



Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"

Anderson Murgi Biazotto

DESENVOLVIMENTO DE UM CLUSTER DE ALTO DESEMPENHO

**Assis/SP
2011.**

Av. Getúlio Vargas, 1200 – Vila Nova Santana – Assis – SP – 19807-634
Fone/Fax: (0XX18) 3302 1055 homepage: www.fema.edu.br

DESENVOLVIMENTO DE UM CLUSTER DE ALTO DESEMPENHO

Projeto de desenvolvimento apresentado ao curso de Bacharelado em Ciência da Computação do Instituto Municipal de Ensino Superior de Assis - IMESA e a Fundação Educacional do Município de Assis – FEMA, como requisito parcial a obtenção do Certificado de Conclusão.

Orientando: Anderson Murgi Biazotto

Orientador: Fabio Eder Cardoso

**Assis/SP
2011.**

Dedicatória

Dedico este trabalho primeiramente a Deus. Meus Pais, familiares, ao meu orientador, e aos amigos de graduação. Os quais me apoiaram constantemente para realização deste trabalho de conclusão de Graduação.

Resumo

Cluster de computadores na atualidade tem sido um tema em evidência diante dos profissionais do ramo. Sua estrutura de fácil acesso e relação custo benefício significativos tem cada vez mais feito as empresas procurarem este recurso.

Um cluster pode ter várias funcionalidades, mediante a necessidade do usuário, este assunto então acaba se tornando complexo para os recentes profissionais da área, com isso o principal objetivo deste trabalho de pesquisa é mostrar de forma simples e didática, os principais conceitos de cluster bem como o funcionamento de um e suas propriedades para que sirva de base para futuros trabalhos da área.

Palavers chive: Cluster, Desempenho, Debian

Abstract

Cluster computers today has been a theme in evidence before the industry professionals. Its structure easily accessible and significant cost benefit has increasingly made the companies look for this feature. A cluster can have several features needed by the user, then this issue turns out to be complex for the latest health professionals, thus the main objective of this research is to show in a simple and intuitive, the main concepts of cluster and as a function of their properties and to serve as a basis for future work in the area.

Keywords: Cluster, Performance, Debian

Sumario

1. INTRODUÇÃO.....	7
2. OBJETIVOS.....	8
3. JUSTIFICATIVA.....	8
4. MOTIVAÇÃO.....	8
5. PERSPECTIVAS DE CONTRIBUIÇÃO.....	8
6. METODOLOGIA DE PESQUISA.....	9
7. CUSTER.....	9
7.1 CLUSTER DE ALTO DESEMPENHO.....	10
7.1.1 CLUSTER BEOWULF.....	10
7.2. CLUSTER DE ALTA DISPONIBILIDADE.....	11
7.3. BALANCEAMENTO DE CARGA.....	13
8. FERRAMENTAS OPEN SOURCE UTILIZADAS.....	13
8.1. LINUX DEBIAN.....	14
8.2 MOSIX	14
9. FERRAMENTAS DE BENCHMARK PARA AVALIAÇÃO.....	15
9.1 WHESTONE.....	16
9.2 DRYSTONE.....	16
9.3 LIMPACK.....	17
9.4 SISOFIT SANDRA.....	18
10. TRABALHO PROPOSTO.....	19
11. APLICAÇÃO DO PROJETO.....	20
11.1 TESTES DO CLUSTER.....	21
12. CONCLUSÃO.....	29
13. BIBLIOGRAFIA.....	30

Lista de Figuras

Figura 1 - Estrutura de um cluster Beowulf.....	11
Figura 2 - Funcionamento do cluster de alta disponibilidade.....	12
Figura 3 - software Linpack em funcionamento.....	17
Figura 4 - Sisoft Sandra.....	18
Figura 5 - Teste no cluster 01.....	21
Figura 6 - Teste no cluster 02.....	22
Figura 7 - Teste no cluster 03.....	23
Figura 8 - Teste no cluster 04.....	24
Figura 9 - Monitoramento nó 01.....	25
Figura 10 - Teste no cluster 05.....	26
Figura 11 - Teste no cluster 06.....	27
Figura 12 - Monitoramento nó 02.....	28

1. INTRODUÇÃO

As médias e grandes empresas, bem como todo segmento que utiliza tecnologia busca a cada dia um melhor desempenho no uso de processamento dos dados. A cada dia a tecnologia evidencia que há desempenho computacional, em relação ao processamento, manipulação, armazenamento e transmissão de dados, devem seguir uma tendência de avanço tecnológico buscando a melhoria no que tange a informação de maneira automática, rápida e precisa.

Dentro deste contexto, pesquisadores desenvolvem técnicas e ferramentas que auxiliam o homem na busca pela informação de forma rápida. O propósito deste trabalho apresenta uma das várias ferramentas que aborda o conceito de poder computacional, cluster de computadores.

Cluster é o nome dado a um sistema formado por dois ou mais computadores que trabalham em conjunto para processar uma aplicação, de forma que esses computadores trabalhem como um só.

Esta área necessita de atenção especial, pois este conceito abrange grande espaço no mercado, devido ao baixo custo. Tendo em vista que mainframes (computadores de grande porte) e até mesmo servidores pesados são de custo extremamente elevado e não são acessíveis a maioria das pequenas empresas.

Este trabalho propõe apresentar como se monta e configura um cluster com o sistema operacional Linux, onde será apresentada, passo a passo, a montagem de modo que qualquer pessoa tenha acesso e possa com pouco conhecimento na área montar seu próprio cluster.

2. OBJETIVOS

O principal objetivo do trabalho é o desenvolvimento de um cluster de alto desempenho de forma a criar um modelo que possa ser utilizado em pequenas empresas e ajudar na propagação deste conceito no mercado de modo mais simples e prático possível.

3. JUSTIFICATIVA

O projeto está sendo desenvolvido com o propósito de suprir a necessidade de empresas de médio porte trazendo assim informações que não são em um todo acessíveis a população, este conceito se mostra promissor, daí a necessidade de desenvolver este projeto.

4. MOTIVAÇÃO

Nos dias atuais, com o mercado competitivo, as médias e pequenas empresas, apresentam dificuldades por vários motivos, dentre eles um agravante é a falta de recursos financeiros e isso leva os empresários a diminuir os custos com equipamentos. Uma vez que equipamentos de grande porte são muito caros, temos a alternativa da utilização de cluster que tem custo bem mais baixo e traz resultados muito satisfatórios.

5. PERSPECTIVAS DE CONTRIBUIÇÃO

Este trabalho contribuirá para área de desenvolvimento de redes e também a área de TI, de forma a apresentar soluções para a montagem de clusters.

Posteriormente, todas as fases serão publicadas para que todos tenham acesso a essa informação de forma a contribuir nas áreas citadas a cima.

Desta forma, o projeto, a implementação, a configuração e a manutenção do cluster vai aprimorar o conhecimento técnico e científico de seus idealizadores.

6. METODOLOGIA DE PESQUISA

Será utilizado para o desenvolvimento do trabalho um conjunto de livros e apostilas que servirá de base para o desenvolvimento. Entre eles o destacamos o livro “CONSTRUINDO SUPERCOMPUTADORES COM LINUX” do autor Marcos pitanga, de onde tiraremos os principais conceitos para fazermos o cluster.

