



Ministério da
**Ciência, Tecnologia
e Inovação**



sid.inpe.br/mtc-m21b/2014/11.27.16.27-MAN

**SISTEMA PARA PREVISÃO OPERACIONAL DA
DINÂMICA DA IONOSFERA BASEADO NO MODELO
SUPIM V2**

Fernando Emilio Puntel
Adriano Petry
Jonas Rodrigues de Souza
Haroldo Fraga de Campos Velho

URL do documento original:
<<http://urlib.net/8JMKD3MGP3W34P/3HFHJTB>>

INPE
São José dos Campos
2015

PUBLICADO POR:

Instituto Nacional de Pesquisas Espaciais - INPE

Gabinete do Diretor (GB)

Serviço de Informação e Documentação (SID)

Caixa Postal 515 - CEP 12.245-970

São José dos Campos - SP - Brasil

Tel.:(012) 3208-6923/6921

Fax: (012) 3208-6919

E-mail: pubtc@sid.inpe.br

CONSELHO DE EDITORAÇÃO E PRESERVAÇÃO DA PRODUÇÃO INTELLECTUAL DO INPE (RE/DIR-204):**Presidente:**

Marciana Leite Ribeiro - Serviço de Informação e Documentação (SID)

Membros:

Dr. Gerald Jean Francis Banon - Coordenação Observação da Terra (OBT)

Dr. Amauri Silva Montes - Coordenação Engenharia e Tecnologia Espaciais (ETE)

Dr. André de Castro Milone - Coordenação Ciências Espaciais e Atmosféricas (CEA)

Dr. Joaquim José Barroso de Castro - Centro de Tecnologias Espaciais (CTE)

Dr. Manoel Alonso Gan - Centro de Previsão de Tempo e Estudos Climáticos (CPT)

Dr^a Maria do Carmo de Andrade Nono - Conselho de Pós-Graduação

Dr. Plínio Carlos Alvalá - Centro de Ciência do Sistema Terrestre (CST)

BIBLIOTECA DIGITAL:

Dr. Gerald Jean Francis Banon - Coordenação de Observação da Terra (OBT)

REVISÃO E NORMALIZAÇÃO DOCUMENTÁRIA:

Maria Tereza Smith de Brito - Serviço de Informação e Documentação (SID)

Yolanda Ribeiro da Silva Souza - Serviço de Informação e Documentação (SID)

EDITORAÇÃO ELETRÔNICA:

Maria Tereza Smith de Brito - Serviço de Informação e Documentação (SID)

André Luis Dias Fernandes - Serviço de Informação e Documentação (SID)



Ministério da
**Ciência, Tecnologia
e Inovação**



sid.inpe.br/mtc-m21b/2014/11.27.16.27-MAN

**SISTEMA PARA PREVISÃO OPERACIONAL DA
DINÂMICA DA IONOSFERA BASEADO NO MODELO
SUPIM V2**

Fernando Emilio Puntel
Adriano Petry
Jonas Rodrigues de Souza
Haroldo Fraga de Campos Velho

URL do documento original:
<<http://urlib.net/8JMKD3MGP3W34P/3HFHJTB>>

INPE
São José dos Campos
2015



Esta obra foi licenciada sob uma Licença Creative Commons Atribuição-NãoComercial 3.0 Não Adaptada.

This work is licensed under a Creative Commons Attribution-NonCommercial 3.0 Unported License.

LISTA DE FIGURAS

	<u>Pág.</u>
2.1 Estrutura do Sistema SUPIM-DAVS	3
2.2 Chamada para enviar notificação por e-mail	5
2.3 Estrutura de diretórios para organizar as simulações já realizadas	7
3.1 Diagrama estrutural do SUPIM-DAVS	9
3.2 Diretório Temporário diaJuliano.PID	14
3.3 Exemplo de imagem para simulação TEC para América do Sul	18
3.4 Exemplo de imagem para simulação TEC para o Globo	19

LISTA DE ABREVIATURAS E SIGLAS

DAVS	–	Data Assimilation and Visualization System
FTP	–	File Transfer Protocol
GrADS	–	Grid Analysis and Display System
GRIB	–	Gridded Binary or General Regularly
IDL	–	Interactive Data Language
MPI	–	Message Passing Interface
SUPIM	–	Sheffield University Plasmasphere Ionosphere Model
SIP	–	Solar Irradiance Platform
TEC	–	Total Electron Content

SUMÁRIO

	<u>Pág.</u>
1 INTRODUÇÃO	1
2 ESTRUTURA DO SISTEMA	3
2.1 Diretórios da pasta “Install”	4
2.1.1 AssimilationData	4
2.1.2 Compiler	4
2.1.3 DAVS	4
2.1.4 Email	4
2.1.5 GenerateDAVSInputs	5
2.1.6 GenerateSUPIMInputs	5
2.1.7 GrADS	6
2.1.8 SOLAR2K	6
2.1.9 SUPIM_CE_07	6
2.2 Diretórios da pasta “Runs”	6
2.2.1 Globe e SouthAmerica	7
3 EXECUÇÃO E OPERAÇÃO	9
3.1 Arquivo de Configuração	9
3.2 Inicialização da Simulação	13
3.3 Execução do código DAVS - Assimilação de Dados	14
3.4 Obtenção de dados de Fluxo Solar	15
3.5 Execução do código SUPIM	16
3.6 Execução do código DAVS - Ajuste de tempo e Interpolação de dados	16
3.6.1 DAVS-LINUX timing adjust	17
3.6.2 DAVS-LINUX interpolation	17
3.7 Geração de imagens	17
3.8 Finalização da Simulação	18
4 CONCLUSÕES	21
REFERÊNCIAS BIBLIOGRÁFICAS	23

1 INTRODUÇÃO

Este manual foi desenvolvido com a intenção de facilitar o entendimento do código de um sistema desenvolvido para previsão da dinâmica ionosférica baseado no modelo SUPIM. É documentado todo o processo de simulação, desde a instalação, arquivo de configuração e a execução.

