma etapa que permite ao engenheiro de *software* criar um modelo experimental do sistema a ser desenvolvido. O modelo pode ter essencialmente três formas:

- em papel ou usando um microcomputador para apresentar a forma que se prevê para a interação entre o ser humano e a máquina;
- *software parcial*, com módulos e funções selecionados entre os que constarão na versão final;
- um *software* com todos os módulos, que executa parte ou todas as funções desejadas, mas que têm características que ainda precisarão ser melhoradas e otimizadas através de uma nova fase de desenvolvimento.

Formas de apresentação de um *software*

Um programa de computador é construído usando-se uma linguagem de programação, que tem repertório finito de vocábulos, estruturação, regras de sintaxe e semântica bem definidas. Um programa de computador pode existir em duas formas:

- interpretada: que depende de um interpretador, para converter os comandos em instruções nativas do computador, à medida que são solicitadas pelo programa;
- executável: que depende de um compilador para converter previamente os comandos em instruções de linguagem de máquina, posteriormente gerando um programa independente, denominado de executável, através de um segundo programa (*linker*).

A desvantagem no uso de linguagens interpretadas é que exige que o usuário possua o programa interpretador, o que implica em custo adicional; maior lentidão na execução, pois o interpretador consome tempo em sua tarefa; e obriga o fornecimento do código fonte com todos os comandos utilizados no *software*.

A forma de apresentação do programa final é um dos aspectos considerados quando se define uma ferramenta de desenvolvimento, como será discutido adiante.

Problemas no desenvolvimento de *software*

O sucesso do desenvolvimento de um *software* depende do gerenciamento adequado de vários fatores e da superação dos problemas. Alguns dos problemas mais significativos são relacionados a seguir:

- Imprecisão na estimativa de prazos e de custos.
- Insuficiência de produtividade dos profissionais das áreas de *software* para acompanhar a demanda.
- Qualidade dos *softwares* desenvolvidos inferior à desejada (índice de falhas muito elevado causando falta de confiança do usuário no sistema).

Alguns problemas que podem ser menos evidentes:

- Tempo insuficiente para colher dados sobre o processo a ser automatizado pelo *software*. O tempo necessário é variável de acordo com o conhecimento e a experiência do entrevistador em relação ao assunto abordado, ou à disponibilidade de conhecimentos prévios para serem usados como referência na entrevista.
- A insatisfação dos usuários com o sistema “final” pode ocorrer muito freqüentemente. Os projetos de desenvolvimento de *software* muitas vezes não levam em consideração as necessidades e/ou características do usuário.
- Suspeitas em relação à capacidade do *software* executar corretamente as funções propostas. Só recentemente é que se começou a entender a importância dos testes sistemáticos de *software*, surgindo conceitos quantitativos concretos de confiabilidade e garantia de qualidade do *software*.
- A tarefa de manutenção do *software* consome grande parte dos recursos financeiros. A capacidade de manutenção do *software* por uma empresa deve ser vista como um critério importante para a sua aceitação.

Algumas considerações em relação aos *softwares*

- O desenvolvimento de *software* é um processo criativo e não um processo mecânico igual à manufatura. Brooks^[7] afirma que “...acrescentar pessoas em um projeto de *software* atrasado torna-o ainda mais atrasado”. Para alguns, esta afirmação pode parecer contra-intuitiva em um primeiro momento, mas é fácil perceber que quando novas pessoas são acrescentadas, as pessoas que estavam trabalhando precisam gastar tempo “educando” os recém-chegados, o que diminui a disponibilidade de tempo para serem dedicados ao desenvolvimento produtivo. Somente sob condições bem planejadas e bem coordenadas é que a inclusão de novas pessoas não causa problema no cronograma produtivo. Este planejamento deve levar em

consideração a necessidade de um período para integração dos novos elementos à equipe.

- Uma declaração geral dos objetivos de um *software* não é suficiente para se começar o desenvolvimento. Os detalhes nem sempre podem ser inseridos mais tarde. Uma definição inicial incompleta é uma das principais causas de fracasso no desenvolvimento de um *software*. Uma descrição formal e detalhada das informações a serem gerenciadas, das funções, do desempenho esperado, do HCI, das restrições do projeto, e dos critérios de validação são fundamentais. Um projeto mal definido inicialmente pode gerar problemas que impedirão que todas as exigências sejam atendidas, à semelhança de um prédio cuja fundação não atenda às necessidades estruturais e seja mal dimensionada para a carga final que deverá suportar.
- Especificações de um projeto não podem ser continuamente modificadas. Flexibilidade de *software* tem limites, e nem sempre todas as mudanças podem ser facilmente acomodadas. É fato que os requisitos de *software* são passíveis de modificações no decorrer do desenvolvimento de um projeto, mas o impacto da mudança varia de acordo com o período em que ela é introduzida (figura 10). Se uma atenção minuciosa for dada à definição inicial, os primeiros pedidos de mudança podem ser acomodados mais facilmente. O usuário pode rever as exigências e recomendar modificações sem causar impacto muito grande sobre os custos nas fases iniciais. Em fases posteriores o impacto será muito maior.

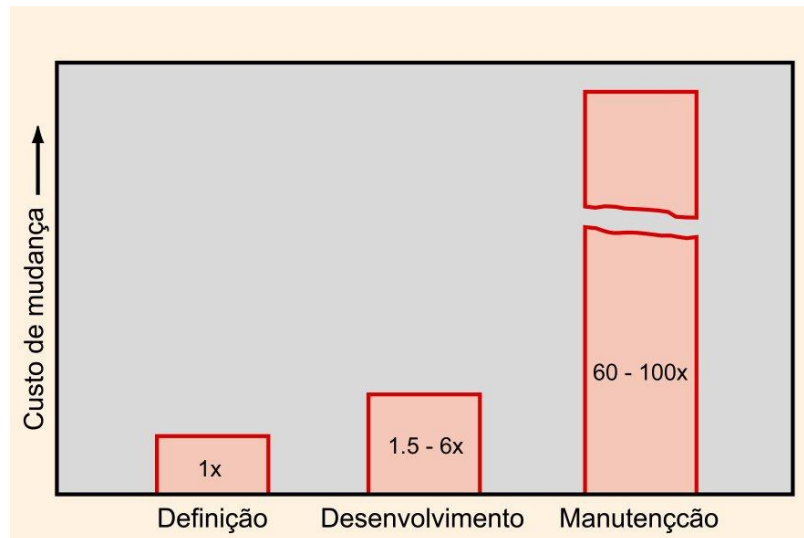


Figura 10: Impacto relativo de custos para a implementação de mudanças em diferentes fases do desenvolvimento de um *software* (modificado de PRESSMAN, 1995^[63]).

- O trabalho de desenvolvimento de um programa não se encerra quando ele é colocado em funcionamento. Lientz^[46], afirma que os dados da indústria indicam que entre 50 e 70% de todo o esforço gasto em um *software* serão despendidos depois que o mesmo é entregue ao cliente.
- É possível avaliar a qualidade de um programa mesmo que ele ainda não esteja em utilização efetiva. Um dos mecanismos importantes para a garantia da qualidade de um *software* é a sua revisão técnica formal^[30], que pode ser aplicada desde o início do projeto. São mais eficientes do que a realização exclusiva de testes para a identificação de defeitos de *software*^[63].

Revisões de *Software*

Num processo de engenharia de *software*, as revisões são um tipo de “filtro” e devem ser aplicadas em vários pontos da programação durante o desenvolvimento do *software* para descobrir defeitos passíveis de correção. Freedman e Weinberg^[31], discutem a necessidade de fazer revisões da seguinte maneira:

“O trabalho técnico precisa de revisão pelo mesmo motivo que os lápis precisam de borrachas: errar é humano. A razão pela qual precisamos de

revisões técnicas é que, apesar das pessoas captarem bem alguns de seus próprios erros, grandes quantidades de erros escapam mais facilmente de quem lhes deu origem, do que de outras pessoas."

Uma revisão é uma maneira de aproveitar melhor a diversidade de um grupo de pessoas para:

- apontar melhorias necessárias ao trabalho de uma única pessoa ou equipe;
- confirmar as partes de um produto em que uma melhoria não é desejada ou não é necessária;
- realizar um trabalho técnico com uma qualidade mais uniforme ou, pelo menos, mais previsível do que aquilo que pode ser realizado sem revisões, de forma a tornar esse trabalho técnico mais administrável.

Impacto do custo de defeitos de *Software*

O grande benefício das revisões técnicas formais é a identificação precoce dos defeitos de *software*, de forma que cada defeito seja corrigido antes de se passar para a fase seguinte do processo de engenharia de *software*. Pressman^[63] relata que uma série de estudos conduzidos pelas indústrias (TRW, Nippon Electric, Mitre Corp., entre outras) apontaram que as atividades de projeto são responsáveis pela inserção de 50% a 60% de todos os erros durante a fase de desenvolvimento. Jones^[40] demonstrou que as técnicas de revisão formal são efetivas em até 75% para a detecção de falhas de projeto. Ao detectar e suprimir uma grande quantidade desses erros, o processo de revisão reduz substancialmente o custo dos passos subsequentes nas fases de desenvolvimento e manutenção.

Confiabilidade de *Software*

Em termos estatísticos, a confiabilidade de *software* é definida como "a probabilidade de operação livre de falhas de um programa de computador num ambiente específico durante determinado tempo"^[56].

A confiabilidade de um *software* de computador é um elemento fundamental para sua qualidade global. Se um programa deixar de funcionar repetidamente, pouco importa se outros fatores de qualidade de *software* são aceitáveis.

Manutenção de *Software*

A manutenção de *software*, que ocorre após a liberação para o usuário, pode representar mais de 70% de todos os esforços gastos por uma organização que os desenvolve.

A manutenção é bem mais complexa do que o simples “consertar erros” e pode ser classificada em quatro atividades:

- Correção de erros.
- Adaptação.
- Aprimoramento.
- Melhoria de confiabilidade.

Estas atividades serão detalhadas posteriormente.

Efeitos decorrentes da manutenção

A modificação de *software* é um processo perigoso e delicado. Provavelmente já se ouviu várias vezes a seguinte expressão: “...mas mudei apenas este comando...”. Infelizmente, toda vez que é feita uma mudança numa complexa estrutura lógica, o potencial de erros aumenta. A consulta à documentação dos projetos e cuidadosos testes ajudam a detectar e eliminar erros, que podem acarretar as seguintes conseqüências:

Conseqüências na codificação

Uma mudança, por mais simples que seja, mesmo que seja num único comando, pode causar importantes alterações no funcionamento do programa comprometendo a sua confiabilidade e eficiência.

Conseqüências nos dados

Podem ocorrer como resultado de modificações feitas na estrutura da tabela de dados do *software* ou na forma como os dados são processados. As rotinas que trabalhavam com os dados, mas que não foram adequadamente preparados para lidar com as novas alterações, podem produzir erros diretos nos dados ou erros de análise.

Conseqüências na documentação

A manutenção deve englobar todos os aspectos do *software* e não somente as modificações no código-fonte. Quando as mudanças não são refletidas na documentação do projeto ou nos manuais destinados ao usuário, o comportamento, a avaliação das conseqüências das alterações, o planejamento e a administração do *software* podem ser comprometidos.

Domínio da Informação

Todos os *softwares* são construídos para processar dados e transformá-los de uma forma em outra, ou seja, aceitar uma entrada, manipulá-la e produzir uma saída. Para que se faça corretamente este processo, é fundamental ter o conhecimento das informações que serão inseridas e o domínio de como devem ser processadas para gerar a saída. O processamento correto é que determina a confiabilidade do sistema.

A interface e o usuário

Os sistemas operacionais e os programas aplicativos tiveram evolução significativa com a popularização dos ambientes gráficos, que, além de introduzirem o conceito de *ícone*, incorporaram capacidade de multimídia. Os ambientes gráficos estimularam o desenvolvimento da área que estuda HCI, que visa a analisar e a elaborar com maior nível de precisão científica os projetos relacionados com a melhoria da comunicação entre o ser humano e o computador, como por exemplo, a padronização de conjuntos de *ícones* para serem utilizadas em programas diferentes, mas definindo as mesmas funções. Isto permitiu maior velocidade de aprendizado dos usuários para os *softwares* de um mesmo sistema operacional. Foi proporcionado pelo *Windows* no ambiente dos microcomputadores *IBM PC* e pelo *Apple Macintosh* e seu sistema operacional.

Informatização de consultórios e clínicas no Brasil

A informatização dos consultórios e clínicas teve impulso no Brasil, por volta de 1985, após a consolidação do microcomputador *IBM PC* como

modelo padrão para uso profissional, aliada a disponibilização cada vez maior de *softwares* e linguagens de desenvolvimento para estes equipamentos.

Neste período, a *interface* de comunicação entre os sistemas operacionais e o ser humano era, fundamentalmente, baseada em caracteres (*MS-DOS*, o mais predominante), apesar do microcomputador *Apple Macintosh* ter sido lançado em 1984, com a *interface* gráfica como a principal forma de comunicação com o usuário, e ter incorporado recursos de multimídia no sistema operacional. Os ambientes gráficos somente se tornaram o principal meio de *interface* com o usuário com a consolidação do *Microsoft Windows* como ambiente gráfico padrão para os microcomputadores *IBM PC* compatíveis.

O crescente interesse pela informatização de consultórios e clínicas começou quando os microcomputadores ficaram mais baratos e suas capacidades de armazenamento e processamento alcançaram uma boa relação custo/benefício para uso diário. Em torno de 1988, a configuração básica de um conjunto computacional, no Brasil, era representada por um microcomputador compatível com *IBM PC XT*, com disco rígido de 20 ou 30 *megabytes*, 640 *quilobytes* de memória RAM, placa gráfica CGA, monitor monocromático, impressoras matriciais de 9 agulhas e 80 colunas, com custo em torno de US\$ 3.000,00.

O desenvolvimento de programas baseados em banco de dados para microcomputadores teve grande impulso com o lançamento da linguagem *dBase II*, por volta de 1982, pela *Ashton Date*, que simplificou a elaboração de programas para lidar de forma ágil com o armazenamento de dados. Esta linguagem e seus sucessores (*dBase III* e *dBase III Plus*) popularizaram-se em pouco tempo e tornaram-se referência.

Por serem linguagens interpretadas, criavam alguns inconvenientes como a necessidade do usuário adquirir versões oficiais do *dBase*, dificuldade de esconder os códigos fontes e menor velocidade de processamento. Com isto, várias empresas começaram a desenvolver alternativas que permitissem aos programadores elaborarem e distribuírem *softwares* em formas executáveis

(programas já traduzidos para instruções nativas do computador), mas que utilizavam sintaxe de programação semelhante ao *dBase* e mantinham o banco de dados em padrão DBF.

Esta alternativa foi amplamente adotada pelos programadores. Em 1985 a empresa *Nantucket* lançou o *Clipper'85 Winter* que teve grande sucesso; depois lançou novas versões como *Clipper'86 Autumn*, *Clipper'87 Summer* (dezembro de 1987), *Clipper 5.0*, *Clipper 5.01*, *Clipper 5.2* e *Clipper 5.3*.

As linguagens de desenvolvimento de banco de dados, que funcionavam em ambiente *Windows*, sucederam as que funcionavam em *MS-DOS* e foram lançadas com aprimoramento significativo dos recursos para a construção da *interface* com o usuário e melhoramento na integridade de dados. Esta nova geração de linguagens viabilizou a elaboração de aplicativos que passaram a incorporar recursos de imagens e multimídia.

O sucesso do padrão de armazenamento do *dBase* foi tão grande que mesmo hoje muitos programas comerciais lêem dados armazenados neste formato, bem como ainda são usadas linguagens de programação que herdaram sua sintaxe. Estas linguagens incorporaram o conjunto de comandos do *dBase*, implementaram novos recursos e melhoraram o gerenciamento do banco de dados, como é o caso do *Microsoft Visual FoxPro*, podendo utilizar rotinas antigas, desenvolvidas em *dBase*, muitas vezes exigindo apenas algumas adaptações simples. Quanto ao formato de banco de dados do *dBase* original, a evolução foi bastante conservadora e não houve perda importante de compatibilidade, o que facilita a migração dos dados antigos.

A aparente facilidade para desenvolver sistemas utilizando um banco de dados, parece ter criado, no período de 1985, uma euforia que gerou distorções no desenvolvimento de sistemas médicos. Neste período não era comum dedicar muito tempo para estudar e padronizar como as informações deveriam ser armazenadas nos programas médicos para clínicas

e consultórios, analisar o uso de termos médicos padronizados, possibilidade de intercâmbio de dados, modularidade e continuidade de desenvolvimento. Ouvia-se com certa frequência que “o filho do meu amigo é muito bom em computador e vai informatizar tudo”, em reuniões técnicas com potenciais usuários. Este erro levou vários médicos a amargarem prejuízos com equipamentos mal aproveitados, gastos com treinamento para uso de sistemas que foram abandonados pouco tempo depois, e perdas indiretas como provocar diminuição de confiança das pessoas envolvidas nos processos de informatização, em relação aos sistemas, e conseqüente resistência às implementações futuras.

A preocupação com treinamento de recursos humanos e manutenção do *software*, incluindo suporte e aprimoramento, também não eram considerados como critérios importantes para decidir sobre a aquisição de um programa computacional.

Alguns, na ânsia de informatizar, decidiram fazer desenvolvimentos independentes com contratação de analistas de sistemas para desenvolvimento específico e particularizado, imaginando que a comunicação e os entendimentos com estes profissionais seriam muito fáceis. O resultado após alguns anos de trabalho, em geral, foi frustrante com a constatação de elevados custos diretos de desenvolvimento, programas inacabados e que muitas vezes não funcionavam, além dos custos indiretos como o tempo pessoal gasto em reuniões e treinamentos.

Este foi um período no qual identificou-se a necessidade de um profissional com formação híbrida, um médico com conhecimentos de informática, que pudesse conduzir um processo de informatização de forma dinâmica e independente, inclusive nas questões referentes à concepção de recursos computacionais para as atividades médicas, com definição de recursos de *software*, *hardware* e capacitação humana para a integração efetiva de um sistema computacional em um ambiente médico. Foi, nesta época, 1987, que foi criada a Residência de Informática Médica, ligada ao grupo responsável pela Disciplina de Informática Médica do

Departamento de Patologia da Faculdade de Medicina da USP, que visava a formar médicos especialistas nesta área.

Evolução dos usos da informática na área médica

Na área médica o uso da informática ocorreu de forma muito irregular. Algumas especialidades tiveram maior interação com a informática, como a Radiologia, Ultra-sonografia, Oftalmologia, Patologia Clínica e Medicina Nuclear, entre outras, devido à incorporação da tecnologia computacional nos equipamentos utilizados. Nestes casos, a evolução foi mais expressiva no suporte a *hardware*, pois o objetivo primordial dos *softwares* era oferecer recursos que facilitassem o uso dos equipamentos pelos usuários. Esta evolução ocorria de forma isolada e independente do contexto da tecnologia de informação médica, uma vez que os fabricantes centravam-se nos equipamentos como instrumentos de medição e avaliação, como aqueles utilizados em laboratórios de análise clínica, sem a preocupação de integrá-los aos bancos de dados que contivessem outros dados dos pacientes. O uso dos computadores como coadjuvantes para a aquisição de dados estava aquém dos seus potenciais. Somente nos conjuntos mais recentes é que se verificou a preocupação com esta integração, enviando-se resultados de exames através das redes para que possam ser utilizados por outros sistemas.

As estruturas que tiveram evolução mais contínua, no contexto da estratégia da informação, foram os ambientes corporativos, como os hospitalares, que investiram em equipes de desenvolvimento e manutenção de sistemas (CPD). Porém, mesmos nestes, em grande parte, a informatização concentrou-se nos aspectos administrativos, deixando em segundo plano a informatização dos aspectos clínicos e assistenciais. O desenvolvimento de sistemas hospitalares tem alto custo e a aquisição de sistemas prontos também é cara, além de, na sua maioria, exigirem adaptações trabalhosas.

Os planos de saúde foram outras organizações que fizeram importantes investimentos em CPD, porém, centrando o desenvolvimento nos aspectos administrativos relacionados com o controle de recebimento de mensalidades

e aspectos ditos como assistenciais, mas que visavam a finalidades de auditoria médica, tais como controle dos serviços médicos consumidos pelos seus usuários e pagamentos à rede credenciada.

Talvez devido ao fator econômico menos expressivo ou talvez pela menor participação da comunidade médica, a informatização em pequenos núcleos médicos, como um consultório e/ou clínica, teve pouca expressividade quanto aos aspectos de aprimoramento tecnológico, continuidade de desenvolvimento e elaboração de bancos de dados de informação médica de apoio. A maioria dos *softwares* médicos também se centrou nos problemas administrativos, deixando os aspectos médicos para segundo plano.

A falta de desenvolvimento e de aperfeiçoamento continuados dos *softwares* médicos disponíveis no mercado marcou esta última década. Os *softwares* apareciam e desapareciam com extrema rapidez. Talvez pelo fato de não atenderem às expectativas financeiras, as empresas rapidamente abandonavam a área médica, causando descontinuidade na evolução dos produtos. Vários usuários foram obrigados a trocar seus sistemas, com conseqüente mudança de banco de dados, além de serem obrigados a fazer novos investimentos em treinamento em decorrência da falta de suporte e/ou impossibilidade de incorporar novos recursos.

Como costuma acontecer no início quando uma nova tecnologia começa a ser utilizada, o desenvolvimento ocorre por surtos, de forma descentralizada, pois cada núcleo desenvolve o seu próprio modelo trazendo falta de padronização para o funcionamento dos sistemas forçando o usuário a ter que se submeter a um novo treinamento ao mudar de um sistema para outro. Ausência de um padrão de protocolo de dados também prejudica o intercâmbio de informações entre os sistemas, ainda que ambos utilizem o mesmo tipo de banco de dados e/ou linguagem de desenvolvimento, fazendo com que as transferências de dados continuem a ser feitas através de documentos em papel. Paradoxalmente, um ambiente médico informatizado, por causa da falta de compatibilidade, pode estar isolado pelo uso desta mesma informática.

O *Tot-Clin*

É neste cenário histórico que desenvolvi o meu primeiro modelo para informatização de consultórios e clínicas, o *Tot-Clin* (inicialmente denominado de G-Clin^[11]), utilizando a linguagem *Clipper'87 Summer* como ferramenta. O fator motivador inicial para este desenvolvimento foi o atendimento das necessidades de uma clínica específica, visando-se a eliminar problemas relacionados com a replicação de trabalhos decorrentes da repetição da transcrição de informações de identificação dos pacientes na ficha de cadastro, na ficha clínica, nos pedidos de exame, etc., além de agilizar a localização dos prontuários dos pacientes. No procedimento manual, a procura da ficha do paciente era feita utilizando-se dois arquivos, um ordenado por nome e outro por número de registro. A transcrição repetida de dados foi eliminada com a emissão de etiquetas de identificação, e a localização dos prontuários com recursos de busca informatizada a partir do nome, sobrenome ou número de registro.

Poucos meses depois, os módulos do *Tot-Clin* foram analisados novamente, com uma abordagem mais ampla. O objetivo agora era a reestruturação do sistema para suprir as necessidades de um número maior de usuários. Em parte devido à insuficiência de recursos tecnológicos dos equipamentos e em parte devido ao maior interesse dos usuários em agilizar os aspectos administrativos e financeiros, o *Tot-Clin*, assim como a maioria dos programas desenvolvidos nesta época, era marcadamente voltado para estas atividades, embora já tivesse vários recursos para uso assistencial. Estes foram gradativamente aprimorados nas versões posteriores e foram implementadas novas características, como será descrito adiante.

A partir de 1989, o desenvolvimento *Tot-Clin* tornou-se mais expressivo, porque passou a ser tratado como ferramenta básica de automação em consultórios e clínicas, quando adquiriu características de um produto, como a elaboração de um manual mais detalhado para o usuário, estojo de acondicionamento, distribuição através de uma empresa, elaboração de módulos adicionais para armazenamento de informações assistenciais e médicas, maior flexibilidade para o gerenciamento de dados administrativos e financeiros, e recursos para levantamentos e emissão de relatórios.

Muitas implementações foram feitas a partir das sugestões dos próprios usuários ou de solicitações específicas de algumas clínicas. Todos os aspectos foram primeiramente analisados e classificados, e posteriormente agrupados para incorporação ao sistema. Este método de trabalho, implementada desde o início, agregou características de manutenção e aprimoramento contínuo ao *Tot-Clin* que, posteriormente, foram adotadas pelo *Tot-Win*. Esta sistemática otimiza o aprimoramento do sistema, uma vez que os custos de aperfeiçoamento do *software* são repartidos entre os usuários existentes. Este modelo também deu origem a um conceito que passamos a chamar de “desenvolvimento baseado em objetivo”; cada implementação tinha que estar inserida num contexto específico, envolvendo elaboração técnica, estruturação da solução e avaliação dos benefícios obtidos.

O treinamento é um fator importante e decisivo para o sucesso de uma informatização. A capacitação do elemento humano é necessária para que cada um tenha proficiência e confiança no uso do computador e do *software* como instrumento de trabalho diário, fator determinante para a efetiva integração às rotinas diárias. Algumas aptidões do usuário fora do contexto de informática podem ser importantes, como por exemplo, a falta de prática em datilografia que pode lentificar a velocidade de inserção dos dados no microcomputador.

O *Tot-Win*

A partir de 1992, quando ocorreu a consolidação dos ambientes gráficos, representados pelo sistema operacional *Windows*, a diminuição dos custos dos equipamentos de multimídia, e o fim da reserva de mercado de informática no Brasil, começou a surgir maior número de programas para *Windows*, e os *softwares* médicos passaram a incorporar recursos de multimídia com armazenamento de fotos, sons e vídeos.

Dois anos mais tarde, em 1994, comecei a desenvolver o *Tot-Win* baseando-me em vários aspectos importantes do modelo do *Tot-Clin*, como a sistemática de desenvolvimento e o conjunto de recursos assistenciais,

administrativos e científicos. Os aspectos assistenciais foram ampliados e passaram a englobar prescrição computadorizada de medicamentos baseada em banco de dados, dados clínicos evolutivos do paciente com imagens digitais, emissão de orientações médicas entre outras características, inseridos num cenário tecnológico e cultural que, embora em rápida evolução, exigia:

- Necessidade de estruturar num *software* os aspectos administrativos, assistenciais e científicos.
- Promover amplo uso da multimídia nos mais diversos *softwares* em decorrência da queda dos preços dos acessórios que incorporavam este recurso aos microcomputadores e do sucesso alcançado pelo *Windows 3.1*. Mais do que uma nova forma de comunicação, a multimídia é uma poderosa ferramenta para a criação de novos recursos para apoio à prática clínica diária, como a utilização de um banco de imagens para orientação e explanação aos pacientes, seqüências de vídeos didáticos, entre outros.
- Necessidade de desenvolver aspectos técnicos que fornecessem substrato para a validação das informações clínicas armazenadas exclusivamente em forma eletrônica. Com a validação legal poder-se-ia pensar em eliminar o prontuário médico físico em papel, o que significaria considerável economia de espaço nas clínicas, e agilidade da recuperação dos dados clínicos dos pacientes. O ambiente gráfico permitiria aos *softwares* guardarem dados textuais e imagens digitais, que poderiam ser fotos ou digitalização de laudos de exames.
- Estruturar um *software* que oferecesse recurso de compartilhamento *on-line* de dados entre diversos usuários simultaneamente.
- Desenvolver ferramentas que pudessem auxiliar o médico durante o atendimento aos pacientes, agilizando processos como prescrição de medicamentos, emissão de orientações e atestados, etc.

- Estabelecer critérios técnicos objetivos a serem consideradas na avaliação um *software* no momento da escolha, para implantação na clínica/consultório.
- Analisar aspectos que permitissem o intercâmbio de dados clínicos de pacientes entre sistemas computacionais, e que dispensassem o texto impresso.
- Avaliar a importância da padronização de termos para a recuperação de dados para fins de levantamento e intercâmbio de dados entre instituições.

Frente a estas necessidades, o *Tot-Win* vem sendo continuamente aprimorado, mas mais do que o desenvolvimento do *software*, este trabalho descreve:

- o modelo que envolveu este desenvolvimento;
- o relacionamento com os usuários e a maneira como suas sugestões são incorporadas sem que haja perda de compatibilidade, descontinuidade ou perturbação propagada para os demais usuários do mesmo sistema;
- a detecção e correção de erros, considerando as conseqüências para este mesmo conjunto de usuários;
- o desenvolvimento de novos conceitos;
- uma discussão sobre como a literatura especializada propõe soluções comparadas à vivência pessoal e à realidade brasileira em implementá-las ao longo de uma década.

Objetivos

- Descrever e discutir os principais aspectos de um modelo para o desenvolvimento de um sistema que reúna as principais necessidades de uma clínica/consultório, nos aspectos assistenciais, administrativos, financeiros e científicos.
- Apresentar as bases necessárias para se elaborar um sistema de prescrição computadorizada de medicamentos e avaliar a sua importância na atividade assistencial.
- Propor, definir, discutir e avaliar quais são os aspectos mais importantes para uma informatização eficiente e explorar os conceitos de eficiência e obsolescência.
- Estabelecer uma sistemática para replicação dos dados e geração de arquivo em formato que possa ser utilizado por outros sistemas computacionais.
- Indicar aspectos relacionados com validade jurídica e estabelecer alguns procedimentos técnicos para sua viabilização, considerando dados armazenados exclusivamente em meio eletrônico.
- Avaliar o papel da modelagem de dados na simplificação da manutenção e no ganho em flexibilidade de um *software*, e suas implicações na confiabilidade dos dados de um sistema.
- Avaliar aspectos relativos à utilização de termos padronizados para garantir a precisão na recuperação dos dados armazenados.
- Avaliar quais são os aspectos mais determinantes da qualidade de um *software*.
- Discutir e esboçar aspectos técnicos para o uso de recursos da *Internet* como ferramenta de apoio pós-consulta dos pacientes.
- Discutir a importância de um profissional para coordenar o desenvolvimento de um sistema médico e a necessidade e as vantagens de que este profissional tenha formação médica e de informática.

Materiais e Métodos

Começamos o desenvolvimento do *Tot-Win* em 1994 e sua evolução tem sido contínua até hoje, resultante da constante interação com os usuários. As trocas de versão das linguagens de programação, bibliotecas de apoio e sistemas operacionais no decorrer do tempo foram conseqüentes à evolução do mundo computacional^[1, 50], e contribuíram para criar a necessidade de manter o *software* atualizado em relação às novas tecnologias. Os materiais a seguir descritos são referentes aos utilizados atualmente. Dispersas ao longo do capítulo anterior, e quando for oportuno no restante deste capítulo, encontram-se algumas descrições relacionadas com a história particular da evolução do *Tot-Clin* e do *Tot-Win*, assim como menção às alterações provocadas pela evolução das linguagens, dos sistemas operacionais e do *hardware* utilizados em cada fase.

Equipamentos

Desenvolvimento

- microcomputador *notebook Pentium 266 MMX (Windows 98)*;
- 32 *megabytes* de memória RAM;
- disco rígido de 4 *gigabytes*;
- monitor colorido *dual scan*;
- drive de 3,5 polegadas;
- *Fax-Modem* de 56 Kpbs;
- leitor de CD.

Estrutura para testes técnicos, composta por

- três equipamentos *desktop Pentium II Celeron* com 64 *megabytes* de memória RAM, e disco rígido de 6 *gigabytes*;
- ZIP Drive de 100 *megabytes*;
- gravador de CD externo (porta paralela);
- impressora *Laser HP*;
- impressora jato de tinta *HP Desk jet 720c*;

- *scanner* de slides da marca *Polaroid*;
- placa *Vídeo Blaster* para digitalização de vídeo;
- *HP Desk Scanner*;
- *Fax-Modem* de 33 kps;
- *Hub* de 8 portas;
- cabo de par trançado, categoria 5 (8 vias) e conectores RJ 45 para conexão dos equipamentos em rede^[71];
- rede local ponto a ponto com *Windows 95*;
- *no-break* inteligente;
- estabilizadores de tensão;
- vídeocassete.