Recursos necessários

Utilizaremos na montagem do cluster: dois microcomputadores, cabos de rede, um switch, sistema operacional Linux.

Utilizaremos também o laboratório de redes para os testes nas máquinas e montagem do respectivo cluster.

7. CLUSTER

De um modo geral denomina se cluster como dois ou mais computadores que trabalha juntos para desenvolver trabalho (executar uma aplicação). Estes computadores trabalham de forma que o usuário tenha a impressão de estar utilizando somente uma máquina.

Cada computador ou estação é chamado de nó, esse conjunto de nós independentes e interligados compõe um sistema único. O cluster é uma opção de arquitetura que combina baixo custo e alto desempenho, como iremos mostrar nos capítulos posteriores.

Existem alguns tipos diferentes de cluster, são eles, de Alto Desempenho, Alta disponibilidade e Balanceamento de carga.

7.1 CLUSTER DE ALTO DESEMPENHO

Esse cluster tem como principal característica utilizar vários computadores para trabalhar como uma única máquina com alto poder computacional, geralmente utilizado para executar uma única aplicação que necessita de muito poder de processamento, como aplicações de astronomia e previsão

O cluster Beowulf proposto a ser desenvolvido é um cluster de alta disponibilidade com alto poder de processamento, nossas pesquisas mais a diante em capítulos posteriores se voltarão a construção de um cluster Beowulf bem como testes no mesmo.

7.1.1 CLUSTER BEOWULF

Segundo Marcos Pitanga (p. 06). “Um dos mais notáveis avanços tecnológicos dos dias atuais, tem sido o crescimento do desempenho computacional dos PCs (Computadores Pessoais).”

Na realidade nos dias atuais os microcomputadores tem o mercado em constante expansão o que faz com que cada vez a produção desta tecnologia fique mais baixa e seu desempenho aumente gradativamente.

O cluster Beowulf foi projetado com a finalidade de criar uma estrutura que por meio de dois ou mais micro computadores pudesse criar um ambiente de alto desempenho com custo benefício viável.

A constante aproximação entre PCs e Workstations é atribuída a crescente evolução dos processadores, também pode ser citada como um ponto positivo nessa questão a diminuição do custo da tecnologia de equipamentos de rede e também a evolução dos sistemas operacionais abertos como o Linux. Tudo isso juntamente com o trabalho de pesquisa que possibilita criar novos métodos e mais eficientes de criar e configurar um cluster de alto desempenho

O Beowulf é caracterizado pelo uso de ferramentas livres de distribuição gratuitas como, por exemplo, o Linux e FreeBSD. Nestes sistemas serão instaladas outras ferramentas que tem a função de dar ênfase ao processamento paralelo. Temos como exemplo as API's MPI e PVM. Essas ferramentas auxiliam a modificar o Linux para torná-lo mais adaptado a processamento paralelo.

A imagem abaixo mostra de modo bem simples a estrutura de um cluster Beowulf.

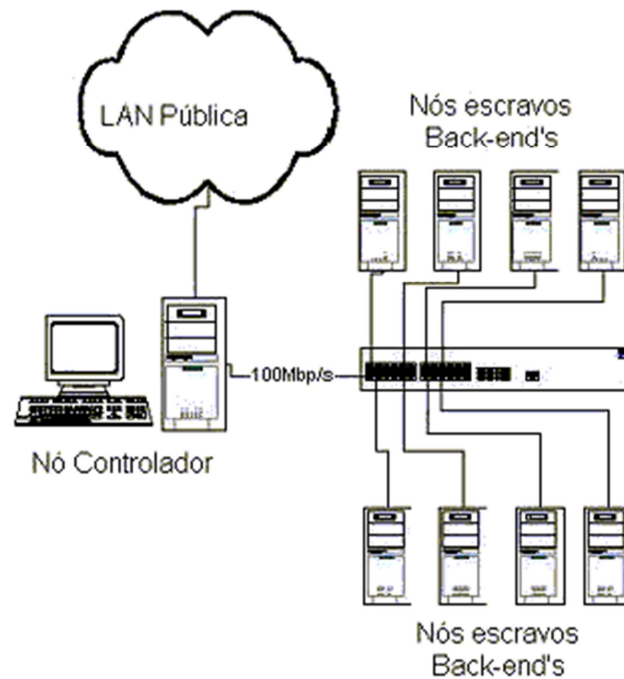


Figura 1 (Retirado de: <http://www.clubedohardware.com.br/imageview.php?image=816>) Ultimo acesso em 13/04/2011.

7.2 CLUSTER DE ALTA DISPONIBILIDADE

Este tipo de cluster é projetado para suprir uma disponibilidade de serviço ininterrupta, de forma que se um nó apresentar falha, o serviço atribuído a ele será disponibilizado em outro nó suprimindo assim a necessidade do serviço de forma imediata.

Esse tipo de cluster é muito utilizado em sistemas críticos que não suportam interrupções como base de dados de missões critica correios, servidores de arquivos e aplicações.

A imagem abaixo mostra o funcionamento do cluster de alta disponibilidade.