Este sistema, chamado SUPIM-DAVS, está em execução desde março de 2011 em um cluster de processadores no Centro Regional Sul de Pesquisas Espaciais (CRS/INPE). As imagens resultantes de cada simulação são disponibilizadas diariamente no sítio do Programa de Clima Espacial do INPE - http://www2.inpe.br/climaespacial/pt/tec_supim. Atualmente o arquivo de configuração para executar a simulação está na versão 7.1.

Inicialmente, o capítulo 2 detalha como o sistema foi estruturado, indicando a criação de novos diretórios e a localização de vários arquivos utilizados durante a execução. No capítulo 3, é detalhada a execução e operação do SUPIM-DAVS, desde os parâmetros do arquivo de configuração, geração e obtenção de arquivos, execução e por fim geração das imagens. Ao final, são apresentadas conclusões e indicadas possíveis atividades futuras.

2 ESTRUTURA DO SISTEMA

Para melhor organização e compreensão, o sistema SUPIM-DAVS foi separado em dois diretórios principais: *Install* e *Runs*. A Figura 2.1 ilustra a estrutura completa de diretórios.

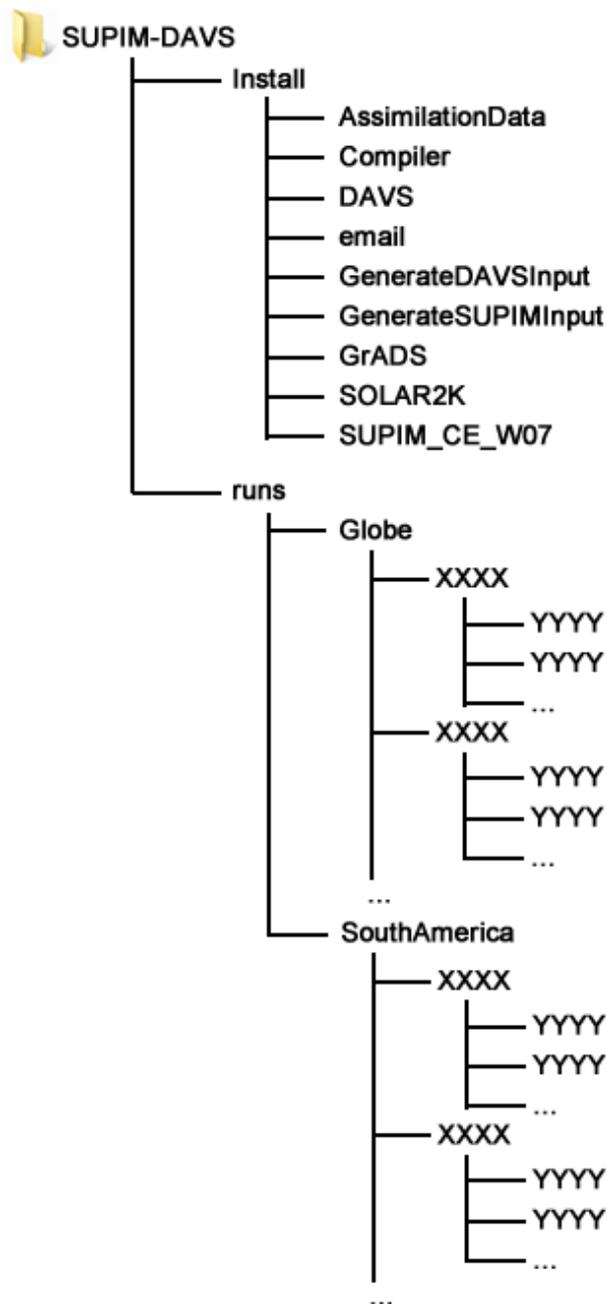


Figura 2.1 - Estrutura do Sistema SUPIM-DAVS

2.1 Diretórios da pasta “Install”

No diretório “Install” estão os arquivos utilizados no desenvolvimento e utilizados para instalação e o “*setup*” do sistema, que são detalhadas nas próximas subseções.

2.1.1 AssimilationData

Arquivos escritos na linguagem C++ e bibliotecas que realizam buscas de dados nas ionossondas e também em sites da internet via FTP.

2.1.2 Compiler

Armazenamento e instalação dos compiladores usados para as linguagens de programação Fortran e C da Intel.

2.1.3 DAVS

Arquivos-fonte escritos na linguagem C++ do *Data Assimilation and Visualization System* (DAVS). O arquivo executável é "SUPIM-DAVS.exe" realiza a assimilação de dados, ajustes da localização temporal dos pontos simulados, cria arquivos para que o GrADS seja capaz de realizar a leitura e é responsável por grande parte do pré e pós-processamento das simulações.

O DAVS possui três funcionalidades principais:

- timeAdjust: ajuste relacionado a correção para diferenças longitudinais entre o local da simulação e os pontos simulados;
- GrADS interpolation: interpola dados de concentração eletrônica a partir dos pontos simulados em grade homogeneamente espaçada em latitude, longitude e altitude;
- AssimilationData: dados adicionais são obtidos através de instrumentos de observação, que são incorporados aos resultados obtidos através das rodadas do SUPIM, deixando assim o resultado mais realista.

2.1.4 Email

Programa escrito na linguagem de programação Java, utilizado para enviar e-mails de notificação durante a execução da simulação. Para isso são utilizadas as bibliotecas javax.mail e javax.mail.internet próprias do Java.

