



MINISTÉRIO DA CIÊNCIA, TECNOLOGIA, INOVAÇÕES E COMUNICAÇÕES
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

RELATÓRIO FINAL: AMBIENTE DE CONTÊINER PARA PESQUISA COM MODELOS COMPUTACIONAIS

Bruno da Silva Alves

Relatório de Iniciação Científica
do Programa PIBIC, orientada por
Dra. Andrea Schwertner Charão e
Dr. Haroldo Fraga de Campos Ve-
lho

URL do documento original:

<<http://urlib.net/>>

INPE
São José dos Campos

2021

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) 3945-6923/6921

Fax: (012) 3945-6919

E-mail: pubtc@sid.inpe.br

**COMISSÃO DO CONSELHO DE EDITORAÇÃO E PRESERVAÇÃO
DA PRODUÇÃO INTELECTUAL DO INPE (DE/DIR-544):****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)

Clayton Martins Pereira - Serviço de Informação e Documentação (SID)

REVISÃO E NORMALIZAÇÃO DOCUMENTÁRIA:

Simone Angélica Del Duca Barbedo - Serviço de Informação e Documentação
(SID)

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

EDITORAÇÃO ELETRÔNICA:

Marcelo de Castro Pazos - Serviço de Informação e Documentação (SID)

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



MINISTÉRIO DA CIÊNCIA, TECNOLOGIA, INOVAÇÕES E COMUNICAÇÕES
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

RELATÓRIO FINAL: AMBIENTE DE CONTÊINER PARA PESQUISA COM MODELOS COMPUTACIONAIS

Bruno da Silva Alves

Relatório de Iniciação Científica
do Programa PIBIC, orientada por
Dra. Andrea Schwertner Charão e
Dr. Haroldo Fraga de Campos Ve-
lho

URL do documento original:

[<http://urlib.net/>](http://urlib.net/)

INPE
São José dos Campos
2021



Esta obra foi licenciada sob uma Licença GNU General Public License - Versão 3.

This work is licensed under a GNU General Public License - Version 3.

LISTA DE FIGURAS

	<u>Pág.</u>
2.1 Fluxograma do sistema de previsão ionosférica.	5

LISTA DE TABELAS

	<u>Pág.</u>
2.1 Disposição dos serviços Docker Swarm.	9

SUMÁRIO

	<u>Pág.</u>
1 INTRODUÇÃO	2
2 DESENVOLVIMENTO	4
2.1 Execução Operacional do SUPIM-DAVS	4
2.2 Configuração do Cluster de Desenvolvimento	6
2.3 Criação da Imagem Docker	6
2.4 Criação dos Serviços Docker	7
2.5 Compartilhamento dos Arquivos de Saída	8
2.6 SUPIM-DAVS <i>on Docker</i> : Executando a Simulação com Contêineres	9
3 CONCLUSÃO	10
REFERÊNCIAS BIBLIOGRÁFICAS	11

1 INTRODUÇÃO

Modelos de previsão são códigos computacionais sofisticados em permanente evolução e desenvolvimento, seja pela incorporação de novos processos físicos (parametrizações, modelos de superfícies) e de novas técnicas computacionais para integração espacial e/ou temporal, seja para adequação a novas arquiteturas de computadores.

Uma tarefa importante da aplicação dos modelos é a aplicação para diferentes execuções de testes de pesquisa. A literatura tem indicado que plataforma de containers é um recurso útil para a pesquisa científica.

Boettiger (2015) lista desafios para reprodutibilidade de pesquisas, como a questão das dependências e a documentação imprecisa. O autor descreve como recursos disponíveis pelo Docker e por containers podem ser a solução para esses e outros problemas citados.

Ambientes de virtualização em containers também têm sido aplicados para modelos de dinâmica da atmosfera como o WRF – Weather Research and Forecasting Model – desenvolvido por pesquisadores do NCAR (National Center for Atmospheric Research, CO-USA) (Hacker et al., 2017). O modelo WRF foi instalado utilizando o Docker volume para obter os arquivos das etapas de pré-processamento, execução e pós-processamento do modelo.

