



MINISTÉRIO DA CIÊNCIA E TECNOLOGIA  
**INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS**



## REPRESENTAÇÃO DE CAMPOS DE DESCARGA ELÉTRICA ATMOSFÉRICA NUVEM-SOLO NA FERRAMENTA OPENDX

### RELATÓRIO ANUAL DE PROJETO DE INICIAÇÃO CIENTÍFICA (PIBIC/CNPq/INPE)

Marilyn Menecucci Ibañez (UNIFEI, Bolsista, PIBIC/CNPq)  
marilyn\_mba@yahoo.com.br

Dr. Odim Mendes Júnior (DGE/INPE, Orientador)  
odim@dge.inpe.br

Dra. Margarete Oliveira Domingues (LAC/INPE, Orientadora)  
mo.domingues@lac.inpe.br

Dr. Stephan Stephany (LAC/INPE, Orientador)  
stephan@lac.inpe.br

Julho 2006



## **AGRADECIMENTOS**

Agradeço ao Instituto de Pesquisas Espaciais - INPE a oportunidade de estudos e utilização de suas instalações.

Ao Programa Institucional de Bolsa de Iniciação Científica do Conselho Nacional de Desenvolvimento Científico e Tecnológico - PIBIC/CNPq a bolsa de iniciação científica.

Aos meus orientadores a orientação e apoio na realização deste trabalho.

Aos meus pais o carinho, amor e compreensão.

A todas as pessoas que participaram de alguma forma do desenvolvimento deste projeto.

E aos desenvolvedores de programas livres.



## **RESUMO**

As descargas elétricas são fenômenos naturais que ocorrem frequentemente e podem causar danos físicos a pessoas, além de danos materiais. Embora estejam presentes no cotidiano do habitat humano desde os primórdios, ainda não se tem um bom entendimento do seu comportamento e da sua previsão de ocorrência na atmosfera. Assim, desenvolver mecanismos que representem adequadamente esses fenômenos é de grande importância para que se possa melhor prever os seus efeitos. E é neste contexto que a visualização científica aplicada por meio da utilização da ferramenta gráfica OPENDX auxilia na compreensão deste fenômeno.



## **ABSTRACT**

Atmospheric electrical discharges are natural phenomena that frequently occur and may cause injuries to persons and material damages. Even though discharges are present at human life since its beginning, there is not a good understanding of their behavior nor of their forecast. Therefore, to model and visualize such phenomena is a way to improve the understanding of its effects. In this scope, scientific visualization can be applied by means of the graphic tool OPENDX to support the studies concerning discharges.

## SUMÁRIO

	<u>Pág.</u>
<b>LISTA DE FIGURAS</b> . . . . .	<b>9</b>
<b>CAPÍTULO 1 – INTRODUÇÃO</b> . . . . .	<b>11</b>
1.1 – <b>Histórico no PIBIC</b> . . . . .	12
<b>CAPÍTULO 2 – DADOS E METODOLOGIA</b> . . . . .	<b>13</b>
2.1 – <b>Dados de Descargas Elétricas</b> . . . . .	13
2.2 – <b>Dados Geopolíticos</b> . . . . .	13
2.3 – <b>Pacote Thor/SLA</b> . . . . .	13
2.4 – <b>Depurador de Programas</b> . . . . .	14
2.5 – <b>Controlador de Versões-RCS</b> . . . . .	14
2.6 – <b>Controlador de Versões-CVS</b> . . . . .	14
2.7 – <b>Construção de um Makefile</b> . . . . .	14
2.8 – <b>Visualizações no OPENDX</b> . . . . .	15
<b>CAPÍTULO 3 – RESULTADOS E ANÁLISES</b> . . . . .	<b>19</b>
3.1 – <b>Visualizações para a Dinâmica Interplanetária</b> . . . . .	19
3.2 – <b>Implementação do pacote Thor/SLA</b> . . . . .	20
<b>CAPÍTULO 4 – CONSIDERAÇÕES FINAIS</b> . . . . .	<b>23</b>
<b>REFERÊNCIAS BIBLIOGRÁFICAS</b> . . . . .	<b>25</b>
<b>APÊNDICE A – Thor/SLA: Programa em C++</b> . . . . .	<b>27</b>
<b>APÊNDICE B – Campo Magnético da Terra: Programação m C e em Script</b> . . . . .	<b>29</b>



## LISTA DE FIGURAS

	<u>Pág.</u>
2.1 Exemplo da janela do <i>Visual Program Editor</i> . . . . .	15
2.2 Exemplo de parte da estrutura do programa de visualização do contorno do Brasil. . . . .	16
2.3 Tela de configuração da ferramenta <i>Import</i> . . . . .	17
2.4 Contorno do Brasil e de todo seus estados. . . . .	17
3.1 Programa de visualização: estrutura do movimento da partícula. . . . .	19
3.2 Visualização resultante: Imagem da Terra com as linhas de campo magnético. . . . .	20
3.3 Visualização resultante: Imagem da movimentação de um partícula ao logo das linha de campo magnético. . . . .	20
3.4 Estrutura do programa de visualização. . . . .	21
3.5 Visualização resultante: Imagem 2D dos campos de descargas elétricas. . . . .	21
3.6 Visualização resultante: Imagem 3D dos campos de descargas elétricas. . . . .	22