O programa recebe dois parâmetros por referência no momento da compilação: o primeiro parâmetro é referente ao assunto do e-mail, e o segundo ao texto. A Figura 2.2 ilustra como é chamado o conjunto de classes SendMail.jar.

```
if test $? -ne 0; then
  echo "Error: no previous simulation to perform data assimilation..."
  cd $DAVS_path/install/email/
  javac -classpath ./mailapi.jar SendMail.java
  jar -cfm SendMail.jar META_INF/MANIFEST.MF *.class
  body=`cat $logFile`
  if test $notifyStatus -eq 1 then
    java -jar SendMail.jar "SUPIM Problem... Error on finding directory structure to perform data
    assimilation... Location: $outputDirectory/$YEAR/$julianDay" "$body"
  fi
  mv $logFile $outputDirectory/$YEAR/$julianDay/
  exit
fi
```

Figura 2.2 - Chamada para enviar notificação por e-mail

Antes de chamar a classe SendMail.jar é preciso abrir o diretório onde estão as classes e fazer a compilação do programa.

2.1.5 GenerateDAVSInputs

Programa escrito na linguagem de programação Java, responsável por criar arquivos de entrada para o DAVS, além de criar arquivos script para geração de imagens, utilizado pelo GrADS.

O primeiro passo para gerar os arquivos de entrada é verificar se os parâmetros passados para o programa estão corretos. Após a verificação são criados os arquivos com os parâmetros, que posteriormente serão utilizados no decorrer da execução da simulação.

2.1.6 GenerateSUPIMInputs

Neste diretório há um arquivo escrito em Java. Este programa é responsável por gerar arquivos de entrada para o SUPIM, que poderá ser executado possivelmente usando gerenciador de recursos OAR ou MPI, implementado através da MPICH2.

Inicialmente é criado o arquivo de comando, e a seguir são criados os arquivos de entrada do SUPIM. A quantidade de arquivos de entrada será correspondente ao valor definido na variável "numberSimulation".

2.1.7 GrADS

Contém os arquivos de instalação do software GrADS, e também dois scripts de execução:

- `scriptPrintImage.gs`: script responsável pela geração de imagens. Aqui são definidos parâmetros como tamanho e cor de fonte, cor da palheta de cores, informações adicionais escritas na imagem final, entre outros;
- `cbar.gs`: script responsável por construir a escala de cores na imagem.

2.1.8 SOLAR2K

Diretório onde são armazenados dados e instalação do SOLAR Irradiance Platform (<http://www.spacewx.com/solar2000.html>).

O diretório com os arquivos e a licença desta versão também estão presentes neste diretório, além de ter outra pasta chamada `checkSOLAR`, que é responsável por fazer download dos dados recebidos e checar a validade dos mesmos.

São três arquivos escritos em linguagem C:

- `downloadSOLAR.c`: Responsável por fazer download dos dados dos sistema SOLAR 2000.
- `checkSOLAR.c`: Responsável por checar a consistência dos dados recebidos;
- `s2k_prev.c`: Responsável pela criação dos valores de previsão de dados de fluxo solar.

2.1.9 SUPIM_CE_07

Diretório que contém os arquivos do *Sheffield University Plasmasphere Ionosphere Model* (SUPIM), escritos na linguagem de programação Fortran, utilizado para simular processos físico-químicos de ionização e neutralização na ionosfera.

2.2 Diretórios da pasta “Runs”

No diretório “Runs” estão os arquivos resultantes das rodadas das simulações referente a cada dia. A Figura 2.3 ilustra a estrutura dos diretórios, que são detalhadas na próxima subseção.

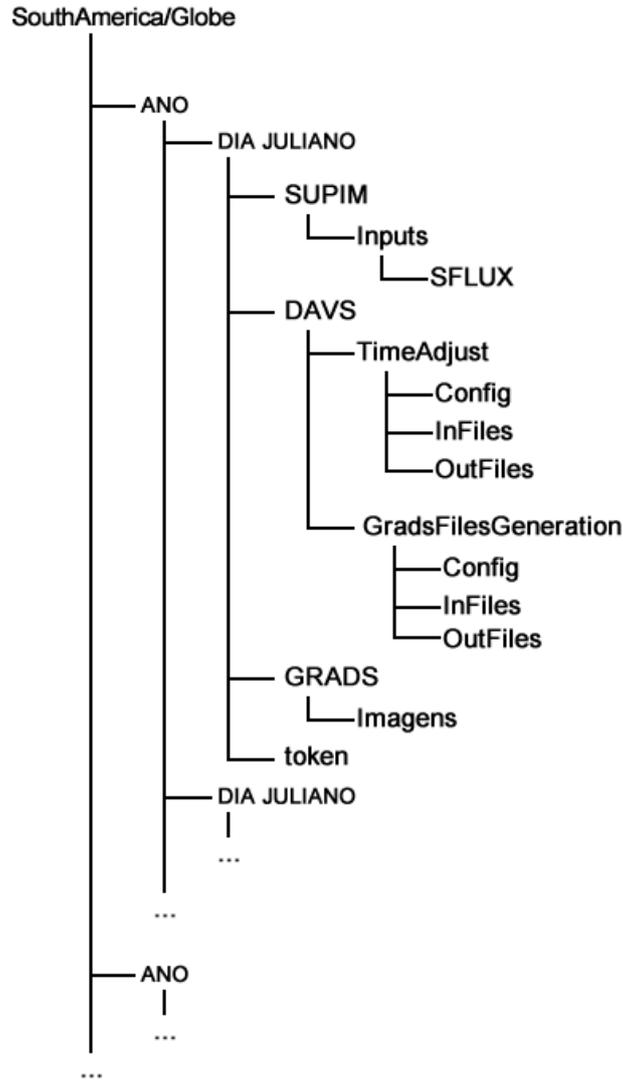


Figura 2.3 - Estrutura de diretórios para organizar as simulações já realizadas

2.2.1 Globe e SouthAmerica

Como já apresentado na Figura 2.3, os diretórios Globe e SouthAmerica possuem a mesma estrutura, o que irá definir em qual pasta será salva a simulação irá depender de como a simulação foi realizada. Caso seja somente para a América do Sul, o resultado será salvo na pasta SouthAmerica. Caso seja para o Globo, a simulação será salva na pasta Globe.

