

Introdução a Programas Físico-Matemáticos Livres

Introduction to Physical–Mathematical Free Software

Margarete Oliveira Domingues

CPTEC/INPE, Cx. P. 515, 12201-970, São José dos Campos, SP

margaret@cptec.inpe.br

Odim Mendes Jr.

CEA/INPE, Cx. P. 515, 12201-970, São José dos Campos, SP

odim@dge.inpe.br, <http://www.dge.inpe.br/wotan>

Recebido em 20 de janeiro, 2003. Aceito em 4 de abril, 2003.

Atualmente encontram-se disponíveis excelentes programas de livre distribuição e/ou de distribuição gratuita para a realização das mais diversas atividades de pesquisa, de ensino ou mesmo de produtividade. Entre esses existem programas físico-matemáticos poderosos. Com o objetivo de motivar professores e estudantes, de vários níveis, e pesquisadores a trabalharem com tais ferramentas, neste trabalho são apresentados os programas GNU/OCTAVE, que lida fundamentalmente com cálculos numéricos; GNU/MAXIMA, que também lida com computação simbólica; OPENDX, que executa visualizações 3D; e RCS, que auxilia o gerenciamento de códigos fontes. Exceto pela última, essas ferramentas são capazes de gerar gráficos e permitem ainda, em decorrência da habilidade crescente do usuário interessado, alcançar níveis de programação avançados e utilizações gráficas sofisticadas.

Nowadays there are excellent free softwares available to many activities in science, in teaching and even to productivity. Among those, there are powerful physical mathematical softwares. In order to motivate teachers and students, in all levels, and researchers to deal with those tools, the GNU/OCTAVE software, that works in principle with numerical calculus, GNU/MAXIMA software, that also works with symbolic computation, OPENDX, for scientific visualization, and RCS, a system for revised version, are presented. Except for the last tool, the other tools can create graphics and allow, with the increasing ability of the interested user, to reach advanced programming levels and sophisticated graphical uses.

I Introdução

Existem disponíveis, atualmente, excelentes programas de livre distribuição e/ou de distribuição gratuita para a realização das mais diversas atividades de pesquisa, de ensino ou mesmo de produtividade. Esses programas são desenvolvidos por milhares de pessoas no mundo inteiro e tornam-se cada vez mais de maior interesse da comunidade científica e do público em geral. A utilização de tais ferramentas computacionais ainda é muito modesta em vários segmentos da comunidade científica brasileira. A divulgação de algumas dessas ferramentas é o objetivo deste trabalho, atento ao lema: investir em conhecimento, aumentar o desempenho e reduzir custos.

Como muitas das atividades científicas e, sobretudo, de ensino nos cursos de Física, Matemática e áreas afins são desenvolvidas utilizando programas proprietários, decorre daí a motivação e o foco deste texto: apresentar alternativas de trabalho. Pois, muitos desses programas possuem similares gratuitos que podem auxiliar a reduzir os custos do uso de tais ferramentas computacionais, ao mesmo tempo que dá acesso irrestrito a ferramentas robustas. Para o desen-

volvimento de suas atividades de pesquisa em Ciências Atmosféricas e Espaciais, os autores têm estruturado um *Laboratório Numérico de Análise e de Simulação* com uso de programas livres. Assim, com este propósito, quatro dos programas de interesse são aqui apresentados. Inicialmente discutem-se o GNU/OCTAVE, que permite lidar com cálculos numéricos; e o GNU/MAXIMA, que permite lidar também com computação simbólica. Um tutorial sobre essas ferramentas foi apresentado em [1]. As outras duas ferramentas são o OPENDX, para a visualização de alta qualidade de dados tridimensionais e também bidimensionais, e o RCS, para gerenciar os códigos de programas e textos \LaTeX .

A finalidade dessa análise é encorajar qualquer usuário, desde o professor ou o estudante do secundário até o pesquisador leigo em informática, uma iniciação no uso desses recursos. Para aqueles que desejem aprofundar-se nessas ferramentas, níveis avançados de programação e de utilizações gráficas sofisticadas podem ser alcançados. Assim, para cada ferramenta, o conteúdo exposto envolve a análise da sua apresentação básica e alguns exemplos.

O programa GNU/OCTAVE, que utiliza o programa

GNUPLOT, e o RCS já vem disponíveis nos CDROMs das principais distribuições GNU/LINUX. O GNU/MAXIMA e o OPENDX devem ser obtido em lugares específicos da Internet. Neste trabalho sobre programas físico-matemáticos, divulgam-se também materiais didáticos já disponíveis para aqueles interessados em se aprofundarem nessas ferramentas. Na Tabela 1 estão apresentados os endereços das ferramentas aqui tratadas e de alguns outros programas e referências de interesse nesse contexto. Em geral, versões

atualizadas são oferecidas com muita frequência, dessa forma é interessante consultar com certa regularidade esses endereços eletrônicos. Embora essas ferramentas apresentem um melhor desempenho em sistemas operacionais Unix (<http://www.unix.org/>), como o GNU/LINUX (<http://www.linux.org/>), em geral elas estão também disponíveis para outros sistemas operacionais. Um endereço interessante em aplicações científicas em GNU/LINUX é o <http://sal.kachinatech.com/>.