O container é um framework baseado em ambiente virtualizado que apresenta características favoráveis para pesquisa, especialmente quando essa faz uso de Computação de Alto Desempenho (CAD). Os aspectos oferecidos pela virtualização e a leveza do container são úteis durante a produção científica e a disseminação de seus resultados.

Este projeto tem como objetivo final portar o modelo atmosférico BRAMS de previsão ambiental e de tempo para um sistema de container. Avanços iniciais em direção a este objetivo mostraram que, antes de focar em um modelo em particular, é necessário preparar um ambiente virtualizado capaz de executar tarefas paralelas e distribuídas, além de gerenciar arquivos de entrada e saída neste ambiente. Assim, decidiu-se utilizar como exemplo inicial um sistema de menor porte, o modelo de previsão ionosférica SUPIM-DAVS.

O monitoramento e previsão da ionosfera é uma atividade da área conhecida como clima espacial que possui um papel importante para humanidade, uma vez que atividades solares podem impactar o ambiente terrestre. Sistemas tecnológicos como

comunicação de alta frequência e o Sistema de Posicionamento Global (GPS) podem sofrer alterações com as mudanças constantes que ocorrem na ionosfera. Com isso, diversos modelos para previsão ionosférica surgiram . O modelo SUPIM estima valores para os fluxos alinhados de campo, concentração e temperatura dos elétrons e dos íons de O_+ , H_+ , N_+ , He_+ , N_+^2 , O_+^2 NO_+ presentes na Ionosfera e Plamosfera.

O sistema de previsão ionosférica adotado pelo Instituto Nacional de Pesquisa (INPE) utiliza o modelo SUPIM juntamente com a busca e assimilação de dados (pré-processamento) e a geração de mapas 2D (pós-processamento). A execução operacional da previsão é feita em um cluster e controlada por meio de um script Shell. Um proceso importante para que a execução do sistema de previsão seja possível é a configuração e controle dos nós computacionais onde o modelo é executado. Esse processo representa uma fase custosa que deve ser feita toda vez que novos nós forem adicionados ao cluster ou quando alguma falha crítica ocorre. Dessa forma, o custo necessário para a configuração do sistema impõe uma barreira para a escalabilidade, reprodutibilidade e flexibilidade do mesmo.

Uma das maneiras para prover um ambiente configurado e pronto para receber as aplicações é a utilização da virtualização, que pode ser feita através do uso de máquinas virtuais ou, recentemente, através do uso de contêineres. Sabe-se que os contêineres são capazes de virtualizar um sistema operacional de forma leve (Kovacs, 2017), (Preeth E N et al., 2015), assim os mesmos se apresentam como uma solução interessante para diminuir os custos de tempo e esforços necessários durante a etapa de configuração das máquinas.

2 DESENVOLVIMENTO

Nesta seção são apresentadas todas as etapas realizadas no processo de desenvolvimento deste trabalho. Na seção 2.1 é apresentada a execução do sistema de previsão ionosférica através da simulação SUPIM, a seção 2.2 detalha o processo de configuração das máquinas do *cluster* de desenvolvimento, a seção 2.3 descreve as imagens Docker criadas, a seção 2.4 descreve os serviços Docker criados, a seção 2.5 apresenta a solução utilizada para o compartilhamento de arquivos na rede e por fim a seção 2.6 apresenta o cenário de testes utilizado.

2.1 Execução Operacional do SUPIM-DAVS

O sistema de previsão ionosférica em execução no Centro Regional Sul do INPE está em funcionamento desde de março de 2011 e os mapas resultantes das simulações são disponibilizados no seguinte *site*: <http://www2.inpe.br/climaespacial/pt/tecsupim>. Para a execução e desenvolvimento do sistema, o centro regional conta com dois *clusters*, um *cluster* operacional e outro de desenvolvimento. O *cluster* operacional é destinado exclusivamente a execução das simulações e geração dos mapas que são enviados ao *site* anteriormente mencionado. Já o *cluster* de desenvolvimento é destinado ao desenvolvimento de melhorias e ajustes que podem, futuramente, ser adicionados ao *cluster* operacional.