Essa organização e nomenclatura foram usados para facilitar a busca por simulações já realizadas. Dentro da pasta referente ao dia Juliano, existem 4 principais diretórios:

- SUPIM: Armazenamento de arquivos gerados durante a execução do SU-

PIM;

- DAVS: Armazenamento do arquivo de configuração e arquivos gerados durante a execução do DAVS;
- GRADS: Arquivos de saída do DAVS;
- token: Armazenamento de arquivos para uso na simulação do dia seguinte.

3 EXECUÇÃO E OPERAÇÃO

O sistema SUPIM-DAVS executa diariamente, com a intenção de gerar mapas com previsões da dinâmica da ionosfera para um período futuro. A sua execução é agendada utilizando o CRON do Linux. A execução das simulações são feitas para a América do Sul e para o Globo. A Figura 3.1 ilustra o diagrama de execução do SUPIM-DAVS.

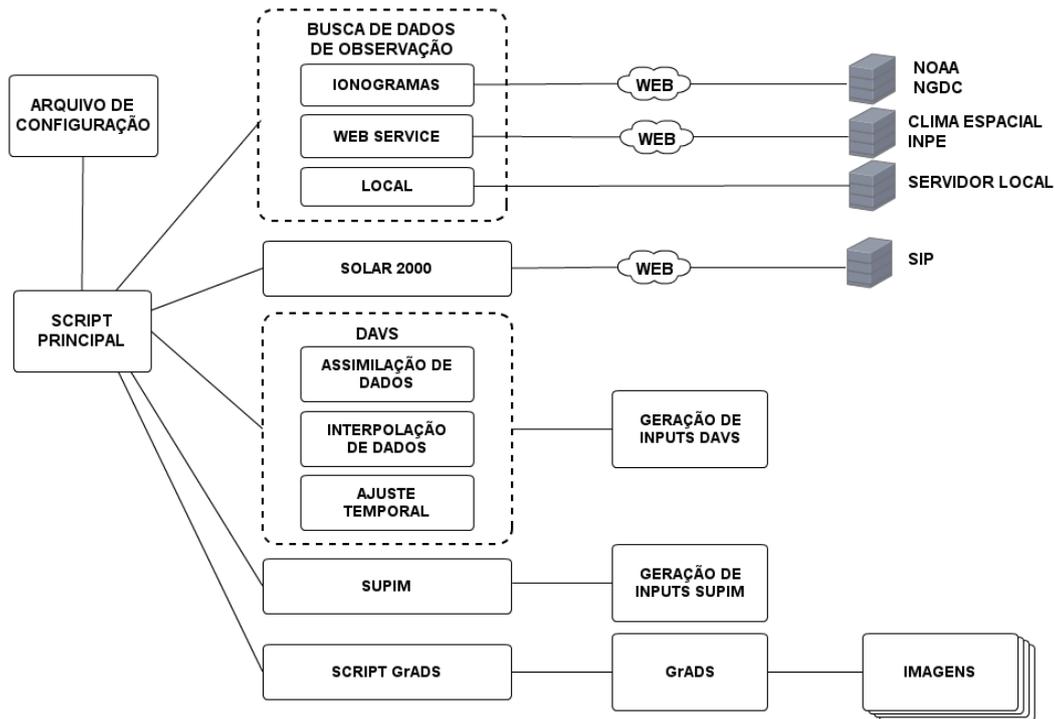


Figura 3.1 - Diagrama estrutural do SUPIM-DAVS

O sistema pode ser dividido em 8 partes distintas, que serão detalhadas nas próximas subseções.

3.1 Arquivo de Configuração

Cada simulação pode ser executada de maneira diferente. O que irá diferenciar é o arquivo de configuração, que define forma, técnicas, algoritmos e parâmetros utilizados.

Para melhor compreensão, cada parâmetro é explicado a seguir:

- waitFor: simulação que não deve estar rodando para iniciar.

- numberSimulation: número de simulações em longitudes diferentes que o SUPIM será executado
- startingLongitude: longitude inicial do programa;
- longitudeStep: define o espaçamento longitudinal entre cada simulação;
- continuity: horário salvo, para que quando a assimilação de dados for executada, saber onde deve começar executar;
- dataAssimilation, runSupim, timeAdjust, gradsFilesGeneration, runGrads, runPutImage: cada uma dessas variáveis representa a execução uma parte autônoma do código. Quando definidas como 1 (um), a parte no código referente será executada, quando definida como 0 (zero) o sistema irá ignorar a parte.
- dateGeneration: esta variável define para qual data será feita a simulação. Ela pode receber três valores:
 - 0 (zero) - a execução será referente a data do sistema;
 - 1 (um) - a execução é feita para o próximo dia, considerando a data do sistema;
 - -1 (menos um) - a data é pré-definida pelo programador utilizando os parâmetros: PredefinedJulinDay, PredefinedDAY, PredefinedMOUNTH e PredefinedYEAR;
- reconstruct: é utilizado como um flag para reconstrução de simulações passadas, caso seja necessário.
- dataAssimilationTime: define se será utilizada a hora atual do sistema (0) ou horário pré-definido (-1) através das variáveis: PrefinedDataAssimilationHour e PrefinedDataAssimilationMinute;
- SolarFile: quando esta variável é definido como 1 (um), representa que o código irá obter os arquivos do SOLAR 2000 da internet; quando é definido como 0 (zero), será utilizado o arquivo local de instalação sem verificação; quando definido como 2 (dois) que tentará baixar para os últimos 2 (dois) dias, ou caso falhar irá utilizar dados em bdSolar.txt.
- DEBUGmode: define se o modo debug ficará desligado (0 - padrão), ou se o debug ficará ligado (1). Caso o debug esteja ligado, não é feito limpeza nos arquivos.