Tabela 1. Endereços na Web de algumas ferramentas e referências de interesse

Nome da Ferramenta	Endereço na Web
<u>Físico-matemáticas</u>	
GNU/MAXIMA	http://www.ma.utexas.edu/users/wfs/maxima.html
GNU/OCTAVE	http://www.octave.org
SCI-LAB	http://www.scilab.org
GNUPLOT	http://www.gnu.org
XGFE	http://www.bl.physik.uni-muenchen.de/rechner/grafik/xgfe
R	http://www.gnu.org/software/r/
OPENDX	http://www.opendx.org
<u>Editoração científica</u>	
\LaTeX	http://www.ctan.org http://biquinho.furg.br
PROSPER(\LaTeX)	http://prosper.sourceforge.net/
Kile(\LaTeX)	http://xml1.net.free.fr/kile
psutils	http://www.go.dlr.de:8081/pdinfo_dv/psutils.html
abntex(\LaTeX)	http://abntex.codigolivre.org.br/
OpenOffice	http://www.openoffice.org/
<u>Organização</u>	
RCS	http://www.gnu.org
CVS	http://www.cvshome.org
Sixpack(BibTeX)	http://www.openoffice.org/
<u>Criação de diagramas</u>	
Dia	http://www.lysator.liu.se/alla/dia/dia.html
XFig	http://www.xfig.org
<u>Tratamento de imagens</u>	
gimp	http://www.gimp.org/
ImageMagic	http://www.imagemagick.org/
<u>Linguagens de programação</u>	
C^{++}	http://gcc.gnu.org/ http://www.research.att.com/~bs/C++.html http://www.cuj.com/
Fortran 90	http://www.liv.ac.uk/HPC/F90page.html
Python	http://www.python.org
gtk	http://gtk-sharp.sourceforge.net/
gawk	http://packages.debian.org/stable/devel/dpkg-awk.html
sed	http://www.gnu.org/software/sed/sed.html
<u>Bibliotecas</u>	
BLAS(Fortran)	http://www.netlib.org/blas/
Atlas (C^{++})	http://www.worldforge.org/dev/eng/libraries/atlas_c++.cpp
Blitz(C^{++})	http://www.oonumerics.org/blitz/

II GNU/OCTAVE e GNUPLOT

O GNU/OCTAVE é uma linguagem de alto nível basicamente voltada para computação numérica. Esse programa provê uma interface por linha de comandos para solução numérica de problemas lineares e/ou não-lineares e para implementar outros experimentos numéricos usando uma

linguagem que é compatível com o **Matlab** (programa comercial). O programa pode ser utilizado também em modo *script* (textos de programação) e permite incorporar módulos escritos nas linguagens C^{++} , C, Fortran e outras. O GNU/OCTAVE foi escrito por John W. Eaton e muitos outros, estando disponível na forma GPL [2].

O GNU/OCTAVE tem ferramentas amplas para soluções

numéricas de problemas comuns de álgebra linear, para a determinação de raízes de equações não-lineares, manipulações polinomiais e integração de equações diferenciais ordinárias e equações diferenciais algébricas.

Quando executado pelo comando **octave** em um terminal, o GNU/OCTAVE resulta em algumas informações e inicia um ambiente de trabalho por linha de comando. Para

se obterem saídas gráficas na tela é necessário estar em um ambiente XWindow (por exemplo, no GNU/LINUX), o GNU/OCTAVE cria automaticamente uma janela separada para a apresentação gráfica (Figura 1). Por outro lado, mesmo em ambiente texto pode-se capturar as saídas em um arquivo que será posteriormente visto em outro aplicativo.

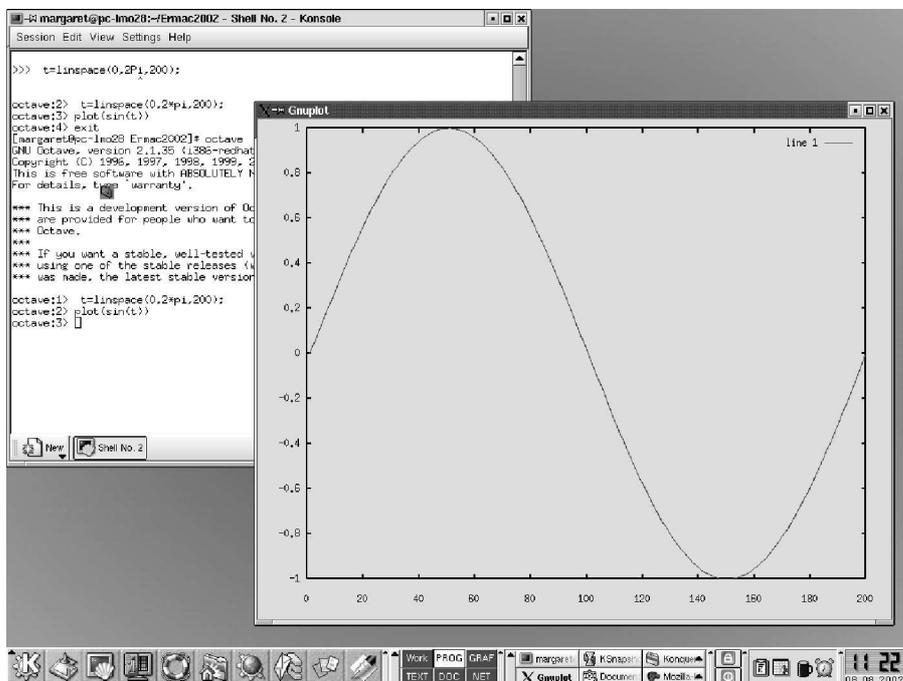


Figura 1. Exemplo da interface GNU/OCTAVE e de uma saída gráfica.