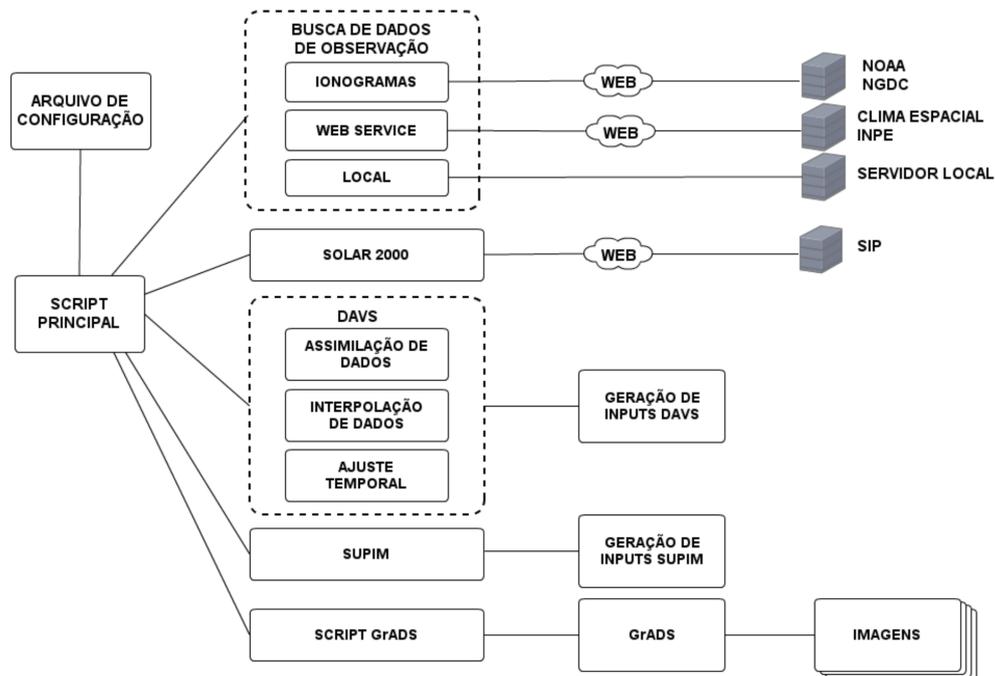
O sistema de previsão executado pode ser dividido em oito partes, conforme mostrado no fluxograma da 2.1.

1. **Leitura do arquivo de configuração.** Neste arquivo de configuração são ajustados os parâmetros que cada simulação deve utilizar. Alguns dos parâmetros que podem ser definidos são: longitude inicial e final da simulação, granularidade da simulação, datas, definição dos dados de entradas, etc.
2. **Inicialização do sistema.** Nesta etapa os parâmetros do arquivo de configuração são verificados e além disso é feita uma verificação do estado dos nós de processamento e do espaço em disco disponível.
3. **Assimilação de dados (DAVS).** É feito o *download* dos dados observacionais, caso tenha sido explícito no arquivo de configuração.
4. **Obtenção do fluxo solar.** Os dados de fluxo solar são obtidos através de um programa escrito em C. Para tal, o software SIP (Solar Irradiance Plat-

form), versão "System Grade"(SIPSystemGrade) (SPACE ENVIRONMENT TECHNOLOGIES, 2014) é utilizado.

5. **Execução do código SUPIM.** A simulação é lançada, podendo utilizar o Slurm, OAR e MPI. Ao final da simulação, os dados processados pelos nós de processamento são retornados através de uma pasta compartilhada pela rede.
6. **Execução do código DAVS.** Nesta etapa, o DAVS realiza ajustes de tempo e interpolação dos resultados da simulação anterior.
7. **Geração de imagens.** Os mapas com o resultado da simulação são gerados através do GrADS (Grid Analysis and Display System) (GRADS, 2000) e disponibilizado no site do INPE.
8. **Conclusão.** Por fim, é enviado um e-mail de conclusão da simulação aos administradores do sistema e é feito um backup local dos resultados.

Figura 2.1 - Fluxograma do sistema de previsão ionosférica.



Fonte: Extraído e traduzido de (PUNTEL et al., 2015).