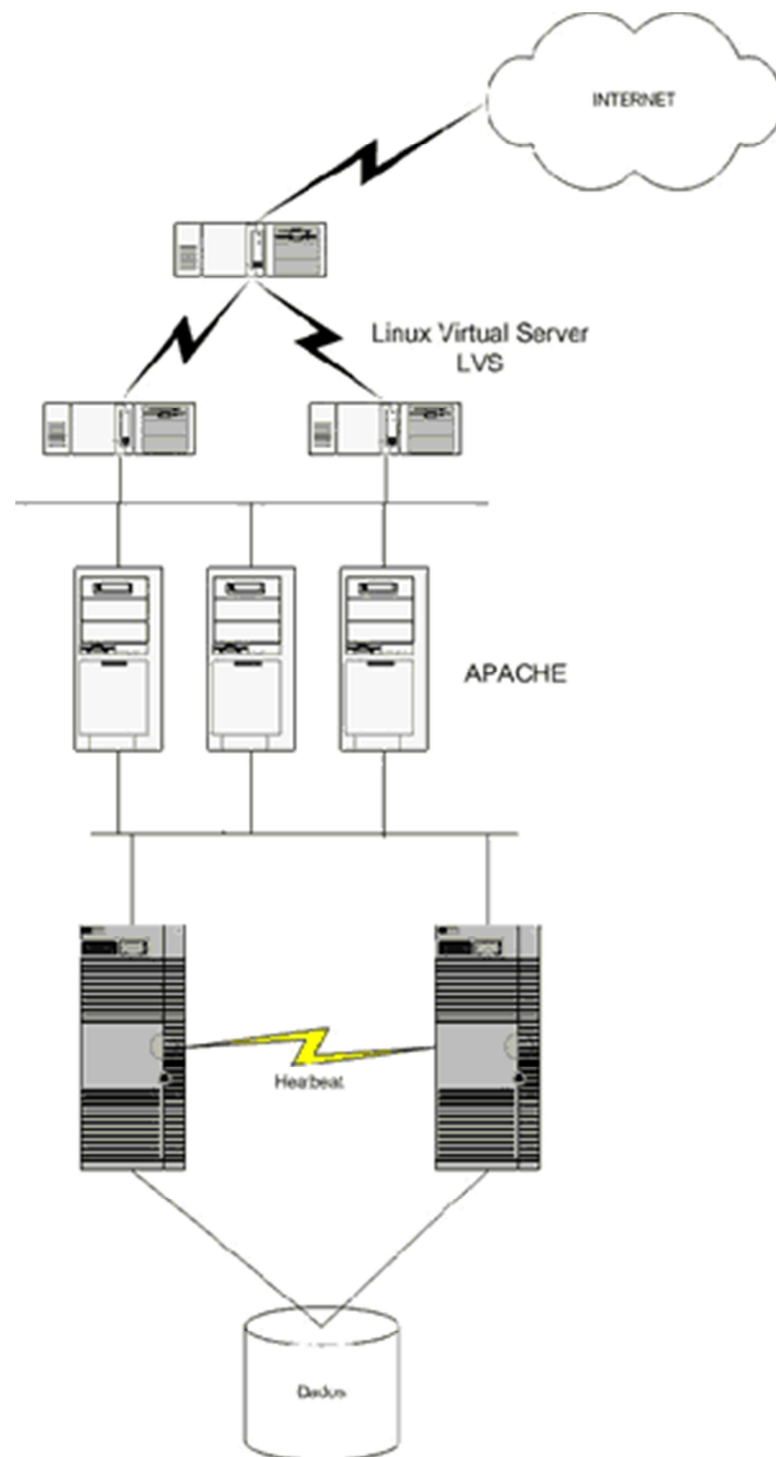


Figura 2 (Imagem retirada de: <http://www.clubedohardware.com.br/imageview.php?image=817>) Ultimo acesso em 15/04/2011.

7.3 BALANCEAMENTO DE CARGA

Esse tipo de cluster distribui o tráfego de processos entre os nós assim todos os nós podem executar os mesmos processos, assim quando um desses nós falha a carga de chamadas para este nó é redistribuída para os outros nós.

Esse cluster costuma ser muito utilizado em Web Farms (fazendas de servidores web).

8. FERRAMENTAS OPEN SOURCE UTILIZADAS

Algumas ferramentas podem ser utilizadas para configurar um cluster e facilitar o trabalho do desenvolvedor.

Dentre todas daremos destaque a algumas mais importantes. Primeiramente falaremos da distribuição de Linux Debian que utilizaremos para montar nosso cluster Beowulf, outro software que não utilizaremos mais também é referencia na pratica de criação de clusters, muitas destas ferramentas auxiliam na configuração e gerenciamento de clusters. Um exemplo de pacote para configuração é o pacote MOSIX o qual falaremos posteriormente.

8.1 LINUX DEBIAN

Debian foi um projeto iniciado em 1993 com o objetivo de criar um Sistema operacional livre que atendesse algumas especificações antes não observadas. O projeto inicial foi desenvolvido por Ian Murdock que tinha a idéia de desenvolver um sistema criado abertamente como o Linux na época. O projeto Debian teve apoio do projeto GNU.

Segundo Garbee, Bdale; Koptein, Hartmut; Lohner, Nils; Lowe, Will; Mitchell, Bill; Murdock, Ian; Schulze, Martin; Small, Craig. Tradução por Biscaia Filho, Israel Mendes. Uma Breve História do Debian. Com a intenção de criar cuidadosamente a distribuição e suporte, o Debian era desenvolvido livre de falhas. O projeto foi iniciado por um pequeno grupo de hackers que gradativamente foi tomando volume e se tornando grande e bem organizada rede de desenvolvedores de software livre.

Debian é considerada uma das versões do Linux mais significativas pelo alto critério e acabamento refinado, como software livre todos podem ter acesso ao código fonte e dar sua contribuição para melhorar esse excepcional sistema operacional.

O Debian possui um conjunto de políticas de distribuição e também vários procedimentos a serem adotados para o empacotamento, com Backups automáticos e documentação detalhada, tudo para garantir a qualidade das distribuições do Debian.

8.2 MOSIX

Consiste em uma ferramenta que por meio da Migração Preemptiva de processos e balanceamento dinâmico, é muito utilizado na configuração de clusters, é desenvolvido para sistemas baseados em Linux e Unix.

O serviço de transportar o serviço de um nó que está sendo utilizado para outro que está desocupado se denomina Migração preemptiva. Este recurso também pode ser denominado um recurso de balanceamento de carga. É um software que promove a migração de processos o tornando ideal para rodar aplicações que necessitam de grande poder de processamento graças ao processamento paralelo que por meio da biblioteca MOPI (biblioteca de entrada e saída) tem os meios necessários para dividir arquivos a serem processados em segmentos.

O Mosix apresenta um gerenciamento de processos onde este escolhe qual melhor processo a ser executado por determinado nó e atribui a este o processo dito. Assim o cluster trabalha de modo ao usuário não veja seu funcionamento em detalhes.

Este software tem o diferencial de poder ser implantado em máquinas que terão outras finalidades e não somente para cluster, por exemplo alguns nós poderão trabalhar como desktop. E o usuário nem perceberá que a máquina ou nó em questão está sendo utilizado em um cluster.

Segundo Rodrigo Mendes Pasini (2009). “Entre as características do Mosix destacam-se as principais:

- Usuários podem fazer login em qualquer nodo do cluster sem a necessidade de saberem da existência do Mosix;
- Não existe nodo controlador;
- Não é necessário modificar ou apontar os aplicativos a bibliotecas especiais;
- Pode-se incluir ou remover algum nodo a qualquer momento sem afetar o funcionamento do sistema;
- Não é necessário copiar arquivos para nodos remotos do cluster;
- Balanceamento de carga usando migração de processos;
- Migração de processos dos nodos escassos de recursos para os mais rápidos;
- Possui algoritmo de monitoramento que identificam, entre outras coisas, a velocidade de cada nó, a carga da CPU, e a memória livre disponível;
- Tem suporte para trabalhos em batch;
- Possui ferramentas de instalação e configuração automáticas.”

Rodrigo Mendes Pasini (2009).

9. FERRAMENTAS DE BENCHMARK PARA AVALIAÇÃO

Existem várias ferramentas cuja finalidade é de testar e avaliar o desempenho de clusters. Dentre tantas falaremos das principais e mais utilizadas, e alguns destes serão utilizadas em nosso cluster posteriormente montado e seus resultados devidamente documentados nesta monografia nos capítulos posteriores.

Deve-se destacar que muita dessas ferramentas tem funcionalidade bem mais ampla do que somente avaliar um cluster, são ferramentas de análise de desempenho, e dependendo do resultado que se espera do cluster algumas se aplicaram melhor a alguns tipos de cluster do que outras, tendo em vista que existem clusters diferentes que possuem focos também específicos.

As ferramentas as quais falaremos serão: Whetstone, Drystone, Linpack e Sisoft Sandra.