# CAPÍTULO 1

## INTRODUÇÃO

Em suas manifestações, as tempestades elétricas produzem tanto efeitos deletérios quanto também benéficos no ambiente em que o ser humano habita. Portanto, atendendo às preocupações da sociedade como um todo, é de grande interesse dos órgãos públicos e privados um melhor entendimento das atividades elétricas e de fenômenos associados. Nesse sentido, o Brasil tem investido em coletar dados de descargas elétricas por meio de sensores sofisticados (por exemplo, LPATS e IMPACT), que constituem a *Rede Integrada Nacional de Descargas Atmosféricas* (RINDAT) [1], como também por meio de sensores experimentais, que se prestam as mais diferentes investigações especializadas. Neste trabalho utilizam-se dados de sensores magnéticos direcionais de baixo custo, pelo seu potencial de uso experimental. Tem-se ainda, para que esses sistemas tenham uma utilização prática, a necessidade de integrar esses dados com outros dados meteorológicos e afins.

Assim, possibilitar visualizações bem elaboradas dos dados de descargas atmosféricas revela-se de grande importância para auxiliar no melhor entendimento da evolução espaço-tempo associada a núcleos de convecções atmosféricas que produzem essas descargas. O objetivo deste trabalho é criar visualizações desses campos de descargas atmosféricas. Tais visualizações representam a ocorrência de descargas elétricas acumuladas durante um período de tempo em regiões especificadas, em que se utilizaram dados de descargas elétricas do projeto Via-lux/INPE.

Para alcançar tal objetivo escolheu-se o OPENDX, que é uma ferramenta de visualização científica, de código aberto, gratuita e multiplataforma, que possibilita criar visualizações com alta resolução gráfica. Por essas características, essa ferramenta é muito promissora para uso na continuidade das pesquisas nessa área

Este relatório está organizado de forma a dar um melhor entendimento sobre o projeto em andamento. No Capítulo 2 abordam-se como os dados são coletados e como eles são utilizados no projeto. No Capítulo 3 mostram-se alguns trabalhos desenvolvidos pela bolsista durante o período da bolsa. Por fim, no Capítulo 4 faz-se um resumo do que se obteve e o que ainda se pretende alcançar no projeto, dentro de um trabalho estratégico de desenvolvimento de recursos próprios de visualização científica. Esse trabalho é uma preparação para uma etapa mais avançada de computação científica.

## 1.1 Histórico no PIBIC

A bolsista iniciou sua participação no projeto em 2003 quando começou a estudar a ferramenta OPENDX . O primeiro contato com a ferramenta foi por meio do VPE (*Visual Program Editor*), o ambiente gráfico do OPENDX . Em paralelo a esse estudo, a bolsista também iniciou a aprendizagem da linguagem de programação C. Esses estudos iniciais tiveram o intuito de desenvolver capacidades de computação científica, que auxiliam os trabalhos de Ciências Atmosféricas e Espaciais. Neste ano de 2003, obtive como resultados preliminares a criação de visualizações de dados de ocorrência de descargas elétricas e do contorno político do Brasil, bem como de seus estados.

No ano de 2004, aprimorou as visualizações existentes dos dados de descargas elétricas, desenvolveu visualizações da topografia dos estados do Brasil e iniciou o estudo do pacote Thor/SLA [8]. Também em 2004, deu início ao estudo da linguagem de programação C++ e de ferramentas que auxiliam na otimização da programação como o depurador de programas GDB [7], os controladores de versões RCS [7] e CVS [3] e o Makefile [7] para poder implementar o pacote Thor/SLA.

No ano de 2005 implementou aos recursos de saída do pacote Thor/SLA as visualizações feitas no OPENDX com dados de descargas elétricas provenientes do sensor *Storm*. Para realização desta implementação iniciou a utilização da programação em *script* do OPENDX .

E no ano de 2006 a aluna concluiu a implementação das funcionalidades do Thor/SLA, desenvolveu visualizações aplicadas a eletrodinâmica da Terra , aprofundou-se nos estudos da computação gráfica e aprimorou as visualizações desenvolvidas com dados de descargas elétricas.

## CAPÍTULO 2

### DADOS E METODOLOGIA

#### 2.1 Dados de Descargas Elétricas

Os dados de descargas elétricas são coletados pelo sensor *StormTracker da Boltek*, de projeto INPE/FAPESP, da Divisão de Geofísica Espacial/CEA. Esse utiliza uma antena magnético–direcional para detectar a direção das descargas elétricas por meio de ondas eletromagnéticas emitidas. A intensidade do sinal é utilizada para estimar a distância que a descarga elétrica ocorreu a partir de um local de origem, neste caso, do sensor localizado no INPE de São José dos Campos. A combinação da direção e distância permitem que a posição da descarga elétrica seja identificada preliminarmente. Na verdade, um conjunto de antenas estarão sendo utilizadas para uma localização melhorada. Esses dados de direção, distância e posição juntamente com um dado de tempo de aquisição são fornecidos no formato .str (definido pelos pesquisadores) pelo software de aquisição de dados. O software de préprocessamento de dados transforma estes dados para arquivos em formato ASCII na forma matricial, utilizado neste projeto. Essas informações são atualmente visualizadas por meio da ferramenta GNUPlot e estão disponíveis na página da internet <<http://www.dge.inpe.br/wotan>>.