2.2 Configuração do Cluster de Desenvolvimento

Considerando o caráter experimental e exploratório, o presente trabalho teve acesso somente ao *cluster* de desenvolvimento. O *cluster* utilizado possui 5 nós de processamento e um nó de controle, onde cada nó de processamento possui a seguinte configuração: 8 CPUs Intel(R) Xeon(R) CPU E5-2609 0 @ 2.40GHz, 74 GB de memória RAM, Sistema Operacional CentOS 6.9. O *cluster* de desenvolvimento replica o funcionamento do *cluster* operacional, onde 1 nó de controle é responsável por submeter as tarefas aos outros nós de processamento, assim as mesmas simulações podem ser executadas nos dois ambientes.

Um dos problemas na utilização dessas máquinas foi que as mesmas possuíam o sistema operacional CentOS 6.9 que não dava suporte a instalação da ferramenta Docker (DOCKER, 2021a), responsável pela virtualização através dos contêineres. Dessa forma, o sistema operacional teve que ser atualizado para o CentOS 7.0, que possui suporte ao Docker. No entanto, o processo de atualização do sistema operacional teve que ser feito de forma incremental, uma vez que os responsáveis pelo sistema de previsão necessitam do *cluster* de desenvolvimento para eventuais testes e cópias de segurança. Além disso, o nó de controle não pôde ser atualizado e por fim destinado aos experimentos, pois o mesmo atuava como *gateway* para os demais nós de processamento, além de executar outras tarefas importantes para o sistema como o gerenciamento de chaves. Assim, somente os nós de processamento foram atualizados e configurados.

O processo de atualização do sistema operacional iniciou-se com apenas um nó, e o mesmo foi completamente formatado, sendo necessário acesso físico ao local. Essa etapa ocorreu de forma lenta devido ao potencial risco ao sistema e também porque o acesso físico foi dificultado devido aos casos de covid-19 na região. Após a formatação e instalação do novo sistema, todas as ferramentas foram reinstaladas e configuradas e novas simulações foram executadas no intuito de garantir que o sistema estava funcional. Assim, o processo foi repetido para mais dois nós do cluster.

2.3 Criação da Imagem Docker

As imagens de contêineres são arquivos onde todas as dependências de uma aplicação são colocadas, dependências como códigos-fontes, pacotes, bibliotecas, etc. As imagens de contêineres podem ser vistas como um ambiente pronto para rodar uma aplicação. Essas imagens são organizadas em um sistema de camadas onde cada imagem pode ser escrita com base em outra imagem previamente configurada. Dessa

forma, as imagens de contêineres permitem modificações e alterações ao longo do tempo, facilitando a manutenção do sistema.

Tanto o *cluster* operacional quanto o de desenvolvimento utilizam a distribuição CentOS para executar o sistema de previsão ionosférica, assim, decidiu-se utilizar a mesma distribuição para a construção das imagens. Dessa forma, a manutenção das imagens é facilitada já que os administradores possuem experiência com o sistema operacional. O CentOS 7 foi utilizado como a primeira camada para a construção da imagem de contêiner e acima dessa camada foram configuradas as bibliotecas, pacotes e códigos fontes.

Nesta versão do sistema, o código SUPIM (Santos P.M.T. et al., 2005) pode executar paralelamente através do MPI, OAR e Slurm. Devido ao conhecimento da equipe com o Slurm, o mesmo foi eleito como o sistema para escalonamento de tarefas no ambiente Docker. Assim, foram criadas três imagens Docker: *base*, *controller* e *worker*. Na imagem *base* são instaladas todas as ferramentas, pacotes e bibliotecas necessárias para a execução do sistema de previsão ionosférica. A imagem *controller* é construída a partir da imagem *base* e nela são configurados arquivos e diretórios do Munge e do Slurm para possibilitar o funcionamento do Slurm em todos os contêineres. Além disso, o banco de dados é configurado para executar no contêiner que utiliza a imagem *controller*. A imagem *worker* também é derivada da imagem *base* e nela é configurado um *script shell* que conecta os contêineres trabalhadores com o contêiner controlador do Slurm.