A utilização de um *notebook* foi importante por oferecer portabilidade para que o ambiente de desenvolvimento (linguagem de programação, programa fonte, bibliotecas de rotina, compiladores e programas auxiliares) fosse transportado aos locais de implantação do sistema, para as manutenções que envolvessem correções de erros e adequações (personalizações) fossem facilitadas e ganhassem mais agilidade.

Linguagem de desenvolvimento e *softwares* de apoio

- *Microsoft Visual FoxPro 6.0* – versão profissional^[52, 53, 54];
- biblioteca de edição de texto em OCX - *High Edit 4.0*^[34];
- *Paint Shop Pro 6.0*^[37];
- *Adobe Premiere*^[2];
- *Microsoft Powerpoint* e *Powerpoint Viewer*;
- *Microsoft Word* e *Word Viewer*;

Escolha da Linguagem de desenvolvimento e ambiente operacional

A linguagem de desenvolvimento escolhida foi a *Microsoft Visual FoxPro 6.0*, por ter herdado a sintaxe de programação das linguagens *dBase/Clipper* e mantido a compatibilidade com o padrão de dados DBF (formato que é acessado pela maioria dos aplicativos atualmente

existentes). Outras características determinantes da escolha foram: velocidade de processamento (devido à tecnologia “*Rushmore*”, segundo as especificações do fabricante da linguagem), baixo custo como ferramenta de desenvolvimento, e baixo custo para implementar um banco de dados, comandos poderosos (através de poucos comandos é possível controlar várias características de um sistema).

Por ser uma linguagem com herança dos comandos *dBase*, é de fácil aprendizado por programadores ou usuários que já possuem conhecimento prévio das linguagens *dBase/Clipper*. Como tecnologia *Front-end*, o *Visual FoxPro*, além de contar com várias facilidades de desenvolvimento, tem capacidade de acesso a banco de dados corporativos, como *Oracle*, *MS-SQL*, *Sybase* entre outros.

Apesar de ter algumas restrições (tabela 2), como por exemplo o tamanho máximo do banco de dados (cada tabela individualmente não pode superar 2 *gigabytes*), estas capacidades foram consideradas como suficientemente amplas, pois o *software* sempre foi dirigido ao porte de dados de uma clínica/consultório.

A capacidade de gerar uma aplicação *stand alone* foi uma característica considerada, pois seria possível gerar um programa executável e distribuí-lo sem gerar outros gastos ao usuário final, como pagamento de *royalties* ou aquisição de gerenciadores de banco de dados.

Tabela 2: Características técnicas do Visual FoxPro 6.0

Arquivos de tabela e índice:	
Nº máximo de registros por arquivo de tabela	1 bilhão
Tamanho máximo de um arquivo de tabela	2 gigabytes
Nº máximo de caracteres por registro	65.500
Nº máximo de campos por registro	255
Nº máximo de tabelas abertas ao mesmo tempo	2551
Nº máximo de caracteres por campo de tabela	254
Nº máximo de bytes por chave de índice em um índice não compactado	100
Nº máximo de bytes por chave de índice em um arquivo compactado	240
Nº máximo de arquivos de índices abertos por tabela	ilimitado
Nº máximo de índices abertos em todas as áreas de trabalho	ilimitado
Nº máximo de relações	ilimitado
Comprimento máximo de expressões relacionais	ilimitado
Características de campos	
Tamanho máximo de campos de caractere	254
Tamanho máximo de campos numéricos (e flutuantes)	20
Nº máximo de caracteres nos nomes dos campos de uma tabela livre	10
Nº máximo de caracteres nos nomes dos campos de uma tabela contida em um banco de dados	128
Valor mínimo de um número inteiro	2.147.483.647
Valor máximo de um número inteiro	2.147.483.647
Dígitos de precisão em cálculos numéricos	16
Matrizes e variáveis de memória	
Nº padrão de variáveis de memória	1.024
Nº máximo de variáveis de memória	65.000
Nº máximo de matrizes	65.000
Nº máximo de elementos por matriz	65.000
Arquivos de programas e procedimentos	
Nº máximo de linhas em arquivos de programas de origem	ilimitado
Tamanho máximo de 3 módulos de programas compilados	64K
Nº máximo de procedimentos por arquivo	ilimitado
Nº máximo de chamadas DO aninhadas	128
Nº máximo de níveis de aninhamento de READ	5
Nº máximo de comandos de programação estruturada aninhados	384
Nº máximo de parâmetros passados	27
Nº máximo de transações	5
Recursos do criador de relatórios	
Nº máximo de objetos em uma definição de relatório	ilimitado
Comprimento máximo de uma definição de relatório	20 polegadas

Nº máximo de níveis de agrupamento	128
Outros recursos	
Nº máximo de janelas abertas (todos os tipos)	ilimitado
Nº máximo de janelas Pesquisar abertas	255
Nº máximo de caracteres por seqüência de caracteres	2 gigabytes
Nº máximo de caracteres por linha de comando	8.192
Nº máximo de caracteres por controle de etiqueta em um relatório	252
Nº máximo de caracteres por linha de substituição de macro	8.192
Nº máximo de arquivos abertos	Limite do sistema
Pressionamentos de teclas máximos em macro de teclado	1.024
Nº máximos de campos selecionados pela instrução SQL SELECT	255

fonte: *Microsoft Visual FoxPro 6.0*, 1999^[54]

Em síntese, a ferramenta de desenvolvimento *Visual FoxPro 6.0* foi escolhida devido aos seguintes aspectos:

- Compatibilidade com o *Microsoft Windows 95* ou superior.
- Capacidade de gerenciar dados sem necessidade de adquirir um servidor de banco de dados.
- Boa velocidade de processamento de dados proporcionada pela tecnologia "*Rushmore*".
- Capacidade de gerar executáveis para distribuição, sem depender da linguagem original de desenvolvimento.
- Rapidez de aprendizado (herança da sintaxe *dBase*) e facilidade para o desenvolvimento de aplicativos de banco de dados, sem necessidade de adquirir bibliotecas de desenvolvimento adicionais.
- Possibilidade de migrar os dados do *Visual FoxPro* para uma arquitetura cliente/servidor e trabalhar com o novo banco de dados utilizando o mesmo programa, sem necessidade de alterar nenhum código original.
- Capacidade intrínseca para gerenciar dados compartilhados em aplicativos utilizados em ambiente com muitos usuários trabalhando simultaneamente.

- Possibilidade de implementar recursos de automação com outros aplicativos *Windows* através do OLE.
- Recursos para programação orientada a objeto na linguagem.
- Capacidade de intercambiar dados com HTML para ambiente de *Internet e Intranet*.
- Capacidade de acessar simultaneamente bancos de dados diferentes como, por exemplo, o acesso a banco de dados *Oracle* e *SQL Server*, para pesquisar informações.
- Capacidade de gerenciar imagens digitais em diversos formatos, como JPG, GIF, BMP, etc.

O *Visual FoxPro* gerencia os bancos de dados em dois formatos::

1. Tabelas livres, semelhantes aos modelos *dBase* e *Clipper*, no qual as informações são armazenadas em arquivos físicos independentes (com extensões DBF, FPT) e arquivos de índice (CDX e IDX). Tem a vantagem da modularidade das informações. Os arquivos DBF do *Visual FoxPro* receberam algumas implementações adicionais tornando-os mais estáveis e confiáveis do que o padrão DBF original.
2. banco de dados relacional (usando arquivos DBC); as informações são incorporadas num único arquivo físico, embora do ponto de vista lógico comportem-se como tabelas independentes. Este é um modelo de gerenciamento que oferece funções de integridade referencial de dados, *triggers*, filtros, visualizadores de tabelas e outros recursos que automatizam e garantem a consistência dos dados. As informações armazenadas neste formato podem ser transferidas diretamente para um banco de dados padrão SQL.

Recursos Humanos

- Médico com conhecimentos de informática, planejamento de informação, análise de sistemas e programação, organização e métodos e administração.
- Dois profissionais de informática, cursando Ciência da Computação, para suporte e treinamento aos usuários.

Método

Fases para administração e desenvolvimento de um *software*

- *Análise e engenharia de sistemas*: levantamento dos requisitos para os componentes de um sistema. O *software* é integrante de um conjunto mais amplo e é preciso considerar suas inter-relações com outros elementos, como, por exemplo, pessoas, *hardware*, e banco de dados.
- *Análise de requisitos de software*: é fundamental ter o domínio da informação e dos processos, para entender a natureza dos módulos a serem desenvolvidos para que se possa definir as funções, o desempenho e as *interfaces* necessárias.
- *Projeto do sistema*: sistematização e documentação das etapas anteriores, com a modelagem das tabelas de dados, elaboração da HCI, etc.
- *Codificação do sistema* (também denominada de programação): fase de desenvolvimento que "materializa" o projeto.
- *Testes do sistema desenvolvido*: detecção de erros.
- *Manutenção* (depois que é entregue ao usuário): reparo de erros detectados durante o uso, necessidade de adaptações devido a mudanças ocorridas no ambiente, necessidades de novas funções e implementações que melhorem o conjunto como um todo e que não haviam sido previstas inicialmente.

O processo de elaboração de um *software* independe, em parte, da técnica de engenharia escolhida. A definição, o desenvolvimento e a

manutenção sempre ocorrem independentemente da área de aplicação, tamanho ou complexidade do projeto.

A fase de definição focaliza-se na expressão "**O Que...?**" precedendo cada pensamento relacionado com a análise do sistema, seus requisitos e planejamento do *software*.

A fase de desenvolvimento (engloba o projeto, a codificação e os testes do *software*), focaliza-se em "**Como...?**", para definir a estrutura de dados, a arquitetura do *software*, os detalhes dos processos a serem implementados, a transcrição do projeto para uma linguagem de programação e os testes que deverão ser realizados, utilizando métodos que variam de acordo com a técnica empregada.

A fase de manutenção focaliza-se nas "**Mudanças**" que estão associadas à correção de erros, às adaptações exigidas à medida que o ambiente do *software* evolui, ao melhoramento funcional, e às ampliações produzidas por novas necessidades dos usuários ^[63].

Fases de desenvolvimento do *Tot-Win*

Basili e Zelkowitz ^[4], definem cinco fatores importantes que influenciam a produtividade do *software*:

- *Fatores humanos*: o tamanho e a experiência da organização que está desenvolvendo o *software*.
- *Fatores do problema*: a complexidade do problema a ser resolvido e o número de mudanças exigidas quando os requisitos ou restrições ao projeto estão sendo estabelecidos.
- *Fatores do processo*: técnicas de análise que são usadas e técnicas de revisão empregadas.
- *Fatores do produto*: confiabilidade e desempenho do sistema.
- *Fatores relacionados aos recursos*: ferramentas de desenvolvimento (CASE), e recursos de *hardware* e de *software* disponíveis.

O processo de desenvolvimento do *Tot-Win* seguiu as fases tradicionais da engenharia de *software*, com algumas implementações adicionais. As

etapas de desenvolvimento e manutenção foram fundidas para criar um núcleo de aprimoramento permanente, formando um “condomínio colaborativo de aperfeiçoamento” para o qual os usuários podiam contribuir com sugestões. Esta estratégia de trabalho visou a garantir o aprimoramento contínuo, porém mantendo o foco no desenvolvimento por objetivos. Além dos aspectos técnicos, foram incorporados conceitos metodológicos ao *software*, como será abordado na discussão. O conceito de “condomínio” diminui os custos diretos de cada usuário com as implementações, e permite que soluções sejam elaboradas antecipando as necessidades dos usuários.

O desenvolvimento do *Tot-Win* envolveu:

- Entendimento das atividades e processos a serem informatizados, levantamento e documentação dos requisitos.
- Escolha da linguagem, banco de dados e bibliotecas de apoio a serem utilizados (instrumentação).
- Capacitação do desenvolvedor para a utilização das ferramentas escolhidas.
- Modelagem dos bancos de dados baseado nos requisitos do *software*, levando-se em conta conceitos advindos de *Data Warehouse*.
- Estruturação e definição dos módulos do sistema e sua dinâmica processual.
- Desenvolvimento e codificação do sistema.
- Implantação do sistema e treinamento dos usuários.
- Avaliação do grau de otimização resultante do processo de automação obtido com o *software*.
- Estruturação de um serviço de suporte ao usuário, para garantir orientação permanente quanto ao uso adequado do programa.
- Estruturação do serviço de suporte como fonte de obtenção de sugestões para o aprimoramento permanente.

- Análise das solicitações de melhorias, planejamento e execução de novas implementações.
- Ajuste e generalização de novos módulos para torná-los compatíveis com os outros recursos do sistema. Quando um dos participantes do “condomínio” fazia uma solicitação, em um primeiro momento procurava-se suprir suas necessidades, sem nos preocuparmos com o grau de especificidade do desenvolvimento. Depois que o novo recurso era implementado e testado por este usuário, o módulo era generalizado para que o benefício pudesse ser estendido aos demais participantes do "condomínio", sem que novos erros fossem introduzidos, e sem perturbar os serviços em andamento.

Aspectos como capacidade de replicar e intercambiar dados, fechamento de registro com possibilidade de reconstituição temporal (histórico), fundamento técnico para viabilizar a auditoria pericial, e técnicas de modelagem dimensional do *Data Warehouse*^[5] foram levados em consideração durante a fase de modelagem das tabelas de dados do sistema. Redundância de informações foram aceitas considerando-se fatores como legibilidade e independência de dados, uma vez que não comprometesse os aspectos funcionais e a capacidade de armazenamento e processamento dos padrões de equipamentos existentes no momento, considerando-se o porte de dados de um consultório ou clínica. A configuração de equipamento considerada padrão muda constantemente^[15, 13]. Em abril de 2000 é um microcomputador *Pentium II Celeron 466* com 64 *megabytes* de memória RAM, *Windows 98*, disco rígido de 6,4 *gigabytes*, placa de vídeo de 4 *megabytes*, placa de *Fax-Modem 56K*, leitor de CD-ROM 40x ou DVD, acionador de disco de 3,5 polegadas, portanto superior à configuração dos computadores que utilizamos para testar o sistema.

Aspectos dos dados que foram levados em consideração:

- Abrangência: amplitude da vinculação dos dados com outros módulos do sistema e liberdade para serem consultados ou utilizados.

- Duração: tempo em que a informação é mantida sem ser alterada ou apagada.
- Periodicidade de atualização: frequência com que a informação é atualizada.
- Confiabilidade: grau de certeza de que os dados reflitam a realidade.
- Categorização: capacidade das informações serem agrupadas segundo critérios escolhidos.
- Modularidade: capacidade das informações manterem a legibilidade sem dependência de outras tabelas.
- Padronização: capacidade de utilizar uma expressão constante para descrever uma mesma situação em momentos diferentes.
- Possibilidade de intercâmbio de dados: característica que permite enviar ou receber dados digitais entre usuários que usam o mesmo sistema e/ou sistemas diferentes.
- Capacidade e facilidade de recuperação: relacionado com a simplicidade para desenvolver módulos para levantamento de dados, uma vez que a padronização dos dados esteja garantida.
- Facilidade de replicação sem dependência de tabelas de apoio: garantida pela modularidade, simplificando o intercâmbio de dados.
- Redução de custos de desenvolvimento e manutenção: facilidade para o desenvolvimento de novos módulos em decorrência do planejamento de dados.

Para armazenar dados inseridos neste formato, foram definidos campos do tipo Memo (texto) na tabela, que no *Visual FoxPro*, tem como característica armazenar os dados num arquivo independente que aumenta de acordo com o tamanho do texto inserido, além da capacidade de ser processado como um campo do tipo caractere, o que permite selecionar registros segundo uma expressão de comparação.

Em relação ao armazenamento de dados clínicos no *Tot-Win* foram considerados os seguintes aspectos:

- Inserção dos dados cadastrais e administrativos dos pacientes, e utilização do número de registro para agilizar a localização e recuperação dos prontuários físicos.
- Concepção de recursos assistenciais que pudessem auxiliar na elaboração da prescrição de medicamentos, emissão de orientações médicas e de atestados aos pacientes.
- Impressão de etiquetas de identificação e de fichas para utilização como história clínica, com os dados cadastrais preenchidos.
- Armazenamento de dados clínicos sumarizados para consulta rápida pelo médico quando necessário, como por exemplo, em esclarecimento de dúvidas de pacientes pelo telefone.
- Estruturação de aspectos que forneçam embasamentos técnicos para uso como critérios em auditorias periciais de dados exclusivamente eletrônicos.

Aspectos de Iconografia

Padrões utilizados no *Tot-Win*

Devido à falta de um padrão bem definido de iconografia computacional para a área médica, além dos *ícones* do *Windows*, foram elaborados alguns outros para uso no *Tot-Win*. Apesar do sistema ser em *Windows*, consideramos que o uso de palavras para definir botões de comandos e legendas de campos ainda são mais eficientes que *ícones*, porém, as palavras precisam ser adequadamente escolhidas.

Padronização adotada para as legendas de campos:

- Campos com comando Ponto (descrito à página 56): possuem as legendas totalmente em letras maiúsculas.
- Legendas vermelhas: expressam campos que são calculados automaticamente.
- Legendas com quadro em azul: para designar a existência de bancos de dados externos ao sistema para pesquisa dos dados, como, por exemplo, lista de CEPs.

Devido à existência de muitas particularidades, divididas em especialidades, e de acordo com as doenças, o desenvolvimento de telas estruturadas e padronizadas para inserção de dados clínicos e assistenciais não foram feitas, uma vez que isto tornaria o *software* complexo demais para o uso no dia a dia. Mesmo dentro de uma especialidade, a padronização total dos formatos da história clínica nem sempre são de consenso.

Para atender mais adequadamente às características de algumas especialidades, foram elaboradas telas de evolução clínica específicas, como as de ginecologia e obstetrícia. Foi também criada uma ficha contábil para uso em agendamento de exames em anatomia patológica.

Imagens digitais - formato de arquivo e modelagem dos bancos de dados

Recurso para o armazenamento de fotografias clínicas dos pacientes juntamente com a ficha clínica evolutiva pode ser importante quando se abordam os aspectos assistenciais, pois ela permite documentar e acompanhar a evolução clínica do paciente com o tratamento. É mais importante em algumas especialidades do que em outras.

A aquisição de imagens pode ser feita por meio de máquinas fotográficas convencionais com posterior *escaneamento* das fotos, ou ser feita através de máquinas fotográficas digitais. Estas últimas tiveram significativa evolução e redução de custo nestes últimos anos.

O formato de arquivo escolhido para armazenamento das imagens digitais foi o JPG, pois é compatível com a *Web*, é de tamanho relativamente pequeno pela capacidade de compactação, armazena as imagens em 16 milhões de cores, e tem resolução suficiente para diagnóstico^[44, 74].

No *Windows*, cada imagem é armazenada como um arquivo independente. Quando uma pasta tem mais que 1.000 arquivos, o acesso começa a se tornar perceptivelmente mais lento, além do inconveniente de dificultar a localização das imagens de um determinado paciente pela grande quantidade de arquivos com nomes nem sempre mnemônicos. Para resolver estes problemas, foram definidos na tabela de dados da evolução clínica seis

campos do tipo Memo Binário, para armazenamento dos arquivos de foto. Como as fotos são capturadas e guardadas dentro da tabela, evitam-se os problemas decorrentes do aumento do número de arquivos no disco, e quando se localiza a ficha evolutiva do paciente, as respectivas fotografias são automaticamente apresentadas.

Este mesmo conceito foi utilizado para os módulos de Banco de Imagem e Memento.

Modelagem de dados – Unidade Celular de Informação

A redundância de dados em tabelas relacionadas teve como objetivo implementar a independência e aumentar a legibilidade das informações, individualmente, em cada tabela, o que passamos a denominar de unidade celular de informação (UCI). Isto possibilitou desenvolver recursos como replicação de dados clínicos através de discos magnéticos^[12], lançamento de pacotes de fechamento de dados clínicos para arquivo de auditoria e viabilização do envio de pacotes de dados para uso da *Web* como contra-referência. São exemplos de redundâncias que foram aceitas: nome do paciente, registro, sexo e idade nas tabelas de prescrição médica (medicamentos prescritos), evolução clínica, ficha financeira da agenda e ficha cadastral (figura 11).

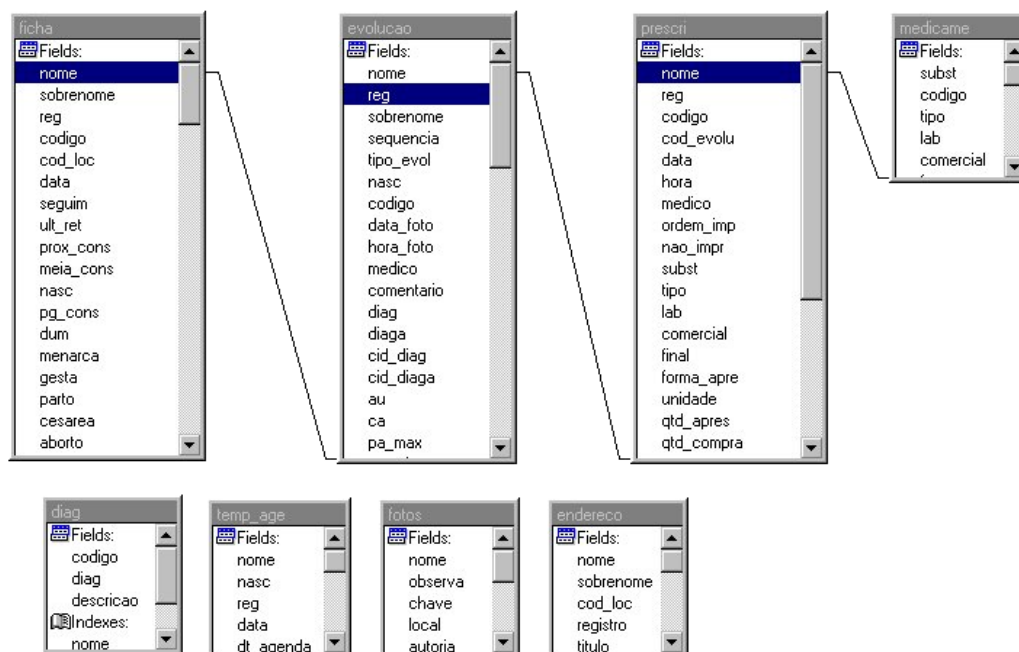


Figura 11: Exemplo de relacionamento entre diferentes tabelas com redundância de campos (acima) e tabelas auxiliares para a padronização de termos (abaixo).

Estruturação dos módulos do sistema

Os recursos do sistema foram subdivididos em 3 grandes grupos:

- Assistenciais – reúne recursos para a atividade de atendimento ao paciente e banco de dados de apoio para padronização de termos médicos ^[38, 69].
- Administrativo - Financeiros – com módulos para o controle gerencial e financeiro da clínica ou consultório, como sistema de segurança de dados, cadastro de convênios, tabela de honorários ^[14], estatísticas gerenciais, entre outros.
- Científico – Didáticos – possuindo módulos que visam a auxiliar a recuperação de dados para levantamentos de casos, protocolos científicos, banco de dados de casos clínicos, diapositivos clínicos, mementos (cadastro de medicamentos com imagens), entre outros.

Bancos de dados de apoio e padronização de termos médicos

Para que o *Tot-Win* pudesse ser utilizado de forma imediata e eficaz, foram preparados alguns bancos de dados de apoio, como cadastro de medicamentos, CID, lista de procedimentos da AMB e SNOMed para utilização nos principais módulos assistenciais do sistema e facilitar a padronização dos termos clínicos e terapêuticos; banco de imagens e seqüências de vídeos; memento como apoio educativo aos pacientes; banco de casos clínicos, doenças e síndromes como literatura de referência aos médicos, etc.

Desenvolvimento de aspectos *Web*

Para estruturação dos recursos operacionais em ambiente *Web*, foi definido o uso do *Microsoft Outlook* como *software* para envio de e-mail e o *Microsoft Internet Explorer* como *browser* das páginas *Web*, baseado no fato de que ambos são *freeware* para usuários que tenham *Windows*. Para implementação de listas de discussão, foi definida a utilização de CGI's baseadas em *Perl*, que estão disponíveis gratuitamente na *Web* para criação de ambiente para debates e esclarecimento de dúvidas.

Resultados

Parte dos resultados deste trabalho relaciona-se como os critérios para o desenvolvimento de um *software*, o *Tot-Clin* e seu sucessor atual, o *Tot-Win*. Encontra-se em anexo (página 159) um guia geral da versão corrente do programa, que permite uma visão mais detalhada de seu uso, que pode facilitar o entendimento de alguns resultados subseqüentes.

O *Tot-Win*, derivado do *Tot-Clin*, teve grande parte dos conceitos e recursos elaborados, do ponto de vista funcional e lógico, a partir das rotinas pré-existentes. As alterações concentraram-se na inclusão de novos recursos nos módulos assistenciais, e novas definições nas tabelas de dados foram incorporadas para ampliar os aspectos administrativos ou para aprimorar funções previamente existentes.

Cada novo recurso assistencial do *Tot-Win* foi projetado e avaliado de forma separada e posteriormente agregado ao conjunto. Esta foi a forma que foi considerada como eficiente para avaliar a objetividade e funcionalidade de cada módulo, pois permitia restringir a quantidade de fatores envolvidos em cada momento. Um exemplo foi o banco de imagens para explanação aos pacientes que, depois de desenvolvido e elaboradas as seqüências de imagens de orientação, foi aplicado para o atendimento de pacientes^[60]. Outros recursos foram previamente prototipados e debatidos em cursos ministrados durante eventos científicos, permitindo recolher e avaliar opiniões em relação às repercussões que teriam se fossem usados durante o atendimento diário ao paciente, como o uso do sistema computadorizado para a prescrição de medicamentos.

O *Tot-Win* foi projetado para ser utilizado por várias especialidades. Para aumentar a abrangência sem sacrificar a praticidade, foram criadas janelas de texto livre para armazenamento de dados clínicos evolutivos. É possível que o usuário crie modelos de texto previamente, e os utilize dentro das janelas de texto livre. Textos totalmente livres sacrificam demasiadamente a padronização de termos e, geralmente, dificultam uma boa recuperação de informações. Textos totalmente padronizados, por outro lado, podem restringir a liberdade do médico. O uso adequado do recurso de

textos pré-definidos pode conciliar parcialmente os extremos, pois permite que se padronize e agilize a inserção de dados, além do que é de fácil aceitação porque se aproxima do que se faz com folhas de papel.

Desenvolvimento de um programa para automação de clínicas e consultórios

O *software* resultante do desenvolvimento (*Tot-Win* – Sistema para automação TOTal de clínicas, em versão *Windows*) é um dos sistemas computacionais para informatização de ambiente médico de pequeno porte; reúne recursos para abordagem dos principais aspectos assistenciais, administrativos e científicos numa clínica ou consultório. O programa foi elaborado para ser executado em ambiente de rede local e roda sob *Windows 95* ou superior.

Desenvolvimento de recursos especiais e funções específicas do programa

O armazenamento e padronização adequados das informações são fatores fundamentais para a análise eficiente dos dados armazenados num sistema, e é ainda mais importante quando se deseja recuperar e analisar dados clínicos inseridos durante a fase de atendimento do paciente, pois além de garantir a padronização dos termos é preciso oferecer ao usuário agilidade na inserção dos dados.

Diversas tabelas de apoio de padronização foram elaboradas para serem utilizadas em áreas diferentes do sistema, seguindo padronização internacional ou nacional, como CID, SNOMed, Lista de Procedimentos da Associação Médica Brasileira, etc.

Para facilitar as inserções dos dados, aumentando a confiabilidade e precisão para a procura posterior e possibilitar o armazenamento de imagens digitais, foram desenvolvidos alguns comandos e funções especiais para uso no *Tot-Win*, relacionados a seguir:

- Comando Ponto: agiliza a digitação de informações em campos específicos, buscando-as a partir de um banco de dados de apoio.

Por evitar a livre digitação repetitiva das mesmas informações e por permitir lançar até uma frase completa apenas digitando os caracteres iniciais. Esta função demonstrou ser um recurso importante e ágil para padronização de termos. Garante que as expressões sempre sejam inseridas da mesma forma. Todos os campos que possuem este recurso têm as suas respectivas legendas escritas em letras maiúsculas. Ao digitar um ponto (.) como primeiro caractere do campo, o sistema permite que o usuário compare a expressão digitada após o ponto com o banco de dados de referência do campo. Caso sejam encontrados mais que um registro, o comando apresenta a relação na tela para que o usuário selecione o registro desejado. Não há necessidade de digitar a expressão por completo, pois este comando faz a procura de acordo com a precisão de caracteres digitada. Quando não se deseja que a expressão digitada seja comparada com o banco de dados, basta digitar normalmente. Exemplos de campos que possuem o comando Ponto: diagnóstico, médico responsável, convênio, médico indicação (que registra o nome do médico que encaminhou um paciente), medicamentos da ficha clínica, entre outros.

- Geração do Cod. Loc. (código de localização): esta função agrega a data, o mês de nascimento e as iniciais do nome, excluídas preposições, para compor um código especial. É considerada pelos usuários como uma boa alternativa de procura de ficha de pacientes quando falham os outros métodos, como procura por nome, sobrenome, e número de registro. Outras alternativas foram também avaliadas, como pesquisa fonética do nome, que se mostrou ineficiente em casos de abreviação de nomes. A procura por CPF é inviável para crianças. Procura por telefone demonstrou-se falha em decorrência das constantes mudanças dos prefixos e números de telefones recentemente.
- Rotina para armazenamento e visualização de imagens digitais em formatos JPG, embora seja possível utilizar as imagens em GIF e BMP, incluindo o desenvolvimento de recursos para a ampliação das

imagens e capacidade de colocar as fotos selecionadas lado a lado para comparação, útil quando se deseja comparar fotos clínicas antes do tratamento e após o tratamento.

- Rotina para retirar espaços em branco inseridos acidentalmente durante a digitação dos nomes de pacientes, cuja identificação fica dificultada em virtude das fontes proporcionais do *Windows*: os nomes podem parecer estar digitados corretamente, porém na pesquisa, o paciente pode não ser localizado devido a espaços extras entre as palavras. Esta rotina converte tudo para um único espaço vazio entre as palavras.
- Rotina para cálculo das idades com precisão para anos, meses e dias. A precisão até número de dias tem importância quando o paciente é uma criança.
- Recurso de filtro, que permite ao usuário, utilizando sintaxe do *dBase*, estabelecer regras de visualização das informações, subtraindo os registros que não atendem às condições. É um comando importante quando se deseja fazer levantamento de casos clínicos.

Aspectos de Iconografia

Exibição de Informações

Quando as informações apresentadas pela HCI são incompletas, ambíguas ou ininteligíveis, a aplicação deixará de atender às necessidades do usuário. As informações podem ser exibidas de muitas maneiras diferentes: por meio de textos, imagens e sons, usando cor e resolução. Até mesmo a omissão de parte das informações em momentos específicos pode aumentar a ênfase, facilitando a assimilação do que está sendo transmitido. Algumas diretrizes para a exibição de informações são:

- Mostrar somente informações que sejam relevantes ao contexto daquele momento. Não se deve forçar o usuário a vagar por dados, menus e gráficos estranhos para obter informações relevantes para uma função específica do sistema.

- Não sufocar o usuário com dados em demasia e usar formatos de apresentação que facilitem a rápida assimilação das informações.
- Usar rótulos (*labels*) e mensagens consistentes, abreviações padronizadas e cores previsíveis. O significado de uma apresentação deve ser óbvio.
- Caso as apresentações gráficas em um computador permitam deslocamentos para cima ou para baixo da tela, é interessante que se exiba uma amostra da imagem original, em forma reduzida, em um canto da tela, para que o usuário conheça a localização relativa da parte da imagem que está sendo vista.
- Emitir mensagens de erros compreensíveis pelo usuário.
- Usar letras maiúsculas e minúsculas, entradas e textos organizados em grupo para ajudar na leitura. Grande parte das informações apresentadas ainda é textual: diagramação e forma do texto exercem impacto sobre a facilidade com que as informações são assimiladas pelo usuário.
- Usar janelas (se forem adequadas) para subdividir diferentes tipos de informação em áreas diferentes. As janelas permitem ao usuário gerenciar tipos diferentes de informação simultaneamente.
- Planejar a área disponível da tela de exibição e usá-la eficientemente. Quando múltiplas janelas tiverem de ser usadas, deve haver espaço disponível para mostrar pelo menos uma parte de cada uma. Deve-se especificar um tamanho de tela adequado para a aplicação a ser desenvolvida.