9.1 WHESTONE

O primeiro software do tipo o Whestone tem como finalidade a medição de desempenho. Sua primeira versão lançada em 1976 produzida na Inglaterra, tinha sido feita em ALGOL, posteriormente muito utilizado em FORTRAM.

É um programa muito simples com poucas linhas de código, e separado em vários módulos, o programa funciona executando estes módulos várias vezes através de loops, serve muito bem para máquinas de pequeno e médio porte.

O whestone mede principalmente a velocidade das aplicações mostrando o resultado em MIPS (milhões de instruções por segundo).

Sua versão oficial curiosamente é feita em Pascal.

9.2 DRYSTONE

Outro software de teste de processamento, desenvolvido em 1984 mede o desempenho do processador em inteiros, este software é utilizado atualmente como parte de outros benchmarks, ele simula execuções de leitura e chamadas de sistema, por fim indica o número de vezes por segundo que o processador é capaz de executar.

Hoje em dia com as novas tecnologias os computadores já possuem processadores que executam mais de um processo por chamada, então a medição deste software depende muito do clock do processador e da memória cache, o resultado varia de acordo com o aplicativo, portanto o resultado do Dhrystone serve somente como referencia de desempenho bruto.

9.3 LIMPACT

O Linpack é um dos mais famosos benchmarks, atualmente muito utilizado. Foi utilizado no teste das 500 máquinas mais rápidas. É um software que apresenta soluções para avaliação de máquinas com sistema linear denso. Utiliza duas rotinas são elas: DGEFA e DGESL que trabalham com 64 bits e AGFA e SGFL para 32 bits, o primeiro faz a decodificação parcial de um vetor, e a outra usa essa decodificação para resolver equações lineares.

A avaliação dele não condiz com a real capacidade da máquina mas traz de forma aproximada sua capacidade, como outras ferramentas do tipo, mas seus resultados se aproximam dos reais através de números obtidos de desempenho de pico do sistema.

A figura abaixo mostra o software Linpack em funcionamento.

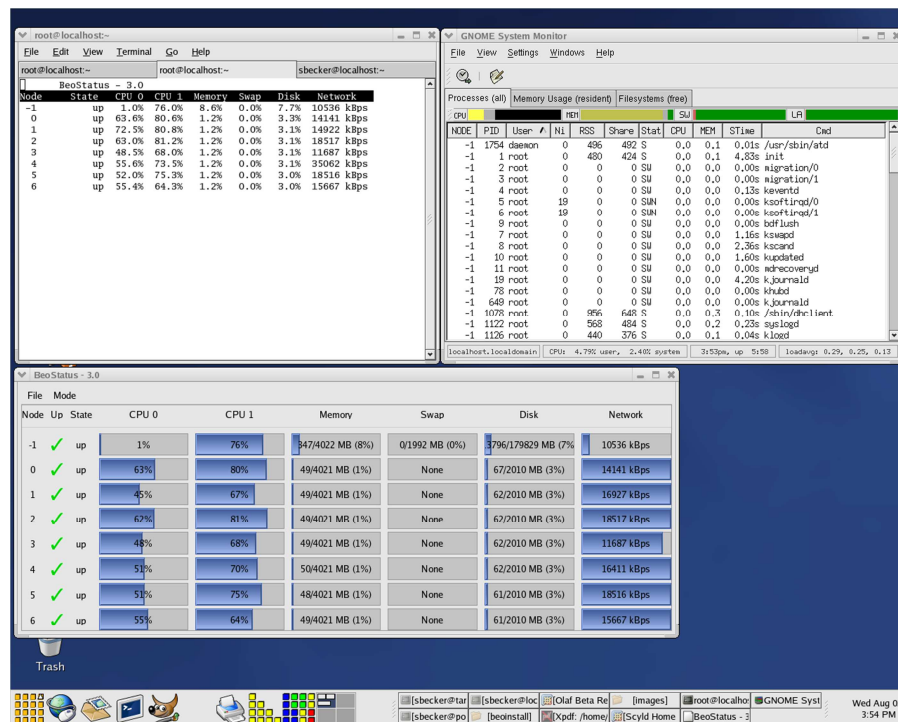


Figura 3 (Retirado de: <http://cougar.triumf.ca/scyld-doc/installation-guide/images/scyld-linpack.png>) Ultimo acesso em 22/09/2011.

9.4 SISOFT SANDRA

O sisoft Sandra é um software novo no mercado muito mencionado blogs por profissionais da área, apesar de haver poucas publicações a respeito desse software, o que se sabe é que o mesmo é formado por vários outros pequenos programas de Banchmark inclusive um deles é o Drystone falado a pouco. Muito utilizado e seus resultados são considerados satisfatórios.

Software em funcionamento mostrado na imagem abaixo.

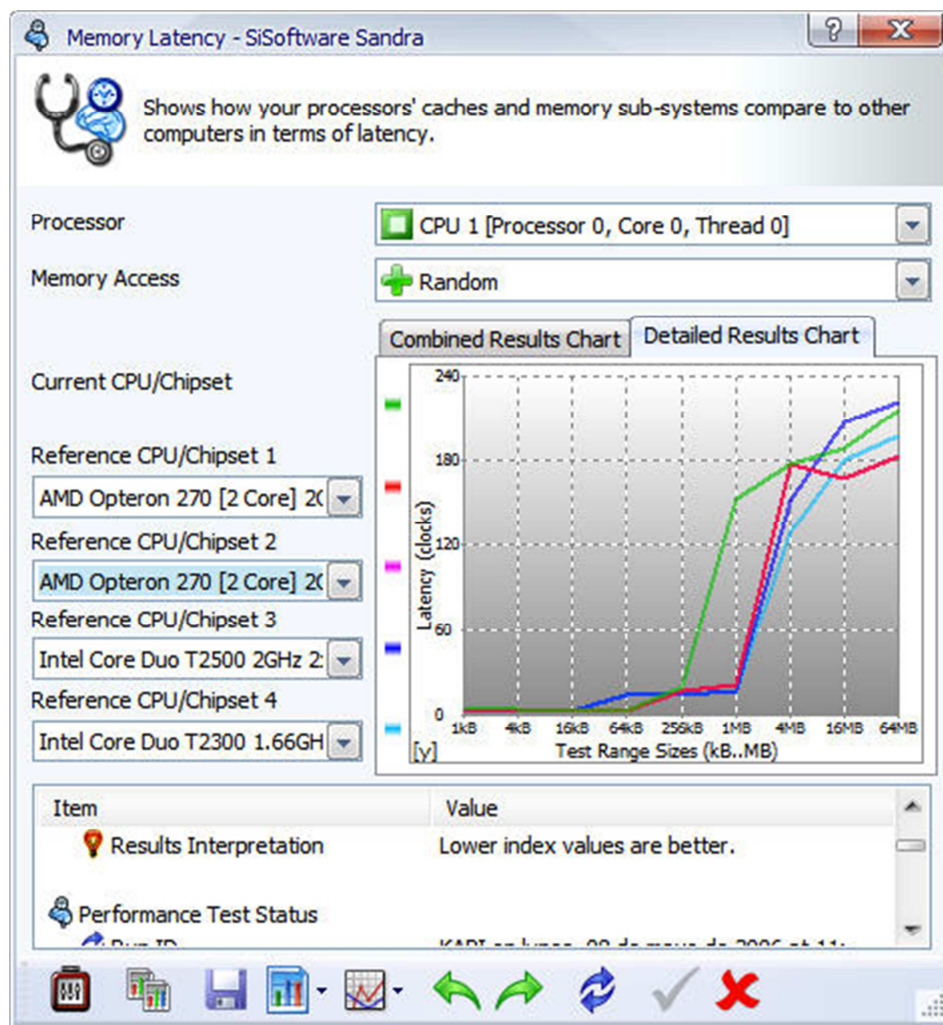


Figura 4 (Retirado de: <http://screenshots.en.softonic.com/en/scrn/51000/51882/sisoft-sandra-21.jpg>) Ultimo acesso em 23/09/2011.