- `destinationServer`: servidor de destino das imagens;
- `notifyStatus`: define se o e-mails serão enviados (1 - padrão) ou não (0);
- `DAVS_path`, `logFile`: caminho completo do sistema e arquivos de log;
- `outputDirectory`: define o diretório de saída;
- `clusterUse`: variável utilizada somente para a execução do SUPIM, pode receber os valores:
 - 0 - quando é utilizado o MPICH2 (GROPP et al., 2007) sendo chamado pelo gerenciador de recursos OAR (NICOLAS, 2014) para paralelismo de atividades;
 - 1 - quando é utilizado MPICH2 diretamente para paralelismo das atividades;
 - 2 (padrão) - quando é utilizado o gerenciador de recursos OAR com simulações independentes.
- `coresPerProcessSUPIM`, `coresPerProcessDataAssimilation`, `coresPerProcessTimesAdjust`, `coresPerProcessGradsFilesGeneration`: parâmetros para otimização do uso do cluster, ignorado para clusters diferentes;
- `APIndex`: parâmetro utilizado em arquivos de entrada do SUPIM, este valor pode ser:
 - 4 - valor padrão, utilizando em dias calmos;
 - 10 - utilizado em dias de tempestades geomagnéticas.
- `JDAY`: pode receber dois valores:
 - 2 - caso seja a continuação de uma execução do dia anterior;
 - 0 - caso seja uma execução que não considera a ionização do dia seguinte.

Define se a simulação será a continuação

- `F10Limit_1`, `F10Limit_2`, `F10Limit_3`, `F10Limit_4`, `F10Limit_5`: define valores de fluxo solar F10.7 para alterar escalas TECU, ou seja, servem para definir o valor máximo da escala de cores usada nos mapas.

A Tabela 3.1 corresponde a configuração padrão do DAVS, para simulações na América do Sul e Globo.

Tabela 3.1 - Parâmetros para Configuração do DAVS

Variável	SouthAmerica	Globe	Significado
numberTimeSteps	24	24	Numero de execuções
timeStep	60	60	Tempo total da execução
initialTime	0	0	Tempo inicial
lonBegin	-90	-180	Valor da Longitude Inicial
lonEnd	-25	180	Valor da Longitude Final
lonStep	1	6	Valor referente ao incremento na Longitude para cada simulação
latBegin	-65	-60	Valor da Latitude Inicial
latEnd	20	60	Valor da Latitude Final
latStep	1	6	Valor referente ao incremento na Latitude para cada simulação
elevBegin	90000	90000	Valor da Elevação Inicial
elevEnd	1000000	1000000	Valor da Elevação Final
elevStep	10000	10000	Valor referente ao incremento na Elevação a cada simulação
interpolationAlgorithm	"IDW"	"IDW"	Algoritmo utilizado para Interpolação de Dados
nearestNeighborAlgorithm	"ANN"	"ANN"	Algoritmo utilizado para procurar os vizinhos mais próximos na Interpolação de Dados
top	400	400	Número de vizinhos utilizados na Interpolação de Dados
explosion	1	-1	Parâmetro Utilizado no algoritmo IE
powerFactor	2	2	Define do expoente para distância utilizado na Interpolação IDW
generateAssimilationDataFile	2	2	Define se os novos arquivos de dados serão gerados
dataAssimilationMethod	"Nudging"	"Nudging"	Método da Assimilação de Dados
influenceRadius	22000	222000	Raio de influência
correctionFactor	0.4	0.4	Fator de correção para Assimilação de Dados

- sshDir: diretório de chaves ssh.
- Após o diretório de chaves ssh são configurados as variáveis do sistema, o caminho para o MPICH2.
- gradsPath, scriptGrADS, imageOrientation: esta variáveis são referentes a configuração do GrADS.
- backup, backupDir: Variáveis referentes ao backup, na primeira variável é setado se será feito backup ou não, e na segunda é o diretório onde o backup estará.
- tecmap: referente ao uso de dados de observação tecmap. Os possíveis valores que podem ser definidos são:
 - 0 - não será utilizado dados tecmap;
 - 1 - será utilizado dados do servidor local;
 - 2 - será baixado e utilizado dados do webservice.
- slidingWindow: Define se a simulação irá gerar 24 (vinte e quatro) horas a frente incluindo dia posterior e ao simulado se necessário, ou apenas da 0 (zero) horas as 23 (vinte e três) horas da data da simulação;
- slidingWindowTime: Se será utilizada a hora do sistema (0) ou horário pré-definido (1).
- PreDefinedSlidingWindowHour, PreDefinedSlidingWindowMinute: valores do tempo pré-definido, só é levado em consideração caso a variável "slidingWindowTime" seja igual a 1 (um).

3.2 Inicialização da Simulação

Inicialmente, a simulação identifica a data (dia, mês e ano) que será executada. O próximo passo é verificar se o número de parâmetros está correto. Caso esteja correto os parâmetros são mostrados na tela. Após isso, é verificado se o espaço em disco é suficiente para a simulação. Caso não tenha espaço suficiente a simulação é abortada e é enviado um e-mail informando a causa. Caso haja espaço suficiente, a simulação verifica se outra simulação excludente já está rodando no cluster que é definida através do parâmetro "waitFor". Caso esteja, espera por no máximo 8 horas para essa simulação concluir.

Após conferir os parâmetros da simulação, é necessário verificar se existem nós de processamento ativos no cluster. Caso não tenha, a simulação é abortada, enviando um e-mail informativo aos responsáveis. Esta verificação é feita através de um comando "ping" para os nós do clusters.