#### 2.2 Dados Geopolíticos

Os dados geopolíticos também são arquivos textos que contém informações, graduadas de 10 em 10 graus, sobre o contorno político (longitude e latitude) de uma dada região. No projeto foram utilizados dados de todos estados do país e do Brasil.

#### 2.3 Pacote Thor/SLA

O pacote Thor/SLA [8] é um programa desenvolvido pelos orientadores da bolsista na linguagem C++ para analisar várias características das descargas elétricas atmosféricas. Por exemplo, são calculadas as estatísticas descritivas e paramétricas de ocorrência das descargas unidimensionais e bidimensionais. Nesse pacote alguns tipos de visualizações unidimensional utilizando pacotes gráficos do L<sup>A</sup>T<sub>E</sub>X foram implementadas para facilitar a interpretação dos dados analisados. Pretende-se também implantar nesse pacote um módulo que permita realizar visualizações bidimensionais de dados fornecidos pelo Thor/SLA utilizando o OPENDX.

Para que a bolsista pudesse aproveitar melhor as ferramentas utilizadas no projeto, ela teve a necessidade de complementar seus conhecimentos estudando a linguagem de programação C++. Dentre os conceitos de C++ estudados estão: classe, hierarquia, polimorfismo, encapsulamento, funções, programação estruturada e também o pacote Thor/SLA. Para se ter um melhor

aproveitamento das ferramentas de programação estudadas, a bolsista estudou ferramentas de otimização de programação, como: GDB (GNU Project Debugger) [7], criação de um Makefile [7], gerenciamento de código fonte com RCS (Revision Control System) [7] e o controlador de versões - CVS (Concurrent Versions System)[3]. Estas ferramentas são explicadas a seguir.

## 2.4 Depurador de Programas

O GDB (GNU Project Debugger) [4], mostra o que ocorre dentro de um programa enquanto ele é executado ou o que o programa estava fazendo no momento em que ocorreu o erro. Ele pode iniciar o programa, especificando o que erro afeta aquele ambiente, parar o programa em uma condição específica, examinar o motivo da parada do programa e fazer correções de erros. O GDB aceita vários tipos de linguagens como: C, C++, Pascal, entre outras e pode ser utilizado tanto em ambientes UNIX como em Microsoft Windows.

## 2.5 Controlador de Versões-RCS

O RCS (Revision Control System) [7] gerencia revisões múltiplas de arquivos. Ele automatiza o armazenamento, a restauração, identificações e a união dos vários arquivos revisados. O RCS é muito utilizado para textos que são revisados constantemente, incluindo códigos fontes, programas, documentações, gráficos, entre outros.

## 2.6 Controlador de Versões-CVS

O CVS (Concurrent Versions System) [3] é um sistema de controle de versões que permite salvar as etapas da modificação de arquivos fontes e documentos. Ele desempenha um papel similar ao RCS, no entanto, possui algumas vantagens como: permitir que vários programadores ou um grupo de programadores mantenham a sua própria versão, que trabalhem no mesmo programa em máquinas diferentes por meio de uma rede de computadores, entre outras.

## 2.7 Construção de um Makefile

O Make [7] é uma ferramenta que controla a geração de arquivos executáveis e arquivos não-fontes de um programa para arquivos de códigos fontes de programas. Ele permite construir o programa por meio de um arquivo chamado *makefile*, que mostra como o arquivo fonte de um programa deve ser processado.

O pacote Thor/SLA atualmente utiliza dados de sensores do RINDAT no formato (*UALF*), definido pela empresa produtora dos sensores. Na implementação do pacote, pretende-se utilizar dados do sensor *StormTracker da Boltek* do INPE/DGE/CEA/FAPESP. O formato dos dados do sensor *StormTracker da Boltek* diferem do formato *UALF*.

## 2.8 Visualizações no OPENDX

Para desenvolver as imagens com dados de descargas elétricas, utilizou-se o OpenDX, que é um visualizador gratuito que teve seu início a partir da liberação do código fonte do DATA EXPLORER pela IBM. Esse programa é uma ferramenta de visualização de dados poderosa, pois lida com uma representação multidimensional, permitindo o manuseio e animação dos dados. Ele é constituído de módulos de programação. Algumas das suas vantagens imediatas, além do seu poder de trabalho, são ser multiplataforma e gratuito. O programa possui vários ambientes de programação, dentre eles *VPE - Visual Program Editor* e o *Script*. Os materiais utilizados para o estudo do programa foram [2], [9], [6] e [5]. O projeto dá maior ênfase ao VPE, que é o ambiente gráfico do OPENDX. O Editor de Programa Visual é o ambiente no qual se constrói o programa de visualização. A maioria das funções pertencentes ao VPE também são encontradas em outros programas gráficos. No entanto, ele apresenta também funções características do OPENDX. Na Figura 2.1 pode-se ver a estrutura do VPE.