A seguir aparece uma discussão mais detalhada dos recursos existentes em vários módulos do *Tot-Win*. Os recursos têm sido implementados principalmente em função das solicitações e da interação com os usuários participantes do "condomínio" para aperfeiçoamento do sistema, já descrito em "Fases de desenvolvimento do *Tot-Win*" (página 46). Cada um dos recursos foi projetado para atender, pelo menos, a necessidade de algum usuário. Todos os recursos estão em uso e são operacionais. Muitas vezes,

por causa da flexibilidade que o sistema oferece, pela incorporação de um conjunto de 40 comandos para recuperação de dados segundo condições específicas (montagem de *queries*) e liberdade para a definição dos relatórios e outros padrões utilizados por vários módulos, diferentes usuários podem encontrar aplicações variadas ao combinar cada um destes recursos, inicialmente imaginados para demandas específicas.

Recursos para finalidades assistenciais

Ficha clínica

Ficha cadastral com informações gerais do paciente (nome, sexo, idade, estado civil, profissão, CPF, indicação, telefone, endereço, etc.), dados administrativos (convênio, número no convênio, data da última consulta, etc.) e foto para identificação do paciente:

- Tela de texto para registro da história clínica, incluindo campos específicos para armazenamento do peso, altura, menarca, data da última menstruação, tipo sanguíneo, diagnósticos principais, SNOMed, medicamentos que estejam em uso crônico. Há também uma janela de texto livre para armazenar dados clínicos. Os campos para diagnóstico(s), medicamentos e SNOMed estão dotados com o comando Ponto e estão vinculados aos seus respectivos banco de dados para padronização dos termos inseridos.
- Dados clínico-evolutivos: cada registro clínico-evolutivo do paciente tem capacidade para armazenar até 6 imagens digitais, texto livre para inserção das intercorrências clínicas e estadiamento tumoral segundo critérios TNM, e está vinculado à ficha clínica do cadastro do paciente através de uma codificação interna, individualizada por data de atendimento e hora. Esta individualização possibilita que um mesmo paciente possa ter duas ou mais fichas evolutivas num mesmo dia, o que é importante quando o paciente passa pela avaliação de dois ou mais médicos. A tela de trabalho da ficha evolutiva foi estruturada visando a disponibilizar os principais módulos que o médico poderia necessitar naquele momento, como o banco de

orientações médicas, o banco de imagens para explicações aos pacientes, a janela para visualização de todas as evoluções clínicas, o banco de dados de endereços para encaminhamentos, textos para lançamento de particularidades do paciente e a tela de dados clínicos.

Apesar do armazenamento de informações clínicas em meios eletrônicos oferecerem importantes benefícios, ainda não se pode eliminar definitivamente os prontuários físicos. Segundo a resolução Nº 1331/89 do CFM, a manutenção do prontuário físico deve ser pelo período de 10 anos.

A necessidade de manter o prontuário físico e inserir dados no sistema computacional gera duplicação de trabalho. Para minimizar este problema, sem prejuízo do aspecto de validade jurídica, foi desenvolvido o recurso de impressão da evolução clínica onde o usuário pode informar a linha a partir da qual deseja que os dados sejam impressos. A vantagem desta rotina é permitir que o usuário insira os dados evolutivos na janela de texto, e depois utilize a mesma folha do prontuário para continuar a impressão, evitando-se a necessidade de reimpressão de toda a ficha evolutiva ou uso de uma nova folha para cada retorno. Como o texto é impresso, datado e assinado pelo médico, todos os aspectos relativos à certificação da veracidade do documento estão preservados, sendo o programa computacional utilizado apenas como meio para a elaboração do prontuário (processo de consulta ao CFM Nº 1345/93).

Outras características importantes da ficha clínica evolutiva são (1) o comando para visualizar, em tela única, todos os acontecimentos evolutivos do paciente e (2) o recurso de reagrupamento de texto a partir de um texto original. Este último consiste em reagrupar parágrafos de texto que se iniciem com um código pré-definido (um rótulo padronizado e acrescentado pelo usuário, para assinalar uma entrada específica, por exemplo "Hb:" ou "Biópsia Gástrica:" antecedendo os valores de hemoglobina ou a transcrição do laudo, respectivamente). É um recurso interessante quando se deseja

visualizar de forma sintetizada a evolução de um determinado exame, por exemplo, biópsias seqüenciais para acompanhamento de um determinado tratamento, ou para elaboração de um documento com os principais aspectos da doença do paciente.

- Atualização e replicação de dados: este recurso permite que sejam transferidos dados clínicos de um paciente para um outro computador que utiliza *Tot-Win*, mas que não estejam interligados entre si para compartilhamento do mesmo banco de dados. Por exemplo: um médico que tenha informações de um paciente no computador da clínica e deseje transferir estes dados ao seu *notebook* para visita hospitalar e depois, atualizar o sistema central com os dados coletados (sincronia das informações). Um outro exemplo seria quando se deseja enviar os dados de um paciente para um outro colega que utiliza o mesmo programa.
- Fechamento de registro de dados: este recurso replica os dados do registro selecionado para um arquivo de auditoria que não é acessível pelo usuário que está usando o programa. Isto implica basicamente em 3 características: (1) criar um banco de dados de *backup* interno no sistema, que permite a recuperação das informações em casos de danos ao banco de dados clínico; (2) possibilitar o rastreamento e acompanhamento do fechamento de dados clínicos no sistema; (3) possibilitar a geração de um arquivo para envio, através da *Internet*, para um outro banco de dados que funcionará como um espelho, e que poderá servir como contra-referência dos dados clínicos do sistema. Estes aspectos, além de aumentarem a segurança dos dados, são processos técnicos que podem ajudar a rastrear a evolução dos dados, e podem fazer parte da validação jurídica de dados armazenados em meio exclusivamente eletrônico, como será discutido adiante.
- Geração de dados clínicos em arquivos TXT (formato texto): este recurso converte os dados clínicos e acontecimentos evolutivos do paciente para um arquivo em formato texto, com inclusão do nome,

tamanho e tipo do campo, de acordo com a estrutura das tabelas do sistema. O padrão TXT foi escolhido pelo fato de ser passível de leitura por praticamente todos os sistemas computacionais, inclusive computadores que executem sistemas operacionais baseados em caracteres, como o dos. A desvantagem deste formato é a incapacidade de agregar diretamente imagens digitais às informações textuais. Para contornar esta limitação, o *Tot-Win* atribui nomes diferentes aos arquivos de imagem, e insere estes nomes no local correspondente do texto visando a permitir a reconstituição da informação por um outro programa com recursos gráficos, desde que este tenha algum módulo que siga este padrão de transferência e recuperação de dados.

- Prescrição de medicamentos: o módulo de prescrição de medicamentos do *Tot-Win* está vinculado com a ficha evolutiva do paciente, e tem as seguintes características (figura 12):
 - ◆ As informações relativas a cada medicamento prescrito são armazenadas em banco de dados, e ao final da prescrição, são transcritos de forma resumida para o texto da evolução clínica.
 - ◆ A pesquisa dos medicamentos pode ser feita através do nome comercial, substância ativa ou nome genérico, categoria e qualquer palavra que esteja dentro destes campos.
 - ◆ Recurso para prescrever medicamentos e fórmulas magistrais.
 - ◆ Medicamentos e fórmulas magistrais são impressos em receitas separadas.
 - ◆ Recurso para impressão de toda a prescrição ou cada medicamento individualmente.
 - ◆ Impressão das orientações vinculadas aos medicamentos, uma vez que estejam cadastradas no banco de dados de medicamentos.

- ◆ Liberdade para alterar as características do medicamento prescrito para cada paciente, sem comprometer o padrão do banco de dados de medicamentos, e sem comprometer as características prescritas do mesmo medicamento para o mesmo paciente, em outras datas.
- ◆ Recurso para agrupar um conjunto de medicamentos como padrão de prescrição.
- ◆ Recurso para armazenar e imprimir códigos. Os códigos podem ser utilizados como contra-referência para evitar que sejam adquiridos medicamentos errados, que podem ser problema quando os nomes comerciais dos medicamentos são muito semelhantes.
- ◆ Recurso para determinar a ordem em que os medicamentos serão impressos na receita.
- ◆ Recurso para alterar a fórmula magistral prescrita ao paciente, sem comprometer a fórmula padrão.

The screenshot displays the medication prescription interface. On the left, the 'Medicamento' section includes a dropdown for 'NOME COMERCIAL' (SCHERING), a text field for 'NOME COMERCIAL' (lina), a text field for 'SUBSTÂNCIA ATIVA' (de oximetazolina), a text field for 'CATEGORIA GERAL', and input fields for 'Qtd a comprar' (1), 'Embalagem' (Frasco), and '20ml'. Below these are fields for 'Verbo de uso' (Aplicar), 'Periodicidade' (2 gotas em cada narina de 12 em 12 hs), and 'Explicação ou complemento' (por no máximo 3 dias). The 'Medicamentos Prescritos' section shows a list with 'AFRIN PEDIÁTRICO' selected. On the right, the 'Na Prescrição' section contains buttons for 'Adicionar medicamento', 'Imprimir a prescrição', 'Copiar para padrão', and 'Sair'. Below this is the 'Dados do Medicamento' section, which includes a checkbox for 'Não imprimir', a numeric input for 'Ordem de impressão na prescrição' (0), a dropdown for 'Via de Uso' (Tópico), a dropdown for 'Apresentação' (Gota), and buttons for 'Ver fórmula magistral', 'Excluir da prescrição', and 'Imprimir o medicamento'. There are also fields for 'uso para ...' (Criança), 'concentração' (0,025%), and a dropdown for 'Medicamen'.

Figura 12: Tela do módulo de prescrição de medicamentos do Tot-Win.

Banco de imagens

Arquivo com acervo de imagens para explanação aos pacientes durante o atendimento médico, com possibilidade de inserção de textos explicativos e recurso de impressão. Este módulo pode ser acessado através do menu principal ou através do painel da tela de evolução clínica.

Banco de seqüências de vídeos

Semelhante ao banco de imagens, destina-se à explanação aos pacientes, porém, neste caso, utiliza-se de seqüências de vídeos digitalizados e inseridos pelo próprio usuário para explicações de tópicos que necessitam de abordagens dinâmicas, como por exemplo, em ortopedia ou fisioterapia, quando se deseja explicar ao paciente como fazer um determinado exercício.

Banco de orientação médica

Módulo para armazenamento de orientações médicas para os pacientes, que podem receber um nome, palavras chaves e serem classificadas. A utilização deste módulo durante o atendimento médico permite fornecer ao paciente um maior conjunto de orientações escritas e personalizadas, em relação à sua doença, tratamento e cuidados adicionais, sem comprometer o tempo de atendimento do médico.

Banco de medicamentos

Arquivo com cadastro dos medicamentos^[8, 82] que serão utilizados para a prescrição e para o módulo de verificação de interação medicamentosa. Podem ser cadastrados aspectos importantes para automatização do processo de prescrição, bem como informações adicionais, como dados de indicação, farmacodinâmica, cuidados especiais, etc.

CID (Código Internacional de Doenças)

Banco de dados de apoio para cadastramento do Código Internacional de Doenças. Possui campo para informar também as características clínicas da doença.

SNOMed (Systematized Nomenclature of Medicine)

Banco de dados de apoio, semelhantemente ao CID, e utilizado para cadastramento de descrições. Esta codificação é muito utilizada pela anatomia patológica.

Doenças e síndromes

Módulo para cadastramento de informações clínicas, referências bibliográficas e palavras chaves para as doenças e síndromes de interesse^[28].

Crítérios TNM de estadiamento tumoral

Módulos para cadastramento dos critérios de estadiamento tumoral segundo TNM. As informações gerenciadas por estes módulos são utilizadas pela tela de evolução clínica^[47].

Interação medicamentosa

Este módulo que permite incluir e verificar a existência de interações entre dois medicamentos que estejam no cadastro de medicamentos (figura 13).

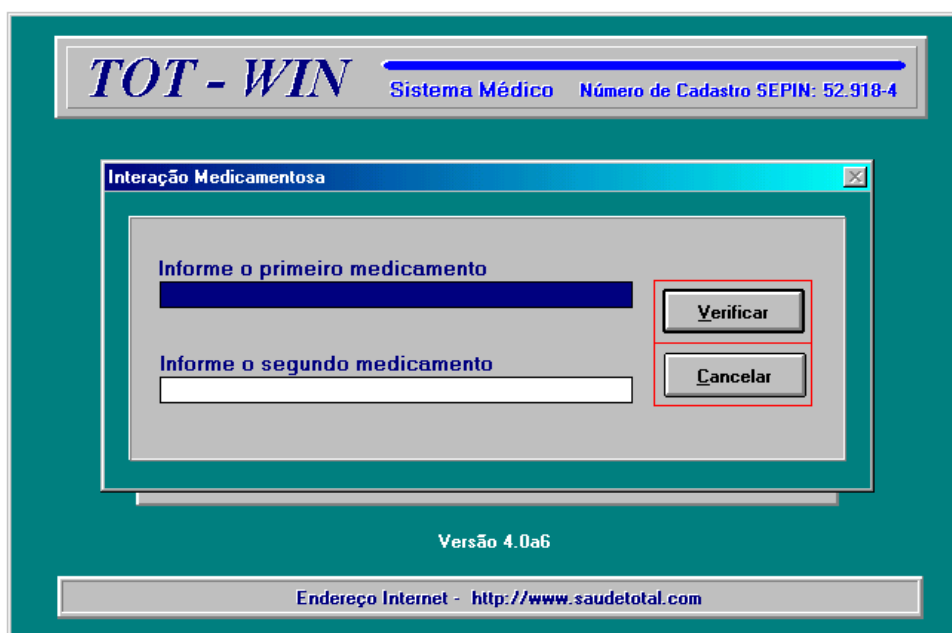


Figura 13: Tela do módulo de Interação Medicamentosa, para verificar a interação entre dois medicamentos.

Memento

Banco de dados para cadastramento dos medicamentos comerciais com as suas bulas, monografias e fotos dos medicamentos. É uma base de dados que se assemelha ao Dicionário de Especialidades Farmacêuticas (DEF) [72].

Emissão de laudos

Este módulo permite a elaboração de laudos de exames usando de modelos padrões para agilização e possibilita intercâmbio de dados com processadores de texto através do recurso de OLE do *Windows*.

Recursos para finalidades administrativas e financeiras

- Cadastro dos médicos: para cadastramento dos médicos da clínica, com informações pessoais e definições das características individuais (períodos de atendimento de acordo com o dia da semana, horários específicos, etc.).
- Tabela de honorários: arquivo para cadastramento dos procedimentos, os preços e suas características, para utilização nos módulos de Contas Hospitalares e Ficha Contábil do paciente (durante a fase de agendamento e atendimento), para gerenciamento dos procedimentos realizados, preparação das contas médicas e controle das cobranças e fechamento financeiro.
- Cadastro de endereços: tabela utilizada para cadastrar endereços de colegas de outras especialidades, laboratórios, clínicas radiológicas, etc., para fins de encaminhamento de pacientes.
- Agendamento de consulta: módulo para marcações de atendimento médico. Características do agendamento:
 - ◆ Integração com ficha contábil para agendamento e lançamento dos procedimentos realizados. A própria ficha contábil é o item da agenda. Os campos de procedimentos desta ficha possuem comando Ponto e estão integrados à tabela de honorários.

- ◆ Gerenciamento dos aspectos financeiros a partir de ficha contábil. Esta característica mantém a integridade física das informações.
 - ◆ Capacidade de compartilhamento da mesma agenda por vários usuários simultaneamente.
 - ◆ Capacidade para pré-definição de horários de atendimento.
 - ◆ Recurso para verificar horários vagos.
 - ◆ Acesso a dados cadastrais do paciente e recurso para impressão de etiquetas identificação e fichas de atendimento.
 - ◆ Sistema de segurança que impede que funcionários da recepção acessem dados clínicos dos pacientes.
 - ◆ Recurso para procurar a data e hora de atendimento do paciente, informando somente o nome.
 - ◆ Recurso para acessar a agenda de determinada data, mas permite alternativamente que a agenda de um determinado dia seja localizada informando-se o intervalo de tempo em número de dias, semanas ou meses (por exemplo, solicitando-se ao sistema abrir a agenda do dia que corresponde a 45 dias a partir de hoje).
 - ◆ Recurso para fechamento financeiro do dia.
 - ◆ Recurso para imprimir a relação de pacientes atendidos, separados por período e por convênio.
 - ◆ Dados da agenda integrados ao módulo de estatísticas gerenciais.
- Contas hospitalares: módulo para preparação de contas de pacientes, com lançamento de materiais, medicamentos e taxas. Os materiais e medicamentos são obtidos a partir do cadastro de estoque. É um módulo projetado para ambientes que realizam atos médicos mais complexos ou cirurgias ambulatoriais.

- Fornecedores: para cadastramento de fornecedores dos produtos constantes no estoque. Importante para otimização dos processos de compra com cotações.
- Módulos de estoque: para cadastro dos produtos e controle das quantidades disponíveis, mínima e máxima, data e valor de última compra, local de armazenamento, etc. Os módulos de Entrada e Saída são destinados ao lançamento de notas de entrada e pedidos de saída, com respectiva alteração do estoque.
- Convênios: cadastramento das informações de cada convênio e suas particularidades administrativas em relação aos associados.
- Notas de despesas: para lançamento e controle de notas e recibos com valor contábil-financeiro.
- Módulo para processamento de estatísticas gerenciais: utilizando as fichas contábeis para emitir relatórios sintéticos em relação aos atendimentos médicos realizados durante um determinado período.

Recursos para finalidades científicas e didáticas

- Slides clínicos: para cadastro e recuperação de slides clínicos, com recursos para inclusão de descrição, palavras chaves, diagnósticos, imagem digital, etc. (figura 14)

Slides Número Pasta Data / / Nro Prontuário

Dados Pessoais

NOME
Endereço
Cidade Estado CEP -
Telefone

DIAGNÓSTICOS

1.
2.
3.

Outros Dados

Região Anatômica
Palavra Chave
Emprestado a... Empréstimo / / Devolução / /

▼ ▲
▼ ▲
Editar Slide
Procurar Slide
Imprimir Slide
Descrição
Incluir Captar
Sair

Figura 14: Tela do módulo para gerenciamento de Slides Clínicos.

- Protocolo Simplificado do Melanoma: protocolo clínico para lançamento de informações de pacientes com diagnóstico de Melanoma, desenvolvido em conjunto com o (GBM)^[33], um grupo multidisciplinar envolvendo dermatologistas, oncologistas, especialistas em anatomia patológica, cirurgiões plásticos, entre outros. É um protocolo de distribuição nacional para constituição de uma base para análise das características desta doença (figura 15). Atualmente (abril de 2000) o banco de dados resultante deste protocolo já acumulou 1600 casos e já foram feitos estudos comparando as características da doença de acordo com a ocorrência do diagnóstico feito em ambiente de consultório, hospitalar ou em hospitais de referência.

The screenshot shows a web-based form for melanoma diagnosis support. The form is organized into several sections:

- Patient Information:** Fields for Name (initials), Birth date, Age, Sex, Color, City, State, and Time of disease (Years and Months).
- Hospital/Clinic/Doctor:** Fields for Hospital/Clinic/Doctor name, Date, and Registro.
- Address:** Fields for Endereço, Cidade, Estado, Cep, and Telefone.
- Anatomical and Clinical Data:** Fields for Local Anatômico da lesão primária (1 and 2), Nível Clark, Espessura Breslow (0,00), and Tipo histopatológico.
- Laboratory and Exam:** Fields for Laboratório, Data, and Nro Exame.
- Location and Contact:** Fields for Cidade, Estado, CEP, and Telefone.
- Buttons:** A vertical sidebar on the right contains buttons for 'Procurar', 'Incluir Ficha', 'Imprimir', 'Transferir', 'Alterar', 'Etiquetas', and 'Enviar p/ GBM'. At the bottom, there are buttons for 'Tumor Primário', 'Linfonodo', 'Metas. Dist.', 'Loc. Metas.', and 'Estadio Não Class', along with a 'Voltar' button.

Figura 15: Aspecto de uma das telas do módulo para o Protocolo Simplificado do Melanoma.

- Apoio ao diagnóstico do Melanoma: roteiro desenvolvido a partir das informações fornecidas por membros do GBM e contém os seguintes tópicos (há uma versão embutida no *Tot-Win* e outra disponível através da *Web* ^[3]):
 - ◆ Introdução e epidemiologia
 - ◆ Fatores de risco
 - ◆ Tipos clínicos
 - ◆ ABCD clínicos
 - ◆ Dermatoscopia
 - ◆ Diagnóstico diferencial
 - ◆ Biópsia frente à lesão suspeita
 - ◆ Correlação clínico-patológica
 - ◆ Estadiamento clínico-patológico
 - ◆ Condutas
 - ◆ Seguimentos

- Jornadas Informatizadas: é um banco de dados que já conta com um conjunto de 250 casos clínicos apresentados em jornadas dermatológicas a partir de 1996, com história clínica, imagens de lesões clínicas e histopatologia, discussões em platéia e referências bibliográficas. Os casos podem ser localizados pela doença, autor ou instituição (figura 16) [61].



Figura 16: Tela de um caso clínico da Jornada Informatizada promovida pela Sociedade Brasileira de Dermatologia - Regional São Paulo.

Aspectos de segurança de acesso

- Com objetivo de oferecer segurança contra acesso do sistema por pessoas não autorizadas, no *Tot-Win* foram criado 5 níveis diferentes de restrição, de 1 a 5. Os dois primeiros níveis são para a equipe administrativa e não permite o acesso a informações clínicas. Os níveis 3 e 4 são destinados à área clínica e o nível 5, com total liberdade para gerenciamento global do sistema.

Elaboração de banco de dados de apoio (banco de imagens, orientações médicas, etc.)

- Foram elaborados vários bancos de imagens, entre eles um conjunto com 450 imagens digitalizadas, entre fotografias clínicas e esquemas, para explanação ao paciente, em dermatologia. Estão relacionadas várias seqüências de procedimentos cirúrgicos.
- Cadastramento de 300 doenças no módulo de Doenças e Síndromes.
- Cadastro de 2.000 medicamentos para uso na prescrição médica.
- Cadastramento do CID-10.
- Cadastramento da Lista de procedimentos AMB.
- Cadastramento de 150 medicamentos para uso em Dermatologia no módulo de Memento. Foram incluídas fotografias dos medicamentos, bulário, e outras informações importantes.

Desenvolvimento de uma área de apoio ao usuário do *Tot-Win* através da *Web*

Foi desenvolvida uma área na *Web* (figura 17) com as dúvidas mais freqüentes, surgidas com o uso do programa, e dúvidas sobre a integração do uso do programa de acordo com o método operacional diário. É um recurso importante, principalmente para usuários de outros estados. Além das páginas estáticas, foram incorporadas listas de discussão baseada em *Web* e apoio através do e-mail para esclarecimentos personalizados e envio de arquivos de atualização ou personalização.

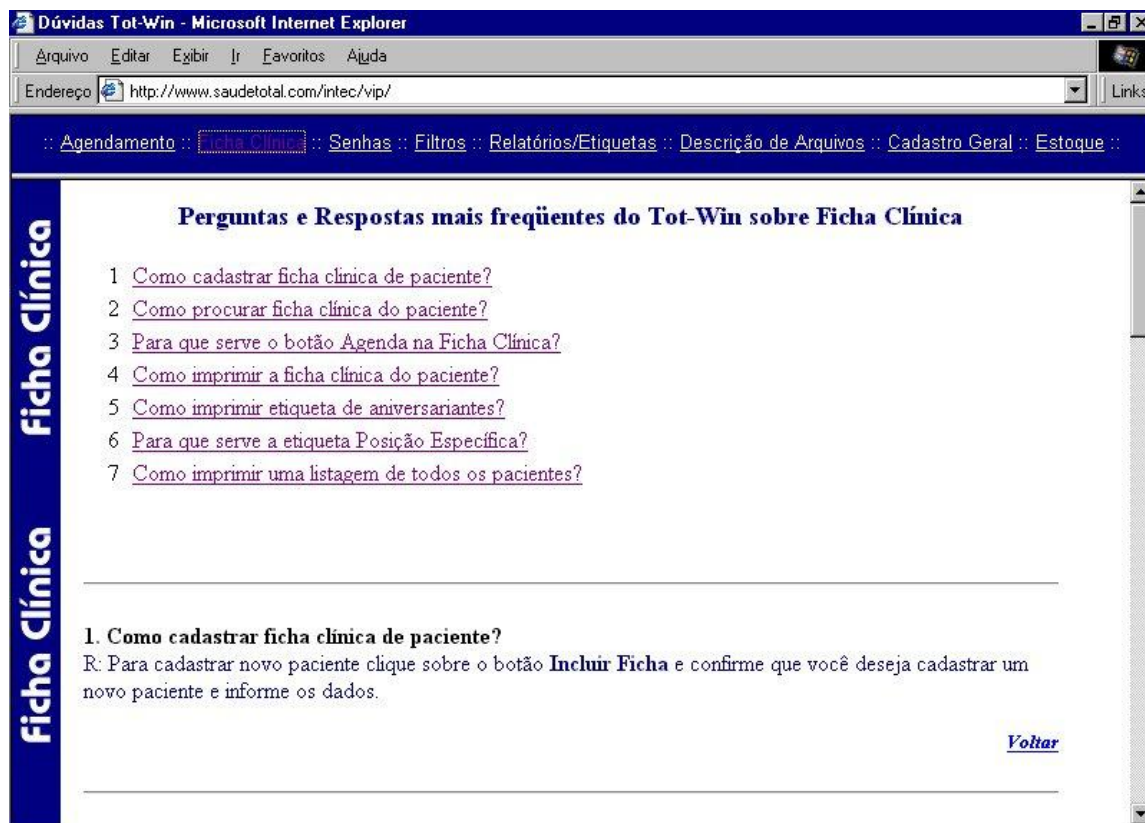


Figura 17: Aspecto de parte do sistema ligado à lista de discussão disponível através da Web^[10].

Discussão

O desenvolvimento de um sistema computacional e a sua efetiva implantação num consultório ou clínica depende de vários fatores que ultrapassam os aspectos técnicos, e necessita da participação da organização como um todo. O sucesso e a aceitação do programa dependem da capacidade de se identificar corretamente os serviços e as informações necessárias para os usuários e da forma como o homem e a máquina interagem. Uma dissociação entre as rotinas da clínica e as concebidas pelo sistema gera grandes inconvenientes que poderão comprometer a eficiência do seu uso.

É provável não se consiga antecipar todas as necessidades de todos os usuários durante o desenvolvimento de sistemas. A estruturação de um método que possibilite reunir críticas e sugestões dos usuários para a correção de erros e o aprimoramento continuado do *software* é ponto crucial para garantir que um sistema esteja sempre adequado às necessidades de uma organização. A informatização deve ser conduzida de forma criteriosa, evitando a dissociação entre a informática e as rotinas operacionais.

Alguns fatores relacionados com o insucesso da informatização

A incorporação de uma nova ferramenta de trabalho numa profissão provoca uma fase de transição, quando os novos métodos e instrumentos são paulatinamente incluídos nas rotinas de trabalho do dia a dia, e gradativamente estes novos elementos são comparados com os métodos convencionais quanto às vantagens e desvantagens decorrentes da sua implementação. É oportuno lembrar que o emprego de técnicas avançadas ou modernas não significa obrigatoriamente melhoria da qualidade do trabalho resultante; outros fatores associados, como o método de trabalho, a capacitação dos recursos humanos para uso eficiente do *software*, e a implementação de estratégias de segurança e contingência, são também fundamentais.

Alguns fatores relacionados com insucesso de uma informatização são:

- Insuficiência de tempo dedicado a estudar e analisar as rotinas operacionais existentes, a identificar os problemas metodológicos passíveis de mudanças, e a elaborar soluções que levem à otimização dos processos.
- Dimensionamento insuficiente de recursos para implementar os novos serviços, gerando pontos de “estrangulamento operacional”. Por exemplo, suponhamos um projeto para informatização da recepção/secretaria de uma clínica com 3 recepcionistas para atendimento aos pacientes, consultórios e médicos, sendo que cada médico tivesse sua própria escala de horário de atendimento; em uma rotina convencional cada médico teria sua própria agenda e a marcação dos atendimentos seria feita em agendas separadas.

Um projeto de informatização de uma clínica com definição insuficiente de recursos poderia prever apenas um único equipamento para a recepção, considerando que as recepcionistas teriam outras atividades, como o preenchimento manual de guias de convênios, a separação dos prontuários médicos, etc., e um computador, devido à velocidade de processamento, seria suficiente para atender às necessidades operacionais.

Neste caso, porém, o computador seria o ponto de “estrangulamento operacional”, pois independentemente da eficiência do *software* ou dos usuários, o resultado final será inferior ao convencional: caso uma recepcionista estivesse agendando o retorno de um paciente no computador e uma outra precisasse também usar o computador para marcar consulta de um outro paciente ao telefone, estes dois eventos não poderiam ocorrer simultaneamente. A limitação não está relacionada com velocidade de processamento das

máquinas, mas com a quantidade disponível para uso. Este problema não ocorreria no método convencional uma vez que os médicos teriam agendas separadas.

A definição mínima deveria ser de 2 equipamentos conectados em rede local, compartilhando o mesmo banco de dados; isto possibilitaria o atendimento simultâneo de pacientes com a vantagem de se poder compartilhar a mesma agenda. A ausência da rede local, neste exemplo, seria considerada especificação insuficiente.

- Enfoque valorizando principalmente os equipamentos e os *softwares*, deixando em plano secundário a capacitação dos recursos humanos da clínica (treinamento); por melhor que sejam os recursos, a incapacidade de utilizá-los adequadamente diminui muito os benefícios, e pode causar grandes prejuízos funcionais.
- Falta de avaliação sobre a qualidade e a confiabilidade dos dados inseridos no sistema. Por exemplo, suponhamos a digitação do protocolo simplificado de Melanoma por uma escriturária: será que a qualidade de serviço gerado atenderia às necessidades de confiabilidade que os dados exigem? Será que não seria melhor ter pelo menos um médico para revisar os dados inseridos?
- Falta de integração dos bancos de dados, causando necessidade de redigitação (duplicação de trabalho). As novas digitações podem introduzir erros além de aumentar o trabalho em decorrência da nova digitação e conferência.
- Análise e armazenamento dos dados de forma não padronizada, impedindo a recuperação e análise correta das informações. A falta de padronização de termos na descrição diagnóstica, com o emprego de sinônimos ou termos equivalentes, poderia dificultar a recuperação precisa, causando perdas em levantamentos posteriores.

- Falta de envolvimento dos usuários como co-participantes e co-responsáveis pelo desenvolvimento, personalização e aprimoramento do sistema.
- Falta de estratégia de segurança dos dados: cópia e segurança contra acesso.
- Falta de análise do método de contingência em relação às eventuais falhas do sistema: suponhamos um ambiente que disponha de um servidor de rede local e microcomputadores clientes. É muito raro uma clínica adquirir dois equipamentos servidores para que um deles seja reserva em caso de defeito no equipamento principal; caso o servidor tenha defeito e seja enviado para a manutenção técnica, é necessário que a clínica disponha de uma alternativa para não comprometer todo o processo de atendimento aos pacientes. Este tópico será discutido em “Qualidade de infra-estrutura física, rede local e estabelecimento de processos de contingência”.

Informatização eficiente

A informatização eficiente, além dos aspectos de *hardware* e *software*, depende de vários fatores, mas pode ser definida como sendo a efetiva integração da realidade diária da clínica com os recursos computacionais.

Uma informatização, por mais elementar que seja, poderá ser considerada bem sucedida quando ela melhora os processos da entidade em relação à situação anterior ao emprego da informática, independentemente do grau de automação obtido.