O próximo passo é verificar a data e o dia juliano corretos, considerando anos bissextos. Também é encontrado o último dia juliano, o dia anterior e o dia antes desse.

A seguir, é testado se será rodada uma simulação com ou sem assimilação de dados. Caso seja uma simulação sem assimilação são criados os diretórios temporários para a execução da simulação. No caso de assimilação, os diretórios já devem estar criados. Todos os arquivos que serão utilizados durante a execução são copiados para este diretório, mantendo assim a integridade dos arquivos principais. Este diretório é criado na diretório "install". O nome do novo diretório é renomeado com o diaJuliano.PID do sistema. A Figura 3.2 ilustra o nome da pasta e também os diretórios dentro dela.

```
-bash-4.1$ cd install/
-bash-4.1$ ls -l
drwxrwxr-x 6 supim-davs supim-davs 4096 Jul 30 14:42 118.22193
-bash-4.1$ cd 118.22193/
-bash-4.1$ ls -l
total 156
drwxr-xr-x 5 supim-davs supim-davs 4096 Jun 6 10:18 AssimilationData
drwxr-xr-x 2 supim-davs supim-davs 4096 Abr 30 15:14 checkSOLAR
drwxr-xr-x 2 supim-davs supim-davs 4096 Abr 30 15:15 GenerateDAVSInputs
drwxr-xr-x 2 supim-davs supim-davs 4096 Abr 30 15:13 GenerateSUPIMInputs
-rwxr-xr-x 1 supim-davs supim-davs 131219 Abr 30 15:13 run_SUPIM-DAVS_v7.0b.sh
-rw-r--r-- 1 supim-davs supim-davs 4544 Abr 30 15:13 v7.assimilationTest.conf
-bash-4.1$
```

Figura 3.2 - Diretório Temporário diaJuliano.PID

Antes de começar a execução da simulação os programas que irão executar são compilados.

3.3 Execução do código DAVS - Assimilação de Dados

A assimilação de dados utiliza um conjunto de técnicas para que seja possível realizar adequadamente inserção de dados de observação em um sistema de previsão de dados.

O primeiro passo na assimilação de dados, é realizar o download dos dados observacionais.

3.4 Obtenção de dados de Fluxo Solar

As informações de fluxo solar são obtidas a partir do software SIP (Solar Irradiance Platform), versão "System Grade"(SIP_System_Grade) (TECHOLOGIES, 2014) da empresa "Space Environment Technologies". Esta versão da plataforma permite realizar downloads de dados desejados através de uma função disponibilizada em Interactive Data Language (IDL), da empresa "ITT Visual Information Solutions"(EXELIS, 2014).

Essa função pode ser invocada a partir de um programa escrito em várias linguagens de programação. Foi escrito um programa na linguagem C (downloadSOLAR.c). Este programa invoca uma função em IDL que via internet obtém os dados de fluxo solar para uma data pré-determinada em um arquivo texto.

Quando obtidos dados de fluxo solar para dia anteriores, os arquivos recebidos são completos, porém quando solicitados dados de fluxo solar para datas futuras, os arquivos podem não conter informações essenciais. Por este motivo foi escrito um programa na linguagem C para avaliar se os dados recebidos estão completos (checkSOLAR.c). A simulação somente dará continuidade quando obtiver todos as informações necessárias para o processamento.

Na verificação dos dados utilizado checkSOLAR.c, é criado o arquivo OK.txt ou notOK.txt, quando o arquivo possui as informações necessárias ou quando está faltando dados, respectivamente. A partir do arquivo criado, a simulação saberá se é necessário fazer download de outros dados ou se poderá utilizar os dados que já foram recebidos.

Caso seja necessário obter dados dos dias anteriores para simulação do dia corrente, é necessário fazer uma estimativa linear dos dados. Para isso é utilizado o programa s2k_prev.c.

Na versão atual do sistema (7.1), caso não seja possível realizar o download dos dados de fluxo solar do dia corrente e do anterior, é invocado uma função para realizar dados manualmente, onde é obtido dados através de um arquivo de texto disponibilizado pela NOAA (<http://www.swpc.noaa.gov/ftplib/indices/DSD.txt>). Este arquivo contém os dados de F10.7 dos últimos 30 (trinta) dias, para as simulações é necessário que tenha o dado para o dia seguinte, para isso é realizado uma projeção de dados do dia ontem e de antes de ontem, resultado no dado desejado, também é feito uma média dos últimos 30 (dias).

Ao final da obtenção de dados é copiado o arquivo obtido (s2k_output.txt e OK.txt) para o diretório correspondente a simulação e verificada a existência do arquivo de fluxo solar no diretório.

3.5 Execução do código SUPIM

Nesta versão do sistema, o código SUPIM (SANTOS et al., 2005) pode executar paralelamente de três maneiras distintas: através da técnica Message Passing Interface (MPICH) versão 2 aplicada diretamente aos nós do cluster; outra maneira também utilizando MPICH2 submetida ao gerenciador de recursos OAR; ou sem o uso do MPICH2 disparando os processos paralelos diretamente através do gerenciador de recursos OAR. Como já explicado anteriormente, a variável "clusterUse" no arquivo de configuração irá indicar de que maneira será executada.

Diferentemente das duas primeiras maneiras de executar, a execução com processos paralelos vem se mostrando mais adequada para execução, já que se uma rodada do SUPIM falhar não compromete a execução dos demais.

O primeiro passo é a geração dos arquivos de entrada para o SUPIM, através de Java.

Ao final é salvo os dados resultantes do SUPIM e os dados de continuidade para o próximo dia.

Foi implementado um mecanismo de proteção de tempo já decorrido, que roda durante toda a simulação. Caso a simulação tenha aguardado mais de 6 (seis) horas pelos resultados, o mecanismo finaliza automaticamente a execução e envia um e-mail aos responsáveis.

