

Relatório Final PIBIC

Orientador: Dr. Adriano Petry

Bolsista: Cássio Dalcin Steffanello

Projeto: Desenvolvimento de infra-estrutura de software para o sistema de previsão operacional da dinâmica da ionosfera

1. Atividades Desenvolvidas

O objetivo principal do projeto é a substituição do sistema que está em funcionamento por um sistema novo e completamente desenvolvido em linguagem Java, que visa o aumento do desempenho do mesmo hardware já utilizado.

Primeiramente foi iniciado a parte que envolve o entendimento do sistema que está em funcionamento no Centro Regional Sul de Pesquisas Espaciais (CRS/INPE), onde o laboratório de computação para clima espacial faz simulações do comportamento da dinâmica da ionosfera terrestre empregando um sistema de previsão operacional da dinâmica da ionosfera baseado no modelo *Sheffield University Plasmasphere-Ionosphere Model* (SUPIM), para este trabalho é necessário uma grande quantidade de processamento paralelo de tarefas e armazenamento de dados, onde ao final imagens são geradas contendo o conteúdo eletrônico total no continente da América do Sul, como vemos abaixo:

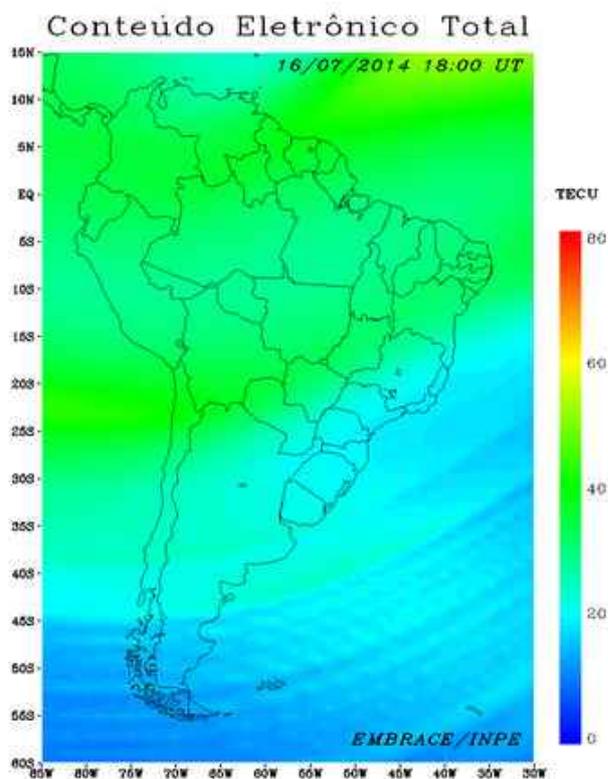


Figura 1: TEC SUPIM(previsão)

O sistema está instalado em um cluster de computadores, em que todos os cerca de 30 nós possuem o sistema operacional Linux CentOS 6.5, onde o escalonador de tarefas denominado OAR faz a distribuição de tarefas entre eles. Para isso, é utilizado o comando “crontab -e” que nos fornece o arquivo de tarefas agendadas, vistos na Figura 2, nestas tarefas estão inclusos os *scripts* que geram mapas, como na Figura 1, vejamos:

```
❏ Avaliação de simulação com assimilação OPERACIONAL v7.0
# Default 19:30h LT = 22:30h UT
#30 19 * * * /home/supim-davs/SUPIM-DAVS/runs/run_SUPIM-DAVS_v7.0b.sh /home/supim-davs/SUPIM-DAVS/runs/v7.tomorrow.conf > /home/supim-davs/SUPIM-DAVS/runs/logV7.assimi
# Default 00:01h LT = 3h UT (must start after 0h LT)
#01 00 * * * /home/supim-davs/SUPIM-DAVS/runs/run_SUPIM-DAVS_v7.0b.sh /home/supim-davs/SUPIM-DAVS/runs/v7.assimilation_3h.conf > /home/supim-davs/SUPIM-DAVS/runs/logV7
ion_3h
# Default 3h LT = 6h UT
#00 03 * * * /home/supim-davs/SUPIM-DAVS/runs/run_SUPIM-DAVS_v7.0b.sh /home/supim-davs/SUPIM-DAVS/runs/v7.assimilation_6h.conf > /home/supim-davs/SUPIM-DAVS/runs/logV7
ion_6h
# Default 6h LT = 9h UT
#00 06 * * * /home/supim-davs/SUPIM-DAVS/runs/run_SUPIM-DAVS_v7.0b.sh /home/supim-davs/SUPIM-DAVS/runs/v7.assimilation_9h.conf > /home/supim-davs/SUPIM-DAVS/runs/logV7
ion_9h
# Default 9h LT = 12h UT
#50 09 * * * /home/supim-davs/SUPIM-DAVS/runs/run_SUPIM-DAVS_v7.0b.sh /home/supim-davs/SUPIM-DAVS/runs/v7.assimilation_12h.conf > /home/supim-davs/SUPIM-DAVS/runs/logV
tion_12h
# Default 12h LT = 15h UT
#00 12 * * * /home/supim-davs/SUPIM-DAVS/runs/run_SUPIM-DAVS_v7.0b.sh /home/supim-davs/SUPIM-DAVS/runs/v7.assimilation_15h.conf > /home/supim-davs/SUPIM-DAVS/runs/logV
tion_15h
```