Um *software* essencialmente “processa” eventos, e como consequência processa dados. O *software* não permanece enclausurado num ambiente isolado, mas para o seu funcionamento correto, depende primordialmente da interação adequada com o meio que o cerca, como equipamentos, sistemas operacionais, usuários, etc. Um *software* deve proporcionar automação das informações e agilizar os processos da organização.

Fatores técnicos

- Qualidade do projeto: um projeto bem desenvolvido é aquele que consegue identificar os objetivos da organização que motivaram ao uso do sistema, analisar os métodos onde o programa será implantado identificando os pontos de atrito e incompatibilidade na interação *software* e a dinâmica diária, identificar as necessidades administrativas e gerenciais da instituição e as angústias de cada setor envolvido.
- Qualidade do programa computacional desenvolvido: envolve o uso da HCI adequada de acordo com o perfil do usuário final e o desempenho de funcionamento do sistema, facilidade de uso e rotinas que garantam a confiabilidade dos dados inseridos (automação de alguns dados e implementação de algumas “armadilhas para detecção de erros”).
- Qualidade da modelagem de dados: a modelagem deve sempre que possível levar em consideração a possibilidade de futuras implementações da organização. A capacidade de reconhecer antecipadamente estas necessidades dependerá em grande parte da experiência e conhecimento da equipe de desenvolvimento em relação às necessidades da organização. A modelagem deve levar em consideração aspectos como:
 - ◆ aquisição dos dados;
 - ◆ arquivamento dos registros de dados;
 - ◆ padronização dos dados;
 - ◆ comunicação e integração;
 - ◆ armazenamento e recuperação de informações;
 - ◆ análise de dados para levantamentos.
- Qualidade de integração das informações: a estruturação e integração de dados com as funções do sistema podem ser comparadas a uma linha de montagem. À medida que uma mesma

informação vai passando pelos diversos setores, torna-se cada vez mais confiável.

Por exemplo, a partir da ficha contábil do *Tot-Win*, o programa permite que a recepção marque a data, a hora e o médico responsável pelo atendimento do paciente; o médico insere os procedimentos realizados, utilizando a tabela de procedimentos para a padronização dos atos e valores; a área de cobranças verifica os procedimentos inseridos e lança os pagamentos; e a área administrativa emite o demonstrativo financeiro diário e gera estatísticas gerenciais no fim de cada mês. A ficha contábil é utilizada pelas diversas áreas, à medida que o processo de atendimento ao paciente ocorre. Novas informações são inseridas de forma progressiva, podendo ser consultadas por outros setores. Este método de trabalho permite que os dados sejam revisados pelas áreas pertinentes, e em consequência, as estatísticas mensais sejam baseadas em dados confiáveis, conferidos diariamente pelo fechamento financeiro diário.

Fatores humanos envolvidos

- Adequada definição do perfil de conhecimento por parte de cada usuário, de acordo com setor e atividade a ser desenvolvida.
- Capacitação dos recursos humanos, isto é, treinamento dos usuários de modo a obter o domínio sobre o sistema, reduzindo as inseguranças durante a utilização.
- Suporte de esclarecimento de dúvidas ao usuário: é uma atividade que diminui progressivamente ao longo do tempo, à medida que os usuários se acostumam com o sistema, embora seja de extrema importância nas fases iniciais de qualquer implantação.
- Treinamento permanente e continuado: a importância cresce na medida que existe maior grau de rotatividade de funcionários na instituição.

- Desenvolvimentos, adequações e aprimoramentos do sistema com co-participação dos usuários. Quanto mais os usuários se sentirem como parte do processo de desenvolvimento, mais ficarão comprometidos com o sucesso do conjunto (motivação dos usuários).
- Integração dos sistemas computadorizados com as rotinas operacionais da organização. O sistema não deve estar dissociado da rotina de trabalho, e deve ser visto como fator de agilização dos processos.

Fatores organizacionais envolvidos

- Avaliação de como o sistema pode ser financiado.
- Designação de um responsável para verificar a qualidade, a confiabilidade e a precisão dos dados inseridos.
- Estruturação de cuidados com a segurança dos dados e contra outros danos, como roubos, incêndios, etc.

Desvantagens dos sistemas computadorizados

A informatização deve levar em conta algumas desvantagens dos sistemas computadorizados:

- investimento inicial: é recuperável ao longo do tempo em decorrência da agilização dos processos e eliminação das duplicações de trabalho. Um parâmetro que pode ser utilizado para avaliar se uma informatização foi válida é acompanhar se todo o valor investido é recuperado em um ano, em decorrência da otimização dos processos, pela redução dos custos funcionais, ou aumento da demanda na clínica em decorrência da melhoria da qualidade dos serviços oferecidos aos pacientes;
- tempo despendido para treinamento do pessoal da organização, consumindo parte do tempo produtivo;
- possível não adaptação de funcionários;

- rotatividade de funcionários que pode gerar uma descontinuidade na capacidade para utilização do sistema; novos funcionários precisam ser previamente treinados;
- quebra de equipamentos ou outras falhas podem causar interrupção do funcionamento do sistema.

Qualidade da infra-estrutura física, rede local e estabelecimento de processos de contingência

O uso de computadores na prática diária deve ser feito criteriosamente, com estratégias bem definidas para o *hardware*, o *software* e o treinamento.

O computador, apesar de oferecer benefícios inquestionáveis, é sujeito a falhas dos mais diversos tipos, que podem gerar como consequência, a paralisação do sistema e mesmo a perda dos dados. Estes aspectos devem ser analisados com cuidado uma vez que a dinâmica funcional de uma clínica informatizada não pode ser interrompida em decorrência dos sistemas computacionais.

Hardware

A escolha dos equipamentos para a informatização deve ser feita criteriosamente, uma vez que microcomputadores e periféricos sem a qualidade adequada podem, freqüentemente, parar de funcionar devido a falhas, prejudicando a informatização e causando perda da confiabilidade dos usuários no sistema. A escolha pode se basear na aquisição de equipamentos de marcas conhecidas e que já foram avaliadas por laboratórios que se dedicam a realizar testes em relação à sua “robustez”, de marcas que foram homologadas em grandes corporações e de fornecedores de equipamentos que disponham de métodos para submetê-los a situações críticas de funcionamento antes de os entregarem aos clientes (por exemplo, submeter os equipamentos a funcionamento ininterrupto durante semanas, executando programas que necessitem de velocidade de processamento e que façam uso de grande quantidade de memória RAM, entre outros).

A escolha de alguns periféricos, tais como impressoras, deve levar em consideração a velocidade e a qualidade de impressão, o tamanho físico do equipamento, o custo dos cartuchos de tinta ou *toner*, facilidade de uso, entre outros.

Infraestrutura ambiental

Fatores ambientais são aspectos que podem causar grandes prejuízos quando negligenciados, mesmo que se adquiram equipamentos de boa qualidade. Alguns aspectos que precisam ser levados em consideração:

- qualidade de rede elétrica: caso esteja sujeita a grandes oscilações, ou se a rede elétrica dos computadores é a mesma utilizada por outros equipamentos que consomem grande quantidade de energia, como freezer, ar condicionado, etc.;
- disponibilidade de fio terra eficiente nas tomadas elétricas;
- disponibilidade de estabilizadores de tensão (para casos de flutuações de tensão elétrica) ou *no-breaks* (para casos de interrupções abruptas de fornecimento de energia elétrica);
- temperatura ambiental adequada aos equipamentos;
- umidade: os computadores podem apresentar falhas mais freqüentes em ambientes úmidos;
- ambiente sem muita interferência eletromagnética;
- instalação dos cabos da rede lógica de computadores de forma adequada, longe de reatores de lâmpadas fluorescentes ou fios elétricos de alta tensão que causem indução magnética;
- cargas eletrostáticas: particularmente os ambientes que usem ar condicionado, tenham piso forrado com carpete, utilizem cadeiras com rodinhas e seja comum o uso de sapatos com sola de couro, estão mais sujeitos a gerar cargas eletrostáticas, que podem até queimar os equipamentos;
- evitar incidência direta dos raios solares sobre os equipamentos;

- ventilação adequada dos equipamentos: é importante que os microcomputadores sejam instalados em locais de fácil circulação para o ar, caso contrário, isto pode contribuir para o superaquecimento dos mesmos.

Rede local

A informatização dos consultórios para automação do processo de atendimento ao paciente ocorreu principalmente após o barateamento dos microcomputadores, o uso cada vez mais freqüente de ambientes para múltiplos usuários e a simplificação para a instalação de redes locais. Atualmente é cada vez menos freqüente a existência de consultórios que não tenham seus equipamentos interligados em rede local.

A redes locais são basicamente de dois tipos:

- Unidade com servidor central de onde os computadores clientes obtêm dados. Esta arquitetura é a recomendada para ambientes que necessitam de maior eficiência e que desejam conectar maior número de equipamentos.
- Ponto a ponto, onde cada microcomputador conectado pode funcionar como um servidor de dados ou compartilhar recursos (impressora, leitor de CD-ROM, etc.).

São comuns as clínicas/consultórios não disporem de equipamentos de reserva para serem utilizados em caso de falhas. Isto ocorre principalmente com o equipamento servidor de rede, devido ao seu valor mais alto.

Processo de contingência em caso de falha do servidor de rede

A realização de cópias eficientes confere segurança contra perda de dados, mas não oferece proteção contra interrupção do funcionamento do sistema. A estruturação de processos de contingência pode garantir a continuidade das atividades do consultório e da clínica em situações de emergência, embora especificações adequadas dos equipamentos utilizados para a informatização visem a minimizar a ocorrência desses colapsos.

Para locais que utilizam rede local baseada em servidor central, a falha neste equipamento pode representar interrupção de todo o sistema até a chegada de um técnico para reparos, principalmente quando o computador precisa ser levado para laboratório de assistência técnica e não se dispõe de equipamento de reserva.

Várias versões do *Windows*, a partir da versão 3.11 *for Workgroups*, além de serem ambientes gráficos, dispõem de recursos para rede ponto a ponto, portanto, os equipamentos clientes podem estar conectados entre si, formando uma rede lógica de comunicação. Mesmo numa estrutura com rede baseada em servidor central, é possível utilizar o *Windows* para criar uma segunda rede lógica de segurança, utilizando a mesma estrutura física. Neste caso, pode-se escolher um equipamento cliente, com maior velocidade de processamento e maior capacidade de disco rígido, para copiar periodicamente todo o sistema do servidor central. Ter-se-ia, assim, um espelhamento das funções que garantem a continuidade do trabalho executado. Em caso de falha do servidor, os outros microcomputadores seriam direcionados para executar o sistema a partir daquele equipamento, preparado para assumir função de um servidor provisório, evitando a interrupção do trabalho por períodos muito longos. Após o restabelecimento do servidor central, os dados seriam atualizados a partir do servidor provisório.

Contingência de cópia de segurança de dados

A realização constante de *backups* garante segurança contra perda de dados, porém existem fatores externos que podem causar prejuízos irremediáveis, como furtos, incêndios, entre outros. Nestes casos, o armazenamento de todas as cópias no mesmo local físico pode não ser a melhor estratégia. Seria importante fazer duas cópias do mesmo dado, na cópia quinzenal ou mensal nos esquemas 7 x 12 ou 9 x 12 (este tema será abordado com mais detalhes na página 112), e guardar um deles na clínica e outro na residência..

Comportamento evolutivo de um *software*

O *hardware* tipicamente apresenta a maior parte de suas falhas nos primeiros meses de funcionamento. Depois que estes defeitos são sanados, um equipamento deve apresentar-se confiável até a deterioração de seus componentes eletrônicos, quando o índice de falhas aumentará progressivamente (figura 18).

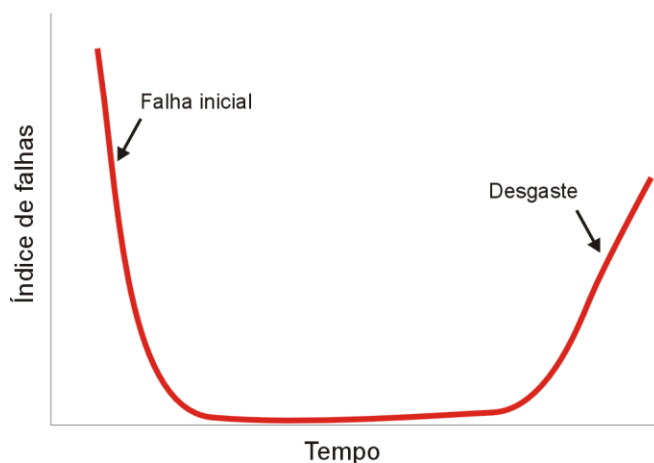


Figura 18: Comportamento do índice de falhas do *hardware* em função do tempo. Tipicamente a maior parte das falhas ocorrem nos primeiros meses de funcionamento. Um bom equipamento deve apresentar baixo índice de erros por um certo tempo, após o qual falhas relacionadas ao desgaste começam a comprometer seu trabalho.

Ao contrário, o *software* não se degrada, mas pode ter dois comportamentos possíveis:

- seguindo o traçado da curva ideal de um *software*, tende a evoluir para a obsolescência no decorrer do tempo, e em consequência, ser abandonado (figura 19);

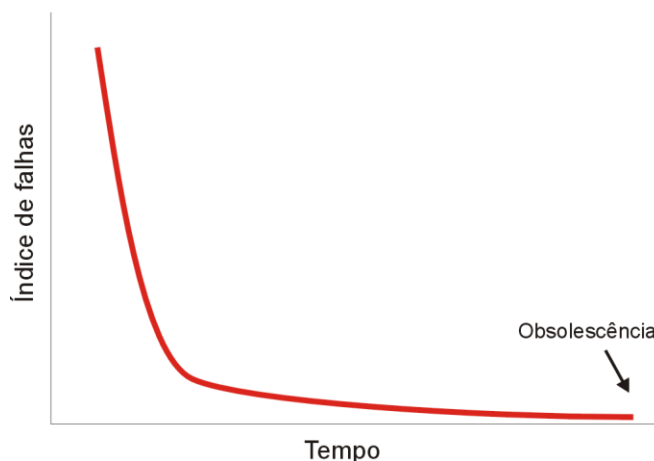


Figura 19: Curva ideal do comportamento do índice de falhas de um *software* em função do tempo. Após a fase inicial de correção da maioria dos erros, o *software* tende progressivamente a ter menos erros, finalizando a sua vida útil na obsolescência.

- deteriorar-se em decorrência de constantes mudanças (figura 20). Em geral a implementação de mudanças acarreta uma elevação do índice de erros durante uma fase. Leva um certo tempo para que as correções necessárias sejam implementadas e o índice de falhas volte a seguir o comportamento da curva ideal. Quando as mudanças começam a ser muito freqüentes, a base da curva de falhas eleva-se com conseqüente diminuição da confiabilidade no sistema. A freqüência de mudanças está diretamente relacionada com a grandeza da dissociação entre os recursos do sistema e as necessidades do usuário. Além da freqüência, quanto mais distante um projeto estiver das necessidades requisitadas, maior será o porte das modificações e maior será o tempo necessário para o desenvolvimento e as correções, com conseqüente aumento do custo. Define-se o nível crítico como sendo o limiar a partir do qual os erros causam tantas falhas que inviabiliza-se o uso do *software*. Caso os erros não sejam solucionados com rapidez suficiente para que o índice de falhas caia abaixo deste limiar, a deterioração leva ao abandono do sistema ou desencadeia um processo de re-engenharia.

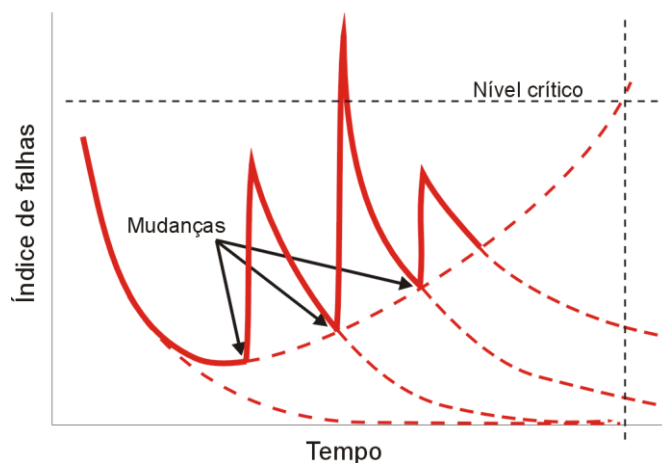


Figura 20: Comportamento real do índice de falhas do *software* em função do tempo. A implementação de mudanças acarreta elevação do índice de erros. Quando as mudanças começam a ser muito frequentes novas falhas são incorporadas, podendo levar a uma deterioração do sistema. A partir do nível crítico, quando há falhas que inviabilizam o uso do *software*, os erros precisam ser rapidamente corrigidos ou o *software* será descartado.

A finalização do desenvolvimento de um *software* e sua entrega ao usuário não devem ser consideradas como o fim dos trabalhos. É o início de uma seqüência de atividades que levarão ao amadurecimento do programa e à estruturação de uma sistemática para o aperfeiçoamento continuado.

Aspectos relacionados com qualidade de um *software*

Avaliar a qualidade de um *software* é uma atividade difícil, pois além dos aspectos objetivos existem questões que são extremamente pessoais, por exemplo, facilidade de compreensão, facilidade de uso, complexidade dos módulos, abrangência do programa, etc. Mas os critérios objetivos a seguir relacionados devem fazer parte dos critérios de avaliação:

- Capacidade do programa em executar corretamente as funções que são exigidas dele: um defeito pode ser definido como uma dissociação entre os requisitos do projeto e o que o sistema efetivamente faz.
- Confiabilidade: é uma medida obtida pela avaliação da precisão resultante da execução de cada função de um programa. É importante não confundirmos com confiabilidade dos dados, que é a certeza de que os dados representam corretamente os fatos ocorridos.

- **Eficiência:** é avaliada através da quantidade de recursos de computação e de código exigida para que um programa execute sua função. Do ponto de vista prático pode ser avaliado verificando a velocidade de desempenho de um sistema numa especificação de equipamento. Exemplo: o *Windows 3.11* funciona de forma eficiente num microcomputador 486 com 16 *megabytes* de memória RAM, enquanto que o *Windows 98* funciona adequadamente num equipamento *Pentium II* com 64 *megabytes* de memória RAM.
- **Segurança:** é o aspecto relacionado com a capacidade do *software* de impedir o uso do programa ou acesso aos dados por pessoas não autorizadas.
- **Facilidade de uso:** é avaliada verificando-se o esforço para aprender, operar, preparar a entrada e interpretar a saída (resultados do processamento) de um programa.
- **Facilidade de manutenção:** é mensurada verificando-se o grau de esforço exigido para localizar e reparar erros num programa.
- **Flexibilidade:** pode ser avaliada analisando-se o esforço exigido para modificar um programa.
- **Facilidade de realizar testes sistemáticos:** é o trabalho necessário para testar um programa a fim de garantir que ele execute as suas funções segundo as especificações projetadas.
- **Portabilidade:** é o esforço necessário para transferir o programa de um ambiente de *hardware* e/ou sistema operacional para outro.
- **Possibilidade de reutilização:** é a avaliação do grau com que um programa (ou partes de um programa) pode ser reutilizado em outras aplicações.
- **Interoperabilidade:** é a avaliação do esforço necessário para se acoplar um sistema a outro.

Desenvolvimento do *Tot-Clin* e *Tot-Win*

A linguagem *Clipper'87 Summer* foi utilizada para o desenvolvimento do *Tot-Clin* num período em que os programas para microcomputadores tinham que ser pequenos, em decorrência da disponibilidade de memória

convencional imposta pelo sistema operacional *MS-DOS* (640 *quilobytes* de memória RAM, subtraídos todos os outros programas residentes nesta memória). Os programadores precisavam reduzir o tamanho dos programas executáveis (EXE) ao máximo, e lançar mão de *overlay* (módulos diferentes do programa que são chamados a partir do disco rígido para compartilhar a mesma área da memória RAM do equipamento à medida que são necessários). A linguagem *Clipper'87 Summer* foi uma ferramenta de desenvolvimento adequada para a época, mas tinha limitações técnicas, como, por exemplo, o tamanho do arquivo de armazenamento do texto livre (DBT) restrito a 16 *megabytes* (16 milhões de caracteres). Um problema inerente à linguagem era a ruptura do vínculo entre a ficha cadastral (DBF) com o correspondente arquivo de texto, quando se apagava um registro usando o comando PACK da linguagem. Quando se apagava um registro do banco de dados, os textos dos registros subseqüentes se desvinculavam, alocando o texto pertencente a um registro para outro. Isto era um problema grave, uma vez que tornava o sistema pouco confiável, justamente no aspecto que o programa tinha de maior flexibilidade no contexto assistencial, que era o armazenamento de dados clínicos em formato de texto livre com possibilidade de recuperação e levantamento.

Foram necessários vários estudos, simulações e testes durante 3 meses para criar alternativas que fornecessem consistência ao sistema. Foi constatado nesta avaliação que caso não fosse utilizado o comando de apagar da linguagem, mas em lugar disto, fosse instruído para copiar para um novo arquivo as fichas que não estivessem marcadas para exclusão, o problema do desvínculo não ocorria.

Em processos de desenvolvimento, independentemente das limitações decorrentes da ferramenta adotada, um engenheiro de *software* precisa sempre avaliar a confiabilidade dos comandos utilizados e, se necessário, elaborar alternativas técnicas para superar problemas das ferramentas de desenvolvimento, sob pena de perder todo o projeto.

Constatamos mais tarde que esta falha de execução da linguagem *Clipper* não foi solucionada conceitualmente mesmo pelas linguagens

sucessoras, como *FoxPro*. Na verdade, elas utilizaram as mesmas idéias que projetamos, porém incorporadas no comando. Por exemplo, o comando PACK, que era causa do desvínculo, ao ser declarado nestas linguagens, passou a copiar os registros não marcados para um outro arquivo e após isto, apagava o arquivo original, e renomeava o arquivo criado com o mesmo nome do original.

A linguagem *Clipper'87 Summer* foi mantida como ferramenta de desenvolvimento até hoje, pois, as primeiras versões do *Clipper 5.0*^[29, 59] (sucessora do *Clipper'87 Summer*) tinham várias falhas técnicas (*bugs*), aumentava o tamanho dos executáveis, os programas resultantes eram mais lentos em equipamentos XT quando comparados ao *Clipper'87 Summer*, e não oferecia novos recursos que contribuíssem substancialmente para o aprimoramento funcional do *Tot-Clin*, exceto pelo aspecto do *overlay* dinâmico, mas esta característica foi implementada no *Clipper'87 Summer*, através da aquisição do *linker* dinâmico *Blinker 1.50*.

Durante o desenvolvimento do *Tot-Clin*, concomitantemente ao *Clipper*, foi lançado o *FoxPro for DOS* que dispunha também de versões para *Xenix* e *Macintosh*, o que era uma vantagem sob o ponto de vista multiplataforma. Nesta época (1991), o *Xenix*, um sistema operacional multiusuário *Unix* para microcomputadores, estava se difundido na área empresarial.

Não foi adotado o *FoxPro for DOS* na ocasião, pois ele inicialmente não tinha capacidade para gerar programas executáveis, e em uma segunda fase, quando passou a gerá-los, estes eram significativamente maiores que os gerados pelo *Clipper'87 Summer*. Mais tarde, após a venda da empresa *Fox* para *Microsoft* foi lançado o *FoxPro 2.5 for Windows*. Paralelamente a isto, o *Windows* consolidou-se como padrão de *interface* gráfica, e o lançamento do *Windows 3.11 for Workgroups* estabeleceu uma alternativa de baixo custo para implementar redes locais.

As implantações, dirigidas a objetivos bem definidos, com acompanhamento *in loco* proporcionam ambientes menos complexos de simulação para detecção de falhas de desenvolvimento, bem como oferecem

modelos que permitem compreender melhor os processos necessários para a integração do sistema com a dinâmica operacional, como por exemplo, definição da forma de treinamento aos usuários e as mudanças necessárias na organização e método.

O *Tot-Win* é a conversão do *Tot-Clin* para ambiente *Windows*, incorporando *interface* gráfica e multimídia para agregar novos recursos às abordagens assistenciais, do qual herdou várias características. Vários aspectos foram reavaliados e aprimorados, e os bancos de dados redimensionados para abranger os novos recursos. O modelo de desenvolvimento do *Tot-Win* seguiu o mesmo padrão instituído para o *Tot-Clin* desde o seu início, que denominamos de informatização e desenvolvimento por objetivo. Este modelo será discutido mais adiante.

O desenvolvimento do *Tot-Win* foi postergado até 1994, porque, para que tivesse sucesso, seria necessário aguardar o *upgrade* do parque de equipamentos dos médicos, de XT para a plataforma de equipamentos 386 com 8 *megabytes* de memória RAM e monitor colorido com 256 cores, e plataforma computacional adequada para rodar o programa. O *Tot-Win* tinha que ser executado em microcomputadores *IBM PC* compatíveis com *Windows*. Esta escolha foi feita porque este conjunto de equipamento e sistema operacional gráfico era o mais difundido no mercado, o que garantia maior número de potenciais usuários para justificar o desenvolvimento. O *Windows*, além de oferecer recursos gráficos e de multimídia, elementos fundamentais para desenvolvimento do *Tot-Win*, tinha um dos maiores acervos de *softwares* e linguagens de programação, e na versão 3.11 passou a dispor de rede ponto a ponto.

Nesta ocasião também foi constatado que o médico já começava a considerar a importância de informatizar o consultório e utilizar o microcomputador como ferramenta de auxílio para atendimento ao paciente, devido ao barateamento dos equipamentos e conjuntos para instalação de rede local e simplificação da forma de uso dos sistemas computacionais.

A primeira versão do *Tot-Win* foi lançada em fins de 1994, baseada no *FoxPro 2.5 for Windows*, que continuou a ser a versão da linguagem utilizada até início de 2000, pelo fato dos sistemas desenvolvidos nesta versão da linguagem exigirem poucos recursos de *hardware* para sua execução: *Windows 3.1* com 4 *megabytes* de memória. A evolução para *Visual FoxPro 5.0*, e posteriormente *Visual FoxPro 6.0*, foi paulatinamente implementada à medida que a configuração básica dos equipamentos evoluiu para microcomputadores *Pentium 200 MMX* ou superior, com 32 *megabytes* de memória RAM e *Windows 95* ou *98*, exigência obrigatória a partir do *Visual FoxPro 5.0*.

Os programas desenvolvidos em ambiente dos funcionam de forma diferente dos programas em *Windows*, assim sendo, foi necessário desenvolver uma nova *interface* com o usuário. Os recursos dos programas *DOS* geralmente são acionados através de teclas de atalho (uso das teclas CTRL ou ALT em conjunto com uma letra) mas no *Windows*, estes podem ser apresentados em forma de *ícones* e acionados através do *mouse*. Para manter certo grau de compatibilidade, além de aumentar a agilidade no uso do programa, minimizando o uso do *mouse* pelos usuários mais experientes, foram criadas também teclas de atalho no *Tot-Win*, embora a maioria dos comandos seja acionada através do *mouse*. Isto teve o objetivo de incentivar uma maior adesão dos usuários do *Tot-Clin* para migrar para o *Tot-Win*.

Com a disponibilização de novas capacidades proporcionadas pelo *Windows*, vários aspectos antes não disponíveis em *DOS* foram implementados como: armazenamento de imagens e seqüência de vídeos e expansão das capacidades oferecidas através da linguagem de desenvolvimento (velocidade de processamento, expansão da capacidade de armazenamento de arquivos de texto livre para 2 *gigabytes*, etc.).

A dificuldade no desenvolvimento de *softwares* para larga distribuição reside principalmente no aspecto de identificar problemas que sejam comuns a todos, e elaborar soluções que atendam às necessidades específicas, sem aumentar demasiadamente a complexidade do sistema. Na fase de

manutenção de aprimoramento a dificuldade aumenta ainda mais, pois a incorporação de novos recursos, além dos aspectos anteriores, não pode estar destoante do restante do conjunto do *software*.

A aceitação de um *software* e conseqüente aumento do seu número de usuários, depende, além da qualidade técnica do programa, da quantidade de bancos de dados de apoio disponíveis (banco de imagens, banco de medicamentos, banco de dados de interação medicamentosa, banco de dados de CID e Lista de Procedimentos da AMB, etc.).

***Internet* como instrumento de apoio ao paciente e para pós-consulta**

O advento da *Internet* gratuita, o acesso através de conexões de alta velocidade com ADSL e TV a cabo, a grande disponibilidade de informações (Medline, Toxline, Cancerline, etc.), e a grande difusão do uso da *Internet* em várias áreas da sociedade atual, obrigam a área médica a incorporar este instrumento como uma nova ferramenta de auxílio do trabalho.

O uso da *Web*, seja através de um *site* dedicado a prestar orientações sobre saúde ou doenças, seja pelo uso do e-mail ou de uma lista de discussão baseada em *site Web* como recurso de apoio em pós-consulta, cada vez ganha mais destaque. A criação de normas e integração à rotina de trabalho é fundamental para definir as linhas gerais de um ambiente computacional na prática clínica.

O e-mail tem grande uso para intercâmbio de informações, e neste momento, a utilização dele como uma ferramenta de apoio permite oferecer suporte diferenciado em pós-consultas, para esclarecimento de dúvidas, envio de orientações adicionais, aviso de retorno à consulta, checagem se o paciente está seguindo corretamente o esquema terapêutico e outros comunicados importantes. Um aspecto importante na *Internet* é a possibilidade de integrar páginas *Web* de orientação com o e-mail ou lista de discussão. Com isto, abre-se um novo canal, que permitirá ao paciente esclarecer dúvidas sem atrapalhar a rotina de atendimento do médico. Tanto o e-mail quanto a lista de discussão podem ser respondidos pelo médico no

momento mais oportuno. Através de um *browser* o paciente pode consultar páginas de orientação que o médico tenha elaborado, ou compartilhar experiências ou dificuldades com outras pessoas que tenham a mesma doença, através de um fórum de discussão que o médico tenha organizado. Esta forma de apoio pós-consulta ganha significado nos pacientes com doenças crônicas (diabéticos, pacientes com Parkinson, hipertensos, cardiopatas, etc.), pois permite implementar aprendizado e compartilhamento de experiências entre eles. É importante que o médico seja o mediador dos debates do fórum de discussão^[62].

Modelagem de dados usando conceitos de *Data Warehouse*

O *Data Warehouse* tornou-se o centro da arquitetura das informações dos anos 90, pois provê uma plataforma sólida que integra vários dados e visa a oferecer dados históricos para utilização em análises baseadas na organização e armazenamento de dados pertinentes para o processamento informatizado e analítico com abordagem histórica ao longo do tempo. Segundo Willian H. Inmon, *Data Warehouse* é “um banco de dados orientado por assunto, integrado, não volátil e histórico, criado para suportar o processo de tomada de decisão”^[5].

Os dados dos *Data Warehouses* são provenientes dos sistemas que gerenciam as atividades operacionais e, antes de serem incorporados no *Data Warehouse*, são analisados (em relação à importância, relevância e confiabilidade) e tratados (inseridos todos os vínculos necessários com outras bases de dados).

No *Data Warehouse* todos os dados que não são importantes para o processo de DSS (Decision Support System - Sistemas de Suporte a Decisão) não são armazenados. O ambiente de *Data Warehouse* utiliza bancos de dados relacionais para armazenamento das informações, mas difere dos sistemas gerenciais pela técnica de modelagem utilizada. Os sistemas utilizados para gerenciamento dos ambientes operacionais são modelados para atender requisitos funcionais, e podem conter dados que sejam importantes ou não para a análise de DSS, porque precisam controlar

atividades do dia a dia. A técnica empregada para a modelagem de dados operacionais é a entidade/relacionamento, onde as tabelas estão vinculadas entre si (relacionamentos) baseando-se nas regras de negócio envolvido.

Os dados do *Data Warehouse* incorporam implicitamente ou explicitamente o aspecto de tempo, que permite a reconstrução histórica, e os relacionamentos entre os dados podem ser múltiplos. Todos os dados são integrados, com padronização dos termos, variáveis, estruturas de código, e assim por diante. Todos os dados no *Data Warehouse* são exatos em algum momento do tempo e não são voláteis.