3.6 Execução do código DAVS - Ajuste de tempo e Interpolação de dados

O principal objetivo da execução do DAVS é construir uma grade homogeneamente espaçada em latitude, longitude e altitude geográfica com estimação de valores de concentração iônica/eletrônica em cada ponto, através dos dados gerados pelas várias rodadas do SUPIM (PETRY et al., 2011). Além disso, o DAVS realiza a assimilação de dados da grade gerada, fazendo com que os dados resultantes possuam as informações corretas.

Antes de começar a execução é necessário resgatar os arquivos gerados pelas rodadas

do SUPIM. O DAVS é executado paralelamente para cada hora simulada, gerando um total de 24 mapas com previsões de conteúdo eletrônico. A execução paralela é feita utilizando o gerenciador de recursos OAR.

O DAVS pode ser dividido em três funcionalidades básicas: a primeira, já explicada anteriormente, responsável pela assimilação de dados; a segunda, é chamada de "timming adjust" que verifica possíveis atrasos ou adiantamentos nas simulações; e a última, parte é chamada "interpolation" para realizar a interpolação dos dados das rodadas do SUPIM.

3.6.1 DAVS-LINUX timming adjust

Antes de aplicar a grade homogênea de concentração eletrônica a cada hora, é verificado o resultado do código SUPIM, que por muitas vezes gera pontos de simulação em coordenadas magnéticas que, ao serem passadas para coordenadas geográficas, acarretavam em um deslocamento longitudinal, com relação ao local simulado.

Com isso era possível perceber que em algumas regiões os atrasos e adiantamentos ultrapassavam 30 (trinta) minutos. Nesses casos, os pontos de simulação com atraso ou adiantamento são transferidos para a hora anterior ou seguinte.

3.6.2 DAVS-LINUX interpolation

Várias rodadas do SUPIM executam em diferentes pontos, que não são pontos geograficamente alinhados. Com isso é necessário fazer a interpolação destes dados, juntando cada um desses pontos, sendo possível assim ter um mapa final completo.

Devido a complexidade para realizar a interpolação de dados foi desenvolvido uma técnica chamada "explosão de perfil", na qual pontos simulados são associados aos pontos da grade próximos a ele (PETRY *et al.*, 2012).

3.7 Geração de imagens

Esta parte do código é onde são gerados mapas com o resultado da simulação, através do GrADS (Grid Analysis and Display System) (GRADS, 2014), para que posteriormente seja disponibilizada no site do INPE.

O resultado do DAVS resulta em arquivos em formato GRIB (Gridded Binary or General Regularly), que é possível realizar a leitura pelo GrADS Para a execução do GrADS foi criado um script para criar mapas para a América do Sul ou mapas Globais de acordo com parâmetros já definidos. Com isso também foi criado um

arquivo script para construção do mapa de cores nas figuras resultantes (PETRY et al., 2014).

As Figuras 3.3 e 3.4 ilustram exemplos de formatos adotados atualmente para a América do Sul e para o Globo, respectivamente.

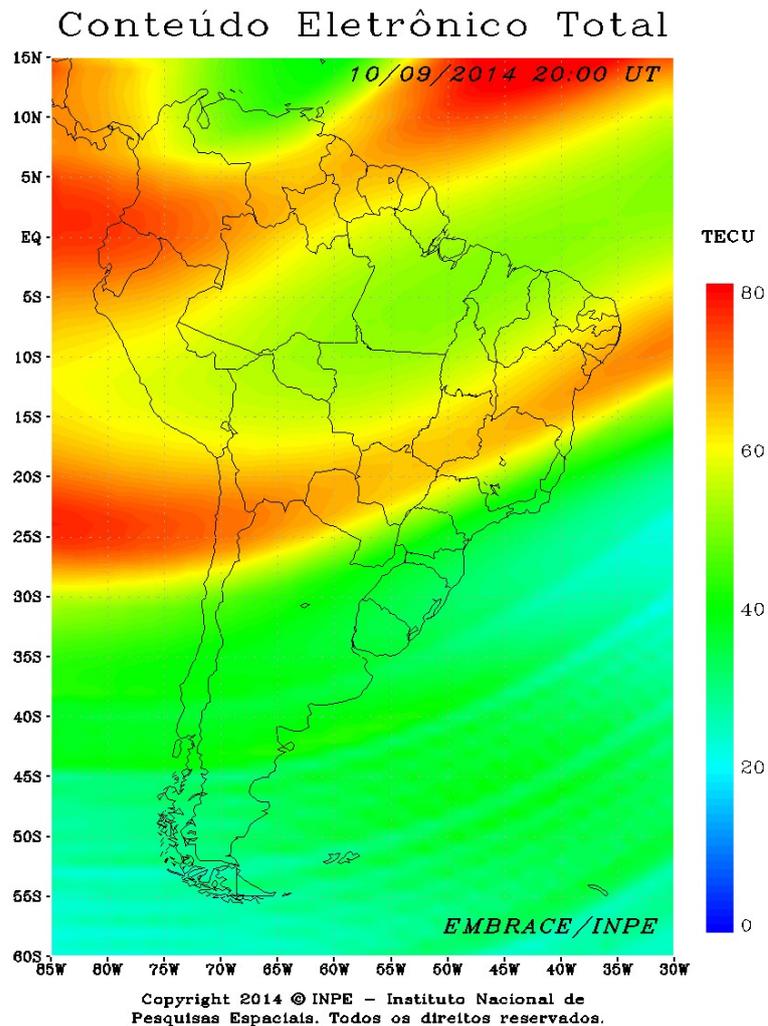


Figura 3.3 - Exemplo de imagem para simulação TEC para América do Sul

3.8 Finalização da Simulação

Para finalizar o código são realizadas algumas operações, como cópia das imagens resultantes para um servidor externo e backup dos arquivos resultantes do SUPIM. Por fim é enviado um e-mail para as contas pré-definidas com a cópia do log da simulação.