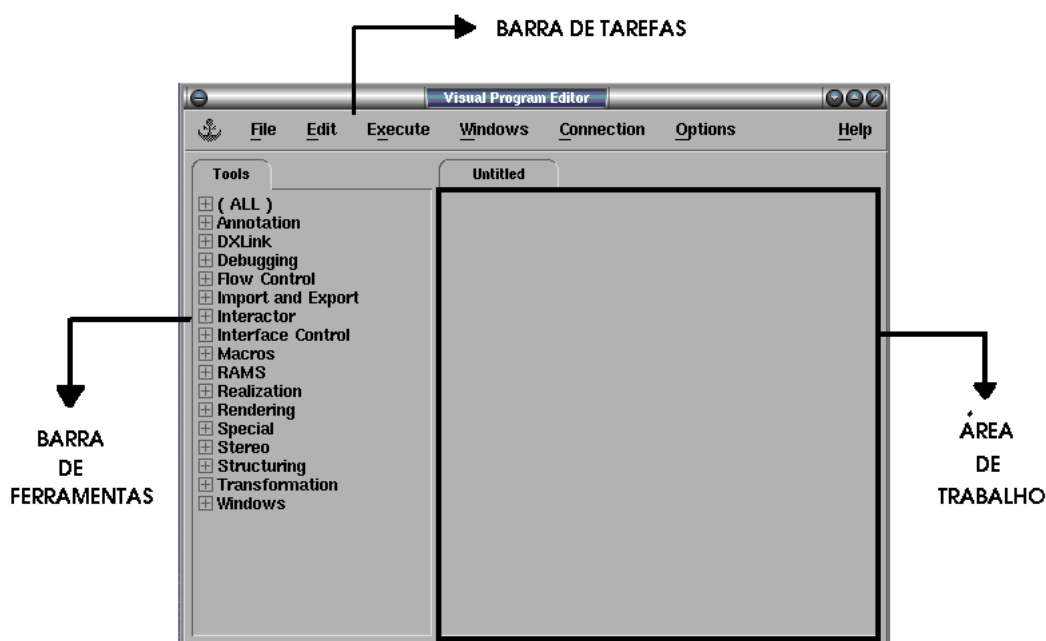


FIGURA 2.1 – Exemplo da janela do *Visual Program Editor*.

O programa de visualização pode ser construído utilizando-se as ferramentas ou macros do VPE. Elas são acessadas por meio da seleção da ferramenta desejada na barra de ferramentas e posteriormente um clique do mouse na área de trabalho. Essas macros são apresentadas na forma de "caixas", que precisam ser interligadas entre si para formar a estrutura do programa de visualização, como mostra a Figura 2.2.







FIGURA 2.3 – Tela de configuração da ferramenta *Import*.

Toda ferramenta do programa possui uma tela de configuração semelhante a da ferramenta *Import*, que pode ser selecionada com um duplo clique do mouse na ferramenta ou por meio do menu *Edit*.

Os dados utilizados na estrutura montada na Figura 2.2 são arquivos de contorno político do Brasil com todos os seus estados no formato nativo do OPENDX - o .dx, como mostra a Figura 2.4.



FIGURA 2.4 – Contorno do Brasil e de todo seus estados.



## CAPÍTULO 3

### RESULTADOS E ANÁLISES

Os resultados obtidos compreendem as visualizações dos dados matriciais obtidas com a ferramenta OPENDX, a publicação de um manual técnico sobre essa ferramenta.

#### 3.1 Visualizações para a Dinâmica Interplanetária

Aluna desenvolveu visualizações que simulam a dinâmica interplanetária da Terra. Para tal utilizou da linguagem de programação C para desenvolver dois programas que geram os arquivos de entrada de dados do opendx para esta visualização. Um dos programas apresenta como resultado valores de coordenadas  $xyz$  que representam pontos no espaço, que unidos formam uma linha de campo magnético. O outro programa apresenta como resultado um arquivo com extensão  $.dx$  que é utilizado para fazer a movimentação da partícula a longo da linha de campo magnético. A Figura 3.1 apresenta uma parte da estrutura do programa de visualização desenvolvido no VPE para estes dados e as Figuras 3.2 e 3.3 mostram as imagens resultante.