Os dados são o centro dos cuidados médicos, pois são cruciais para o processo de DSS. Em última instância, pode-se inferir que todas as atividades relacionadas com cuidados médicos envolvem o processo de reunir, analisar ou usar os dados. São eles que ajudam a decisão médica. Os dados médicos cobrem uma grande gama de informações, como narrativas, dados textuais, medidas numéricas, registro de sinais e até imagens.

O foco central da informática médica é a base da informação que constitui a substância da medicina. Profissionais que trabalham na área da tecnologia de informação têm tentado esclarecer a diferença entre 3 termos que freqüentemente são usados para descrever sistemas baseados em computador^[69]:

- Dados: é um simples ponto de observação, um valor específico de um parâmetro para um objeto particular, num determinado tempo.
- Conhecimento: é resultante de uma análise (interpretação) dos dados. Isto inclui os resultados de um estudo formal e os aspectos resultantes do bom senso, hipóteses, raciocínio, ou qualquer procedimento que possa refletir a experiência ou tendência de pessoas que interpretam dados primários.
- Informação: é mais genérico, e engloba dados organizados e conhecimentos.

A grande maioria dos sistemas de informação atuais são baseados em banco de dados relacional e a modelagem segue técnicas baseadas no

modelo entidade/relacionamento, com inter-relação das tabelas através de chaves. Esta técnica visa a remover quaisquer redundâncias de dados, considerando que a ausência de redundância aumenta a eficiência de uma transação, pois bastaria modificar apenas num ponto do banco de dados para que o restante do sistema tenha os atualizados. As consistências das informações são obtidas através de uma série de recursos de gerenciamento da integridade de dados. Esta técnica permite aproveitamento mais eficiente dos espaços físicos de armazenamento, simplifica a atualização das informações e aumenta substancialmente a velocidade de processamento de transações; mas como consequência do alto grau de inter-relacionamento, as tabelas são extremamente dependentes umas das outras. A apresentação de uma seqüência completa de dados ao usuário, na verdade, é resultante de uma série de processamentos e validações internas.

Os modelos do tipo entidade/relacionamento são técnicas eficientes para modelagem e desenvolvimento de sistemas de processamento OLTP (*On-line Transaction Processing*), porém criam dificuldades para consultas e obtenção de dados, pois não podem ser entendidos facilmente pelos usuários e nem analisados e avaliados facilmente pelos *softwares* de DBMS (Data Base Management System).

Com objetivo de tornar os bancos de dados do *Tot-Win* mais legíveis e cuidando para que tenham enfoque temporal, a modelagem das tabelas baseou-se nas idéias das modelagens dimensionais do *Data Warehouse*, cujo objetivo é de estruturar um banco de dados simples, que fosse de fácil compreensão pelos usuários, que pudesse armazenar informações com aspecto de tempo para reconstituições históricas e que tivesse confiabilidade suficiente para levantamentos estatísticos dos dados.

Considerando a relação custo/benefício, foi aceito certo grau de redundância de dados para aumentar o grau de independência das tabelas, melhorando a “legibilidade de dados” e a documentação temporal, que denominamos anteriormente como requisitos mínimos para UCI (Unidade Celular de Informação). A eficiência transacional oferecida pela técnica entidade/relacionamento foi mantida com o uso de códigos internos do

sistema para relacionamento entre tabelas. No *Tot-Win* não existe a preocupação em atualizar todos os registros das tabelas com dados replicados, quando uma determinada informação é atualizada. Os dados antigos das tabelas filhas servem como documentação temporal da evolução da informação e não comprometem a qualidade de vínculo entre elas.

As redundâncias na estrutura do banco de dados não geram implicações relevantes, pois considerando que o *software* é estruturado para o porte de dados de consultórios e clínicas, a capacidade dos equipamentos disponíveis no mercado atende plenamente as dimensões para armazenamento dos dados, por preços razoáveis. Um bom exemplo deste aspecto de modelagem são os relacionamentos entre as tabelas de Ficha Clínica, Evolução, Prescrição Médica e Ficha Contábil. A Evolução Clínica é uma tabela que tem inserção de um novo registro a cada vez que o paciente retorna para uma consulta. São armazenados alguns dados replicados, como nome do paciente, sexo, idade e outras informações como texto clínico evolutivo, imagens, data e hora do atendimento, e médico responsável. A vinculação com a Ficha Clínica é feita através de um código interno, único, gerado pelo sistema no momento da criação do registro evolutivo. Mesmo que as informações da evolução clínica sejam consultadas de forma independente da ficha cadastral, os dados nela contidos são suficientes para a identificação do paciente e permitem situar em que momento foi feita a observação clínica e por qual médico.

Este mesmo conceito é aplicado também para a prescrição de medicamentos. Cada medicamento prescrito é gravado em registros independentes numa tabela que está vinculada à ficha evolutiva através do código do paciente, data e hora. Este detalhamento temporal permite que a prescrição de medicamento tenha um nível de precisão absoluto em relação à evolução clínica. Semelhantemente à evolução clínica, é possível identificar os medicamentos prescritos para um determinado paciente numa data e hora específicas, sem dependência da ficha cadastral ou da evolução clínica, pois em cada registro são transcritos os campos de nome, sexo e idade. Esta independência facilita o desenvolvimento de módulos de levantamentos estatísticos e torna os registros de dados legíveis por si só.

A Ficha Contábil está associada à ficha de evolução através do código do paciente e data de atendimento. Isto possibilita a vinculação entre os registros de atendimento clínico com dados financeiros contábeis.

Os descritivos dos diagnósticos e os respectivos CIDs constantes nas fichas clínicas e fichas evolutivas são obtidos a partir do banco de dados do CID, que é utilizado para padronizar as nomenclaturas. A replicação dos nomes das doenças pode parecer simples redundância se seguirmos os critérios dos modelos entidade/relacionamento, uma vez que o programa poderia apresentar os nomes das doenças pesquisando na tabela do CID, não havendo necessidade de desperdiçar espaço extra para armazenar os mesmos no registro clínico; mas quando esta análise leva em consideração o fator tempo e o aspecto de facilidade de intercâmbio de dados, a redundância adquire importância, pois em consequência do encapsulamento das informações, ela é resistente às modificações feitas nas tabelas de apoio (CID, SNOMed, Lista de Procedimentos da AMB, etc.), mantendo a legibilidade ao longo do tempo, e simplifica o envio dos dados clínicos, não havendo necessidade de enviar nenhuma tabela de apoio juntamente com as fichas clínicas.

Uma doença pode ser designada de várias formas dependendo da instituição e até mesmo em departamentos diferentes dentro de uma mesma instituição, embora especifiquem a mesma doença. Esta é uma situação de difícil manipulação por um sistema computacional se não lançarmos mão de uma codificação. O armazenamento de código é importante para garantir a fidelidade na recuperação das informações, pois, quaisquer que sejam os termos utilizados, basta associá-los a um mesmo código para que o sistema computacional possa tratá-los como equivalentes. Caso o *Tot-Win* não incorporasse o nome das doenças no registro do paciente, o programa não permitiria a utilização de múltiplos termos, pois utilizando um simples relacionamento entre as tabelas, o mesmo código sempre atribuiria um único nome, e seria difícil reconstituir o termo utilizado originalmente. Se não existisse a possibilidade de múltiplos termos, a implantação de uma codificação seria dificultada. Esta liberdade permite a criação de um

banco de dados de sinônimos, sem prejuízo da padronização ou precisão da recuperação de dados.

O armazenamento do nome da doença na ficha simplifica a apresentação dos dados pelo programa, pois o sistema precisa apenas apresentar o conteúdo de um campo para informar as doenças do paciente, sem precisar consultar um banco de dados para apresentar o significado de um código.

Envolvendo o aspecto temporal, pelo fato de mudança na terminologia utilizada não ser atualizada no registro, pode-se reconstituir a evolução dos termos utilizados para designar uma determinada doença, uma vez que os registros podem ser levantados pelo código da doença, e cada registro tem a data, hora e médico que inseriu os dados.

Em caso de envio dos dados eletrônicos do paciente para uma outra instituição que utilize o mesmo sistema, não há necessidade de enviar o banco de dados de termos, pois todas as informações suficientes já estão definidas no registro.

Considerando a relação custo/benefício, este nosso conceito de UCI, tem sido vantajoso operacionalmente.

Desenvolvimento baseado em objetivo

O *Tot-Win* é um sistema resultante do aprimoramento do *Tot-Clin*, e que adotou o mesmo método de desenvolvimento utilizado desde o início da elaboração do *Tot-Clin*.

O princípio básico é a estruturação de uma sistemática de desenvolvimento fundamentado na co-participação dos usuários no aprimoramento do programa. Neste modelo, o desenvolvimento de novos módulos, ou implementações de novos recursos, visam a solucionar necessidades específicas, e melhorar a dinâmica funcional através de cada recurso implementado (avaliação do nível de incorporação de novos valores à dinâmica operacional). O aumento da eficiência global do setor onde foi feita uma implantação é utilizado como parâmetro para avaliar a eficiência de cada desenvolvimento. Consideramos que a qualidade de um *software* não deve

ser avaliada exclusivamente sob critérios técnicos, mas deve-se levar em consideração os aspectos metodológicos a ela agregada. Quanto mais enfoque metodológico um *software* tiver agregado a si, maior será a sua contribuição organizacional para o local onde será utilizado. Um *software* obriga, de forma direta ou indireta, que o meio tenha que implementar uma sistemática básica compatível com o que foi utilizado como modelo para a elaboração do sistema. Por isto, além do aspecto técnico do desenvolvimento do *Tot-Win*, avaliações dos métodos da rotina operacional precisam ser analisados, buscando-se aumento de eficiência juntamente com os usuários.

O desenvolvimento de um *software* não pode e não deve estar dissociado das necessidades da dinâmica operacional a que se destina. Um *software* não pode existir tecnicamente para si mesmo, estruturado num modelo teórico de funcionamento, mas deve ser um instrumento para agilização, baseado numa seqüência metodológica de trabalho, cujo objetivo final é, além de implementar automação das informações, inserir uma melhoria na dinâmica funcional. O sucesso de um desenvolvimento está na capacidade de entender os processos e elaborar rotinas operacionais para simplificação dos mesmos, integrando-os com as atividades diárias. Por isto é fundamental o domínio das informações e dos processos envolvidos para a elaboração de um bom projeto que agregue valores à dinâmica funcional e que diminua os custos de desenvolvimento de um *software*.

Através da participação de grande número de usuários, pelo entendimento detalhado dos processos envolvidos, e pela preocupação em melhorar a qualidade global da organização, este modelo permite estruturar um núcleo de aperfeiçoamento continuado do *software*, tanto no aspecto técnico quanto no aprimoramento das rotinas operacionais. A área de suporte ao usuário é a área mais importante para garantir a constância desta participação. Até certo ponto, este pode ser considerado como um núcleo de fomento e elaboração de soluções com antecipação das necessidades da comunidade de usuários. As necessidades atuais de um poderão ser as necessidades do outro amanhã, e uma vez desenvolvidas, os custos para a implantação de soluções reduz-se drasticamente.

O desenvolvimento técnico com implantação e correções técnicas *in loco* são características importantes da informatização por objetivo, pois oferece agilidade na elaboração de soluções. Em certo grau é a fusão entre um modelo de prototipagem e a manutenção de correção e aprimoramento. Para que pudéssemos oferecer este serviço de forma eficiente, foi fundamental a nossa capacitação na linguagem de desenvolvimento, domínio de todos os aspectos do *software* a ser desenvolvido, conhecimento dos processos envolvidos na dinâmica funcional de uma clínica, dos assuntos envolvidos (problemas), e do conhecimento de organização e método.

Iconografia e *Interface HCI*

Com a consolidação do *Windows* como ambiente gráfico padrão, os *softwares* desenvolvidos para este sistema operacional começaram a usar largamente *ícones* para definir funções ou recursos. Como quaisquer recursos, os *ícones* devem ser utilizados de forma criteriosa, pois *ícones* não padronizados podem tornar os programas bem menos compreensíveis.

Os *ícones* fundamentam-se no princípio da comunicação associativa, onde uma figura representa a função a ela associada. Quanto mais comum ou integrada ao dia a dia do usuário for a imagem, mais intuitiva ela será. O uso de repertórios de imagens que não sejam comuns na rotina diária do usuário apenas cria dificuldades de entendimento quando empregados num *software*. Nestes casos, é preferível o uso de palavras para designar funções, pois melhora a *interface HCI*.

A idéia associada a um *ícone* é também tempo dependente. Exemplo: há 10 anos a imagem de um disco flexível de 5,25 polegadas era razoavelmente inteligível quanto à sua utilidade; hoje, o uso da mesma simbologia poderá ser menos compreensível, pois as mídias de armazenamento comuns são discos de 3,5 pol, CD, discos de ZIP, fitas DAT, etc.

Atualmente ainda não existem conjuntos de *ícones* específicos, suficientemente padronizados para utilização de forma abrangente na área médica. Para que isto ocorra, há necessidade de estudos pormenorizados, e difusão dos padrões para que as simbologias sejam assimiladas por um

maior número de médicos. O uso de expressões textuais ainda é vantajosa para designar funções do sistema. O uso de *ícones* deve se restringir aos conjuntos já consagrados e largamente utilizados pelos *softwares* comerciais.

A escolha de palavras para designar comandos deve também ser feita com cuidado, pois algumas expressões podem ser facilmente compreendidas por um conjunto de profissionais e não por outros, o que acontece quando há utilização de jargões técnicos. Para indicar que um botão tem função de excluir uma ficha, é conveniente utilizar a palavra APAGAR ou EXCLUIR no lugar de DELETAR.

Por este motivo, apesar do *Tot-Win* se basear em ambiente gráfico *Windows*, ele usa principalmente expressões textuais para designar as funções dos botões e dos menus de acesso. Foi evitado, na medida do possível, o uso de expressões técnicas.

Aspectos de estruturação de telas

O *Windows* facilitou vários aspectos da *interface* dos programas, tornando-os mais intuitivos, mais bonitos e mais fáceis de serem utilizados, principalmente por principiantes, pois o *mouse* oferece ao usuário a possibilidade de acionar a função que ele esteja visualizando, diminuindo a necessidade de memorização de teclas como os programas em *DOS* exigiam. Porém, justamente o uso intensivo do *mouse* pelos programas baseados em *Windows* pode lentificar a velocidade de inserção de dados, pois obriga maior movimentação da mão para acionar uma função. Quando a agilidade é imprescindível, o uso de teclas de atalho (teclas Alt ou Ctrl em conjunto com outra letra) pode ser fundamental. Por este motivo, as funções mais utilizadas em cada tela possuem seu conjunto de teclas de atalho, representado pela tecla Alt e o caractere sublinhado no botão de comando.

Como o *Tot-Win* tem por objetivo abranger o maior número de especialidades possíveis, a modelagem das tabelas levou em consideração a incorporação de campos para Ginecologia, Obstetrícia e Oftalmologia. A apresentação de todos os campos numa tela de evolução clínica poderia

umentar demasiadamente sua complexidade e torná-la confusa. Para evitar este inconveniente e manter a abrangência do sistema, criou-se um campo para definir a especialidade de cada usuário, de modo que, quando se informa o nome e a senha ao *Tot-Win*, o programa automaticamente ativará a tela de evolução clínica correspondente à especialidade do usuário.

A estruturação das telas do *Tot-Win* é baseada em 2 grandes princípios:

1 – *Disponibilizar na tela todos os principais recursos que o usuário poderá necessitar naquele instante para executar as tarefas pertinentes ao módulo.* É comum os usuários usarem somente os recursos disponíveis na tela. Comandos que são acessados através de Menus de Opção são menos freqüentemente utilizados, pois não são tão óbvios, exigem consulta ao manual do *software*, e muitas vezes passam anos despercebidos pela maioria dos usuários.

2- *Evitar compartimentalização demasiada das telas.* Em muitos sistemas era comum a elaboração de múltiplas telas para a inserção de dados de uma ficha clínica. A prática mostrou que, principalmente nos dados clínicos, a compartimentalização demasiada lentifica a inserção de dados e, em decorrência, todo o trabalho. Muitos médicos trocaram de sistema pela dificuldade de adequar o sistema à velocidade necessária para o atendimento ao paciente. No *Tot-Win* somente foram criadas telas adicionais quando o tamanho do monitor obrigava a isto, quando grande quantidade de informações apresentadas na tela poderia confundir o usuário ou quando se desejava utilizar a compartimentalização para criação de segurança contra acessos não autorizados.

Como exemplo de estruturação de telas, a ficha cadastral inicial é composta por diversos campos e um conjunto de botões de comando, entre eles o botão de acesso aos dados pessoais, que quando acionado, apresenta uma tela com informações cadastrais complementares do paciente. Esta separação foi motivada pela grande quantidade de informações a ser exibida, e para evitar que dados pessoais fossem facilmente vistos por outros. Os dados clínicos, como a tela de observação clínica, foram colocados em tela separada, pois com isto pode-se evitar, se necessário, acesso daquelas

informações por pessoas não médicas quando for ativado o sistema de segurança por senhas.

Aspectos sobre a prescrição computadorizada de medicamentos

A prescrição de medicamentos é uma atividade importante para o processo de cuidados assistenciais aos pacientes e representa uma das ações médicas possíveis. Porém, a grande quantidade de fármacos e produtos comerciais disponíveis no mercado, a alta frequência de novos lançamentos e a enorme quantidade de interações e efeitos adversos produzidos por estes medicamentos, faz com que esta importante etapa do processo de atendimento seja susceptível a erros^[27]. Levantamentos realizados, estimaram que os custos relacionados com a morbidade e mortalidade nos EUA, relacionados ao uso de medicamentos, esteja em torno de 136 bilhões de dólares ao ano^[39] e que as reações adversas às drogas esteja possa ser classificada numa faixa entre quarta a sexta maior causa de morte, em 1994, nos hospitais americanos^[45].

Outros estudos, abordando retrospectivamente a incidência de efeitos adversos a medicamentos, num período de 6 meses em dois hospitais da Universidade de Harvard, mostraram que das 4.031 admissões hospitalares estudadas, foram detectados efeitos adversos em 6,5% dos casos e identificados potenciais efeitos adversos antes da administração do medicamento em 5,5%. Os efeitos adversos levaram ao óbito em 1% das admissões; ofereceram risco de vida em 12%; e 30% foram considerados graves^[5]. O que chama a atenção é o fato de que, destes efeitos adversos, 28% foram considerados como passíveis de prevenção durante a fase da prescrição do medicamento.

Apesar do uso de sistemas informatizados para prescrição de medicamentos seja um instrumento importante para auxiliar na tomada de decisão, o seu uso ainda é incipiente. Estima-se que, nos Estados Unidos, somente 1% das prescrições foram feitas pelos médicos utilizando computadores no ano de 1998^[67].

O uso de um sistema computadorizado para prescrição de medicamentos associados a efeitos adversos já foi pauta de discussões. Nos EUA, foi inclusive analisado se este tipo de programa deveria ser avaliado e aprovado preliminarmente pela FDA (Food and Drug Administration). Após os debates, concluíram que existe uma analogia entre o sistema de informação computadorizada e outras fontes de informação médica, como livro texto, e que a FDA não precisaria regularmentar *software* para prescrição de medicamentos, uma vez que estes funcionariam como fonte de informação computadorizada, sem interferir diretamente no contexto do processo de conduta médica ^[69].

Nos nossos estudos, as questões centralizaram-se principalmente em três aspectos:

1 - A agilização que este novo recurso traz é superior à prescrição manual?

A agilidade está muito relacionada com a disponibilização de medicamentos e padrões de orientação previamente cadastrados e com a familiarização do médico com o programa. No *Tot-Win*, além da disponibilização do recurso de prescrição, foram cadastrados inicialmente 2000 medicamentos para uso. Sua efetiva utilização no processo de atendimento ao paciente em geral somente ocorre após treinamento adequado do médico e a disponibilização de suporte telefônico para atendimento relacionado com dúvidas em relação ao programa. Este recurso do *Tot-Win* implementa ainda maior agilidade para a prescrição de medicamentos formulados.

2 – Por ser a prescrição um procedimento muito pessoal, a prescrição computadorizada de medicamentos não afetaria a relação médico-paciente?

Embora, já na década passada, a idéia do uso do computador como instrumento de apoio ao atendimento médico já não era mais vista como fator que prejudicasse o relacionamento médico-paciente, atualmente a noção do uso do computador está mais consolidada e, portanto, não interfere negativamente neste relacionamento. O uso do computador como instrumento de apoio para a prescrição de medicamentos pode ser

considerado vantajoso. Algumas características das prescrições computadorizadas de medicamentos:

- A impressão da receita garante a legibilidade do texto e evita erro na compra de medicamentos decorrente da interpretação do que está escrito. Vários médicos já prescreviam medicamentos usando máquina de escrever para evitar problemas de legibilidade.
- Maior clareza quanto ao texto referente à forma de uso, uma vez que estes textos podem ser elaborados previamente e corrigidos, minimizando os riscos de interpretação errada.
- Pesquisa dos medicamentos por nome comercial, substância ativa ou nome genérico, categoria e outras expressões. Este recurso oferece flexibilidade para a pesquisa de medicamentos alternativos durante o atendimento médico, sem prejudicar a relação médico-paciente.
- Implementação de sistema de alerta para verificação de eventuais interações entre os medicamentos prescritos.
- Possibilidade de emitir orientações adicionais em relação aos medicamentos prescritos, sem aumentar o tempo utilizado para a consulta.
- Possibilidade de conhecer a quantidade do medicamento na apresentação comercial, facilitando a prescrição da quantidade a ser adquirida para completar o esquema terapêutico.
- Possibilidade de imprimir a prescrição no conjunto ou separadamente. Esta é uma característica importante quando se deseja que duas receitas sejam independentes, como no caso da prescrição de medicamentos e medicamentos formulados.
- Agilização na prescrição de medicamentos formulados.

3 – A prescrição pelo computador acarreta problemas éticos?

Não existe problema segundo o CFM, no que diz respeito ao uso do computador como instrumento de apoio à prescrição de medicamentos. É oportuno lembrarmos que o uso de máquinas de escrever para elaborar

receitas já era fato existente em épocas passadas, e o uso computador não é diferente do ponto de vista conceitual, porém agrega quantidade imensa de valor adicional.

Módulos para prescrição de medicamentos no *Tot-Clin* e no *Tot-Win*

O *Tot-Clin* já possuía o módulo de prescrição de medicamentos desde sua concepção inicial. Era uma conceituação simples, usando janela de texto livre que permitia chamar padrões pré-definidos. O conceito era semelhante ao uso de uma máquina de escrever para a preparação das receitas, porém agregava o valor da agilização, permitindo a rápida elaboração de novos textos.

Este conceito foi totalmente remodelado no *Tot-Win* uma vez que se desejava implementar recursos de verificação de interações medicamentosas, levantamentos estatísticos de medicamentos prescritos, incorporação de textos explicativos vinculados a cada medicamento, pesquisa de medicamentos por diversas chaves, como nome comercial, categoria, substância ativa, entre outros. Para atender aos novos requisitos, o módulo foi desenvolvido baseado em banco de dados. Cada medicamento prescrito é armazenado num registro independente, com parte das informações guardadas em campos tipo caractere, outras em campos de texto livre e outras em forma numérica. Todas as informações necessárias para o relacionamento com a tabela da evolução clínica do paciente estão dentro do registro, além de informações replicadas como nome do paciente, sexo, idade, data e hora de prescrição. Esta modelagem baseada em certo grau de redundância de dados tem como objetivo diminuir o grau de dependência entre o arquivo de prescrição e o das fichas clínicas e evolutivas, tornando-o por si só compreensível e passível de análise histórica. O arquivo, mesmo sem vinculação com as tabelas de origem da parte de dados, permite gerar novamente a prescrição, conhecendo-se no nome do paciente e a data da prescrição; é possível realizar levantamentos cruzando o nome do medicamento com sexo e idade; é possível acompanhar os principais medicamentos prescritos por cada médico, etc.

A utilização de recursos de alerta para interações entre os medicamentos prescritos é de importância indiscutível. Quando este recurso não está integrado ao ato prescritivo, o uso na prática diária poderá não ser habitual devido à quantidade de consultas adicionais necessárias durante a elaboração das receitas. Na modelagem do banco de dados de prescrição de medicamentos do *Tot-Win*, foram analisados aspectos referentes à verificação de interação durante o ato de elaboração da receita. Necessita-se, ainda, da preparação da base de dados de informações para a efetiva implementação deste recurso.

Aspectos importantes sobre uso de padrões de codificação

A padronização de termos num banco de dados é um aspecto de grande importância quando se deseja ter precisão para recuperação das informações, agilizar o processo de inserção de dados e normatizar aspectos, como a tabela de honorários profissionais. Durante o desenvolvimento do *Tot-Clin* já foram incorporados aspectos de padronização de dados, como o uso do CID, Tabela AMB e cadastro de medicamentos. O comando Ponto, recurso para acessar dados de uma tabela, foi criado para facilitar a inserção dos dados.

A adoção de padrões internacionais de codificação é fundamental para manutenção da intercambiabilidade de dados, mas alguns aspectos merecem ressalvas:

- Os bancos de dados contendo todos os termos podem ser complexos demais para uso no dia a dia, portanto é preciso avaliar o grau de precisão que se deseja utilizar: por exemplo, é possível utilizar apenas o modelo simplificado do CID-10, que é constituído de 3 dígitos.
- Os termos utilizados podem não ser os comumente utilizados na instituição. Isto pode causar dificuldade no seu uso, principalmente quando o termo é utilizado para classificação, como é o caso do CID. Talvez a criação de sinônimos fosse uma alternativa adequada para resolver este problema.

- Nem sempre padrões utilizados são de conhecimento de todas as especialidades, como por exemplo, o SNOMed, que é mais difundido em Anatomia Patológica, é pouco conhecido por muitas outras especialidades, mas a sua adoção pode ajudar a correlacionar os aspectos clínicos com a classificação patológica, quando a ficha contiver campos para classificação com CID e SNOMed simultaneamente.

Capacidade de intercâmbio de dados

É muito comum a idéia de que o uso da informática simplifica e agiliza o intercâmbio dos dados entre os usuários. O fato, porém, é que as informações precisam ser antecipadamente projetadas para esta finalidade, e que os programas sejam elaborados com esta função, senão o intercâmbio de dados será mais difícil ou lançar-se-á mão da metodologia convencional para a troca de dados.

Em geral, hoje, a troca de dados entre sistemas computacionais é difícil, porque não tem sido considerada como um aspecto prioritário nos *softwares* que informatizam clínicas e consultórios. Paradoxalmente, o uso da informática tem aprisionado os dados aos programas e aos computadores do usuário pelos seguintes motivos:

- Ausência de padronização para intercâmbio de dados entre sistemas diferentes.
- A maioria dos programas comerciais não dispõe de recursos para intercâmbio de dados. Mesmo quando dois usuários utilizam o mesmo programa, muitas vezes não conseguem enviar os dados clínicos de pacientes, de um para outro, como acontece, por exemplo, com clínicas que tenham duas unidades fisicamente separadas e não estão conectadas *on-line*.
- Muitos *softwares* fragmentam demasiadamente os dados, criando um ambiente de dependência de tabelas, de forma que ao transferir um registro de um ambiente para outro, pode-se perder a consistência da informação e os dados podem tornar-se

incompreensíveis. Isto ocorre porque parte das informações está armazenada numa tabela, e partes noutras, e relacionam-se apenas por um código de sistema. Quando o registro é transferido para um outro ambiente onde não existe o código correspondente ou quando o mesmo código é utilizado para outra finalidade, a consistência e confiabilidade dos dados ficam comprometidas. Nestes casos, seria obrigatória a transferências das tabelas de apoio juntamente com o dado clínico, o que aumenta a complexidade, e exige maior cuidado durante o desenvolvimento do sistema.

No *Tot-Win* foram desenvolvidos vários recursos para resolver estes problemas, reunidos nas funções seguintes:

- Botão “Exportar” da ficha cadastral do paciente: para exportar os dados clínicos em formato de texto (ASCII). Esta opção foi escolhida por considerarmos o ASCII como o formato com compatibilidade de maior abrangência para os sistemas computacionais. Antecedendo cada informação são incluídos parâmetros de controle indicando nome do campo, tipo de dado e tamanho. Estes parâmetros são inseridos entre colchetes e permitem que outros sistemas os utilizem para gerar um outro banco de dados.
- Botão “Replicar”: recurso acessado através da ficha cadastral do paciente. Gera um arquivo com todos os dados clínicos e evolutivos do paciente para gravação em disquete ou envio por e-mail para replicação dos dados num outro computador que utilize o *Tot-Win*.
- Botão “Atualizar”: tem a função inversa ao anterior, isto é, pesquisa os dados do disquete e inclui as alterações nas fichas clínicas do computador que recebe os dados. A este processo denominamos de Replicação e Sincronização de dados. É particularmente interessante quando se deseja utilizar um *notebook* para fazer um acompanhamento *in loco* dos pacientes, como por exemplo, no hospital.

O processo de replicação de dados no *Tot-Win* é simplificado através da técnica de modelagem de dados utilizados (UCI), que encapsula todos os dados importantes no próprio registro do paciente, sem dependência de nenhuma tabela de apoio.

Confiabilidade de sistema computacional

À medida que se utiliza um sistema computacional para centralização dos processos de trabalho, a integridade dos dados e a estruturação de estratégias de contingência são aspectos primordiais para se manter o uso do computador como ferramenta de trabalho.

Definimos como estratégia de contingência a estruturação de alternativas de trabalho em caso de falha parcial ou completa do sistema computadorizado.

Entendemos como integridade de dados: os aspectos relacionados com a consistência dos bancos de dados, a realização de cópias de segurança eficientes e a segurança contra acesso por pessoas não autorizadas.

Estratégia de cópia de segurança de dados

A utilização dos recursos de informática exige cuidados especiais, principalmente relacionados à segurança dos dados e à continuidade de funcionamento dos sistemas. Se, por um lado, a informatização otimiza e agiliza os processos de uma organização, por outro lado, expõe o usuário a fraquezas.

A adequada estruturação da rotina de segurança de dados é fundamental tanto para garantir a integridade dos trabalhos, bem como para eventual processo de validação das informações. A cópia de segurança não deve ser vista como uma simples cópia dos dados para recuperação em caso de falha de sistema, mas deve estar inserida num contexto mais amplo, envolvendo aspectos como segurança de dados, duração das informações antes da sobregravação, periodicidade de atualização, dimensão, vulnerabilidade à degradação ambiental e roubo, entre outros. Um esquema mais consistente para armazenamento de cópias de segurança talvez seja um modelo misto entre cópias em mídias magnéticas e associadas a modelos em *Laser*.

As perdas de dados podem basicamente ser divididas em duas categorias:

- Perda por falha abrupta de sistema – quando os erros ocorrem e pode-se constatar imediatamente a perda de dados.
- Perda por deterioração: é uma gradativa perda de dados que, em geral, é detectado tardiamente. Estas perdas podem ser decorrentes de falhas de *hardware*, falhas do sistema operacional, erros do programa, destruições por vírus do computador ou por erros no uso do programa. Quando as falhas são detectadas, muitas vezes já foram reproduzidas para as cópias de segurança de dados.

Uma rotina adequada de cópia de segurança consegue proteger eficientemente os dados dos problemas causados pelas falhas abruptas. Uma vez identificado o problema, pode-se restaurar os dados e reativar o sistema.

As falhas de deterioração são na verdade as falhas mais problemáticas. Para proteger os dados deste tipo de falha é preciso criar uma estratégia adequada.

A seguir serão discutidos dois modelos de estratégia de segurança que garantem a integridade para os dois tipos de falhas:

Estratégia de segurança de dados – Modelo 7 x 12 anual

Este modelo de cópia de segurança de dados baseia-se no uso de 7 mídias para utilização durante um mês e 12 mídias para armazenamento mensal dos dados durante o ano, sem riscos de sobregravação.

As 7 mídias para utilização dentro do mês são estruturadas da seguinte forma:

- 1 mídia para cada dia da semana (segunda a sábado).
- 1 mídia para ser utilizada no meio de cada mês (acumulador parcial - AP).

As seis mídias da semana garantem o *backup* diário, pois a cada dia os dados são gravados em mídias independentes. Isto significa que a precisão de segurança é diária e a duração dos dados é semanal.