Figura 3: Tarefas agendadas com o uso do crontab

O *script* principal “run_SUPIM-DAVS_v7.sh” que se encontra na pasta “runs” está dividido em 7 etapas, são elas, a etapa de “Inicialização”, que realizada a leitura dos parâmetros de configuração, bem como a verificação dos nós de processamento e do espaço disponível em disco. Além disso, essa etapa também verifica simulações em andamento e executa outros procedimentos básicos ao funcionamento. A seguir, a etapa de “Assimilação de Dados”, responsável por realizar a assimilação de dados de observação para melhorar os resultados da simulação. Antes do código SUPIM executar paralelamente, a etapa de “Obtenção de dados de fluxo solar” garante a disponibilidade e validade das informações de fluxo e atividade solar para o dia simulado. Em seguida, a “execução paralela do código SUPIM” irá executar paralelamente várias rodadas de código, em longitudes distintas, de forma a fornecer a quantidade suficiente de pontos de simulação dentro de uma área de cobertura pré-definida. Já a “execução paralela do código DAVS” pode ser dividida em duas etapas menores: TimeAdjust e GradsFilesGeneration. Essa etapa faz uso das saídas da etapa anterior para obter valores de concentração eletrônica e/ou iônica dentro de uma grade homogeneamente espaçada, que varia em longitude, latitude, altitude e tempo (TimeAdjust). Os resultados obtidos são gravados em arquivos no formato apropriado para que a etapa de “geração de imagens” possa gerar mapas bidimensionais que contenham a informação apropriada a cada espaço de tempo (GradsFilesGeneration). Por fim, a etapa de “finalização da simulação” encarrega-se de enviar os mapas gerados ao destino, bem como realizar um backup dos arquivos da simulação três dias anterior e notificar a conclusão do processo.

Posteriormente, fez-se um estudo aprofundado nas características e propriedades da camada ionosférica com intenção de compreender área. Ganhando conhecimento desde sua localização e de sua composição, até conhecimentos mais específicos, como sua propagação eletromagnética, a variação da densidade iônica, e ainda a sua reflexão de ondas de rádio.

O entendimento do *Apache Hadoop* consistiu no próximo passo, nesta etapa foi estudada a forma como a ferramenta utiliza o hardware, para isso, utilizou-se o próprio site da *Apache Hadoop* e o livro “*Hadoop in action*” do autor “*Chuck Lam*”. Sendo

assim, passamos para o próximo passo, a instalação de teste. Nesta etapa foi feita uma instalação provisória do *Apache Hadoop single-node* em uma máquina local com sistema operacional semelhante ao do cluster a fim de avaliar possíveis erros a serem corrigidos futuramente. Para a configuração do mesmo, várias etapas foram criadas ao decorrer da instalação. O primeiro passo foi a instalação do Java versão 1.7.0_60, pois a plataforma *Hadoop* roda em cima de uma máquina virtual Java. Ao final do processo de instalação do Java, foi dado um “set” nas variáveis locais com o seguintes comandos:

```
# export JAVA_HOME=/opt/jdk1.7.0_60
# export JRE_HOME=/opt/jdk1.7.0_60/jre
# export PATH=$PATH:/opt/jdk1.7.0_60/bin:/opt/jdk1.7.0_60/jre/bin
```

O próximo passo foi o de configuração e criação de um usuário chamado “hadoop”. Para isso utilizou-se comandos Linux de criação de usuários como o “adduser” e o de criação de senha para o mesmo, como o comando “passwd”. Logo após, foi criada uma pasta com o caminho “opt/hadoop”. Então fez-se o download do *Hadoop-1.2.1* onde foi extraído para a pasta criada.