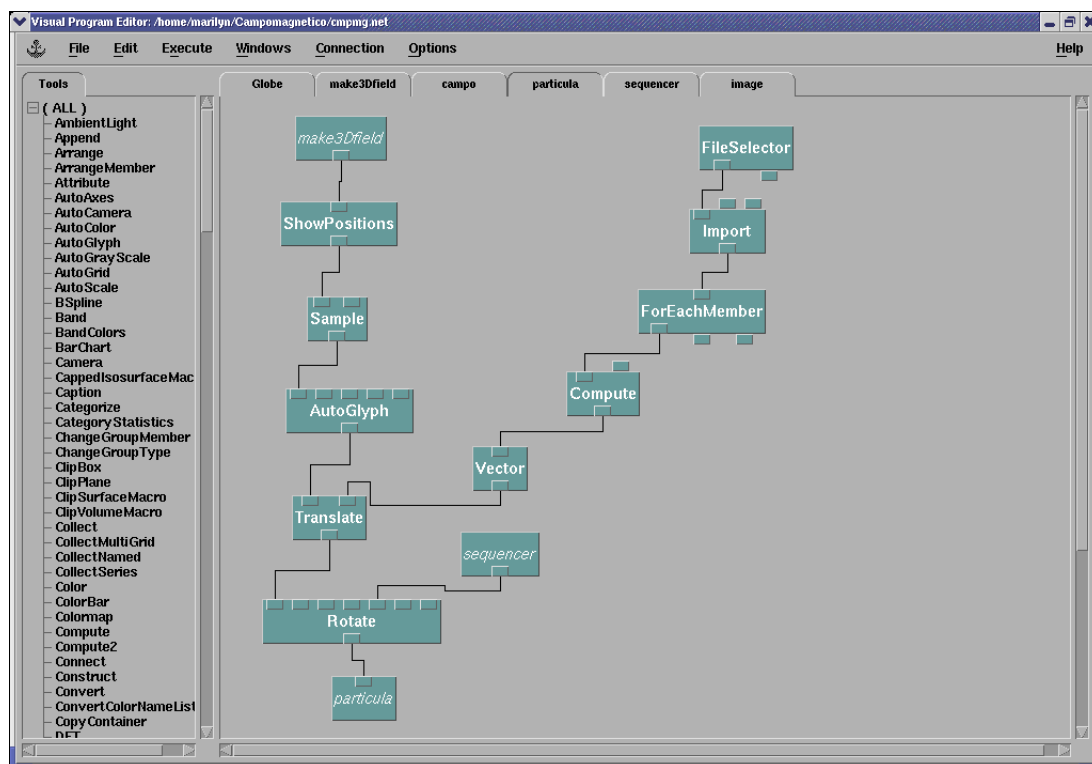


FIGURA 3.1 – Programa de visualização: estrutura do movimento da partícula.

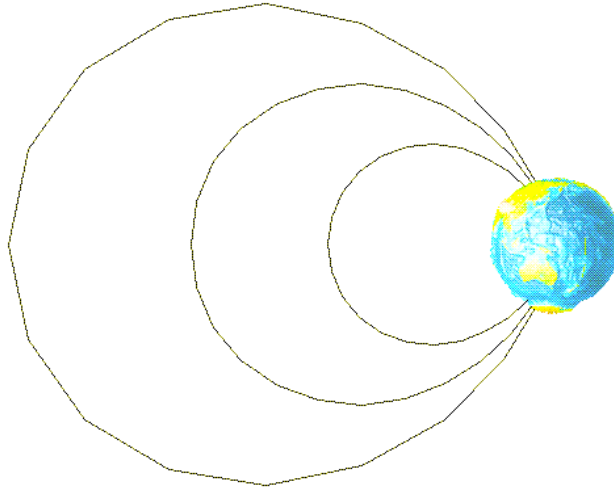


FIGURA 3.2 – Visualização resultante: Imagem da Terra com as linhas de campo magnético.

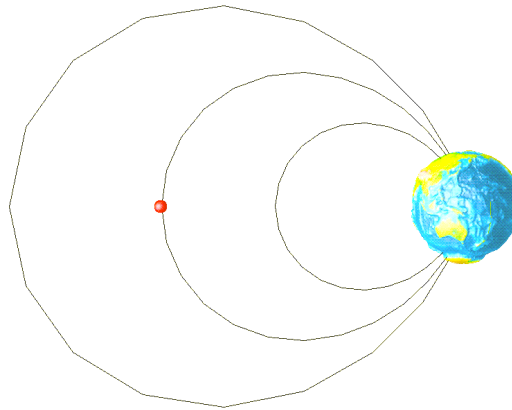


FIGURA 3.3 – Visualização resultante: Imagem da movimentação de um partícula ao longo das linha de campo magnético.

### 3.2 Implementação do pacote Thor/SLA

A aluna finalizou implementaçã à saída do pacote Thor/SLA visualizações feitas no OPENDX com dados de descargas elétricas do sensor STORM. Esta saída representa um matriz de densidade de descargas elétricas de ordem  $n \times n$  que é gerada por meio de um programa na linguagem de programação C++ que foi desenvolvido pela bolsista. Com os dados da matriz de densidade de descargas elétricas foram criadas visualizações 2D e 3D na ferramenta OPENDX . A Figura 3.4 apresenta parte da estrutura do programa de visualização desenvolvido no ambiente gráfico do OPENDX e as figuras 3.5 e 3.6 as imagens resultantes.

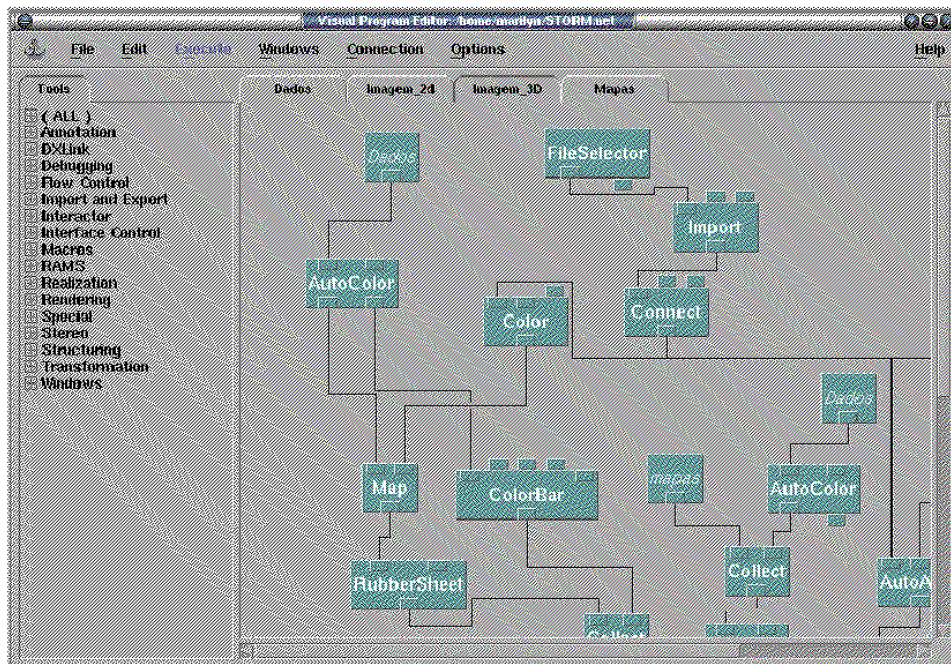


FIGURA 3.4 – Estrutura do programa de visualização.