Todas as imagens foram descritas através de arquivos Dockerfile, que são arquivos de texto que contém todos os comandos necessários para a construção da imagem. Assim, através dos arquivos Dockerfile o processo para a construção da imagem de contêiner pode ser facilmente automatizado e adições de dependências podem ser feitas através da modificação destes arquivos.

2.4 Criação dos Serviços Docker

Em cenários em que várias máquinas trabalham em conjunto para completar uma tarefa é comum que algumas tarefas falhem uma vez que podem ocorrer problemas de comunicação, disco, entre outros. Dessa maneira, tanto as tarefas quanto os contêineres distribuídos pelos nós do *cluster* devem ser gerenciados. Entre as ferramentas disponíveis pelo Docker, o Docker Swarm (DOCKER, 2021b) permite conectar diversas máquinas em modo Swarm. Neste modo, cada máquina pode exercer 3 papéis: *manager*, *worker* e ambos. Os nós *manager* trabalham para manter o estado definido

para determinado serviço, gerenciando as tarefas. Assim, caso algum contêiner falhe em sua execução, o *manager* é capaz de reescalonar a tarefa para outro nó trabalhador conectado em modo swarm. Múltiplos nós *manager* podem ser definidos por motivos de redundância. Os nós *workers* recebem as tarefas enviadas pelo manager e as processam.

Os serviços podem ser definidos por meio de arquivos chamados de Docker-compose, e neles diversas opções podem ser configuradas, como: número de réplicas (quantos contêineres devem estar ativos executando o serviço), configurações de rede, recursos mínimos para executar o serviço, entre outros. Para a execução do sistema de previsão ionosférica foram definidos dois serviços: o serviço *worker* e o serviço *controller*. O serviço *controller* utiliza a imagem *controller* definida pelo Dockerfile e é responsável por lançar as tarefas Slurm geradas pela execução da simulação. O serviço *worker* utiliza a imagem *worker* e executa as tarefas submetidas pelo *controller*. Ainda, o serviço *worker* é configurado para executar uma réplica em cada nó trabalhador do sistema, e o serviço *controller* possui apenas uma réplica. Assim, o Docker Swarm fica responsável por orquestrar os contêineres executando a simulação e o Slurm atua no controle das tarefas.

2.5 Compartilhamento dos Arquivos de Saída

Na versão do sistema que executa no *cluster* operacional, todos os nós possuem uma pasta compartilhada pela rede através do NFS (Networking File System), onde todos os trabalhadores executam suas tarefas e salvam os arquivos de saída nesta pasta compartilhada. Dessa forma, ao final da execução da previsão, todos os arquivos de saída gerados por todos os trabalhadores podem ser acessados pelo nó que lançou a simulação.

Assim, o compartilhamento dos arquivos de saída precisa ser adaptado ao cenários dos contêineres. Neste cenário, existem algumas alternativas para o compartilhamento de arquivos através da rede, como GlusterFS, CephFS, HDFS, NFS, entre outros. A solução adotada foi compartilhar uma pasta do sistema de arquivos nativo através do GlusterFS e vincular tal pasta ao sistema de arquivos dos contêineres em execução. O GlusterFS foi escolhido por dois principais motivos: funcionamento similar ao NFS e pelo bom desempenho quando comparado à outras soluções (Juve et al., 2010), (DONVITO et al., 2014). Utilizar uma ferramenta similar ao NFS, permite que os administradores compreendam e deem manutenção ao sistema, ao mesmo tempo que facilita a identificação de cada versão da simulação. Dessa forma, as duas versões da simulação podem ser mantidas, uma vez que o NFS em execução

no *cluster* é responsável pela compartilhamento na versão bare-metal da simulação, enquanto que o GlusterFS está relacionado à versão que utiliza os contêineres.

2.6 SUPIM-DAVS *on Docker*: Executando a Simulação com Contêineres

Após toda a etapa de configuração do cluster, das imagens de contêineres e dos serviços para executar a simulação, foi feita uma rodada de testes. Para o cenário de testes foram utilizados dois nós de processamento do *cluster* de desenvolvimento. Os dois nós foram colocados em modo Swarm, sendo o nó 01 o manager e trabalhador e o nó 02 trabalhador, somente. Ainda, o serviço controller foi executado no nó 01 e o serviço worker foi configurado para executar com duas réplicas, uma em cada um dos nós. O limite de cores para cada serviço foi definido conforme apresenta a tabela 2.1.