No meio do mês (ao redor do dia 15), os dados são gravados na mídia AP. Como esta mídia somente será reutilizada no mês seguinte, a durabilidade dos dados será de um mês. Quando integrada ao conjunto das seis mídias, esta estratégia mantém os dados com a precisão de quinze dias.

Ao final do mês, como os dados são gravados nas mídias mensais, garante-se a manutenção das informações mesmo que o conjunto de mídias seja reutilizado. Como existem 12 mídias, a reutilização, se for esta a opção do usuário, somente ocorrerá no próximo ano. Isto significa que a manutenção dos dados gravados é por 1 ano.

Ao final do ano, ter-se-á, um conjunto de 12 mídias que são a evolução dos dados durante todo o ano, que além de proporcionarem segurança aos dados, permitem um certo grau de reconstituição da evolução temporal dos mesmos.

Nesta estratégia, a identificação de falhas até o prazo de uma semana permitirá a reconstituição com precisão de um dia. Quando a identificação da falha ocorrer até quinze dias, a reconstituição pode ter como consequência uma defasagem de quinze dias dos dados.

Estratégia de segurança de dados – Modelo 9 x 12 anual

Este modelo é muito semelhante à estratégia anterior, sendo diferente no fato deste esquema utilizar 9 mídias proporcionando segurança semanal.

As 9 mídias para utilização dentro do mês são estruturadas da seguinte forma:

- 5 mídias para cópias diárias (segunda a sexta)
- 4 mídias semanais (aos sábados)

As quatro mídias de sábado possuem sobrevida mensal, pois somente serão sobregravadas no próximo mês.

As 12 mídias mensais funcionam de forma idêntica ao esquema anterior.

Gravação em CD-R

É importante considerar a opção de gravação de dados em CD-R, que pode acrescentar valor técnico quando se deseja enfatizar aspectos de inviolabilidade de dados, pois podem ser gravados somente uma única vez. São diferentes dos CD-RW que podem ser apagados e regravados várias vezes.

A gravação em CD-R tem ainda as seguintes vantagens:

- Capacidade de armazenamento de dados (650 *megabytes*) superior a outros meios de armazenamento como *ZIP Drive* (capacidade de 100 ou 250 *megabytes* num único disco) e ao *Super Disk* (capacidade de 120 *megabytes* e compatibilidade com discos de 3,5 polegadas convencionais).
- Baixo custo das mídias de gravação.
- Possibilidade de ser utilizado em maior número de computadores: existem muito mais equipamentos que dispõem de leitor CD do que equipamentos com *ZIP Drive*, *Super Disk* ou *Iomega Jaz Drive* (dispositivos que podem armazenar de 1 a 2 *gigabytes* em cartuchos especiais próprios).
- São resistentes à umidade.
- São resistentes, diferentemente dos disquetes e fitas, a campos magnéticos. Exemplo: ímã, alto falante de rádios, etc.

Uso de fita DAT

Para ambientes de maior porte e que utilizam servidores baseados em NT, uma opção é a utilização de unidade de fita DAT para esquemas de cópia de segurança de dados. Estas mídias são as que oferecem maior capacidade de armazenamento (12/24 *gigabytes*) e vários *softwares* de gerenciamento oferecem o recurso de se programar o horário para o início da realização da cópia de segurança. É possível utilizar 12 fitas de 12/24 *gigabytes*, e implementar cópias alternando esquema de Cópia Completa e

Cópia Diferencial, que transfere apenas arquivos que foram modificados desde a última gravação Completa.

Validação jurídica dos dados eletrônicos

O registro médico é um documento legal e por conter dados como observações clínicas, hipóteses diagnósticas e a terapêutica escolhida, é o fundamento para determinar se um cuidado apropriado foi despendido ao paciente. Para ter efeito legal, a maioria das informações clínicas deve ser assinada pelo médico responsável.

Talvez induzidos por uma noção errada de que a informatização poderia eliminar todos os documentos convencionais, era comum ouvir frases como “agora vamos acabar com todos os prontuários em papel”, “vamos economizar espaço, pois tudo pode ser armazenado no computador”, e outras expressões deste gênero. Os sistemas computacionais podem armazenar diferentes tipos de dados, desde informações textuais até imagens digitais, seqüências de vídeos e modelos tridimensionais, mas alguns documentos resultantes da atividade assistencial médica precisam ser mantidos. Os sistemas computacionais, nestes casos, podem ser utilizados para ajudar a elaborar estes documentos, mas não podem eliminá-los antes de cumpridos os prazos e formalidades regulamentadas, como é a situação dos prontuários de pacientes.

Apesar de existirem vários pareceres do CFM e CREMESP estimulando o uso dos meios eletrônicos para armazenamento de dados, o prontuário médico é um documento indispensável para aferir a assistência médica prestada, servindo também como instrumento de defesa legal, conforme resolução do CFM N^o 1.331/89^[20].

Artigo 1: o prontuário médico é um documento de manutenção permanente pelos estabelecimentos de saúde.

Artigo 2: Após decorrido prazo não inferior a 10 (dez) anos, a partir da data do último registro de atendimento do paciente, o prontuário poderá ser substituído por métodos de registro, capazes de assegurar a restauração plena das informações nele contidas.

Em relação à consulta 1.326/92, há parecer de que os exames laboratoriais e subsidiários, conforme legislação estadual, através do decreto nº 12.479 de 19/10/79, estabelecendo em cinco anos o prazo de guarda de resultados obtidos em exames citopatológicos.

Acrescenta-se que os laudos dos exames laboratoriais, anátomo-patológicos e radiológicos devem ser anexados ou transcritos ao prontuário do paciente, e uma vez cumprida essa formalidade, não existe obstáculo para que sejam fornecidos ao paciente os filme radiográficos, fitas de ECG, etc., sendo que essa entrega deve ser feita mediante recibo.

Isto significa que, mesmo que os aspectos eletrônicos sejam relevantes, a existência de um prontuário físico em papel ainda é obrigatório, independente do fato de se armazenar os dados também em ambiente computacional.

É importante lembrar que vários aspectos que conferem validade jurídica ao prontuário em papel são decorrentes da possibilidade de realização de exames periciais: detecção da idade do papel e da tinta, análise da escrita, detecção de rasuras, etc., que permitem analisar a veracidade do documento.

Devido à extrema versatilidade dos sistemas computacionais, a validade das informações eletrônicas em ambientes de consultórios ou clínicas ainda precisa ser melhor estruturada, para que venha a adquirir regulamentação oficial que confira validade jurídica aos dados exclusivamente eletrônicos, apesar de já existirem várias consultas ao Conselho Federal de Medicina e Conselho Regional de Medicina do Estado de São Paulo com pareceres favoráveis^[19,20,24,21,22,25]. Para a viabilização da validação, em relação à pessoa física, é necessário desenvolver estudos técnicos que permitam reconstituições periciais das informações quando necessárias, como rotinas de “criptação” e proteção contra violabilidade de informações, sistema de contra-referência de dados baseado na *Web*, etc.

Há duas situações diferentes no que se refere a questões de validade jurídica: 1. Ambientes corporativos e hospitalares; 2. Ambiente de consultórios e clínicas.

Ambos podem diferir quanto à validade jurídica embora utilizem sistemas computacionais para armazenamento de dados. No primeiro caso, por ser uma instituição de médio ou grande porte, em geral os médicos que assistem o paciente não possuem ascendência hierárquica sobre os funcionários responsáveis pelo CPD. Assim sendo, um sistema adequadamente projetado, que identifique o médico e registre seus apontamentos clínicos e prescrições, poderá armazená-los num banco de dados ao qual ele não tenha acesso direto para realizar alterações posteriores. As cópias de segurança, nas instituições, em geral são feitas em monobloco (dados assistenciais gravados em conjunto com dados administrativos) usando Fitas DAT. Isto pode representar mais um fator técnico de apoio para a validação jurídica das informações eletrônicas nas instituições, pois estas podem ser utilizadas como contra-referência para assegurar a fidelidade e integridade das informações. Estas características não existem no ambiente de consultórios ou clínicas. Embora o *Tot-Win* possa ser utilizado para auxiliar a elaboração de apontamentos clínicos, imprimindo no mesmo papel, logo abaixo dos apontamentos anteriores do paciente e com assinatura do médico, a estruturação técnica para suportar a validação jurídica dos dados exclusivamente eletrônicos é a seguinte:

- Permite ao usuário inserir os apontamentos clínicos e prescrições de medicamentos na ficha clínico-evolutiva e, ao encerrar o trabalho com cada ficha, este procedimento impede que os dados possam ser novamente alterados. Ao final do expediente, ao acionar o comando de fechamento, o programa realiza uma cópia das fichas encerradas para um arquivo de "auditoria". Este arquivo não pode ser acessado pelo usuário e é um espelho dos dados clínicos. Os registros nele gravados contêm além dos dados, a data e hora da gravação, e atributos de algum arquivo do *Windows* (alguns arquivos específicos do *Windows* são atualizados sempre que o sistema operacional é acionado, com isto a data e hora do arquivo também são alterados segundo o relógio do microcomputador).

- Simultaneamente ao fechamento dos dados é gerado um arquivo contendo as mesmas informações que foram replicadas para o arquivo de auditoria, para ser enviado a um *Site* de contra-referência.
- Neste *Site*, o usuário poderá abrir uma conta para armazenamento das informações. Os arquivos enviados serão vinculados com o nome do dono da senha da conta, a data e hora de envio de acordo com o relógio do servidor, que estarão fora do controle do usuário. Este procedimento gerará um conjunto de contra-referência e também o torna como mais uma cópia de segurança de dados. A fim de garantir o sigilo destes dados, o sistema poderá ser estruturado de forma que os arquivos possam somente ser acessados quando o usuário os libere através da senha de autorização.
- Anualmente, o provedor de armazenamento, poderá gravar as informações em CD-ROM, sendo uma unidade enviada ao usuário e a outra permanecendo na empresa pelo período legal pertinente.

Os dois primeiros procedimentos descritos já estão implementados no sistema, cabendo estudos técnicos posteriores para a implementação do sistema de contra-referência através da *Web*. O *Site* de contra-referência ou provedor de armazenamento poderia ser uma instituição privada, universitária ou pública.

Os resultados de exames subsidiários podem ser transcritos para o prontuário, ou digitalizados e armazenados em formato JPG, e depois devolvidos aos pacientes mediante recibo, conforme o parecer do CFM Nº 1.326/92.

Manutenção de *Software*

Muitos têm a noção errônea de que a manutenção de um *software* restringe-se tão somente a procedimentos de correção de problemas técnicos resultantes da fase de desenvolvimento. No entanto, a manutenção é bem mais complexa e envolve diversos outros procedimentos que podem fundamentalmente ser agrupados em três atividades (Swanson, 1976) ^[73]:

- Correção de erros: apesar do uso de métodos adequados de revisão técnica formal e aplicação de testes de *software*, é improvável que se consiga identificar todos os erros de um sistema, principalmente quando este é complexo. Durante o uso do sistema, após a entrega ao usuário, podem surgir falhas de desenvolvimento resultantes de erros logísticos não identificados ou decorrentes de eventualidades não previstas durante a fase de desenvolvimento. A eficiência da manutenção de correção é um fator decisivo para manter a confiança do usuário no sistema e garantir a aceitação do mesmo pela instituição.
- Aprimoramento: é uma atividade muito importante, principalmente quando um *software* é bem-sucedido. Ocorre à medida que o *software* é usado e sugestões de novos recursos, melhorias nas funções existentes e ampliações gerais são recebidas dos usuários. É uma atividade delicada e os métodos são iguais aos utilizados no desenvolvimento de um sistema, talvez com um agravante: a necessidade de manter compatibilidade com todo o restante do sistema, sem prejuízo a nenhum dos recursos já implementados. A manutenção de aprimoramento pode ser a responsável pela maior parte dos esforços envolvidos na manutenção. Aliada a uma boa infraestrutura de apoio ao usuário, pode se tornar um poderoso instrumento de aperfeiçoamento continuado do *software*, garantindo a evolução permanente para que o sistema se molde cada vez mais de acordo com o segmento de profissionais para o qual foi projetado.
- Adaptação: é a atividade que modifica o *software* para que tenha uma *interface* adequada com um novo meio computacional, em decorrência das rápidas mudanças, como o lançamento de novas gerações de *hardware*, novos sistemas operacionais e novos equipamentos periféricos, entre outros. Isto ocorre porque um *software* pode ter vida útil mais longa que o *hardware*, ultrapassando 10 anos. Este é um fato bem conhecido e um exemplo amplamente divulgado pela mídia ocorreu quando o centro das

preocupações era com o *bug* do ano 2000. O problema em essência era decorrente de sistemas que tinham sido desenvolvidos há 30 anos atrás, e que, pelo fato de que armazenavam o ano com dois dígitos, poderia confundir 1900 com 2000.

Além disto, uma quarta atividade pode ser incluída nesta lista ^[63]:

- Modificações para melhoria da confiabilidade: são trabalhos que visam a adequar os sistemas para comportarem futuras ampliações. Algumas vezes esta atividade é denominada de manutenção preventiva, embora este seja um termo mais comumente utilizado para designar a manutenção de *hardware*.

O *software* não deve ser considerado como um produto estático, que uma vez desenvolvido torna-se imutável ao longo do tempo. Um programa de computador é um produto dinâmico que precisa ser periodicamente reavaliado, e nele incorporados novos recursos para atender às necessidades de um ambiente ou de uma instituição. A continuidade de desenvolvimento de um *software* é um aspecto importante que pode determinar se ele é robusto e confiável o suficiente para ser utilizado como instrumento de trabalho na prática diária. Este aspecto é particularmente importante quando se leva em consideração que ambientes de pequeno porte, como consultórios e clínicas, não possuem, em geral, infraestrutura de CPD próprio, com analistas e programadores para desenvolvimento de sistemas, como ocorre nas grandes corporações, mesmo que em muitas situações necessitem de desenvolvimentos para automatização de aspectos específicos da prática diária. A contratação de programadores e analistas temporários pode criar riscos para que ocorram desenvolvimentos de programas incompatíveis entre si, causando replicações de trabalho e sem que haja perspectiva de aperfeiçoamento e, a contratação de uma equipe para tal finalidade é inviável do ponto de vista financeiro para o porte destas instituições. Nestes casos, a utilização de um *software* bem sucedido, com grande número de usuários e uma eficiente infraestrutura de apoio e manutenção de aprimoramento, pode oferecer as mesmas facilidades de um CPD de uma grande empresa, a custos relativamente baixos. Embora, em valores absolutos, os custos de

desenvolvimento sejam muito semelhantes aos de um ambiente corporativo, num ambiente baseado em *software* de grande distribuição, o valor individual reduz-se muito, pois os custos são divididos pelo número de usuários que potencialmente utilizarão os mesmos recursos.

Aspectos sobre obsolescência de *software*

O termo obsolescência deve ser analisado com muito cuidado, quando se está sob o prisma da informatização de uma clínica ou consultório e deve ser avaliado sob dois ângulos:

- Obsolescência técnica: quando os aspectos são avaliados somente do ponto de vista técnico; versão da linguagem, capacidade de gerenciamento do banco de dados, novos recursos de comandos da linguagem, capacidade de execução em ambiente *Windows*, etc.
- Obsolescência funcional: quando se analisam os aspectos funcionais e sua inserção dentro da dinâmica operacional da clínica ou do consultório, verificando se atende plenamente às suas necessidades. A confiabilidade do sistema é importante para avaliar se o *software* pode ser adotado como um instrumento de automação dos processos.

O sistema operacional *DOS*, apesar de ser muitas vezes apontado como um sistema operacional ultrapassado, tem a grande vantagem de ser ágil, o que possibilita que equipamentos com configuração modesta possam executar programas com significativa eficiência.

Atualização do *Tot-Clin* para processamento do ano com 4 dígitos

Estes aspectos sobre obsolescência podem ser exemplificados pelo *Tot-Clin*, que pode ser considerado tecnicamente obsoleto, ainda baseado na linguagem *Clipper'87 Summer*, e executado sob *DOS*, continua a ser utilizado por vários usuários. Em dezembro de 1999, 28 usuários que se adaptaram ao programa, não desejavam mudar de sistema, uma vez que já tinham prática para seu uso, o programa tinha a confiabilidade desejada, e não necessitavam de novos recursos para o trabalho do dia a dia. Isto motivou a atualização do sistema para o ano 2000. Portanto, um sistema que

é tecnicamente ultrapassado pode ser funcionalmente adequado. O aspecto técnico não é critério determinante para definir que um sistema está obsoleto ou que o seu uso esteja inadequado. Como discutido anteriormente, um *software* bem dimensionado agrega em si, além do projeto técnico, valores de métodos de trabalho incluindo funcionalidade e rotinas operacionais.

O fator mais importante para definir obsolescência seria a verificação de que o *software* não atende mais as necessidades do usuário. Por exemplo, quando a clínica cresce, e os controles administrativos se tornam complexos de forma que o *software* não consegue gerenciar os controles, e não há possibilidade de fazer uma adequação e aprimoramento do sistema.

Importância do envolvimento de um profissional de informática médica

O processo de desenvolvimento de um sistema computacional pode ser descrito como sendo a identificação e estruturação dos processos, e a codificação através de uma linguagem de programação, para que o método de trabalho possa ser executado por um conjunto computacional. A quantidade de erros que existem num *software* finalizado depende de vários fatores, destacando-se o método de desenvolvimento utilizado, o grau de conhecimento da equipe em relação aos processos que estão sendo automatizados pelo *software*, o número de pessoas envolvidas, e o grau de complexidade do sistema e assim por diante. Como no desenvolvimento já está implícito o entendimento do serviço a ser informatizado, a capacitação técnica (conhecimentos técnicos de informática) é apenas um dos itens do processo de desenvolvimento de um sistema.

Quanto aos aspectos assistenciais, administrativos, científicos ou didáticos num ambiente médico, se o profissional de informática envolvido tiver formação médica, é facilitado seu entendimento, por causa da sua familiaridade, estimulando o debate e permitindo que aspectos mais profundos sejam abordados.

Os desenvolvimentos na área médica podem ser divididos em dois grandes grupos:

- *Específicos*: quando há necessidade de conhecimentos especializados numa determinada área ou num determinado assunto para o desenvolvimento de um sistema, como por exemplo, a elaboração de um *software* para avaliação de testes de sensibilidade cutânea. Neste caso, o conhecimento detalhado dos vários aspectos da temática é fundamental.
- *Gerais*: aspectos estratégico-organizacionais, quando o enfoque está relacionado às estratégias gerenciais e administrativas, aplicação de conceitos de tecnologia da informação, desenvolvimento de sistemas assistenciais de caráter institucional e corporativo (situado em posição comum a todas as especialidades, como é o caso da elaboração de prontuários eletrônicos, elaboração de sistemas de prescrição computadorizada de medicamentos, etc.).

Em qualquer um dos dois casos expostos, a existência de um profissional médico com preparo em informática seria fundamental, porque podem facilitar a coordenação de projetos de desenvolvimento de informática:

- conhecimento das capacidades e técnicas oferecidas pela informática e pela telemedicina;
- familiaridade com alguma linguagem de programação ou técnicas de análise;
- conhecimento de aspectos relacionados com organização e método;
- conhecimento de processos de engenharia de *software*;
- conhecimento sobre desenvolvimento organizacional;
- e conhecimento de estratégias de armazenamento de informação usando conceitos de *Data Warehouse* e *Data Mining*.

Este profissional poderia ser chamado de Especialista em Informática Médica, e seria determinante para a viabilidade de um desenvolvimento computacional na área médica. Pela falta de profissionais com este perfil, muitas instituições atualmente têm procurado preencher esta lacuna com profissionais de outras áreas.

Como discutido anteriormente, a entrega de um *software* para o usuário não é o fim de um projeto computacional. É o início de um processo resultante da fusão da manutenção de aprimoramento com o suporte ao usuário, que é a fase que garante o uso efetivo de um *software*. Esta fase é permanente, e deve ser administrada constantemente, com cuidados adicionais quando comparada à fase de desenvolvimento, pois a necessidade da garantia da compatibilidade dos novos módulos com o restante do sistema precisa sempre ser analisada. Estudo da lucratividade, do desenvolvimento de modelos de acompanhamento e da eficiência organizacionais, desenvolvimento de estratégias de integração didática e científica, e construção de modelos de bioinformática seriam focos de atuação deste profissional.

O tipo de atenção requerida pelos sistemas como descrevemos, caracteriza a necessidade de um serviço constante, à semelhança do que ocorrem com ambulatórios, serviços de radiologia ou laboratórios de análises clínicas, que atendem seus usuários. Da mesma forma que estes, percebe-se a necessidade de um aprendizado em serviço, uma forma que pode ser alcançada pelo modelo de residência médica. A existência de um modelo de capacitação em informática médica precisaria ser reavaliada.

Conclusão

Objetivo: Descrever e discutir os principais aspectos de um modelo para o desenvolvimento de um sistema que reúna as principais necessidades de uma clínica/consultório, nos aspectos assistenciais, administrativos, financeiros e científicos:

A integração dos aspectos administrativo-financeiros, assistencial e científico num único *software* é viável, sem prejuízo de nenhum dos aspectos, como é demonstrado pelo programa *Tot-Win*, uma vez que todas as formas de funcionamento de cada uma das abordagens sejam totalmente compreendidas.

O desenvolvimento deve levar em consideração aspectos da dinâmica operacional e o perfil dos recursos humanos. Quanto mais integrado um *software* ao seu meio, maior será a sua aceitação. Programas que não abordam adequadamente estes aspectos têm mais probabilidade de ser abandonados pelas dificuldades em utilizá-los como instrumento de agilização dos trabalhos diários.

Objetivo: Apresentar as bases necessárias para se elaborar um sistema de prescrição computadorizada de medicamentos e avaliar a sua importância na atividade assistencial:

A prescrição computadorizada de medicamentos é uma ferramenta que pode agilizar o atendimento médico, garantir a melhor legibilidade das receitas, e não tem nenhuma consequência legal quanto ao seu uso no dia a dia, como consta em parecer do CFM N^o 1.345/93. Estes sistemas podem oferecer a vantagem de permitir verificação de interações entre medicamentos prescritos, o que pode implicar em diminuição de custos da assistência médica.

Objetivo: Propor, definir, discutir e avaliar quais são os aspectos mais importantes para uma informatização eficiente e explorar os conceitos de eficiência e obsolescência:

Uma informatização pode ser considerada eficiente quando implementa melhoria da qualidade de funcionamento da clínica ou do consultório em relação à situação que existia antes do uso da informática. A eficiência fundamenta-se em dados confiáveis, gera como efeito agilidade na recuperação de dados e diminuição dos custos (quando se compara a

execução dos mesmos serviços, no mesmo grau de qualidade, entre o novo sistema implementado e o anterior).

O uso da informática deve considerar vários aspectos de segurança, bem como estratégia de contingência, isto é, alternativas de trabalho que permitam ao consultório/clínica funcionarem mesmo em caso de falha do sistema.

A estruturação de uma estratégia eficiente de cópia de segurança de dados deve proteger um ambiente computacional contra as perdas abruptas ou por deterioração gradativa do sistema. Os esquemas de cópia podem ser 7 x 12 ou 9 x 12. A inclusão de gravação em CD-R pode melhorar ainda mais estas estratégias.

A obsolescência de um sistema computacional não deve ser avaliada baseando-se somente nos aspectos técnicos; deve-se considerar a adequação do sistema ao ambiente, e se mudanças forem pretendidas, considerar se estas oferecerão ganho funcional.

Objetivo: Estabelecer uma sistemática para replicação dos dados e geração de arquivo em formato que possa ser utilizado por outros sistemas computacionais:

A incorporação de rotinas para replicação de dados pode facilitar o intercâmbio entre usuários do mesmo sistema. A implementação de recursos para gerar arquivos contendo dados do paciente em formato ASCII, agregando imagens em JPG, oferece compatibilidade para outras plataformas computacionais. Inserindo-se outras informações, como nome do campo, tamanho e finalidade do arquivo gerado, pode-se oferecer dados técnicos suficientes para que sejam desenvolvidos módulos em outros programas, capazes de captar estas informações e incorporá-las nos seus respectivos bancos de dados.

Objetivo: Avaliar o papel da modelagem de dados na simplificação da manutenção e no ganho em flexibilidade de um *software*, e suas implicações na confiabilidade dos dados de um sistema:

A modelagem de um banco de dados com certo nível de redundância é importante, e o gasto de espaço adicional com o armazenamento, em consequência do uso desta técnica não causa consequências significativas, quando se leva em consideração as capacidades dos equipamentos

atualmente disponíveis e a relação ao porte de dados de um ambiente de clínica/consultório.

As redundâncias podem facilitar o intercâmbio de dados entre sistemas, evitando a necessidade de desenvolvimento de programas tecnicamente mais complexos, bem como facilita a análise de dados com o uso de programas disponíveis no mercado, como planilhas eletrônicas, sem necessidade de analistas para elaboração de rotinas de programa.

Objetivo: Avaliar aspectos relativos à utilização de termos padronizados para garantir a precisão na recuperação dos dados armazenados:

O uso de banco de dados de padronização de terminologia médica, como CID-10, SNOMed, Lista de Procedimentos da AMB, entre outros, têm a vantagem de oferecer precisão na recuperação dos dados, pois permitem que termos com o mesmo significado, porém escritos de forma diferente, tenham o mesmo código. Porém o uso dos termos deve ser implementado com cuidado. Utilização de segmentos dos dados, referentes a cada especialidade, diminui o tamanho do arquivo a ser consultado no dia a dia e pode agilizar a procura das informações. A inserção de sinônimos ou jargões utilizados na instituição pode facilitar o uso, mas os novos termos precisam manter o padrão de codificação.

Objetivo: Indicar aspectos relacionados com validade jurídica e estabelecer alguns procedimentos técnicos para sua viabilização, considerando dados armazenados exclusivamente em meio eletrônico:

É possível a estruturação de etapas técnicas que possibilitem a reconstituição histórica de informações clínicas e permitam o uso de um modelo de contra-referência baseado na *Web*, como fundamento técnico para viabilizar a validação jurídica de informações armazenadas em meio exclusivamente eletrônico. Estudos futuros mais detalhados para o desenvolvimento dos aspectos jurídicos e usos relacionados com a *Web* são necessários.

Objetivo: Avaliar quais são os aspectos mais determinantes da qualidade de um *software*:

A disponibilidade de manutenção para o *software* deve ser considerada como um dos critérios para definir sua aquisição, pois é ela que garante a

evolução constante. Através da manutenção de aprimoramento é possível constituir um núcleo de desenvolvimento compartilhado, que reduz os custos individuais para o desenvolvimento, e incorpora soluções metodológicas além das tecnológicas.

Objetivo: Discutir e esboçar aspectos técnicos para o uso de recursos da *Internet* como ferramenta de apoio pós-consulta dos pacientes:

A *Internet* pode ser um instrumento para melhorar a qualidade do atendimento pós-consulta, através de páginas de orientação, *e-mail* para interação e lista de discussões, propiciando ambiente de debate e compartilhamento de experiências entre pacientes. É importante que o médico seja o mediador destes ambientes compartilhados, vigilante sobre os aspectos éticos envolvidos.

Objetivo: Discutir a importância de um profissional para coordenar o desenvolvimento de um sistema médico e a necessidade e as vantagens de que este profissional tenha formação médica e de informática:

Quanto mais aspectos assistenciais médicas um *software* tiver, maior será a importância do envolvimento de um profissional de informática médica para desenvolvimento do projeto. Conhecimentos de programação, modelagem de dados, aspectos relativos à dinâmica de atendimento dos pacientes, e aspectos administrativo-financeiros na área médica, agilizarão o processo de desenvolvimento, facilitarão a compreensão do trabalho a ser desenvolvido e a troca de idéias entre os médicos e a equipe de informática.

Glossário

Este glossário destina-se a ser um conjunto de definições que sirva de apoio para a leitura do texto ^[16].

ActiveSync

Operação que sincroniza o banco de dados de um micro *Windows CE* com a réplica localizada na máquina *desktop*.

ActiveX

Veja Linguagem de programação

ADSL

Asymmetric Digital Subscriber Line. É uma nova tecnologia que possibilita a transmissão de dados, em altas velocidades, utilizando cabos telefônicos comuns. O sistema trabalha com velocidades assimétricas, ou seja, diferentes em cada sentido. No *downstream* a ADSL atinge de 1,5 a 9 Mbps. E no *upstream*, as taxas vão de 16 a 640 Kpbs. É necessário instalar *Modems ADSL* nas duas pontas.

AGP

Sigla de Accelerated Graphics Port, ou Porta Gráfica Acelerada. Trata-se de uma tecnologia da *Intel*, projetada especialmente para obter melhor rendimento com imagens em 3D. O sistema AGP é 32 *bits* e transfere dados à razão de 266 *megabytes* por segundo (Mbps).

Applet

Programa escrito para rodar dentro de outro programa (por exemplo, macros do *Word* ou do Excel). Um *Applet* não pode ser executado de forma autônoma. Os *Applets* escritos em *Java* responsáveis por boa parte da dinâmica da *Web*, em geral rodam no ambiente do *browser*.

Área visível

Os tamanhos de monitores são determinados por sua área visível. Modelos de mesmo tamanho freqüentemente têm áreas visíveis diferentes. Por exemplos: um monitor de 14 polegadas pode ter uma área visível de 13 a 13,2 polegadas; monitores de 17 polegadas podem variar de 15,6 a 15,9 polegadas.

ARJ

Compactador de arquivos. Depois da consolidação do *Windows*, perdeu espaço para os programas baseados no sistema ZIP, que adotaram primeiro a *interface* gráfica. Hoje o ZIP é um padrão de fato, largamente usado na *Internet*.

Arquivo de log

Os servidores *Web* mantêm arquivos com o registro (log) de todas as solicitações que recebem. Esse documento permite analisar itens como a procedência do usuário, a freqüência com que retornam ao *site* e seus hábitos de navegação dentro do *site*.

ASCII

Abreviatura de *American Standard Code for Information Interchange*, desenvolvido pelo *American National Standards Institute*. ASCII é um esquema de notação binária para dados lidos em computadores. Os códigos entre 0 e 127 são padronizados, incluindo todo o alfabeto ocidental, o que permite a passagem de textos (formato TXT) entre computadores baseados em diferentes plataformas.

ASP

Active Server Pages: são páginas criadas dinamicamente pelo servidor *Web*, orientado por um programa em *VBscript* ou *Javascript*. Quando um *browser* solicita uma página do tipo ASP, o servidor constrói uma página HTML e a envia ao *browser*. A diferença entre uma página ASP e um documento HTML clássico, é que o segundo corresponde a um documento estático, que já se encontra no servidor no formato que em que será exibido no navegador. A página ASP, ao contrário, não existe no servidor: é montada a partir de uma solicitação específica. Pode ser, por exemplo, um documento HTML criado como resultado de uma pesquisa num banco de dados.

Backup

Jargão técnico utilizado para designar cópia de dados, geralmente feita para fins de segurança.

Banco de dados

Genericamente, é qualquer coleção de informações organizadas de tal forma que seja possível localizar itens escolhidos. Os bancos de dados são estruturados em campos, registros e arquivos.

Banco de dados distribuído

Estrutura com dois ou mais arquivos armazenados em diferentes locais numa rede de computadores. Para manter a consistência, as informações são sincronizadas.

Banco de dados relacional

Banco de dados cujas informações são armazenadas no formato de tabelas inter-relacionadas, enquanto que o banco não-relacional (flat-file database) resume todo o seu conteúdo numa única tabela.

Bit

Binary digit. É a unidade computacional de informação. Cada dígito binário pode ser somente duas situações (0 ou 1), assemelhando-se a um lâmpada, que pode estar acesa ou apagada.

Bitmap ou BMP

Uma representação binária na qual um *bit* ou conjunto de *bits* corresponde a parte de um objeto como uma imagem ou fonte. Por exemplo, nos sistemas monocromáticos, um *bit* representa um *pixel* na tela.

bps

bits por segundo, medida da velocidade de transferência de dados em uma linha de comunicação (cabos, *Modems*, etc.). Para Kpbs, Mbps, etc., por analogia, consulte *byte*.