No GNU/OCTAVE, expressões são blocos básicos de construção de assertivas. Uma expressão calcula um valor, que pode ser impresso, testado, armazenado, passado a uma função ou designar um novo valor para uma variável por meio de um operador de atribuição. O GNU/OCTAVE conhece operações com números complexos, em que $i = \sqrt{-1}$. Outras funções básicas também são pré-definidas, como \cos , \sin , \tan , \log , $\sqrt{\quad}$ e \exp , e expressas da forma usual. Por exemplo:

```

octave:1> sqrt(-6.0);      octave:1> cos(pi)
ans = 0.00000 + 2.44949i   ans = -1
  
```

As operações matriciais utilizam uma notação de comandos muito similar à notação tradicional. Determinantes, transpostas e matrizes inversas também são diretamente obtidas, quando existentes. Por exemplo, seja A uma matriz $m \times m$, o determinante de A pode ser obtido pelo comando $\det(A)$ e sua matriz transposta pode ser obtida pelo comando A' . Além disso, várias matrizes muito utilizadas, como a matriz identidade e matrizes diagonais, podem ser construídas diretamente por funções pré-definidas. Também é possível fazer operações elemento a elemento entre matrizes e calcular a solução de sistemas lineares $Ax = b$.

O GNU/OCTAVE possui recursos no manuseio de funções. Por exemplo, pode-se definir uma função com o

comando **function** e executá-la posteriormente na linha de comando. Pode-se calcular a solução de um conjunto de equações não-lineares e fazer o cálculo da integral definida em um intervalo para uma variável.

A utilização do GNU/OCTAVE apresenta um potencial maior quando se faz uso de recursos de programação. Com este propósito, é importante conhecer as formas de controlar os fluxos de cálculos **if**, **for**, **while** e **switch**. O comando **break** pode ser utilizado para sair de qualquer um desses controles. O comando **continue** pode ser utilizado para os laços **for** e **while** quando se deseja retornar à condição desse laço. Em geral, é mais conveniente a criação de programas em arquivos externos. Os comandos do GNU/OCTAVE podem ser salvos em um arquivo texto de extensão **.m** ou **.oct**, como por exemplo **prog.m**, para posterior processamento. Para o leitor interessado, uma boa referência sobre formas de se controlar fluxos é o livro de Córtes [3].

Toda a parte gráfica do GNU/OCTAVE é feita pelo programa GNUPLOT [4]. É possível gerar gráficos bidimensionais e alguns tipos de gráficos tridimensionais. Na Figura 2 é apresentado um exemplo de gráfico 3D e de curvas de nível no GNU/OCTAVE.

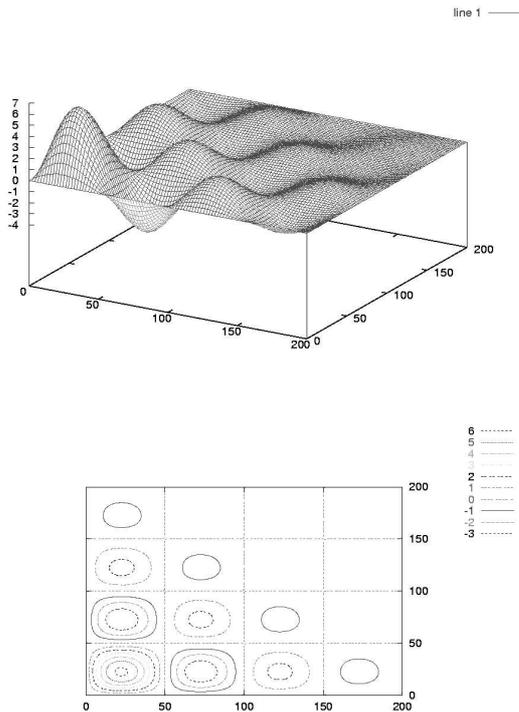


Figura 2. Gráficos 3D e de curvas de nível no GNU/OCTAVE.

O GNUPLOT pode ser utilizado independentemente do GNU/OCTAVE. Ele é bem simples de ser utilizado e pode produzir figuras simples com boa qualidade gráfica na tela e em formatos de imagens compatíveis com o \LaTeX , como o *postscript*¹ e o *png*². Ele pode ser utilizado diretamente na linha de comandos ou por *scripts*, acionando o comando **gnuplot** seguido do nome do *scripts* na linha de comandos do terminal. Seus comandos são bem intuitivos, como o **plot** para gerar gráficos. O programa **XGFE** é um programa tipo tela de usuário (*front end*) para o GNUPLOT, que auxilia a criar os gráficos. O help do GNUPLOT é muito bom o que facilita a criação de figuras mais elaboradas. Dentro do GNU/OCTAVE, o GNUPLOT fica sendo uma ferramenta sub-jacente. Muitos outros recursos gráficos podem ser obtidos no GNU/OCTAVE, incluindo animações.