10. TRABALHO PROPOSTO

O trabalho proposto nesta monografia consiste em construir um cluster Beowulf (alta performance), para isso utilizaremos de referencia principalmente os trabalhos descritos no livro de Marcos Pitanga, reconhecido como um dos profissionais da área de rede mais bem sucedidos na atualidade.

Para formulação do cluster será utilizado dois micro-computadores, um switch, conexão ethernet, cabos par trançado, softwares de configuração de rede, softwares de medição.

Configuração das máquinas

Máquina 01 (Nó Mestre)

Processador	PROCESSADOR Core 2 Duo ULV SU9400
Memória	2048 Mb
HD	500 Gb
Dispositivo de rede	Ethernet Integrado a placa mãe

Máquina 02 (Nó Escravo 01)

Processador	PROCESSADOR Core i3 M350 2,27Ghz
Memória	3,0 Gb
HD	320 Gb
Dispositivo de rede	Ethernet Integrado a placa mãe

Máquina 03 (Nó Escravo 02)

Processador	PROCESSADOR Dual Core E5400 3.0 Ghz
Memória	2048 Mb
HD	250 Gb
Dispositivo de rede	Ethernet Integrado a placa mãe

Primeiramente será utilizado o laboratório de redes da FEMA como local físico para formulação do cluster. Os computadores serão ligados por um switch 3com, será efetuada a formação dos computadores onde ocorrerá a instalação do sistema operacional Linux na versão Debian.

Esta versão do Linux foi escolhida principalmente por possuir ferramentas de configuração bem úteis para configuração do cluster Beowulf.

Depois de instalado o sistema operacional será efetuado várias configurações e comandos onde denominaremos uma máquina como principal e a outra como escrava, tendo em vista que o cluster pode ser feito com mais de duas máquinas, mas sempre somente uma será a principal e as outras secundárias (escravas).

Após isso será utilizado pacotes e ou programas os quais farão estas duas máquinas trabalharem de forma simultânea visando sempre o aumento significativo de poder computacional.

Por fim serão efetuados testes para calcular o desempenho do cluster, serão utilizadas varias ferramentas e posteriormente comparados os resultados, também será testado um computador com as mesmas configurações e depois de obtido os resultados, faremos uma comparação com os números obtidos das duas arquiteturas provando assim a eficiência do cluster e deixando claras as vantagens e desvantagens de formulá-lo.

11. APLICAÇÃO DO PROJETO

Primeiramente deixaremos claro que o cluster a ser montado tem como foco alto desempenho. Tendo em vista isso, nossos testes e configurações serão efetuados também com esse fim.

Como dito anteriormente utilizaremos uma distribuição do Linux Debian com o kernel Pelican, este kernel já é preparado para configuração do cluster, ou seja, já possui ferramentas instaladas próprias para criação e configuração do cluster.

Primeiramente é feita a instalação do Debian na máquina principal, também denominada nó mestre, nele será instalado um programa de gerenciamento, neste caso usaremos os próprios recursos desta distribuição do Debian, e depois de feita a instalação do nó mestre, iniciaremos a ligação dos nós escravos.

Neste protótipo montado utilizei duas máquinas com configurações diferentes e arquiteturas diferentes, iniciaremos estas máquinas dando o boot pela placa de rede, claro após todas as máquinas (inclusive o nó mestre) estarem devidamente ligadas em um hub ou switch.

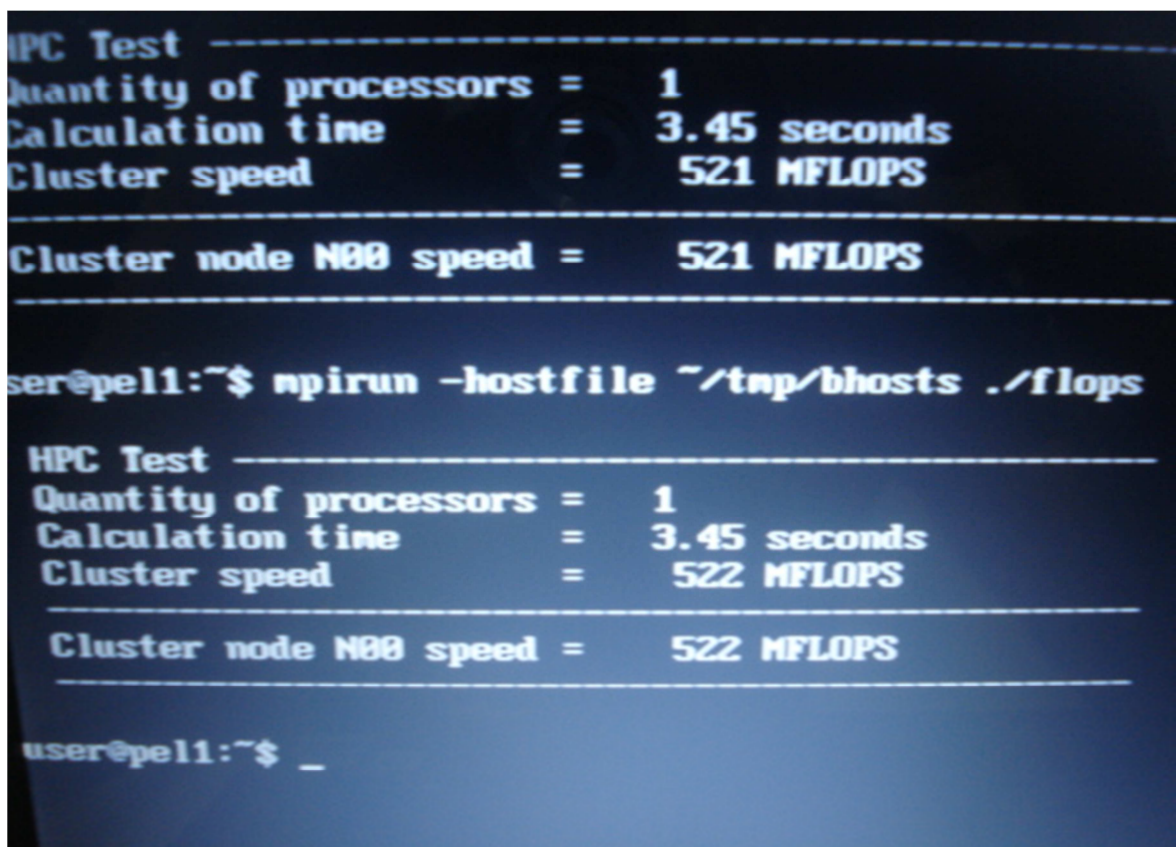
O nó mestre fará o papel de disponibilizar a instalação para as outras máquinas. Então utilizaremos do software do cluster para reconhecer e conectar os nós escravos ao nó mestre e assim finalizar a configuração mais básica de um cluster de alto desempenho. Como o foco do cluster é o processamento paralelo o nó mestre somente utilizara o poder de computação das maquinas escravas (nós escravos) não efetuando nenhuma gravação em disco rígido.