Tabela 2.1 - Disposição dos serviços Docker Swarm.

Nó	Serviço Docker	Limite de Cores
01	Controller	1 core
01	Worker	7 cores
02	Worker	8 cores

3 CONCLUSÃO

A análise da ionosfera nos permite medir e prever as interações entre o sistema Sol e Terra. O presente trabalho fez esforços para containerizar um sistema de previsão ionosférica que está em execução no Centro Regional Sul do INPE. Para tal, várias etapas foram necessárias como um estudo inicial do sistema, configuração das máquinas do *cluster* de desenvolvimento, construção das imagens de contêineres, criação dos serviços Docker e configuração de um sistema de arquivos pela rede. Dessa forma, o sistema pôde ser portado para rodar em contêineres Docker e com isso adquiriu uma maior flexibilidade, uma vez que as imagens de contêineres podem ser enviadas para outras máquinas, facilitando a configuração inicial do ambiente. Além disso, o sistema passa ter maior escalabilidade e pode até ser executado em ambientes de computação em nuvem, já que o suporte ao Docker é comum nesses ambientes.

REFERÊNCIAS BIBLIOGRÁFICAS

DOCKER. **Docker**. 2021. Disponível em: <<https://www.docker.com/>>. Acesso em: Março de 2021. 6

_____. **Docker Swarm**. 2021. Disponível em: <<https://docs.docker.com/engine/swarm/>>. Acesso em: Março de 2021. 7

DONVITO, G.; MARZULLI, G.; DIACONO, D. Testing of several distributed file-systems (HDFS, ceph and GlusterFS) for supporting the HEP experiments analysis. **Journal of Physics: Conference Series**, IOP Publishing, v. 513, n. 4, p. 042014, jun 2014. Disponível em: <<https://doi.org/10.1088/1742-6596/513/4/042014>>. 8

GRADS. **Grid Analysis and Display System (GrADS)**. 2000. Disponível em: <<http://iges.org/grads/>>. Acesso em: 09 de setembro de 2014. 5

Juve, G.; Deelman, E.; Vahi, K.; Mehta, G.; Berriman, B.; Berman, B. P.; Maechling, P. Data sharing options for scientific workflows on amazon ec2. In: **SC '10: Proceedings of the 2010 ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis**. [S.l.: s.n.], 2010. p. 1–9. 8

Kovacs, A. Comparison of different linux containers. In: **2017 40th International Conference on Telecommunications and Signal Processing (TSP)**. [S.l.: s.n.], 2017. p. 47–51. 3

Preeth E N; Mulerickal, F. J. P.; Paul, B.; Sastri, Y. Evaluation of Docker containers based on hardware utilization. In: **2015 International Conference on Control Communication Computing India (ICCC)**. [S.l.: s.n.], 2015. p. 697–700. 3

PUNTEL, F. E.; PETRY, A.; VELHO, H. F. d. C.; SOUZA, J. R. d. Preliminary evaluation with ensemble prediction for the supim model. In: PAN-AMERICAN CONGRESS ON COMPUTATIONAL MECHANICS, 1. (PANACM), 2015, Buenos Aires. **Proceedings...** International Center for Numerical Methods in Engineering (CIMNE), 2015. v. 1, p. 1322–1330. ISBN 9788494392825. Disponível em: <<http://congress.cimne.com/PANACM2015/frontal/doc/EbookPANACM2015.pdf>>. Acesso em: 03 abr. 2021. 5

Santos P.M.T.; J.R. Souza; J.H.A. Sobral; G.J. Bailey; C.G.M. Brum.

Development of the sheffield university plasmasphere ionosphere model-enhanced (supim-e). In: CEDAR-GEM JOINT WORKSHOP, 2005, Santa Fe. [S.l.], 2005. 7

SPACE ENVIRONMENT TECHNOLOGIES. **S. E.SOLAR IRRADIANCE PLATFORM**. 2014. Disponível em:

<<http://www.spacewx.com/solar2000.html>>. Acesso em: 18 agosto de 2014. 5