III GNU/MAXIMA

O GNU/MAXIMA é um programa de computador, sob a licença GPL, para lidar com sistemas algébricos. Wil-

liam F. Shelter desenvolveu esse programa em LISP baseado na implementação original do *Macysma* no MIT. O GNU/MAXIMA pode ser utilizado para cálculos matemáticos, manipulação simbólica, computação numérica e criação de gráficos [5,6,7]

De uma forma geral, todas as funções numéricas disponíveis no GNU/OCTAVE existem no GNU/MAXIMA. Por outro lado, o GNU/MAXIMA tem seu ponto forte na parte simbólica, o que faz com que a parte numérica seja um pouco mais lenta que os algoritmos utilizados pelo GNU/OCTAVE. É possível combinar esses dois programas, utilizando os algoritmos do GNU/OCTAVE diretamente no GNU/MAXIMA.

Quando executado pelo comando **maxima** em um terminal, o GNU/MAXIMA resulta em informações iniciais e inicia um ambiente de trabalho por linha de comando. Nesta forma de execução, para se obter saídas gráficas é necessário estar em um ambiente XWindow. Pode-se trabalhar também por meio de uma interface gráfica conhecida como XMAXIMA, apresentada na Figura 3, que é acionada com o comando **xmaxima**. Há duas janelas nessa interface, na janela superior está localizada a linha de comando interativo. Na outra janela está localizado um tutorial de utilização do GNU/MAXIMA, basta clicar na palavra **Result** para ativar o GNU/MAXIMA e executar os comandos em questão.

No GNU/MAXIMA, podem-se calcular fatoriais, fatorar em números primos, expandir e fatorar polinômios, simplificar expressões algébricas, calcular a decomposição parcial fracionária para expressões fracionárias de uma forma simples. Por exemplo,

$$\begin{aligned} \text{expand}((x+7)*(x-7)); & \Rightarrow x^2 - 49 \\ \text{factor}(x^2-49); & \Rightarrow (x-7)(x+7) \\ \text{partfrac}(1/(x-5)*1/(x-7),x); & \Rightarrow \frac{1}{2(x-7)} - \frac{1}{2(x-5)} \end{aligned}$$

As operações usuais com matrizes também podem ser feitas simbolicamente. Por exemplo, seja

$$A = \begin{bmatrix} a & b & c \\ a-b & b-c & c^2 \\ 1 & 2 & 3 \end{bmatrix}.$$

Assim, para calcular o determinante da matriz A , utiliza-se o comando **determinant(A)**, que resulta na expressão algébrica $-b(3(a-b) - c^2) + a(3(b-c) - 2c^2) + c(c-b + 2(a-b))$.

Além disso, é possível definir funções, resolver equações ou sistemas para uma variável designada, calcular a derivada com respeito a uma variável, diferenciar implicitamente, integrar simbolicamente ou numericamente e resolver equações não-lineares. Por exemplo,

¹(<<http://www.adobe.com>>)

²(<<http://www.libpng.org/pub/png/>>)

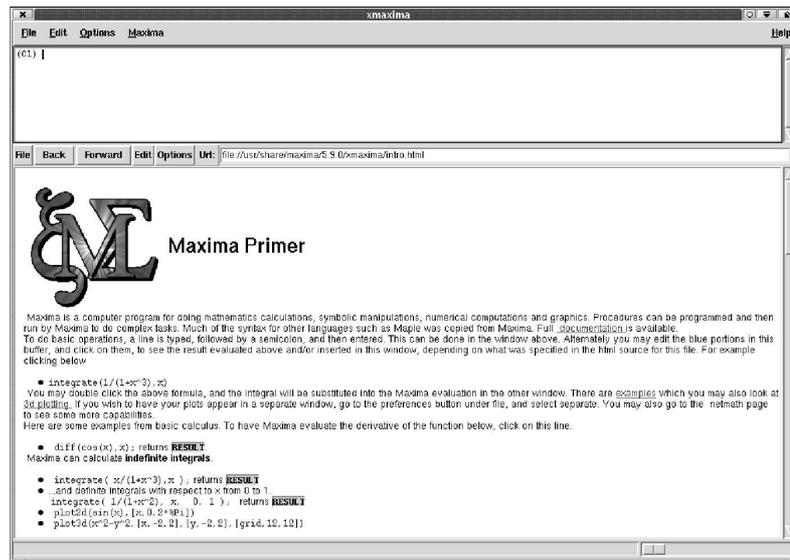


Figura 3. Exemplo da interface gráfica XMAXIMA.

```

depends (y,x);           ⇒ [y(x)]
diff (x^2 * y = 9, x);  ⇒ x^2 dy/dx + 2xy = 0

eq1: x^3 + 5 * x * y + y^2 = 0
eq2: 3 * x + 2 * y = 1
solve([eq1,eq2]);       ⇒ [[y = - 0.02483745477298, x = 0.34989161849711],
                           [y = 0.71257391845627, x = -0.14171594563751],
                           [y = - 7.062736205593349, x = 5.041824295922656]]

linsolve([3*x+4*y-z=7,2*x+a*y+b*z=13,x+y+z=10],[x,y,z]);