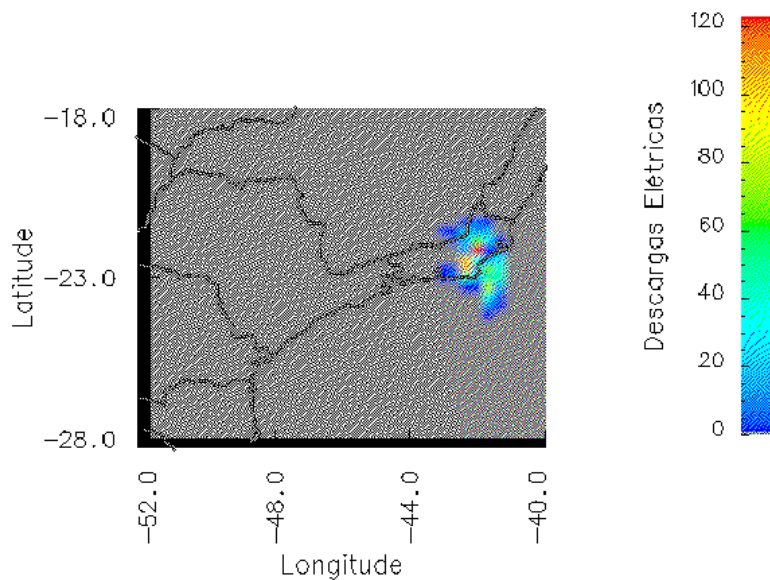


FIGURA 3.5 – Visualização resultante: Imagem 2D dos campos de descargas elétricas.

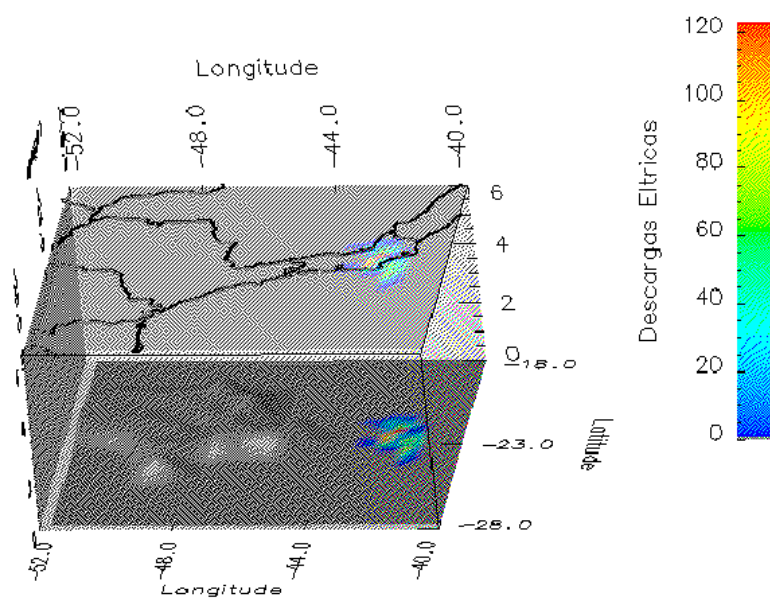


FIGURA 3.6 – Visualização resultante: Imagem 3D dos campos de descargas elétricas.

## CAPÍTULO 4

### CONSIDERAÇÕES FINAIS

A bolsista obteve como resultados a finalização da integração à saída do pacote Thor/SLA visualizações de dados estatísticos de descargas elétricas, o aprimoramento das visualizações de descargas elétricas, o desenvolvimento de visualizações da eletrodinâmica da Terra e o início dos estudos da teoria da computação gráfica .

Os fenômenos atmosféricos e espaciais requerem, dada sua complexidade espacial e dinâmica de processos, abordagens de representações gráficas mais elaboradas, mas provendo interatividade. Integrar recursos de representação espacial em áreas extensas a essa dinâmica é um desafio, pois requer a compreensão dos fenômenos eletrodinâmicos a serem visualizados. O plano previsto de trabalhar com aplicações em Eletrodinâmica Planetária permitirá adquirir a vivência necessária nesses tipos de visualizações. Isso se inseriria nos objetivos estratégicos do INPE.

Por outro lado torna-se conveniente explorar plenamente os recursos de computação gráfica da ferramenta OPENDX . Dessa forma, no biênio que se inicia, a aluna pretende se aprofundar no estudo da computação gráfica para melhor assimilar e aproveitar recursos de animação dessa ferramenta, o que permitirá desenvolver visualizações mais aprimoradas dos fenômenos abordados.