11.1 TESTES DO CLUSTER

Nos testes que mostraremos a seguir utilizamos o teste de capacidade do próprio pacote de gerenciamento do cluster para mostrar os picos de processamento e os resultados de alguns testes mediante as máquinas utilizadas.

Primeiramente utilizamos o nó mestre para obter um resultado de uma única máquina mesmo este não representando um cluster, pois não há no momento nenhum nó escravo, mas servira de base para que possamos entender a capacidade de uma máquina, claro que a capacidade da máquina interfere no resultado, mais isso também será levado em consideração.

Observe a foto tirada do teste efetuado no nó mestre abaixo:



```

HPC Test -----
Quantity of processors = 1
Calculation time      = 3.45 seconds
Cluster speed         = 521 MFLOPS
-----
Cluster node N00 speed = 521 MFLOPS
-----

user@pell1:~$ npirun -hostfile ~/tnp/bhosts ./flops

HPC Test -----
Quantity of processors = 1
Calculation time      = 3.45 seconds
Cluster speed         = 522 MFLOPS
-----
Cluster node N00 speed = 522 MFLOPS
-----

user@pell1:~$ _

```

(Figura 5.)

Repare que a quantidade de processos é igual a um e o teste está sendo executado em um único computador (no nosso caso o nó mestre), observando também que foram feitos dois testes iguais para garantir que os resultados sejam compatíveis.

Então se obteve o seguinte resultado, o teste durou 3,45 segundos para ser completo e a velocidade calculada de 522 MFlops, ou seja, Megaflops (Flops: Floating point Operations Per Second - Operações de Ponto Flutuante por Segundo) que é uma unidade que se usa para medir velocidade computacional.

Depois utilizamos o mesmo teste mas agora dividimos ele em 10 processos diferentes para vermos os resultados.

Abaixo imagem do teste com apenas um nó dividido em 10 processos:

```

er@pel1:~$ mpirun -hostfile ~/tmp/bhosts -np
HPC Test -----
Quantity of processors = 10
Calculation time      = 3.49 seconds
Cluster speed         = 515 MFLOPS
-----
Cluster node N00 speed = 51 MFLOPS
Cluster node N01 speed = 52 MFLOPS
Cluster node N02 speed = 52 MFLOPS
Cluster node N03 speed = 52 MFLOPS
Cluster node N04 speed = 52 MFLOPS
Cluster node N05 speed = 52 MFLOPS
Cluster node N06 speed = 52 MFLOPS
Cluster node N07 speed = 52 MFLOPS
Cluster node N08 speed = 51 MFLOPS
Cluster node N09 speed = 52 MFLOPS
-----

```

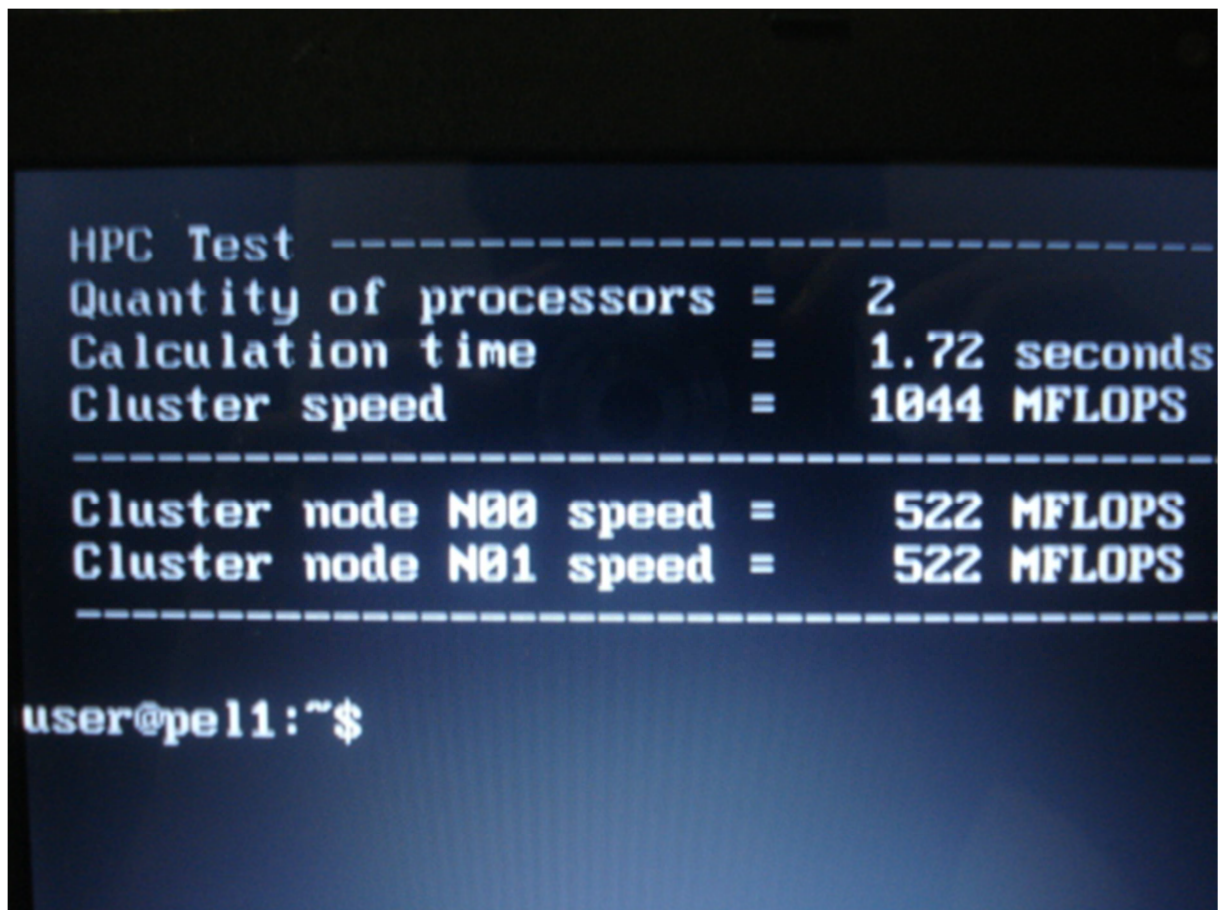
(Figura 6)

Observa-se então que é utilizado basicamente o mesmo poder de processamento em processo. Não alterando de forma significativa o tempo de execução bem como o poder computacional registrado pelo teste, tendo em consideração o teste com apenas um processo.