A etapa seguinte consistiu em configurar o *Hadoop*, para trabalhar de forma *single node*. Primeiro foi editado o arquivo “*core-site.xml*” adicionando as seguintes linhas:

```
<property>
  <name>fs.default.name</name>
  <value>hdfs://localhost:9000</value>
</property>
<property>
  <name>dfs.permissions</name>
  <value>>false</value>
</property>
```

Logo após, foi feita a edição do “*hdfs-site.xml*”:

```
<property>
  <name>dfs.data.dir</name>
  <value>/opt/hadoop/hadoop/dfs/name/data</value>
  <final>>true</final>
</property>
<property>
  <name>dfs.name.dir</name>
  <value>/opt/hadoop/hadoop/dfs/name</value>
  <final>>true</final>
</property>
<property>
  <name>dfs.replication</name>
  <value>2</value>
</property>
```

E ainda, a edição do “*mapred-site.xml*”:

```
<property>
  <name>mapred.job.tracker</name>
```

```
<value>localhost:9001</value>
</property>
```

Por fim, a edição do “hadoop-env.sh”, que é o script principal da plataforma:

```
#export JAVA_HOME=/opt/jdk1.7.0_55
#export HADOOP_OPTS=-Djava.net.preferIPv4Stack=true
```

Ao final desta etapa, foi formatado o sistema de arquivos simulado, com o seguinte comando “*hadoop namenode -format*”.

Esta instalação proporcionou rodar exemplos como um contador de palavras (*WordCount*) e acompanhar todo o processo de uma interface *Web*. Além de otimizar o processamento, a ferramenta possui um modelo de programação simples, o que facilitará o processo de migração do sistema antigo para o novo. Vejamos:

De início carregamos alguns arquivos de texto qualquer para o *hdfs*, utilizando os comandos:

```
#hadoop/bin/start-all.sh
#bin/hadoop dfs -copyFromLocal /tmp/cassio /user/hadoop/cassio
```

Após, podemos visualizar os arquivos carregados para o *hdfs*:

```
#bin/hadoop dfs -ls /user/hadoop/cassio
-rw-r--r-- 3 hadoop supergroup 674566 2014-05-10 11:38 /user/hadoop/cassio/pg20417.txt
-rw-r--r-- 3 hadoop supergroup 1573112 2014-05-10 11:38 /user/hadoop/cassio/pg4300.txt
-rw-r--r-- 3 hadoop supergroup 1423801 2014-05-10 11:38 /user/hadoop/cassio/pg5000.txt
```

Logo, foi utilizado um dos exemplos que integram a plataforma, o *WordCount*, vejamos:

```
#bin/hadoop jar hadoop*examples*.jar wordcount /user/hadoop/cassio /user/hadoop/cassio-output
```

Tendo a saída gravada na nova pasta do usuário *hadoop* a *cassio-output*. Por fim, foi analisado a saída e se a plataforma realmente contou as palavras:

```
# head /tmp/cassio-output/cassio-output
"(Lo)cra" 1
"1490 1
"1498," 1
"35" 1
"40," 1
"A 2
"AS-IS". 1
"A_ 1
```

"Absoluti 1
"Alack! 1

Além de tudo, ainda pode-se acompanhar o processo por uma interface Web:

localhost Hadoop Map/Reduce Administration

State: RUNNING
Started: Sat May 08 17:32:20 CEST 2010
Version: 0.20.2, r11707
Compiled: Fri Feb 19 08:07:54 UTC 2010 by chrisho
Identifier: 201005081732

Cluster Summary (Heap Size is 15.19 MB/966.69 MB)

Maps	Reduces	Total Submissions	Nodes	Map Task Capacity	Reduce Task Capacity	Avg. Tasks/Node	Blacklisted Nodes
0	0	1	1	2	2	4.00	0

Scheduling Information

Queue Name	Scheduling Information
default	N/A

Filter (Jobid, Priority, User, Name)
Example: 'user:mrh 2007' will filter by 'user:mrh' only in the user field and '2007' in all fields

Running Jobs

none

Completed Jobs

Jobid	Priority	User	Name	Map % Complete	Map Total	Maps Completed	Reduce % Complete	Reduce Total	Reduces Completed	Job Scheduling Information
job_201005081732_0001	NORMAL	hadoop	word count	100.00%	3	3	100.00%	1	1	N/A

Failed Jobs

none

Local Logs

[Log directory](#), [Job Tracker History](#)
[Hadoop](#), 2010.

2.Resultados Alcançados

- Conhecimento avançado de linguagem *Shell Script*.
- Conhecimento aprofundado do sistema para previsão operacional do conteúdo eletrônico da ionosfera.
- Compreensão do funcionamento de um sistema *cluster*.
- Entendimento da plataforma *Apache Hadoop*.
- Criação dos primeiros protótipos e processamento dos primeiros testes.

Santa Maria, 15 de Julho de 2014.

Cássio Dalcin Steffanello
Bolsista

Dr. Adriano Petry
Orientador