## REFERÊNCIAS BIBLIOGRÁFICAS

- [1] C. A. A. Benetti, E. L. Alvim, S. M. G. Andrade, L. A. R. Assunção, A. F. Cazetta, and R. J. Reis. Rindat - rede integrada nacional de detecção de descargas atmosféricas. In *XI Congresso Brasileiro de Meteorologia*, RJ, 2000. SBMET. [CDROM]. 11
- [2] R. Blaz, M. O. Domingues, and O. Jr. Mendes. *Introdução à entrada de Dados no OPENDX : Formatos DX , .general e .grb*. INPE, Brasil, 2003. 15
- [3] GNU. Cvs, 2005. <<http://www.gnu.org/software/cvs/cvs.html>>. 12, 14
- [4] GNU. Gnu project, 2005. <<http://www.gnu.org>>. 14
- [5] IBM. *Visualization data explorer. Quickstar Guide SC34-3262-02*. IBM, Estados Unidos, setembro 1997. <<http://www.opendx.org>>. 15
- [6] IBM. *Visualization data explorer. Programmer's reference SC38-0486-03*. Estados Unidos, maio 1997. <<http://www.opendx.org>>. 15
- [7] Mike Loukides. *Programando Com Ferramentas GNU*. CONECTIVA INFORMATICA, 2000. 269p. 12, 14
- [8] O. Mendes Jr and M. O. Domingues. Programa para análises de relâmpagos - Thor/SLA. In *XI Congresso Brasileiro de Meteorologia*, RJ, out 2000. SBMET. [CDROM]. 12, 13
- [9] D. Thompson, J. Braun, and R. Ford. *OpenDX Path to Visualization*. Visualization and Imagery Solution, Inc, Missoula, MT, 2001. 207p, <<http://www.vizsolutions.com>>. 15



## APÊNDICE A

### Thor/SLA: Programa em C++

Para a implementação dos recursos de saída do pacote Thor/SLA foi desenvolvido um programa na linguagem de programação C++, que gera uma matriz de densidade de descargas elétricas. Desta matriz cria-se uma visualização com a ferramenta OPENDX, que é a saída do pacote Thor/SLA. A seguir, é listada parte do programa que representa a função responsável pela criação da matriz de densidade de descargas elétricas. A construção da matriz é realizada utilizando o recurso de manipulação de arquivos da linguagem C++. Os dados de latitude e longitude, lidos de um arquivo .str e processados no programa, são convertidos nos índices  $i$  e  $j$  da matriz, respectivamente. A posição  $i \times j$  repetida é acumulada e ao final gera-se um arquivo no qual cada posição da matriz contém a quantidade de repetições que se obteve lendo o arquivo .str.

---

```
void TStorm::SelectSTR(char*aux,char*FileName, char*FileOut){

ifstream pIn;
ofstream pOut;

//Abertura do arquivo .str para leitura

pIn.open(FileName,ios::in);

if(!pIn) TLight::Mens("Error opening Data File",1);

while(pIn){

pIn>>fSTORMDistcorr>>fSTORMDist>>fSTORMAngle>>fSTORMLong>>fSTORMLat
    >>fSTORMTime>>fSTORMQuant>>fSTORMFreq>>fSTORMCity>>aux10>>aux11
    >>fSTORMDate>>fSTORMPadrao1>>fSTORMHora>>fSTORMPadrao2;

fSTORMDLong = atof(aux10);
fSTORMDLat = atof(aux11);

i=abs((int)((fSTORMDLat - posminy)/incry));
j=abs((int)((fSTORMDLong - posminx)/incrx));
```

```
        matriz[i][j]++;
    }
    pIn.close();

//Criacao de um arquivo .dat para gravacao

pOut.open(FileOut,ios::out);

if(!pOut)TLight::Mens("Error opening Data File",1);

for(i=0;i<maxi;i++){
    for(j=0;j<maxj;j++){
        pOut<<matriz[i][j]<<"\t";
    }
    pOut<<"\n";
}
pOut.close();

} //Fim SelectSTR()
```

---

## APÊNDICE B

### Campo Magnético da Terra: Programação m C e em Script

Para a construção da entrada de dados para o OPENDX para o problema de simulação do Campo Magnético da Terra foram desenvolvidos dois programas na linguagem de programação C. Um para produzir as linhas de campo magnético e outro para que produz um arquivo nativo do OPENDX (.dx). Com a intenção de aplicar interação com o usuário também foi desenvolvido um programa em linguagem *script* que possibilita ao usuário a escolha de quantas linhas de campo magnético gostaria de ter na visualização. Esse programa gera como resposta além dos dois arquivos de dados de entrada a imagem resultante destes dois arquivos. A seguir são apresentados os programas desenvolvidos.