Tendo esse resultado foi reiniciado o cluster com dois nós, ou seja, o nó mestre e um nó escravo.

Ligados e funcionando em paralelo utilizaremos o comando htop para monitorar em tempo real a reação de processamento do nó escravo quando acionamos o teste de desempenho.

Abaixo a imagem do teste do cluster com dois nós:



```
HPC Test -----
Quantity of processors = 2
Calculation time      = 1.72 seconds
Cluster speed         = 1044 MFLOPS
-----
Cluster node N00 speed = 522 MFLOPS
Cluster node N01 speed = 522 MFLOPS
-----

user@pel1:~$
```

(Figura 7)

Nota-se então que a velocidade do cluster literalmente dobra e há uma queda brusca no tempo de resposta do teste. Cada nó do cluster é medido com a velocidade de 522 megaflops cada, no entanto são máquinas de configurações diferentes o que leva a crer que este resultado é diretamente influenciado pelo meio de transmissão (estamos utilizando uma rede ethernet 10 megabits).

Neste ponto de testes tem-se a impressão de que os dois nós possuem mesmo poder computacional, fica claro que com tão poucos nós não se pode calcular com clareza a real performance do cluster.

Agora é realizado um novo teste nesta mesma configuração de cluster e novamente dividiremos o processo em dez para analisarmos alguns pontos do processamento.

Abaixo o resultado do teste com o processo dividido em dez partes:

```

mrc test
Quantity of processors = 10
Calculation time      = 1.76 seconds
Cluster speed         = 1021 MFLOPS
-----
Cluster node N00 speed = 102 MFLOPS
Cluster node N01 speed = 103 MFLOPS
Cluster node N02 speed = 103 MFLOPS
Cluster node N03 speed = 103 MFLOPS
Cluster node N04 speed = 102 MFLOPS
Cluster node N05 speed = 103 MFLOPS
Cluster node N06 speed = 103 MFLOPS
Cluster node N07 speed = 103 MFLOPS
Cluster node N08 speed = 102 MFLOPS
Cluster node N09 speed = 103 MFLOPS
-----
user@pe11:~$

```

(Figura 8)

A este ponto notamos que os processos são processados com duas vezes mais velocidade em relação ao mesmo teste com apenas um nó.

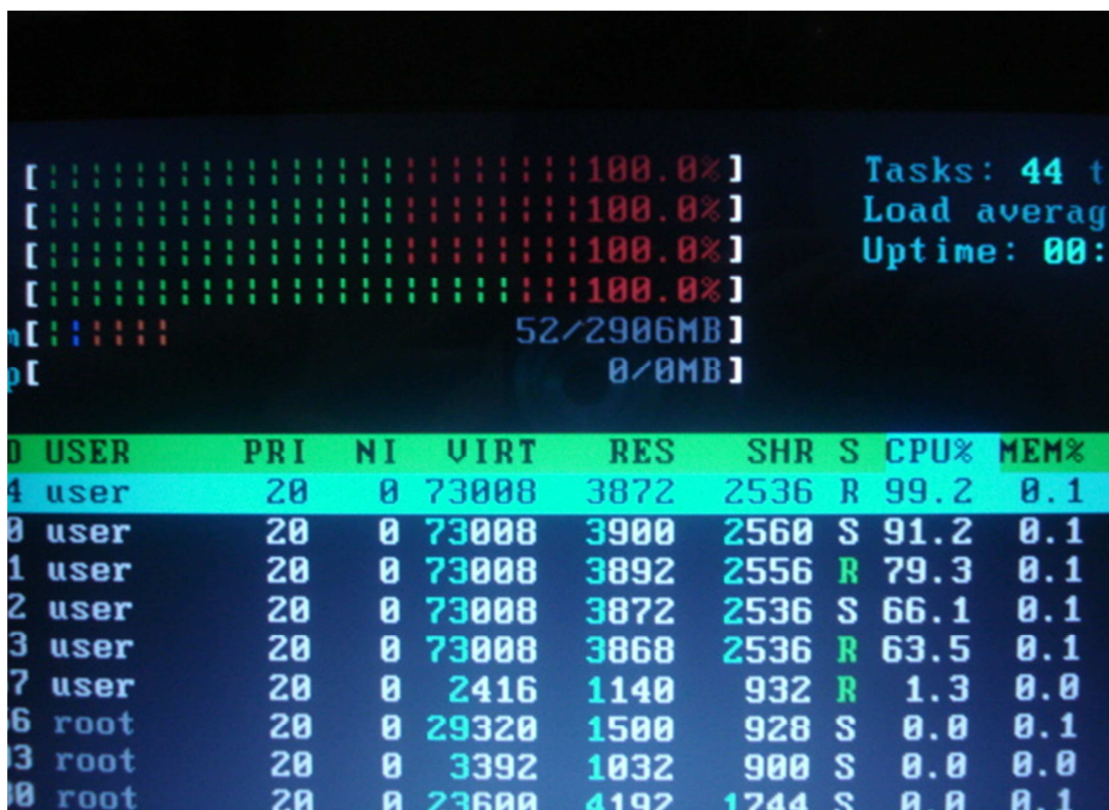
Agora observando o nó escravo mediante a estes dois testes, acompanhando o processamento em tempo real pode-se notar alguns detalhes interessantes.

Concentrando se apenas no nó escravo e observando a configuração do mesmo, sabe-se que este tem um processador com arquitetura de núcleo repartido, mais precisamente em quatro núcleos de processamento. Tendo por base este conceito nota se que na tela de monitoramento deste nó são mostrados os quatro núcleos separadamente.

Então ao executar o primeiro teste com apenas um processo, observa-se que o nó escravo utiliza apenas um núcleo para suprir o processo do cluster, o que é mais do que suficiente tendo em vista que este é um teste de pequeno porte.

No entanto ao dividir o processo notamos que todos os núcleos do processador são utilizados e que mediante isso para que se possa obter Maximo poder computacional neste tipo de arquitetura mais moderna, o software a ser executado no cluster teria que desenvolver também esta função de dividir o processamento para que este seja executado em um processador com vários núcleos, obtendo resultados finais mais satisfatórios.

Abaixo a imagem do nó escravo durante a execução do teste com processamento dividido:



(Figura 9)

O nó escravo tem picos de cem por cento de utilização de todos os núcleos do processador, embora esteja utilizando pouca memória já podemos ter uma noção do que esta configuração de cluster já pode fazer.

Faremos os testes com a última formação, um nó mestre e dois nós escravos, como feito anteriormente reiniciaremos todos os nós, inclusive o mestre.

Logo, seguindo os mesmos procedimentos de ligamento e configuração. Vamos rodar o programa do cluster até que este reconheça os dois nós escravos. A partir daí faremos os testes.