```

$$x = \frac{33b + 17a - 65}{b + 4a - 10} \quad y = \frac{23b - 18}{b + 4a - 10} \quad z = \frac{23a - 53}{b + 4a - 10}$$

Os comandos do GNU/MAXIMA podem ser salvos em um arquivo texto extensão **.mac**, como por exemplo **prog.mac**, para posterior processamento. Para executar esse arquivo dentro do ambiente, utiliza-se o comando **batch**.

Fazer um gráfico bidimensional ou tridimensional no GNU/MAXIMA também é simples, como mostrado no seguinte exemplo:

```

plot2d([cos(2*x), x^3], [x, -%pi, %pi]);
plot3d(sin(x^2+y^2)/(x^2+y^2), [x, -5, 5], [y, -5, 5], [grid, 45, 45]);

```

Esses comandos geram um gráfico apresentado na própria janela do GNU/MAXIMA, que possui na lateral esquerda um menu que possibilita fazer várias tarefas. Entre essas tarefas está salvar o arquivo no formato postscript. Um exemplo dessas saídas é apresentado na Figura 4. Muitos outros recursos gráficos podem ser obtidos no GNU/MAXIMA, incluindo animações.

IV OPENDX

Para trabalhos que necessitem de uma melhor qualidade gráfica dos resultados, uma opção é o programa gratuito OpenDX (<<http://www.opendx.org>>), que é semelhante ao Data Explorer da IBM. Ele consiste de um pacote de ferramentas para manipular, processar, transformar, visualizar e animar dados em diversas plataformas computacionais e pode ser utilizado também para visualização de dados no paradigma da computação de processamento paralelo.

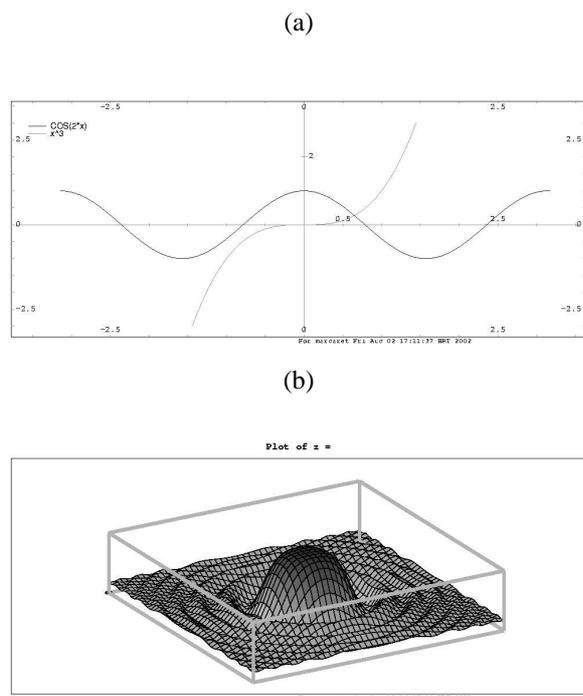


Figura 4. Saída *postscript* de gráficos no GNU/MAXIMA. (a) 2D e (b) 3D.

OPENDX é uma ferramenta de visualização muito poderosa pela forma de lidar com dados e permitir suas visualizações. Tem como finalidades principais a visualização de informações formatadas em 2D e 3D, de forma interativa ou automática (por meio de *scripts* ou pela inclusão no código fonte de funções em linguagem C ou Fortran), e a criação de filmes de visualização temporal de uma base de dados em formato *mpeg* (<<http://www.mpeg.org>>). Uma das grandes vantagens desse recurso é sem dúvida a facilidade de programação em ambiente *visual*, conhecido pela sua sigla em inglês VPE, “Visual Program Environment”. Na Figura 5 é ilustrado o VPE e esquematizado como é o processo de programação nesse ambiente.

Essa ferramenta tem uma enorme gama de aplicações, desde as áreas de ciências em geral, por exemplo a Meteorologia e a Geofísica Espacial (<<http://www.inpe.br>>), até a indústria. Na Figura 6 são apresentados alguns exemplos de visualizações de dados no OPENDX.

Maiores informações sobre esse programa podem ser obtidas na página do OPENDX e em outras páginas da Internet³. Tutoriais sobre o OPENDX são apresentados em [8,9,10].

V Gerenciamento de código-fonte

O desenvolvimento das atividades de pesquisa ou didáticas exige muitas vezes a criação de programas e textos e a atualização constante destes. O Sistema de Controle de Revisão (RCS) é uma ferramenta muito útil no auxílio ao

desenvolvimento de programas ou textos ASCII, como os do \TeX e \LaTeX . Com essa ferramenta é possível manter um histórico da sequência de etapas de construção de um código, de forma organizada e sem repetições desnecessárias. Desta forma, a qualidade do trabalho em desenvolvimento é assegurada e os contratempos decorrentes de dificuldades de administração do desenvolvimento minimizados [11,12,13].