#### Programação na Linguagem C

---

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

int main(void){

//Variaveis Locais
float rterra=6370, alpha, phi, lambda_max, lambda_min, lambda;
int quant, quanti, i, j, k, w, numlinha, l;
float Ri, x, y, z, delta;
FILE *fp, *pa;

//Inicio do bloco do calculo das coordenadas xyz
printf("Quantas linhas de campo magnetico deseja inserir? ");
scanf("%d", &numlinha);

for(l=0;l<numlinha;l++){

//Aquisicao de dados
printf("Digite a quantidade de dados:");
scanf("%d",&quant);
printf("Digite o valor de alpha:");
```

```

scanf("%f",&alpha);
printf("Digite o valor de phi:");
scanf("%f",&phi);
printf("Digite o valor maximo de lambda:");
scanf("%f",&lambda_max);
printf("Digite o valor minimo de lambda:");
scanf("%f",&lambda_min);

//Calculo das variantes
lambda=3.14159216*(lambda_min)/180;
delta=3.14159216*((lambda_max - lambda_min)/quant)/180;

//Criacao dos arquivos
fp=fopen("Arqcampm.dx", "w");

if(!fp) return 0;

pa=fopen("Arqcamp.dat", "w");
if(!pa) return 0;

fprintf(pa, "x y z");
fprintf(pa, "\n");

//Loop para o calculo de valores
for(i=0;i<=quant;i++){

Ri=alpha*rterra*cos(lambda)*cos(lambda);

x=Ri*cos(lambda)*cos(3.14159216*phi/180);
y=Ri*cos(lambda)*sin(3.14159216*phi/180);
z=Ri*sin(lambda);

//Dados para o arquivo de coordenada
fprintf(pa, "%f %f %f", x, y, z);
fprintf(pa, "\n");

//Dados para oarquivo .dx

```

```

fprintf(fp,"object %d class array type float rank 1 shape 3 items 1 data
fprintf(fp,"\n");
fprintf(fp,"%f %f %f",x,y,z);
fprintf(fp,"\n");
fprintf(fp,"attribute \"dep\" string \"positions\");
fprintf(fp,"\n");

lambda=lambda+delta;

}
fclose(pa);
quanti=i;

for(j=quanti;j<=quant+i;j++){

fprintf(fp,"object %d class field", j+1);
fprintf(fp,"\n");
fprintf(fp,"component \"positions\" value file \"coneccao.dat\",1");
fprintf(fp,"\n");
fprintf(fp,"component \"connections\" value file \"coneccao.dat\",2");
fprintf(fp,"\n");
fprintf(fp,"component \"data\" value %d", (j-quanti)+1);
fprintf(fp,"\n");
}

fprintf(fp,"object \"series\" class series");
fprintf(fp,"\n");

for(k=0;k<quant;k++){

fprintf(fp,"member %d value %d position 0",k, quanti+k+1);
fprintf(fp,"\n");
}

fprintf(fp,"end");
fclose(fp);

```

```
}  
}
```

---

## Programação na Linguagem C

---

```
#include <stdio.h>  
#include <stdlib.h>  
#include <math.h>  
  
int main(void){  
  
    //Variaveis Locais  
    float rterra=6370, alpha, phi, lambda_max, lambda_min, lambda;  
    int quant;  
    float Ri, x, y, z, delta, i;  
    FILE *fp;  
  
    //Aquisicao de dados  
    printf("Digite a quantidade de dados:");  
    scanf("%d",&quant);  
    printf("Digite o valor de alpha:");  
    scanf("%f",&alpha);  
    printf("Digite o valor de phi:");  
    scanf("%f",&phi);  
    printf("Digite o valor maximo de lambda:");  
    scanf("%f",&lambda_max);  
    printf("Digite o valor minimo de lambda:");  
    scanf("%f",&lambda_min);  
  
    //Variantes  
    lambda=3.14159216*(lambda_min)/180;  
    delta=3.14159216*((lambda_max - lambda_min)/quant)/180;  
  
    fp=fopen("Arqcampm.dat","w");  
  
    if(!fp) return 0;
```



```

//Calculo dos valores das coordenadas
for(i=0;i<=quant;i++){

Ri=alpha*rterra*cos(lambda)*cos(lambda);

x=Ri*cos(lambda)*cos(3.14159216*phi/180);
y=Ri*cos(lambda)*sin(3.14159216*phi/180);
z=Ri*sin(lambda);

fprintf(fp,"%f %f %f",x,y,z);
fprintf(fp,"\n");

lambda=lambda+delta;

}
fclose(fp);
}

```

---

## **Programação em Script**

---

```

#!/bin/bash
# Programa para a criação de arquivos com dados para gerar
# linhas de campo magnetico
#
#Marilyn Menecucci ibanez
#PIBIC - INPE -2006
#
export Opc=0
while [ "$Opc" -lt 1 -o $Opc -gt 4 ]
do
clear

```

```
echo -e "
```

```
*****
```

```
*      Campo Magnetico      *
```

```
*****
```

```
OPCAO ACAO
```

```
1 Manter os dados anteriores no arquivo
```

```
2 Criar novos dados
```

```
3 Finalizar
```

```
Entre com a opcao desejada: \c "
```

```
read Opc
```

```
#Selecao de opcoes
```

```
if [ $Opc -eq 3 ]
```

```
    then
```

```
exit
```

```
    fi
```

```
    if [ $Opc -eq 1 ]
```

```
        then
```

```
g++ -o cpmg1 /home/marilyn/Campomagnetico/campomagdx3.c
```

```
./cpmg1
```

```
dx -script /home/marilyn/Campomagnetico/cmpmg.net
```

```
    fi
```

```
    if [ $Opc -eq 2 ]
```

```
        then
```

```
g++ -o cpmg2 /home/marilyn/Campomagnetico/campomagdx2.c
```

```
./cpmg2
```

```
dx -script /home/marilyn/Campomagnetico/cmpmg.net
```

```
    fi
```

```
done
```