Browser

A palavra "browse" significa examinar casualmente e um *browser* é um programa que permite a navegação da Internet e a visualização das páginas *Web*. Os *browsers* mais difundidos são, Netscape Communicator e o *Internet Explorer*.

Byte

Binary Term. É composto por oito *bits*, é a unidade utilizada para codificar um caractere. Um *byte* permite codificar 256 elementos diferentes.

CD-R

Compact Disc-Recordable. Diferente do CD-ROM, o CD-R aceita gravação do usuário uma única vez. Com a mesma capacidade para 640 *megabytes* de um CD-ROM, tem como vantagens a confiabilidade da mídia, sua universalidade (pode ser lida na maioria dos drives para CD-ROM) e preço relativamente baixo.

CD-ROM

Compact Disc Read-Only Memory: disco óptico de armazenamento de dados (na ordem de 640 *megabytes*), não volátil com o mesmo formato físico de um CD de áudio.

CD-RW

Tipo de CD com a mesma capacidade do CD-R, mas que aceita regravação. A regravação não se faz como num disco magnético, porque não se pode apagar um arquivo e gravar outro em seu lugar. Para sobrescrever, é preciso, antes, limpar todo o conteúdo do disco. Existem *softwares* que permitem gravar no CD-RW como se faz num disco magnético comum, no entanto, isso causa perda de 150 MB do espaço útil da mídia. A maioria dos novos gravadores de CDs trabalha com mídias CD-R e CD-RW.

CGI

Common Gateway Interface. É uma especificação que gerencia a transferência dados entre um servidor *Web* e o *browser* do usuário. Normalmente o programa CGI é executado no servidor, em resposta a uma solicitação do *browser* (por exemplo, quando se preenche um formulário). O código CGI pode ser escrito em diferentes linguagens, entre as quais *C*, *Perl*, *Java* ou *Visual Basic*. Outras formas de fazer esse processamento utilizam programas que rodam na máquina cliente: *Applets Java*, *Javascripts* ou controles *ActiveX*.

Chave primária

Campo usado para identificar registros. Numa tabela, cada valor de chave primária corresponde a um – e somente um – registro, e vice-versa.

Clustering

Técnica que consiste em conectar dois ou mais computadores para trabalhar como se fossem um, dividindo, assim, a carga de trabalho e aumentando a resistência a erros e o desempenho da rede. A palavra vem de cluster (agrupamento). Esses aglomerados de máquinas são usados especialmente em computação paralela (processamento paralelo).

Comando Ponto

Criação do autor, destina-se a agilizar a digitação de dados em campos específicos, buscando-as a partir de um banco de dados de apoio: ao digitar um ponto (.) como primeiro caractere do campo, o sistema permite que o usuário compare a expressão digitada após o ponto com o banco de dados de referência associado àquele campo e apresente a relação na tela para que o usuário selecione o registro desejado.

Compactação de dados

Processo no qual utilizam-se programas especiais para reduzir o tamanho dos arquivos. Os formatos de arquivos mais comumente utilizados nas compactações são ZIP e ARJ.

Compactação em vários volumes

Recurso dos programas de compressão que permite com que arquivos compactados grandes possam ser gravados em múltiplos discos.

Compressão de dados

A compressão de dados é especialmente útil nas comunicações porque permite a transmissão da mesma quantidade de informações em menor volume – e, em conseqüência, menor tempo. As técnicas comuns de compressão envolvem a eliminação dos espaços em branco e a substituição de padrões repetitivos por símbolos menores, que poupam espaço.

Consulta (*Query*)

Qualquer solicitação de pesquisa no banco de dados. Uma consulta típica pode ter a seguinte descrição: exibir todos os municípios do Brasil que possuem mais que 40.000 habitantes.

Cookie

Mensagem enviada ao *browser* por um servidor *Web*. Normalmente, essa mensagem é gravada no micro do usuário como um arquivo texto. Sempre que o usuário volta a acessar aquele *site*, a mensagem é reenviada ao servidor. O objetivo do *cookie* é identificar, pelo menos parcialmente, o usuário ou suas preferências e permitir exibir páginas personalizadas.

Correção de erros de *Modems*

Sistema utilizado por dois *Modems* conectados para verificar se os dados enviados numa ponta chegaram inalterados na outra. Quando um erro é detectado, o emissor reenvia a informação.

DAT

Digital Audio Tape (fita de áudio digital): – esta tecnologia foi originalmente criada para a gravação de sons. Menor que um cartucho de audiocassete, um cartucho de fita DAT de 4 milímetros armazena até 24 *gigabytes* (o equivalente a cerca de quarenta CD-ROMs) e transfere dados à uma velocidade de até 2 Mbps.

Data Mining

É uma expressão correspondente à mineração de dados. Refere-se ao uso de “ferramentas” que auxiliam a análise de dados de um *Data Warehouse*, proporcionando conclusões antes desconhecidas.

Data Warehouse

Banco de dados organizado para dar suporte à tomada de decisões estratégicas na gestão da empresa. O *Data Warehouse* usa dados de toda a corporação, rearranjados para facilitar a tomada de decisão.

DHTML

HTML dinâmico. Refere-se a páginas *Web* cujo conteúdo se modifica sem intervenção direta do webmaster. Uma mesma página pode ser vista de forma diferente, dependendo de variáveis como a localização geográfica do internauta, a hora da visita, as páginas anteriores já visitadas e as preferências do usuário. Várias tecnologias são usadas para produzir HTML dinâmico: scripts CGI, Server Side Includes (SSI), *cookies*, *Java*, *Javascript* e *ActiveX*.

Disco rígido removível

Drive contido em cartucho especial que lhe permite ser encaixado e retirado de seu local de funcionamento, como os discos flexíveis. Combina portabilidade com alta capacidade de armazenamento.

Distância focal

É a distância entre o foco principal e o centro da lente.

DNS

Domain Name System - sistema de mapeamento de nomes em endereços de rede, usado na *Internet*. É um sistema de associação distribuído que faz a correspondência de nomes de nós da rede (endereços do tipo HTTP://...) em endereços *Internet* (dado por números IP) e vice-versa. Consulte também TCP/IP

Downstream

Antônimo de *upstream*. Tráfego de dados no sentido do provedor para o usuário ou de um equipamento maior (um grande servidor) para um menor (um microcomputador pessoal ou um pequeno servidor de rede local).

Estabilizador de tensão

Aparelho que protege o computador ou outros equipamentos contra variações na rede elétrica como a supertensão e a subtensão. O dispositivo mantém a tensão estável 115 ou 220 volts (medida de força no circuito elétrico), conforme o caso.

Extranet

Rede de negócio para negócio que une empresas parceiras por meio de suas intranets. Por usar os padrões abertos da *Internet*, os parceiros não precisam usar o mesmo sistema operacional, *hardware* ou *browser*.

Fax-Modem

Veja *Modem*

Filtro de linha

Dispositivo destinado a suprimir ruídos (sinais externos que interferem nas transmissões elétricas e podem ser originados por linhas de alta tensão, produtos elétricos desajustados ou surtos de força) da rede elétrica que podem interferir no funcionamento de equipamentos sensíveis como o computador.

Firewall

Barreira de segurança baseada em *hardware* e *software* que protege a rede corporativa contra acessos externos não autorizados, como por exemplo os hackers da *Internet*. É o ponto de conexão da rede com o mundo externo – tudo o que chega passa pelo *firewall*, que decide o que pode ou não entrar, dependendo do nível de segurança criado pela empresa.

Flicker free (sem tremulação)

Monitor que não produz imagem trêmula que possa ser percebida pelos olhos humanos.

Frequência de varredura horizontal

Número de linhas iluminadas na tela de vídeo em um segundo. Uma resolução de 400 linhas renovada 60 vezes por segundo requer uma frequência de varredura de 24 KHz.

FTP

File Transfer Protocol: serviço para transferência de arquivos

GBM

Grupo Brasileiro Multidisciplinar e Multicêntrico para Estudo do Melanoma, formado por dermatologistas, oncologistas, especialistas em anatomia patológica, cirurgiões plásticos, etc.

Gigabyte

Gigabyte: 1024 *megabytes*. Veja *byte*.

GIF

Graphics Interchange Format, foi desenvolvido pela Comuserve e pode comprimir figuras a até 1 centésimo do tamanho original. A taxa de compressão varia muito, dependendo da imagem; quanto mais redundante a figura, maior a compressão. Para imagens complexas, sem padrões repetitivos, o melhor que se consegue é reduzir o tamanho a cerca de 80% do original. A principal limitação deste formato é que suporta apenas 256 cores, e entre as vantagens estão os recursos de transparência de fundos e animações.

Gravação multisessão

Possibilidade de gravar novos dados num CD-R ou CR-RW que já tenham dados gravados. Após cada gravação, cada sessão é fechada, mas o disco deve ficar aberto, para que seja possível adicionar mais informações.

Grupos de discussão

Espaços de trabalho na *Intranet* nos quais os participantes discutem um assunto, como o desenvolvimento de linhas de pesquisa ou casos clínicos de difícil diagnóstico. Os grupos podem ser abertos ou fechados (acesso restrito) e separados por assuntos.

GUI

Graphical User Interface. Usando imagens mais que apenas palavras para representar entradas e saídas de um *software*, a GUI é desenvolvida sob sistemas baseados em janelas, utilizando *ícones*, botões, caixas de diálogo, etc., que permitem ao usuário o controle das funções de um programa.

Hand helds

Equipamentos de dimensões que os permitem ser acomodados na palma da mão e que dispõem de recursos computacionais como os palmtops. (exs: Pilot III, V e VII, Cassiopéia E105, Compaq Aero, etc.).

Hardware

Componentes físicos de um ambiente computacional (CPU, discos, *Modem*, cabos, etc.).

HCI

É um conjunto de *software* e *hardware* que permite a um usuário interagir com um computador (consulte *Interface*)

HDTV

Televisão de alta definição

Hiperlink

Veja *link*.

Hipertexto

Banco de dados em formato especial, sobre o qual é possível visualizar e pesquisar as informações baseando-se em texto, imagens, arquivos de multimídia e programas ligados entre si por *hiperlinks*. Exemplos: páginas da *Web* e a Ajuda do *Windows*.

Hot swapping

Recurso que permite conectar um equipamento a outro dispositivo em funcionamento, sem que haja necessidade de desligá-lo para realizar a troca.

HotSync

Operação que sincroniza os bancos de dados do Palm localizados no *desktop* e no micro de mão.

HTML

Hypertext Markup Language: formato para apresentação de informações em sistema de hipermídia, que inclui recursos de hipertexto (texto que usa *hiperlinks*), animação, som, formulários para captação de dados, etc., usado na *Web*, e interpretado por um *browser*.

HTTP

Sigla de Hypertext Transfer Protocol. Usado para acessar informações na *World Wide Web*.

Ícone

Pequenas imagens que sugerem a atividade a ser realizada.

IDE

A *interface* IDE destina-se a conectar o computador a periféricos de armazenamento de dados. O padrão IDE original transferia dados à velocidade de 3,3 MB por segundo e estava limitado a discos de no máximo 528 MB. Uma evolução, o Enhanced IDE (EIDE), ou Fast IDE, opera ao ritmo de 12 MB por segundo e aceita discos de até 8,4 MB. Com esses números, o padrão aproxima-se da velocidade do padrão SCSI.

Imagens digitais

Imagens armazenadas em formato digital. A viabilização da sua utilização é decorrente da melhoria dos sistemas computacionais atuais.

Instant on

Os micros de mão não consomem tempo de inicialização. Basta ligar e estão prontos para o uso.

Interface

Jargão de informática utilizado para designar a fronteira através da qual dois sistemas se comunicam. Pode ser um conector de *hardware* usado para conexão com outros equipamentos, ou uma convenção para a comunicação entre dois *softwares*. Normalmente é usado para designar todo o conjunto envolvido, *hardware* ou *software*, na comunicação entre um usuário e o computador.

Interpolação

Método usado para aumentar a resolução de uma imagem utilizando um *software*. O aplicativo adiciona *pixels* à figura com base nos *pixels* ao redor – processo que, em certos casos, pode distorcer a imagem.

Inversor

Circuito que converte a corrente contínua fornecida por uma bateria em corrente alternada, exigida pela maioria dos equipamentos de computação.

IP

Internet Protocol: é o protocolo responsável pelo serviço da camada de inter-rede na arquitetura *Internet* TCP/IP

ISDN

Sigla de Integrated Services Digital Network, ou rede digital de serviços integrados (RDSI). Trata-se de um padrão de comunicação para o envio de voz, dados e vídeo em linhas telefônicas digitais que utiliza a mesma estrutura de cabos da telefonia convencional. O ISDN exige cabeamento metálico e *Modems* especiais. Trabalha com dois canais de 64 Kpbs, o que permite transmitir a 128 Kpbs.

Java

Veja Linguagem de programação

Javascript

Veja Linguagem de programação

Jaz Drive

Produzida pela Iomega, essa unidade removível trabalha com dois tipos de cartucho, de 1 e 2 GB. O produto pode ser usado para fazer *backup* ou para o trabalho no dia-a-dia.

JPG

O padrão JPEG foi criado pelo Joint Photographic Experts Group. As imagens em JPG aceitam que o grau de compactação seja definido pelo usuário. Quanto menor o arquivo obtido, menor também será a qualidade da imagem, embora o número de cores seja maior (16,7 milhões). O padrão JPEG é mais adequado para imagens mais complexas, como fotos que apresentam muitas cores e detalhes.

Jscript

script *Java*. Veja Linguagem de programação.

Kpbs

Quilobps. Veja bps.

Kbyte

Quilobyte: 1024 bytes. Veja *byte*.

LCD

Acrônimo de Liquid Crystal Display, ou seja, Tela de Cristal Líquido. LCD é a tecnologia utilizada na maioria dos visores de *notebooks*, agendas, calculadoras e outros aparelhos portáteis. Até pouco tempo atrás, o LCD era usado somente em equipamentos portáteis. A partir de 1997, no entanto, começaram a surgir micros de mesa com monitores de cristal líquido. A principal vantagem é a economia de espaço. Em contrapartida, os sistemas de LCD têm preços muito mais altos.

Line interactive

Tipo de *no-break off-line* que monitora a rede elétrica e só aciona as baterias quando há problemas. No caso de uma queda total, ocorre uma interrupção de alguns milissegundos, tempo necessário para que o inversor seja acionado e entre em fase com a alimentação que provinha da rede elétrica.

Linguagem de programação

Linguagem formal na qual programas de computador (*softwares*) são escritos. A definição de uma linguagem particular consiste de sua sintaxe (como os vários símbolos podem ser combinados) e semântica (o significado das construções da linguagem).

Link

Seqüência de código que permite vincular um documento ao outro.

Mainframe

São computadores com grande capacidade de processamento, para sua época, de alto custo, utilizado em empresas.

Mbps

Megabps. Milhões de bps. Veja bps.

Megabyte

Megabyte: 1024 *quilobytes*. Veja *byte*.

Memória RAM

Random access memory, memória volátil, de acesso não seqüencial, interna ao computador, onde os programas são executados, da ordem de centenas de milhões de caracteres

Memória ROM

Read only memory, memória apenas para leitura, que é instalada no *hardware* de computadores para conter rotinas básicas ligadas ao seu funcionamento.

Modem

Modulator/demodulator: equipamento eletrônico que converte dados seriais de um computador em um sinal de áudio, tipicamente utilizado para transferir dados através de linhas telefônicas.

Modem a cabo

Periférico que se comunica usando as linhas de TV a cabo. Sua vantagem em relação ao sistema telefônico é a velocidade: pode alcançar até 2 Mbps.

Modem PC Card

Dispositivo do tamanho de um cartão de crédito e compatível com o padrão PCMCIA. Usado principalmente em equipamentos portáteis.

Monitor

É um periférico de saída e é o principal meio para visualização do usuário quando este utiliza um microcomputador. É neste dispositivo que normalmente é apresentada a *interface* para o usuário. Tradicionalmente é construído como um tubo de raios catódicos e componentes eletrônicos associados, mas atualmente também são disponíveis em outras modalidades (consulte LCD, tela de gás-plasma, tela de matriz ativa, tela de matriz passiva).

MPEG

Acrônimo de Moving Picture Experts Group. É um formato de compressão de arquivos de vídeos que possibilita apresentar vídeos com resolução de pelo menos 30 quadros por segundo. O padrão MPEG-2 é usado nos DVD-ROMs, e pode operar com imagens de até 1.280 por 720 *pixels*, a 60 quadros por segundo e som com qualidade de CD.

MPEG-3

MPEG camada 3, sistema de gravação de áudio definido para a HDTV. A capacidade de compactação sem praticamente perda de qualidade é na razão de 12 para 1. Os arquivos de áudio com esse padrão têm extensão MP3.

MTBF

Medium time Between Failures - Tempo médio entre falhas

MTTR

Medium Time to Repair - Tempo médio de reparo

Multimídia

Recurso computacional que permite utilizar texto, imagem, som e animação para maior interação com o usuário.

Multiprocessamento

Trabalho simultâneo executado por dois ou mais processadores na mesma máquina, ou dois ou mais computadores interligados.

Multithreading

Vários processos (multitarefa) são executados dentro de um mesmo programa. Essa característica, embora deva ser implementada no aplicativo, precisa ser suportada pelo sistema operacional. Cada processo deve ser projetado de tal forma que todos possam rodar ao mesmo tempo sem interferir uns nos outros.

No-break

Termo em inglês para UPS externo. Curiosamente, essa expressão é usada somente no Brasil.

No-break microprocessado

No-break capaz de se comunicar com o microcomputador por meio de uma porta serial. Usa um *software* de controle que monitora as condições da rede elétrica e toma decisões como desligar um servidor de rede. Também conhecido como UPS Inteligente.

No-break off-line

Quando ocorre uma queda na rede elétrica, o dispositivo pára de fornecer energia por alguns milésimos de segundos, o que não determina o desligamento dos aparelhos a ele conectados. Há dois tipos de *no-breaks off-line*: *stand by* e *line interactive*.

No-break on-line

Ao contrário do modelo *off-line*, esse tipo de *no-break* não interrompe o fornecimento quando há uma queda de energia. A saída é alimentada constantemente por uma combinação de recursos da rede elétrica ou da bateria.

Notebooks

Equipamentos portáteis com capacidades semelhantes aos equipamentos de mesa (*desktops*), pesando, na maioria dos modelos, abaixo de 3 Kg.

OCR

Sigla de reconhecimento óptico de caracteres. Quando um documento de texto é digitalizado no computador, ele se transforma numa imagem digital. O *software* OCR analisa as áreas de luz e escuras da imagem para identificar cada uma das letras e dos dígitos. Os caracteres reconhecidos são convertidos em texto ASCII.

Palm tops

Computadores de dimensões semelhantes às de agendas eletrônicas, incorporando recursos (processador de textos, planilhas eletrônicas, banco de dados, acesso a *Web*, videoconferência, etc.) que eram habitualmente encontrados apenas em microcomputadores de maior porte.

PCMCIA

Padrão de conexão para os dispositivos do tipo *PC Card* (Cartão PCMCIA).

PCMCIA, cartão

Personal Computer Memory Card International Association: um módulo removível com dimensões similares às de um cartão de crédito para expandir as capacidades dos computadores *notebook*. Os cartões incluem funções de disco removível, placa de rede, *fax-Modem* e várias outras.

PCS

Personal Communications Service. É um sistema digital de telefonia móvel também chamado de banda C, que permite transmissão de vídeos.

PDF

Portable Document Format, ou Formato de Documento Portátil, usado pelo *software Adobe Acrobat*. Esse programa converte qualquer documento gráfico em arquivos PDF, os quais podem ser lidos ou impressos em qualquer sistema onde haja o Acrobat Viewer (visualizador), sem a necessidade da aplicação original que gerou o documento.

Perl

Veja Linguagem de programação

Pico de tensão

Também chamado de impulso, o pico é o aumento momentâneo e drástico na tensão. Um pico pode penetrar em um equipamento eletrônico através de linhas de corrente alternadas, linhas seriais de redes ou telefônicas e danificar ou inutilizar completamente os componentes. É causado tipicamente por queda de um raio nas proximidades, mas também pode ocorrer quando o fornecimento de energia é restabelecido após ter sido interrompido. Isto causa danos irreversíveis de *hardware* e perda de dados.

Pixel (picture element)

Elemento de imagem. A menor unidade que pode ser endereçada na tela. Quanto maior for a resolução (quanto mais linhas e colunas de *pixels*), tanto mais informação pode ser exibida.

Placa de vídeo

Uma placa de expansão que se conecta no micro e converte as imagens criadas no computador em sinais eletrônicos. Determina a resolução e a taxa de renovação máxima, além do número de cores que podem ser enviados.

Plug and play

Ligue e opere, em português. A expressão é utilizada para identificar sistemas fáceis de usar e instalar, já configurados e prontos para a operação do usuário final.

Porta infravermelha

Via de comunicação entre micros e periféricos. Permite a transmissão de dados de um aparelho para outro sem o uso de cabos, na forma de raios infravermelhos.

Portas paralelas

Interfaces para conectar dispositivos externos. No *PC*, a porta paralela padrão (Standard Parallel Port, ou SPP) era utilizada quase exclusivamente para ligar o micro a impressoras. Há dois tipos mais recentes: a EPP (Enhanced Parallel Port) e a ECP (Extended Capabilities Port). Ambas suportam comunicação bidirecional e operam com taxas de transferência dez vezes maiores que a SPP. Existem muitos periféricos que usam a porta paralela. Exemplos disso são o *ZIP Drive* e equipamentos como *scanners* e gravadores de CD.

Portas seriais

Interfaces de comunicação que transmitem informação em série, ou seja, um *bit* e depois outro. No *PC*, essas portas são muito usadas por *mouse* e *Modems*.

Potência (em *no-break*)

Normalmente a potência dos *no-breaks* é indicada em VA (Volt Ampère). Em geral, o consumo da carga não deve ultrapassar 90% da potência do *no-break*.

Processamento paralelo

O mesmo que computação paralela. Esse conceito envolve dois entendimentos. No primeiro, um computador executa mais de uma operação ao mesmo tempo. No outro, vários computadores fazem as tarefas em conjunto. Em princípio o processamento paralelo faz um programa rodar mais rápido, porque usa mais CPUs. Na prática, não é simples dividir o programa em partes para tirar proveito dessa arquitetura.

Profundidade da cor

Número de cores que pode ser representado em cada *pixel*. Uma imagem em preto e branco tem profundidade 2 (2 elevado à potência 1). Nela, cada *pixel* está aceso ou apagado. Da mesma forma, numa figura de 24 *bits*, a cor em cada ponto pode ser qualquer uma do total de 2 elevado à vigésima quarta potência (cerca de 16,7 milhões de tonalidades).

Provedor de acesso

Um provedor de acesso à *Internet* é uma empresa que fornece aos seus clientes, a conexão à *Internet*.

Quedas curtas de tensão

Esta ocorrência é representada por queda breve na tensão. É causada quando ligamos muitos dispositivos elétricos (motores, compressores, elevadores, etc.) ao mesmo tempo. Uma queda de tensão interrompe o suprimento de energia por um intervalo de tempo muito curto, o que pode travar o teclado e o processamento do sistema, resultando em perda ou corrupção de dados. As quedas curtas de tensão também reduzem a eficiência e a vida útil de equipamentos elétricos e podem queimar as saídas seriais.

Quilobytes

Veja *Kbyte*.

Resolução

Número de *pixels* usados para capturar ou exibir uma imagem. A resolução VGA padrão é 640 *pixels* na horizontal e 480 na vertical. O Super VGA (SVGA) trabalha com valores de 800x600 e 1024x768. Há quem chame de SSVGA as resoluções acima de SVGA, como 1600x1200. Quanto maiores os números, maiores os detalhes da imagem.

Resolução óptica

A resolução de um dispositivo de digitalização, contrastando com uma resolução interpolada, em que a imagem é otimizada pelo *software* (interpolação). Ambos são dados em dpi (dots per inch, ou pontos por polegada).

Ruído

Os distúrbios causados por ruídos elétricos interferem com o fluxo que distorce a onda senoidal pura que se espera no fornecimento de energia. É causado por muitos fatores e fenômenos, inclusive raios, chaveamentos de carga, geradores, transmissores de rádio e equipamentos industriais. Podem ser intermitentes ou permanentes. Isto gera problemas e erros em programas executáveis e arquivos de dados e pode ocasionar a queima de componentes de *hardware*.

SCSI

Abreviatura de Small Computer System *Interface*. É uma modalidade de *interface* de alta performance de comunicação, permitindo grandes taxas de transferência de dados.

Servidor de impressão

Equipamento da rede de computadores encarregado de controlar impressoras e filas de impressão.

Servidor Web

Central que fornece informações quando se faz uma pesquisa na *Internet* utilizando um *browser*.

SGML

Standard Generalized Markup Language. Linguagem genérica para formatação de documentos. O XML corresponde a uma versão reduzida do SGML aplicado somente à *Web*.

Short-break

O mesmo que *no-break off-line*. Sigla de Standby Power Supply. Equivale ao UPS do tipo line-interactive.

S-HTTP

Extensão do protocolo HTTP para enviar dados com segurança na *Web*.

Sistema operacional

software que controla o funcionamento de um equipamento computacional

SMTP

Simple Mail Transfer Protocol - que oferece serviço store-and-forward para mensagens que carregam correspondências contendo texto

Software

Uma série de instruções para o computador que executam uma tarefa particular.

SQL

Structured *Query* Language, ou linguagem estruturada de consultas. Criada pela *IBM*, é uma ferramenta para extrair informações de bancos de dados.

SSI e SS Extensions

Sigla de Server Side Include. São extensões da linguagem HTML incluídas no código da página *Web* que permitem que o servidor *Web* insira novas informações na página no momento em que ela é solicitada.

SSL

Sigla de Secure Sockets Layer. Protocolo para transmissão de documentos com segurança via *Internet*.

Standby by, no-break

Tipo de *no-break off-line*. Quando há queda na rede elétrica, as baterias entram em ação, fornecendo energia ao inversor.

Super Disk

Criado pela Imation, é uma tecnologia de armazenamento que trabalha com disquetes de alta densidade. Ao contrário do *ZIP Drive*, o *Super Disk* é compatível com os disquetes comuns de 1,44 MB, ou seja, seu drive – chamado LS-120 – lê e escreve nesses discos e também nos seus próprios, que armazenam até 120 MB.

Surto

Aumento substancial da tensão durante uma pequena fração de segundo, em geral causado quando aparelhos de grande potência são desligados.

Surto de tensão

É o aumento breve na tensão, durante uma fração de tempo de pelo menos 1/120 de segundo. É causado por motores elétricos de alta potência, como condicionadores de ar e eletrodomésticos em áreas próximas. Quando esse tipo de equipamento é desligado, uma tensão extra é transmitida através da linha de transmissão. Os computadores e dispositivos eletrônicos são projetados para receber energia dentro de uma determinada faixa de tensão. Qualquer ocorrência fora dos níveis de tensão médio esperado causará uma sobrecarga sobre componentes delicados, que podem queimar, ou provocar uma falha prematura do sistema.

Taxa de recarga

É o tempo que leva para o *no-break* carregar totalmente as baterias depois delas terem sido totalmente descarregadas.

TCP

Transmission Control Protocol - é um protocolo de transmissão de dados confiável.

TCP/IP

Sigla de Transmission Control Protocol/*Internet* Protocol. Protocolo usado na comunicação entre computadores de redes diferentes. Compõe uma suíte de protocolos (HTTP, FTP, NNTP e mais de 100 outros) que formam a espinha dorsal dos padrões da *Internet*. O IP garante o endereçamento de todas as máquinas na *Internet* e o encaminhamento das mensagens entre essas várias máquinas.

Tecnologia de impressão

Sistema no qual se baseia uma impressora: matricial, a jato de tinta, a *Laser*, térmica. Atualmente, as máquinas a jato de tinta são as mais vendidas seguidas das máquinas a *Laser*.

Tela de gás-plasma

Tipo de tela usada em micros portáteis. Baseia-se num dispositivo que mantém gás neon entre duas placas condutoras, uma contendo fios que formam as linhas horizontais e a outra, as verticais. Com a ação da corrente elétrica, o gás brilha nas interseções dos fios formando os *pixels* (pontos da tela). As imagens nas telas são monocromáticas e apresentam-se como objetos alaranjados sobre um fundo preto.

Tela de matriz ativa

Monitor plano que usa tecnologia LCD no qual cada ponto da imagem é controlado por um transistor e atualizado com mais frequência que nas chamadas telas de matriz passiva. Os *notebooks* de produção mais recente são normalmente equipados com telas de matriz ativa. Com isso oferecem imagens de melhor qualidade, além de reduzir a sensação de tremor.

Tela de matriz passiva

Uma tecnologia LCD comum usada em computadores *notebooks*. Os monitores de matriz passiva (DSTN, CSTN etc.) não são tão claros e definidos e não têm um ângulo de visão tão amplo quanto os monitores de matriz ativa.

Telnet

Programa e protocolo utilizados para fazer ligações remotas entre vários computadores.

Tolerância a falhas

Habilidade de um sistema de responder adequadamente a falhas inesperadas de *hardware* ou *software* no servidor, garantindo a integridade dos dados.

Transiente

Um problema que ocorre freqüentemente de forma intermitente. Um exemplo seria a tensão fornecida ao computador ficar instável devido a problemas na rede elétrica.

TXT (formato texto)

São arquivos gravados sem formatação, utilizando padrão de código ASCII, e podem ser lidos por diferentes plataformas computacionais.

UCI

Unidade Celular de Informação. Definição original do autor, é decorrente do método utilizado para organizar os dados em tabelas de um banco de dados, muitas vezes com alguma redundância, de forma a aumentar o grau de independência entre elas. Entre as vantagens, considera-se que o uso de UCI facilita a legibilidade dos dados, atividades que dependam de documentação temporal (*Data Warehouse*), transmissão de dados para outros sistemas, etc.

UPS

Sigla de Uninterruptible Power Suply. Fonte de alimentação que utiliza baterias para manter o fornecimento de energia mesmo quando ocorre uma falha da rede elétrica. Além disso, pode também ter as funções de filtro de linha e estabilizador. Normalmente, o UPS é apresentado como um equipamento externo, ao qual se ligam computadores ou outros equipamentos. A finalidade básica do dispositivo não é substituir a rede elétrica, e sim manter os equipamentos funcionando por tempo suficiente para que se possa desligá-los de forma normal, salvando os arquivos.

Upstream

Tráfego de dados no sentido do usuário para o provedor.

Upstream

Veja *Downstream*

USB

Sigla de Universal Serial Bus, ou barramento serial universal. Novo padrão para conectar externamente ao micro uma série de periféricos, como *mouse*, *Modem* e teclado, sem a necessidade de instalar placas e reconfigurar o sistema. A tecnologia USB é suportada pelo *Windows 98*. Uma entrada USB aceita a conexão simultânea de até 127 periféricos. Além disso, as conexões podem ser feitas a quente, ou seja, com a máquina ligada.

VBscript

Visual Basic. Veja Linguagem de programação

Visual Basic

Veja Linguagem de programação

W3

Veja *Web*

Web

Sistema de informação distribuído, de arquitetura cliente-servidor, originado dos Laboratórios de Física de Alta Energia em Genebra, Suíça. No início dos anos 1990 a *Web* começou a espalhar-se pelo mundo e em 1993 passou a compartilhar informações utilizando a estrutura física da *Internet*. Originalmente na *Web* tudo é representado para o usuário como hipertextos, usando o formato HTML. O programa cliente, conhecido como *browser* (*NCSA Mosaic*, *Netscape Navigator*, *Internet Explorer*, etc.) é executado no computador do usuário e executa duas operações básicas: seguir um *link* ou enviar uma solicitação a um servidor. Novos recursos de multimídia, e interação facilitada com outros serviços disponíveis na *Internet* têm sido progressivamente acrescentados aos *browsers*. Sinônimia: WWW, W3, *World Wide Web*.

Windows CE

Sistema operacional que tem três versões, uma para cada tipo de micro WinCE: o *HP/PC*, de mão, horizontal; o *P/PC*, também de mão, vertical como o *Palm*; e o *H/PC Pro*, do tamanho de um subnotebook.