O RCS é baseado em uma estrutura de árvore. Dessa forma, a primeira revisão é a raiz da árvore, que receberá a designação de revisão 1.1. As revisões posteriores descendem linearmente na árvore e passam a ter as designações 1.2, 1.3, ..., conhecidas como ramos. É possível também iniciar uma nova sequência de revisões para cada um desses ramos estabelecidos, conhecida como galhos (Figura 7). Utiliza-se então a ferramenta **rcsmerge** para auxiliar na consolidação de versões a partir de diferentes ramos de desenvolvimentos.

A primeira etapa a ser feita é criar um sub-diretório com o nome **RCS** no diretório onde está o código-fonte. Introduz-se no código-fonte desejado uma sequência de identificação da revisão, sempre entre comentários em relação ao compilador em que este código estará sendo processado. Os principais caracteres de identificação são: **\$Header\$**, **\$Author\$**, **\$Date\$**, **\$Locker\$**, **\$Revision\$**, **\$Source\$** e **\$State\$**. É possível ainda colocar um identificador no programa executável resultante. Para isso, deve-se inserir no código-fonte

```
char rcsid[]=@(#);
```

Os dois comandos fundamentais para a utilização do RCS são **ci** e **co**. O comando **ci** introduz uma nova versão no sub-diretório RCS e o comando **co** retorna a versão mais recente guardada no diretório RCS. Nesse sub-diretório há um arquivo com o nome do programa-fonte original acrescido de uma nova extensão ,**v** (vírgula letra v minúscula). Por exemplo os comandos:

```
$ ci -l nome-do-arquivo
```

geram uma revisão mais atualizada do arquivo. Com esse comando, um indicador (*prompt*) surge e espera uma linha de comentário sobre a revisão que está sendo guardada. A fim de utilizar um arquivo que foi colocado no RCS, pode-se extrair, para uso, o arquivo-fonte com o comando:

```
$ co -l nome-do-arquivo
```

É possível também recuperar uma certa versão de revisão do arquivo-fonte estabelecida pela árvore de revisão. Os indicadores, após a chamada do primeiro **co**, geram uma série de informações sobre a revisão no código-fonte. Há muitos outros comandos e opções que facilitam essa tarefa de gerenciamento. Assim, aprender o RCS é muito promissor, evitando perda de tempo, perda de espaço em disco e preservando o conteúdo de versões anteriores.

³<<http://www.csse.monash.edu.au/~donny/mc2/visualization/miscinfo/dxlinks.html>>

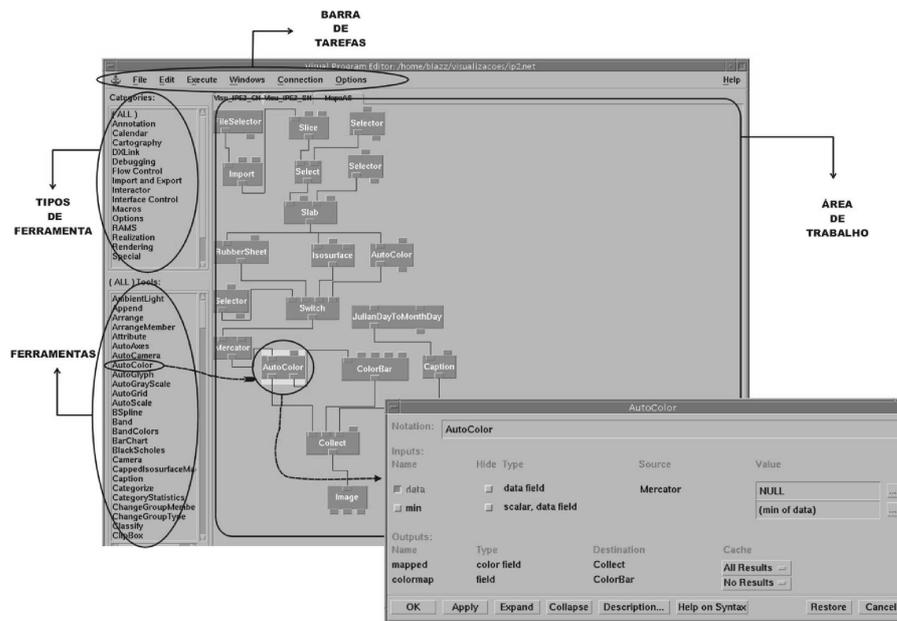


Figura 5. Ambiente de programação visual do OPENDX.