Conteúdo Eletrônico Total

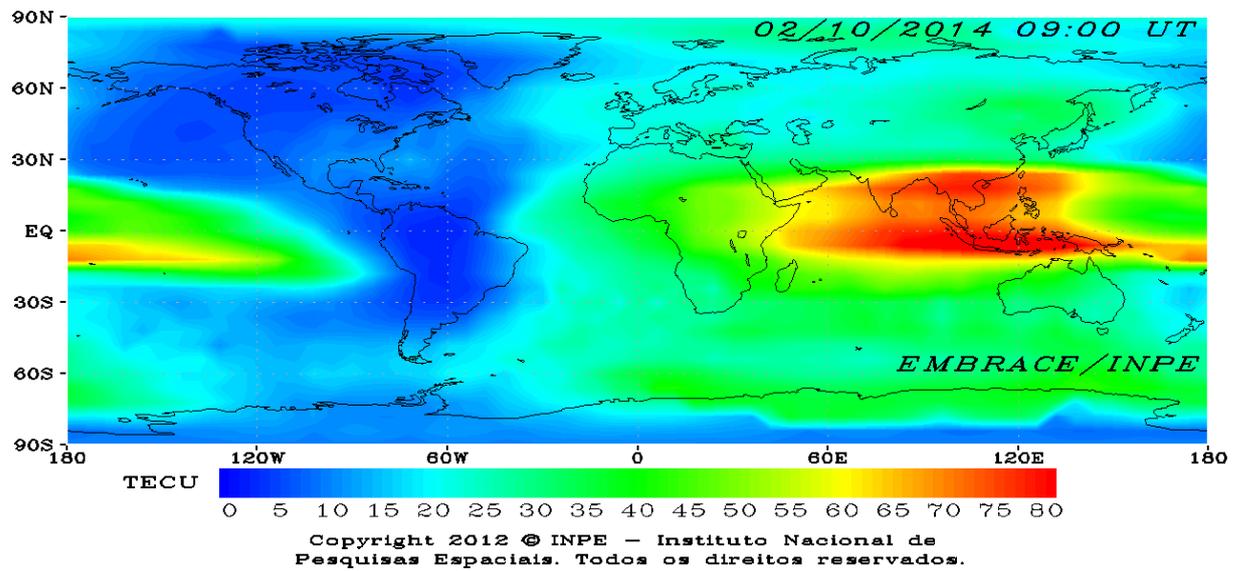


Figura 3.4 - Exemplo de imagem para simulação TEC para o Globo

4 CONCLUSÕES

Este manual documentou a estrutura e funcionamento de um sistema usado para simular e prever o comportamento da ionosfera terrestre, chamado como SUPIM-DAVS. Esta atividade é desenvolvida dentro do Programa de Clima Espacial do INPE. Inicialmente foi apresentada a estrutura com seus diretórios, bem como a funcionalidade de cada um. Após isso todo o funcionamento do código, desde o arquivo de configuração até a geração de imagens e finalização da simulação.

Espera-se que este manual sirva como guia para a equipe responsável pela manutenção e operação diária nas previsões ionosféricas da América do Sul e do Globo. O código pode ser alterado, bem como alguns diretórios adicionados, porém acreditamos que a estrutura principal do Sistema deve ser mantida.

REFERÊNCIAS BIBLIOGRÁFICAS

- EXELIS. **Scientific Data Visualization Software form ITTVIS**. EUA, Acessado em 18 de agosto de 2014. 2000. Disponível em: <<http://www.spacewx.com/solar2000.html>>. 15
- GRADS. **Grid Analysis and Display System (GrADS)**. EUA, Acessado em 09 de setembro de 2014. 2000. Disponível em: <<http://iges.org/grads/>>. 17
- GROPP, W.; LUSK, E.; ASHTON, D.; BALAJI, P.; BUNTINAS, D.; BUTLER, R. **MPICH2 - User Guide**. 1.0.6. ed. Argonne National Laboratory, 9 2007. 11
- NICOLAS, E. J. C. **OAR Documentation - User Guide**. 2.3.0. ed. A, 08 2014. Laboratoire d'Informatique de Grenoble Bat. ENSIMAG. 11
- PETRY, A.; PEREIRA, A. G.; VIERO, F.; SOUZA, J. R. **Image generation and visualization system for ionosphere dynamics**. São José dos Campos: Instituto Nacional de Pesquisas Espaciais, 2012. Disponível em: <<http://urlib.net/sid.inpe.br/mtc-m19/2011/05.30.13.07>>. Acesso em: 09 set. 2014. 17
- PETRY, A.; SOUZA, J. R. d.; VELHO, H. F. d. C. **Sistema para previsão operacional da dinâmica da ionosfera baseado no modelo SUPIM**. São José dos Campos: Instituto Nacional de Pesquisas Espaciais, 2011. 40 p. Disponível em: <<http://urlib.net/sid.inpe.br/mtc-m19/2011/06.03.18.08>>. Acesso em: 06 set. 2014. 16
- PETRY, A.; SOUZA, J. R. de; VELHO, H. F. de C.; PEREIRA, A. G.; BAILEY, G. J. First results of operational ionospheric dynamics prediction for the brazilian space weather program. **Advances in Space Research**, Elsevier, v. 54, n. 1, p. 22–36, 2014. 18
- SANTOS, P.; SOUZA, J.; SOBRAL, J.; BAILEY, G.; BRUM, C. Development of the sheffield university plasmasphere ionosphere model-enhanced (supim-e). 2005. 16
- TECHOLOGIES, S. E. **SOLAR IRRADIANCE PLATFORM**. EUA, Acessado em 18 de agosto de 2014. 2000. Disponível em: <<http://www.spacewx.com/solar2000.html>>. 15