World Wide Web

Veja *Web*.

WWW

Veja *Web*.

XML

Sigla de eXtended Markup Language. Linguagem de descrição de páginas extensível e personalizável, que pretende colocar um fim nos problemas de dialetos incompatíveis de HTML; uma espécie de HTML estendido que está sendo desenvolvida pela W3C. Essa linguagem permite que o designer de páginas *Web* crie tags próprias não existentes no HTML. Com o XML pode-se criar *links* apontando para mais de um documento, diferentemente do que ocorre com os *links* do HTML, que sempre se referem a um endereço único.

ZIP

O mais conhecido formato de arquivos compactados. Tornou-se popular com os programas *PKZip* e *PKUnzip*, da PKWare (WWW.pkware.com), *WinZip*. Não deve ser confundido com *ZIP Drive*

ZIP Drive

Disco flexível de alta capacidade desenvolvido pela Iomega. São mais espessos e ligeiramente maior que os disquetes de 3,5 polegadas; têm versões de 100 e 250 MB. O *ZIP Drive* é hoje uma das mídias mais populares para *backup* ou transporte de dados. Não deve ser confundido com o formato ZIP para compactação de dados.

Zoom digital

Este tipo de zoom apenas muda a forma de apresentar a imagem. A porção central da figura é exibida, em resolução menor, na área total que ela ocupa, dando a impressão de que foi ampliada.

Zoom óptico

Mudança da distância focal para permitir que o objeto se torne mais próximo sem que o fotógrafo saia do lugar ou o foco seja perdido. Esse efeito é obtido mediante uma combinação de lentes, que amplia a imagem antes de ela ser detectada pelo sensor. Por essa característica, o *zoom óptico* é mais eficiente que o digital.

Referências Bibliográficas

1. Adams P, Powell J. *FoxPro 2.5 - Guia de Desenvolvimento Avançado para Multiusuários*. Tradução de von Riegger R. Berkeley Brasil Editora, 1994.
2. Adobe. *Adobe Premiere LE – User Guide*, 1995.
3. *Apoio ao Diagnóstico do Melanoma*. Grupo Brasileiro Multidisciplinar e Multicêntrico para estudo do Melanoma, Saúde Total, 1997-2000 (vários autores).
<URL:<http://www.saudetotal.com/melanoma/apoiodia.htm>>
4. Basili V, Zelkowitz M. Analysing medium scale *software* development. *Proc. 3rd Intl. Conf. Software Engineering, IEEE*, pp. 116-123, 1978.
5. Bates DW, Cullen DJ, Laird N, Petersen LA, Small SD, Servi D, Laffel G, Sweitzer BJ, Shea BF, Hallisey R, Vliet MV, Nemeskal R, Leape LL. Incidence of Adverse Drug Events and Potential Adverse Drug Events. **JAMA**, 274:29-34, 1995.
6. Böhm GM. Aula inaugural do curso de Disciplina de Informática Médica. *Hosp Clín Fac Med São Paulo*: 42(5):191-194, 1987.
7. Brooks F. *The Mythical Man-Month*. Addison-Wesley, 1975.
8. Caetano N. *Guia do Paciente - Guia de Remédios em Linguagem Clara*. 2ª edição, BPR, 1997.
9. Celestino F. A history of the computer. *Flavio Celestino Home Page (Politecnico di Torino)*.
<URL:http://www.polito.it/~fcelesti/comp_hist>
10. Chao LW. *Apoio para uso do Tot-Win para o Usuário VIP*. Intec-System Ltda, 2000.
<URL:<http://www.saudetotal.com/intec/vip/>>
11. Chao LW, Silveira PSP, Massad E. GClin - a system for physician's office management. *Medinfo '89 (VI World Congress on Medical Informatics)*. Cingapura, República de Cingapura, de 10 a 14 de dezembro. *Congress Program*, p. 49-50, 1989.
12. Chao LW. Como estar em dois lugares ao mesmo tempo. *Med.Web*, 3:23, 1999.
13. Chao LW. Informatize-se agora. *Med.Web*, 1:28-29, 1999.
14. Chao LW. Processador de faturamento hospitalar segundo tabela AMB. *II Congresso Brasileiro de Informática em Saúde, III Exposição Técnica de Informática em Saúde e II Congresso Nacional de Informática Aplicada*. São Paulo, 24 a 28 de outubro, 1988.
15. Chao LW. Que equipamentos colocar no consultório? *Revista Farmacêutica Kairos*, 89:54-28, 1996.
16. Coleção *InfoExame*. Editora Abril, fascículos mensais de jan/1996 a fev/2000.
17. *Computação*. MeWolf & Cia Inc.
<URL:<http://www.elogica.com.br/users/nsn/computa.htm>>

18. Computers: history and development. *Departamento de Engenharia Informática, Instituto Superior de Engenharia, Instituto Politécnico do Porto.*
<URL:http://www.dei.isep.ipp.pt/works/it_10dei>
19. Conselho Federal de Medicina. *Processo Nº 1.345 / 93, assunto: Informatização de Consultórios e Clínicas.* 6 de maio de 1993.
20. Conselho Federal de Medicina. *Processo Nº 1.331 / 89, em relação à manutenção do prontuário médico.* 21 de setembro de 1989.
21. Conselho Regional de Medicina do Estado de São Paulo. *Consulta Nº 44.432 / 96 – Em relação ao período retroativo dos prontuários a serem guardados em decorrência do encerramento das atividades do Hospital Umberto I.* 27 de dezembro de 1996.
22. Conselho Regional de Medicina do Estado de São Paulo. *Consulta Nº 28.549 / 96, assunto: Microfilmagem de Prontuários – Informatização de ficha de Prescrição Médica.* 25 de setembro de 1996.
23. Conselho Regional de Medicina do Estado de São Paulo. *Consulta Nº 013 / 92, assunto: Laboratório de Análises Clínicas - Responsabilidade.* 27 de outubro de 1992.
24. Conselho Regional de Medicina do Estado de São Paulo. *Consulta Nº 1.326 / 92, assunto: Tempo de guarda de prontuários médicos e exames laboratoriais.* 26 de outubro de 1992.
25. Conselho Regional de Medicina do Estado de São Paulo. *Consulta Nº 1.435 / 92, assunto: Perigo de consultas furtivas, burlando o esquema de segurança de restrição no computador.* 09 de março de 1992.
26. Cringely RX. *Triumph of the nerds: history of the computer.* PBS
<URL:<http://www.pbs.org/nerds/timeline>>
27. Del Fiol G. *Sistema de Apoio à Prescrição: prevenindo eventos adversos relacionados ao uso de medicamentos.* Dissertação de Mestrado apresentada à Pontifícia Universidade Católica do Paraná, Curitiba, 89 pp., 1999.
28. Edição especial Roche. *Fichário Clínico Terapêutico.* Editora Massan do Brasil Ltda, 1984.
29. Ferreira M, Jarabeck FJ. *Programação Orientada ao Objeto em Clipper 5.0.* Editora Makron Books, 1991.
30. Freedman DP, Weinberg GM. *Handbook of Walkthroughs, Inspections and Technical Review*, 3^a ed. Dorset House, 1990.
31. Freedman DP, Weinberg GM. *Handbook of Workthroughs, Inspections, and Technical Reviews.* 3rd ed., Dorset House, 1990.
32. Goldschmidt A, Akera A. John W. Mauchly and the development of the ENIAC computer: an exhibition in the Department of Special Collections, Van Pelt Library, Department of History and Sociology of Science, University of Pennsylvania.
<URL:<http://www.library.upenn.edu/special/gallery/mauchly/jwmintro.html>>

33. Grupo Brasileiro Multidisciplinar e Multicêntrico para estudo do Melanoma. *Saúde Total*, 1997-2000.
<URL:<http://www.saudetotal.com/melanoma>>
34. HeilerSoftware. *HighEdit 4 OCX – Guide*. 1996.
35. Howe D. *FOLDOC: Free On-Line Dictionary Of Computing*. 1993-2000.
<URL: <http://wombat.doc.ic.ac.uk/foldoc>>
36. Intel Corporation. *Intel Products and Services*.
<URL: <http://www.intel.com/intel/product>>
37. Jasc Software. *Paint Shop Pro 6.0 – User’s Guide*. 1999.
38. Javitt J. *Computers in Medicine - Applications and Possibilities*. W. B. Saunders Company, 1986.
39. Johnston JA, Bootman JL. Drug-related morbidity and mortality: a cost of illness model. *Arch Intern Med* **155**:1949-1956, 1995.
40. Jones TC. *Programming Productivity*. McGraw-Hill, 1986.
41. Kimball R. *Data Warehouse Toolkit*. Tradução de Rosemberg M. Editora Makron Books, 1998.
42. Klatzky R. *Human Memory*. 2ª edição, W. H. Freeman and Co., 1980.
43. Kneale D. Coping with technofright, technology in the workplace. *The Wall Street Journal*, 16 setembro, p. 98, 1985.
44. Krupinski EA, Lesueur B, Ellsworth L, Levine N, Hansen R, Silvis N, Sarantopoulos P, Hite P, Wurzel J, Weinstein RS, Lopez AM. Diagnostic accuracy and image quality - using a digital camera for teledermatology. *Telemedicine Journal* 5:257-263, 1999.
45. Lazarou J, Pomeranz BH, Corey PN. Incidence of adverse drug reactions in hospitalized patients. *JAMA*, 1998; 279:1200-1205.
46. Lientz B, Swanson E. *Software Maintenance Management*. Addison-Wesley, 1980.
47. Massad E, Böhm GM, Chao LW, Silveira PSP. O universo da informática e o ensino médico. *Educação Médica*. Savier Editora de Livros Médicos Ltda, 1998, 211-222.
48. Melayu B. History of the computer industry: the first record keepers and their tools. *SMKTS World Online*.
<URL:<http://www.cybermalaysia.com/smkts/online/hismain.htm>>
49. Meyers J. A short history of the computer. *Softlord Dot Com*.
<URL:<http://www.softlord.com/comp>>
50. Microsoft Official Curriculum. *Programming in Microsoft Visual FoxPro 3.0 for Windows*. Student Workbook, 1995 – Course 455 – PN 5748A.
51. Microsoft. *Getting Started with FrontPage 98*. Microsoft Corporation, 1997.
52. Microsoft. *MSDN Visual Studio 6.0 Visual FoxPro for Windows – User’s Guide*. Editado em CD-ROM por Microsoft Inc., 1998.

53. Microsoft. *MSDN Visual Studio 6.0 Visual FoxPro for Windows – Programmers Guide*. Editado em CD-ROM por Microsoft Inc., 1998.
54. Microsoft. *MSDN Visual Studio 6.0 Visual FoxPro for Windows - Reference*. Editado em CD-ROM por Microsoft Inc., 1998.
55. Monk A. *Fundamentals of Human-Computer Interaction*. Academic Press, 1984.
56. Musa JD, Lannino A, Okumoto K. *Engineering and Managing Software With Reliability Measures*, McGraw-Hill, 1987.
57. Museu dos Computadores *Apple II II (1976-1990)*
<http://www.triang.com.br/waldolar/museu1.htm>
58. NSF SUCCEED Engineering Visual Database. *Charles Babbage's Analytical Engine*.
<URL:<http://www.ce.vt.edu/evd/Htmls/P403735.html>>
59. Occhiogrosso J. Tradução ARX Publicações Ltda. *Biblioteca do Programador Clipper 5.01*. Editora Makron Books, 1992.
60. Oliveira Filho J, Chao LW. Banco de Imagens para Elucidação Pré-Cirúrgica. Apresentação Oral para o VII Congresso Brasileiro de Cirurgia Dermatológica, São Paulo, SP, 19 a 22 de julho, 1995.
61. Oliveira Filho J, Chao LW. Jor-Info: coletânea de casos clínicos das jornadas dermatológicas (1983-1995) e jornadas informatizadas de dermatologia (1995-1998) da Sociedade Brasileira de Dermatologia (Regional São Paulo) em CR-ROM. Sociedade Brasileira de Dermatologia, Regional São Paulo, 1998.
62. Oliveira Filho J, Sampaio AM, Chao LW. Em discussão, a informática médica - software e programas de gerenciamento médicos disponíveis. *Dermatológica - Galderma*, **18**:4-11, 1996.
63. Pressman RS. *Engenharia de Software*. Editora Makron Books, 1995.
64. Ricky's Home Page. *História da Computação*.
<URL:<http://www.geocities.com/SiliconValley/Haven/8692/comput.htm>>
65. Sanford G. History of *Apple II Computers*, 1996-1999.
<URL:<http://www.Apple II II-history.com>>
66. Savola T. *Usando o HTML*. Editora Campus, 1995.
67. Schiff GD, Rucker TD. Computerized prescribing: building the eletronic infrastructure for better medication usage. *JAMA*, **279**:1024-1029, 1998.
68. Shneiderman B. *Designing the User Interface*, Addison-Wesley, 1987.
69. Shortliffe EH, Perreault LE. *Medical Informatics – Computer Applications in Health Care*. Addison-Wesley Publishing Company, 1990.
70. Smith RJ, Gibbs M. Tradução - Kurban A. *Navegando na Internet*. Berkeley Brasil Editora, 1995.
71. Soares LFG, Lemos G, Colcher S. *Redes de Computadores*. Editora Campus, segunda edição, 1995.

72. Sociedade Brasileira de Dermatologia Regional SP, Chao LW. Agenda do Dermatologista 2.000. 1999.
73. Swanson EB. The dimensions of maintenance, *Proc. 2nd Intl. Conf. Software Engineering, IEEE*, pp 492-497, outubro de 1976.
74. Tachakra S. Colour perception in telemedicine. *Journal of Telemedicine and Telecare*, 5:211-219, 1999.
75. The MacTutor History of Mathematics Archive. *Blaise Pascal*. Turnbull WWW Server, School of Mathematical and Computational Sciences, University of St Andrews.
<URL:<http://www-groups.dcs.st-and.ac.uk/~history/Mathematicians/Pascal.html>>
76. The Science Museum London. *Babbage's Calculating Engines - 1832-71*.
<URL:<http://www.nmsi.ac.uk/on-line/treasure/objects/1862-89.html>>
77. *The Virtual Museum of Computing*.
<URL:<http://www.nma.gov.au/vlmp/computing.html>>
78. Tobia E. History of the operating system. *Bryant College*, 1999.
<URL:<http://acad.bryant.edu/~ept1>>
79. Tracz W. *Software reuse: emerging technology. IEEE Computer Society Press*, 1988.
80. University of Pennsylvania. *ENIAC-on-a-Chip*. Moore School of Electrical Engineering.
<URL:<http://www.ee.upenn.edu/~jan/eniacproj.html>>
81. Werkgroep Geschiedenis der Elektrotechniek. *The History of the Computer*. Delft University of Technology
<URL:<http://historia.et.tudelft.nl/wggesch/geschiedenis/computer>>
82. Zanini AC. *Dicionário de Medicamentos Genéricos*. Editora Ipex, 1999.

Índice Remissivo

A

ActiveSync · 130
ActiveX · 130, 132, 134
Adobe · 40, 141
ADSL · 14, 15, 94, 130
AGP · 130
 Aiken, Howard · 6
Apple · 9, 10, 28, 29
 Macintosh · 10, 28, 29, 91
Apple II · 9
Applet · 130, 132
ARJ · consulte Compactação de dados, formato ZIP
ARPA · 7
 ARPANET · 13
ASCII · 111, 127, 131, 140, 145
 TXT · 62, 131, 145
Ashton Date · 29
ASP · 131
Atlas · 7

B

Babbage, Charles · 4, 5
 Máquina Analítica · 4, 5
 Máquina de Diferenças · 4, 5
Backup · 62, 85, 114, 131, 138, 147
 Banco de dados · 32, 33, 44, 47, 50, 51, 54, 65, 66, 67, 77, 92, 94, 95, 97, 109, 112, 127, 131, 134, 136, 144
 Chave primária · 132
 DBF, padrão · 30, 40, 44, 90
 distribuído · 131
 Query (consulta) · 60, 133, 144
 relacional · 95, 131
 SQL · 41, 43, 44, 144
 Server · 44
 Trigger · 44
 Berners-Lee, Tim · 13
 Bit · consulte Byte, Bit
 Bitmap · consulte BMP
 Bits por segundo · consulte bps
Blinker
 1.50 · 91
 BMP · consulte Imagens digitais
 bps · 14, 39, 130, 131, 133, 137, 138, 139
 Browser · consulte Web, Browser
Byte · 42, 131, 132, 135, 138, 139
 Bit · 9, 10, 130, 131, 132, 142
 gigabyte · 11, 39, 41, 42, 43, 48, 93, 115, 133, 135
 megabyte · 11, 14, 29, 39, 48, 89, 90, 92, 93, 115, 130, 132, 135, 139
 quiloobyte · 9, 14, 29, 90, 138, 139, 142

C

Cartão PCMCIA · consulte PCMCIA

CD · 39, 102, 115, 132, 139, 142
 CD-R · 115, 127, 132, 135
 CD-ROM · 48, 84, 119, 132
 CD-RW · 115, 132
CDX · consulte Fox, Visual FoxPro, CDX
CERN · 13
CGI · 132, 134
 Chave primária · consulte Banco de dados
Chip · 7, 8, 9
Clustering · 132
CMOS · 8
 Código Fonte · consulte Software, Código fonte
Colossus · 6
 Comando Ponto · 50, 56, 60, 67, 109, 133
 Compactação
 de dados · 133, 147
 em vários volumes · 133
 Compactação de dados
 formato ARJ · 130, 133
 formato ZIP · 39, 102, 115, 130, 133, 142, 144, 147
 Compilador · consulte Software, Compilador
 Compiler · consulte Software, Compilador
 Confiabilidade · consulte Software, Confiabilidade
 Consulta (a um banco de dados) · consulte Banco de dados, consulta (query)
Cookie · 133, 134
 CPD · 32, 118, 121

D

DAT · 102, 115, 118, 133
Data Mining · 124, 133
Data Warehouse · 47, 48, 95, 96, 97, 124, 133, 134, 146
 dBase · consulte Linguagem de Programação, dBase
 DBC · consulte Fox, Visual FoxPro, DBC
 DBF · consulte Banco de dados, padrão DBF
 Desktop · consulte Hardware, Desktop
DESK · 10
DESKView · 10
 DHTML · consulte Web, DHTML
Digital Research · 10
 Disciplina de Informática Médica · consulte Informática Médica, Disciplina
 Disco rígido removível · 134
 Display · consulte Monitor
 DNS · 134
Downstream · 15, 130, 134, 146
 DVI · 14

E

ENIAC · 6, 7, 8
 Estabilizador de tensão · 40, 83, 134
Ethernet · 9
 Extensions, SSI e SS · consulte SSI e SS Extensions
Extranet · 134. (consulte também Internet e Intranet)

F

Fax-Modem · consulte Modem
 Filtro de linha · 134, 146
 Firewall · 135
 Forest, Lee de · 6
 Fox · 91
 FoxPro for DOS · 91
 FoxPro for Windows · 91, 93
 Visual FoxPro · 30, 40, 41, 42, 43, 44, 49, 91, 93
 CDX · 44
 DBC · 44
 FPT · 44
 IDX · 44
 FPT · consulte Fox, Visual FoxPro, FPT
 FTP · 135, 145

G

GBM · 70, 71, 77, 135
 GClin · consulte Tot-Clin
 GEM · 10
 GIF · consulte Imagens digitais
 Gravação multisessão · 135
 Grupo Brasileiro de Melanoma · consulte GBM
 Grupo Brasileiro Multidisciplinar e Multicêntrico para estudo do Melanoma · consulte GBM
 Grupos de discussão · 135
 GUI · consulte Graphical User Interface

H

Hand Held · consulte Hardware, Hand Held
 Hardware · 15, 16, 17, 18, 31, 32, 39, 45, 46, 78, 82, 86, 89, 93, 113, 120, 121, 134, 135, 136, 137, 139, 141, 143, 145
 Desktop · 39, 130, 136, 140
 Hand held · 3, 136
 Jaz Drive · 115, 138
 Palm top · 3, 140
 ZIP Drive · 39, 115, 142, 144, 147
 HCI · consulte Interface, HCI
 HDTV · 136, 139
 High Edit · 40
 Hiperlink · consulte Web, Link
 Hipertexto · consulte Web, Link
 Hollerith, Hermann · 5, 6
 Hot swapping · 136
 HotSync · 136
 HTML · consulte Web, HTML
 HTTP · 134, 136, 143, 145
 Hub · 40
 Human Computer Interface · consulte Interface, HCI

I

IBM · 6, 7, 10, 28, 29, 92, 144
 PC · 28, 29, 92
 Ícone · 9, 10, 28, 50, 93, 102, 136
 IDE · 137

IDX · consulte Fox, Visual FoxPro, IDX
 IEEE 1394 Serial Bus · consulte Firewire
 Imagens digitais · 36, 44, 51, 56, 57, 60, 63, 69, 116, 137, 140
 BMP · 44, 57, 131
 GIF · 44, 57, 135
 JPG · 44, 51, 57, 119, 127, 138
 Máquina fotográfica · 51
 Distância focal · 134, 148
 Zoom digital · 147
 Zoom óptico · 148
 Impressora
 Deskjet · 39, 145
 HP · 39, 40, 147
 Laser · 39, 112, 145
 Informática Médica
 Disciplina · 31
 Residência · 31
 INRA · 13
 Instant on · 137
 Intel · 8, 9, 11, 12, 130
 Interface · 9, 10, 19, 20, 21, 28, 29, 30, 45, 91, 92, 93, 102, 103, 120, 130, 132, 136, 137, 139, 142, 143
 GUI · 136
 HCI · 19, 20, 24, 28, 45, 58, 79, 102, 136
 Internet · 12, 13, 14, 15, 38, 44, 54, 62, 94, 129, 130, 132, 134, 135, 137, 142, 143, 144, 145, 147
 Protocol · 134, 137, 145
 Internet Explorer · consulte Microsoft, Internet Explorer
 Interpretador · consulte Software, Interpretador
 Intranet · 44, 135. (consulte também Internet e Extranet)
 ISDN · 14, 137

J

Jacquard, Joseph Marie · 4
 Jato de Tinta, Impressora · consulte Impressora, Deskjet
 Javascript · consulte Linguagem de Programação, Javascript
 Jaz Drive · consulte Hardware, Jaz Drive
 Jobs, Steve · 10
 JPG · consulte Imagens digitais

K

Kbyte · consulte Byte, Quilobyte

L

LCS · consulte Monitor, LCD
 Leibnitz, Gottfried · 4
 Leibnitz, Gottfried Wilhelm von · 4
 Line interactive · 138
 Linguagem
 Semântica · 22, 138
 Sintaxe · 22, 30, 40, 43, 58, 138

Linguagem de programação · 130, 137, 138, 141, 146
Clipper · 30, 34, 40, 41, 44, 89, 90, 91, 122
dBase · 29, 30, 40, 41, 43, 44, 58
dBase II · 29
dBase III · 29
dBase III Plus · 29
Java · 130, 132, 134, 137, 138
Javascript · 131, 134, 137
Perl · 54, 132, 141
Visual Basic · 131, 132, 146
 Link · consulte Web, Link
 Linker · consulte Software, Linker
Lisa · 10
Long-Term Memory · 19
 Lovelace, Augusta · 5
 LSI · 9
 LTM · consulte Long-Term Memory

M

Macintosh · consulte Apple, Macintosh
Mainframe · 11, 15, 138
 Máquina fotográfica · consulte Imagens digitais, Máquina fotográfica
Mark I · 6
 Melanoma · consulte GBM
 Memória
 RAM · 9, 11, 29, 39, 48, 82, 89, 90, 92, 93, 139
 ROM · 139
Microsoft · 10, 11, 13, 29, 30, 40, 43, 54, 91
 Internet Explorer · 54, 132, 147
 Outlook · 54
 Powerpoint · 40
 Viewer · 40
 Word · 40, 130
 Viewer · 40
 MIT · 8, 13
Modem · 14, 39, 40, 48, 130, 131, 133, 134, 136, 137, 139, 141, 142, 146
 a cabo · 139
 PC Card · 139
 Monitor · 9, 11, 29, 39, 92, 104, 130, 135, 138, 139, 145
 Flicker free (sem tremulação) · 135
 Frequência de varredura horizontal · 135
 LCD · 138, 139, 145
 Placa de vídeo · 11, 48, 141
 Tela de gás-plasma · 139, 145
 Tela de matriz ativa · 139, 145
 Tela de matriz passiva · 139, 145
 Monitor de vídeo · consulte Monitor
 Moore, Lei de · 8, 12
Mosaic · 11, 13, 147
Motorola · 10
 MPEG · 139
 MPEG-2 · 139
 MPEG-3 · 139
 MS-DOS · consulte DOS
 MTBF · 139
 MTTR · 139
 multimídia · 13, 28, 29, 30, 35, 36, 92, 136, 140, 147
 Multiprocessamento · 140

Multithreading · 140

N

Nantucket · 30
NCSA · 13, 147
 Netscape
 Navigator · 11, 13, 147
No-break · 40, 83, 138, 140, 142, 143, 144
No-break microprocessado · 140
No-break off-line · 138, 140, 143, 144
No-break on-line · 140
Notebook · 3, 39, 40, 62, 111, 138, 140, 141, 145. consulte Hardware, Notebook

O

OCR · 140
 OCX · 40
 OLE · 44, 67
Oracle · 9, 41, 44
 Outlook · consulte Microsoft, Outlook

P

Paint Shop Pro · 40
 Palm top · consulte Hardware, Palm top
 PARC · 9, 10
 Pascal, Blaise · 3, 4
 Pascalina · 3, 4
PC · 10, 141, 142, 147. consulte IBM, PC
 PC Card · consulte Modem
 PCMCIA · 139, 141
 PCS · 141
 PDF · 141
Pentium · 11, 12, 39, 48, 89, 93
 200 MMX · 93
 266 MMX · 39
 Celeron · 39, 48
 II · 11, 12, 39, 48, 89
 III · 12
 Pro · 12
 Perl · consulte Linguagem de Programação, Perl
 PET · 9
 Pico de tensão · 141
Pixel · 131, 137, 139, 141, 142, 143, 145
 PKUnzip · consulte Compactação de dados, formato ZIP
 PKware · consulte Compactação de dados, formato ZIP
 PKZip · consulte Compactação de dados, formato ZIP
 Placa de vídeo · consulte Monitor, Placa de vídeo
Plug and play · 141
Polaroid · 40
 Ponto a ponto, Rede · consulte Rede, Ponto a ponto
 Porta infravermelha · 141
 Portas paralela · 39, 142
 Portas serial · 140, 142
 Powerpoint · consulte Microsoft, Powerpoint

Processamento paralelo · 17, 132, 142
 Profundidade da cor · 142
 Provedor de acesso · 14, 142

Q

Quaterdeck · 10

R

Rede
 Cliente/servidor · 43
 Ponto a ponto · 40, 84, 85, 92
 Residência de Informática Médica · consulte
 Informática Médica, Residência
 Resolução óptica · 143
 Revisão técnica formal · consulte
 Software, Revisão técnica formal
Rushmore · (consulte também Visual FoxPro)
Rushmore, tecnologia · 41, 43

S

Scanner · 40, 51. (consulte também pelo
 fabricante)
 HP · 40
 SCSI · 137, 143
 Semântica · consulte Linguagem, Semântica
 Servidor · 43, 78, 84, 85, 115, 119, 130, 131,
 132, 133, 134, 140, 143, 144, 145, 147
 de Dados · 43, 84
 de impressão · 143
 de Rede · 78, 84, 134, 140
 Web · 130, 131, 132, 133, 143, 144
 Servidor de Banco de Dados · consulte Servidor
 de Dados
 SGML · 143
 Shockley, William · 7
Short-break · 143
Short-Term Memory · 19
 S-HTTP · 143
 Sintaxe · consulte Linguagem, Sintaxe
 Sistema operacional · 10, 28, 29, 35, 39, 63, 78,
 89, 90, 91, 92, 102, 113, 118, 120, 122, 134,
 140, 143, 147
 DOS · 93, 103, 122
 Windows · 10, 11, 13, 28, 29, 30, 35, 36, 39,
 40, 43, 44, 48, 50, 51, 54, 56, 58, 67, 85,
 89, 91, 92, 93, 102, 103, 118, 122, 130,
 136, 146, 147
 1.0 · 10
 2000 · 11
 3.0 · 10
 3.1 · 11, 36, 89, 91, 93
 3.11 · 89, 91
 95 · 11, 40, 43, 56, 93
 98 · 11, 39, 48, 89, 146
 CE · 130, 147
 for Workgroups · 85, 91
 Xenix · 91
 SMTP · 143

Software · 10, 15, 16, 17, 18, 21, 22, 23, 24, 25,
 26, 27, 28, 29, 31, 32, 33, 35, 36, 37, 38, 39,
 40, 41, 45, 46, 47, 51, 54, 55, 56, 75, 76, 77,
 78, 79, 82, 86, 87, 88, 89, 90, 92, 93, 94, 97,
 98, 100, 101, 102, 103, 104, 106, 110, 115,
 119, 120, 121, 122, 123, 124, 125, 126, 127,
 128, 129, 132, 135, 136, 137, 138, 140, 141,
 143, 144, 145
 Código fonte · 22
 Compilador · 22, 40
 Confiabilidade · 23, 26, 27, 28, 38, 46, 49, 56,
 77, 79, 81, 82, 87, 88, 90, 95, 97, 111,
 112, 121, 122, 127, 132
 Interpretador · 22
 Linker · 22, 91
 Manutenção · 18, 23, 26, 27, 28, 31, 32, 35,
 38, 40, 45, 46, 47, 49, 61, 78, 89, 94, 102,
 109, 114, 116, 119, 120, 121, 125, 127,
 128
 Adaptação · 27, 81, 120
 Aprimoramento · 27, 30, 31, 33, 35, 47,
 75, 78, 81, 91, 94, 100, 101, 102, 120,
 121, 123, 125, 129
 Correção de erros · 27, 37, 46, 75, 120,
 133
 Revisão técnica formal · 25, 26, 120
 Software house · 16
 Software Arts · 10
 SRI · 8
 SSI e SS *Extensions* · 134, 144
 SSL · 144
 Standby by · 144
 STM · consulte Short-Term Memory
Super Disk · 115, 144
 Sybase · 41

T

TCP · 134, 137, 144, 145
 TCP/IP · 134, 137, 145
Telnet · 145
TopView · 10
Tot-Clin · 34, 35, 39, 55, 89, 91, 92, 93, 100,
 108, 109, 122
Tot-Win · 35, 37, 39, 46, 47, 49, 50, 54, 55, 56,
 59, 62, 63, 64, 71, 72, 73, 80, 89, 92, 93, 97,
 98, 99, 100, 101, 103, 104, 106, 108, 109,
 111, 112, 118, 126
 Trigger · consulte Fox, Banco de dados, Trigger
TRS-80 · 9
 TV
 a cabo · 14, 15, 94, 139
 TXT · consulte ASCII, TXT

U

UCI · 52, 97, 100, 112, 146
 Unidade Celular de Informação · consulte UCI
UNIVAC · 7
 UPS · 140, 143, 146
Upstream · 130, 134, 146
 USB · 146

V

Válvula · 3, 6, 7
VBscript · consulte Linguagem de Programação, VisualBasic
Vídeo Blaster · 40
Visicalc · 10
VisiCorp · 10
VisiOn · 10

W

Web · 11, 12, 13, 14, 15, 51, 52, 54, 71, 73, 74, 94, 117, 119, 128, 130, 132, 134, 136, 140, 143, 144, 146, 147
Browser · 11, 13, 54, 95, 130, 131, 132, 133, 134, 136, 143, 147
DHTML · 134
HTML · 131, 134, 136, 144, 147
Link · 136, 138, 147

WinZip · consulte Compactação de dados, formato ZIP
Word · consulte Microsoft, Word
World Wide Web · (procure apenas Web)
WWW · consulte Web
WYSIWYG · 9, 10

X

Xerox · 9
XML · 143, 147

Z

ZIP
formato
PKZip/PKUnzip · 147
WinZip · 147
Zip Drive · consulte Hardware, Zip Drive
Zuse, Conrad · 6

Anexo

<http://www.saudetotal.com/tesechao>