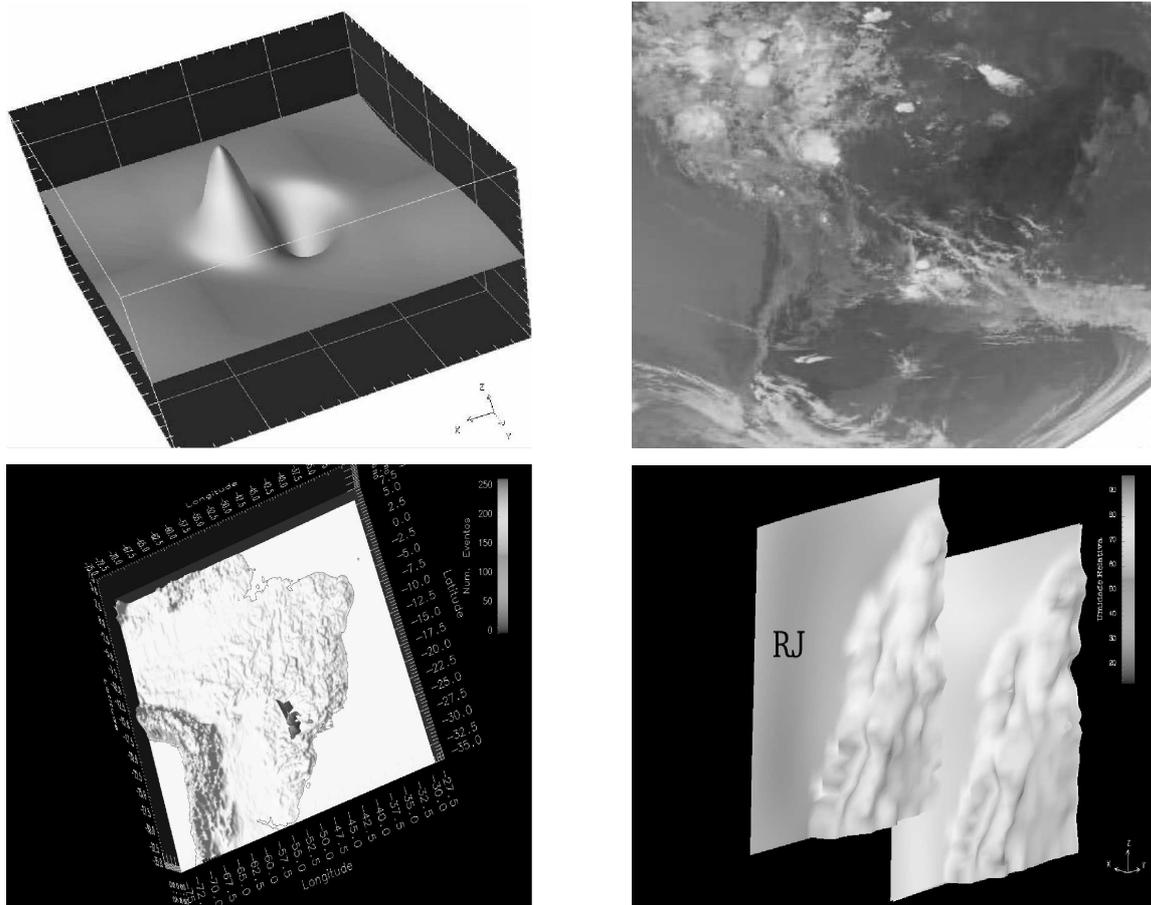


Figura 6. Alguns exemplos de visualizações de dados no OPENDX.

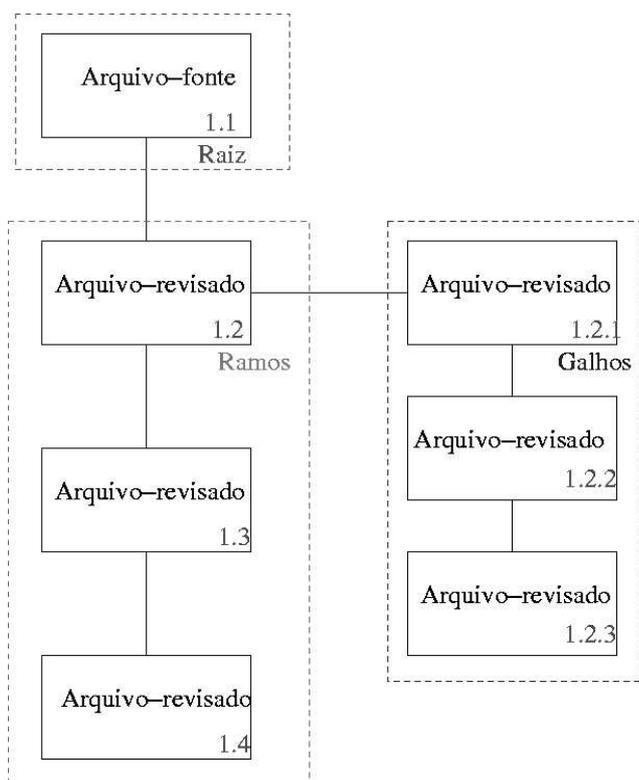


Figura 7. Árvore de revisão do RCS.

VI Comentários Finais

As novidades tecnológicas têm sido marcantes no estilo de vida da sociedade atual. Um dos aspectos principais dessa evolução acelerada de recursos materiais a ser analisado, apontado por algumas pessoas proeminentes, é a qualidade de vida. Dizem os especialistas que ela deveria ser a questão prioritária nos próximos anos.

Essa preocupação motiva-nos a uma análise de duplo caráter das ferramentas computacionais hoje disponíveis: do ponto de vista pessoal e do ponto de vista institucional.

No sentido pessoal, é uma questão de escolha de recursos de informática, que dá maior ou menor liberdade, maior ou menor poder a criatividade e que motiva a uma postura mais exclusivista ou mais cooperativa.

No sentido institucional, há a possibilidade, que já pode ser pressentida, de uma dependência forte de uma empresa com respeito a outras e uma fragilização de sua capacidade de atuação, e isso, indiscutivelmente, afeta o seu poder de decisão. Por isso, uma instituição necessita ter mais consciência das ferramentas de trabalho que precisa. Isso exige antecipar (antever) as suas necessidades institucionais no tempo.

Dentro do quesito “qualidade de vida” para uma instituição, o momento presente requer análise da compra, instalação e uso de produtos e serviços básicos de informática. Essas aquisições devem atender a:

- a) aumento ou estabilidade de produtividade na instituição;
- b) aumento ou garantia de segurança intrínseca dos serviços da instituição;

- c) garantia de comunicação integrada e harmoniosa entre serviços;
- d) inserção eficiente da instituição na sociedade já reaparelhada tecnologicamente;
- e) eliminação de gastos desnecessários para permitir economia pública ou, o que é preferível, reinvestimentos com benefício público.

Assim, algumas perguntas devem ser feitas inteligentemente e respondidas habilmente por todos os membros ativos da comunidade. Hoje a época é outra, iniciativas individuais têm consequências globalizadas. Para as decisões é necessário haver negociação, transparência, responsabilidade, retreinamentos e avaliação de serventia de produtos e serviços. Quanto a questões gerais, pergunta-se: Que posição deve-se ter perante essa inevitável revolução tecnológica e de mudança de paradigmas? Qual a posição a se ter perante programas proprietários comprados e programas de livre distribuição com licença pública geral? Que critérios de custo/benefício devem ser atendidos na aquisição de produtos com financiamento público?

Do ponto de vista prático, escolhas adequadas de ferramentas de informática representam diferenças de milhões de reais ao longo do tempo. Ressalta-se ainda que a escolha adequada de plataformas de trabalho traduz-se em perspectivas de real produtividade, segurança, eficiência e de colaboração para a formação de competência local.

Como uma consideração final, ressalta-se que o uso dessas alternativas de livre distribuição possibilita ao futuro profissional (aluno formado ou mesmo em formação) levar com ele esses programas em que está sendo treinado. Isso garante continuidade mais facilitada de projetos e a ampliação das soluções dentro do universo das ferramentas adotadas. Essas são oportunidades que abrem perspectivas no trabalho científico ou de ensino-aprendizagem muito significativas.

Agradecimentos

Os autores agradecem a SBMAC e aos organizadores do ERMAC de Natal de 2002 a motivação e a FAPESP o suporte financeiro dado às pesquisas do Projeto Via-Lux (FAPESP 1998/3860-9). Agradecem também ao revisor deste artigo as sugestões muito oportunas.

Referências

- [1] M. O. Domingues and O. Jr. Mendes. Introdução a programas científicos de distribuição gratuita: GNU/Octave, GNU/Maxima, \LaTeX e GNU/RCS. Technical Report INPE-9381-PRE/5041, INPE, 2002. <<http://www.dge.inpe.br/wotan>>.
- [2] J. Eaton. Octave manual. <<http://www.octave.org/doc/>>, 2001.
- [3] P. L. Côrtes. *C auto explicativo*. Érica, São Paulo, 1992. pp. 17-54.
- [4] T. Williams, C. Kelley, R. Lang, D. Kotz, J. Campbell, G. Elber, A. Woo, and et al. Gnuplot FAQ. <<http://www.ucc.ie/gnuplot/>>, 2001.
- [5] W. Schelter. Maxima manual. <<http://www.ma.utexas.edu/users/wfs/maxima.html>>, 2001.

- [6] W. Schelter. Basic Maxima commands. <http://math.cochise.cc.az.us/Doc/maxima/basic_maxima.html>, 2001.
- [7] ICM Institute for Computational Mathematics. Interactive demos of mathematical computations. <<http://icm.mcs.kent.edu/research/demo.html>>, 2002.
- [8] D. Thompsom, J. Braun, and R. Ford. *Paths to Visualization*. Visualization and Imagery Solutions, Inc, Estados Unidos, 2001. <<http://www.vizsolutions.com>>.
- [9] E.C. Silva. Introdução ao IBM Data Explorer. Technical report, CENAPAD-SP, Campinas, SP, Ago 1997. <<http://www.cenapad.unicamp.br>>.
- [10] R. Blaz, M. O. Domingues, and O. Jr. Mendes. Introdução à entrada de dados no OpenDX: formatos dx, general e grib. Technical Report INPE-9558-NTC/351, INPE, 2003.
- [11] GNU. RCS manual. <<http://www.gnu.org>>, 2002.
- [12] R. Petersen. *Linux - Programmer's Reference*. Mc-Graw-Hill, Berkeley, 1998.
- [13] M. Loukides and A. Oram. *Ferramentas GNU*. O'Reilly, 1998.