Primeiramente faremos o teste com somente um processamento. Abaixo a imagem que registra o teste com apenas um processamento:

```
Cluster node N00 speed = 520 MFLOPS
Cluster node N01 speed = 583 MFLOPS
Cluster node N02 speed = 521 MFLOPS
-----

user@pel1:~$ mpirun -hostfile ~/tmp/bhosts ./

HPC Test -----
Quantity of processors = 3
Calculation time = 1.15 seconds
Cluster speed = 1565 MFLOPS
-----

Cluster node N00 speed = 521 MFLOPS
Cluster node N01 speed = 583 MFLOPS
Cluster node N02 speed = 521 MFLOPS
-----

user@pel1:~$ _
```

(Figura 10)

Nota-se novamente o crescimento gradativo da capacidade do cluster de um modo geral. Observando agora que com mais um nó no cluster, o primeiro nó escravo já deixa sobressair sua superior capacidade computacional.

Observa-se também que o tempo de execução também diminui gradativamente de acordo com o aumento de poder computacional de uma forma proporcional. Em seguida será observado o ultimo teste, desta vez com vários processos.

Abaixo, teste com os vários processos:

```

HPC Test -----
Quantity of processors = 10
Calculation time      = 1.05 seconds
Cluster speed         = 1712 MFLOPS
-----
Cluster node N00 speed = 171 MFLOPS
Cluster node N01 speed = 387 MFLOPS
Cluster node N02 speed = 173 MFLOPS
Cluster node N03 speed = 174 MFLOPS
Cluster node N04 speed = 174 MFLOPS
Cluster node N05 speed = 173 MFLOPS
Cluster node N06 speed = 368 MFLOPS
Cluster node N07 speed = 378 MFLOPS
Cluster node N08 speed = 172 MFLOPS
Cluster node N09 speed = 171 MFLOPS
-----
user@pe11:~$

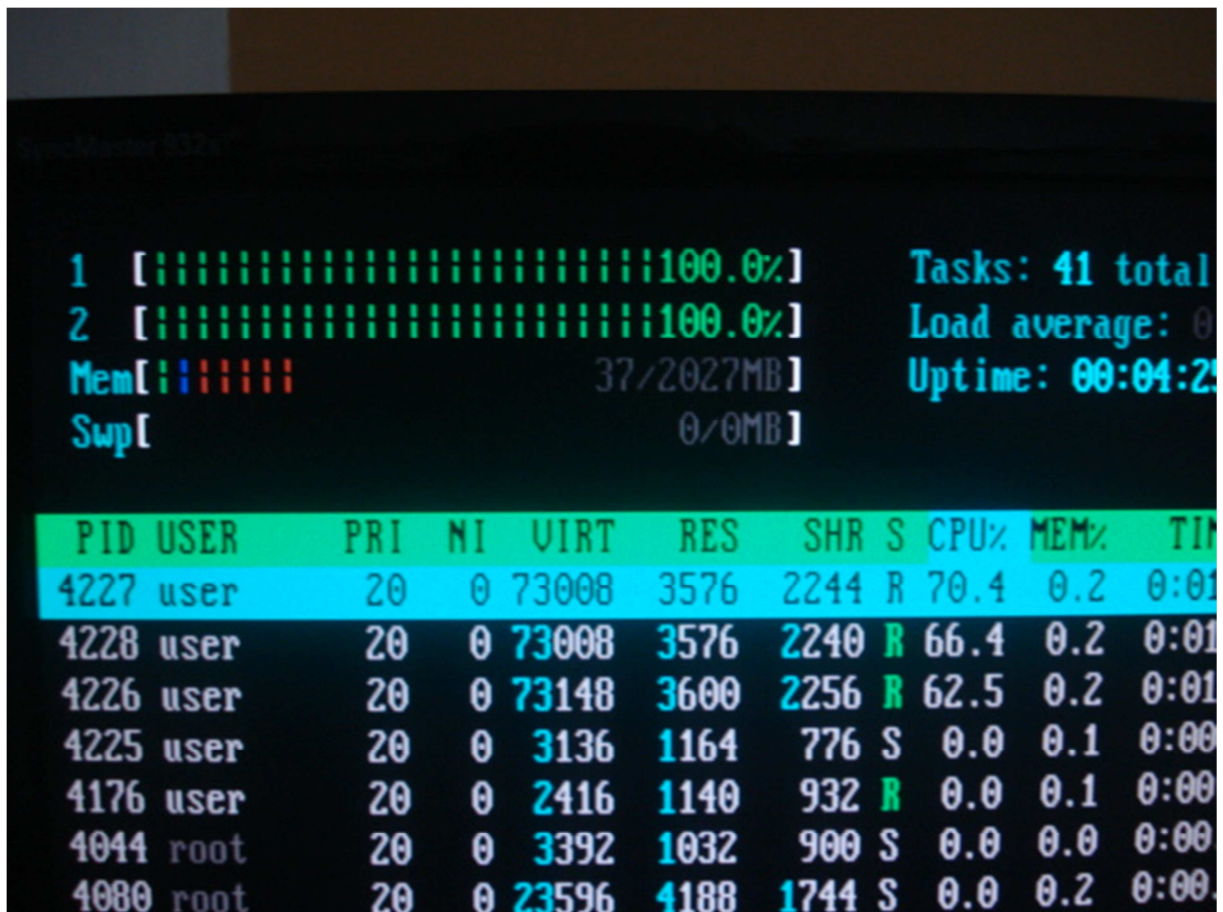
```

(Figura 11)

Agora temos uma visão interessante onde podemos tirar várias conclusões. A velocidade total do cluster obteve aumento, bem como diminuição do tempo de execução. Devido a arquitetura de processador com vários núcleos de processamento agiliza o tempo do teste obtendo uma melhora gradativa no cluster.

Alguns nós também já começam a sobressair a outros, no nosso caso especificamente os nós que executam os processos 01, 06 e 07.

Abaixo a imagem do último nó na hora deste processamento.



(Figura 12)

Na imagem acima podemos observar o processamento do nó escravo numero 2.

Executando o teste com processamento dividido, o nó em questão utiliza todos os núcleos. Da mesma forma que no teste anterior, com o outro nó escravo constatamos que é necessária uma aplicação para dividir os processos.

O teste com processamento dividido foi novamente executado, desta vez com um número menor de processamentos, neste caso 5 processamentos, notou-se observando o quadro de monitoramento dos nós que estes não tiveram 100 por cento de seus núcleos utilizados, na realidade é rigorosamente utilizada a mesma quantidade de processos e núcleos. No caso de o número ser maior os processos que sobram são executados pelos núcleos que primeiro ficam ociosos (terminam primeiro o processamento).

12. CONCLUSÃO

Ao término deste trabalho, foram apresentadas algumas considerações sobre os conceitos e resultados obtidos pelo uso de clusters de alto desempenho.

Como o cluster é feito com equipamento acessível, seu custo indiscutivelmente se torna menor, a dúvida então estava em se o mesmo conseguiria suprir as necessidades de um equipamento de alto desempenho, já desenvolvido para esta finalidade. A resposta é sim.

Com os testes apresentados adquiriu-se a noção de como um cluster pode alcançar um poderoso poder de processamento, tendo em vista que seu aumento de capacidade está diretamente relacionado com quantidade de nós e poder computacional destes nós, o aumento da capacidade é gradativo e proporcional. Outro ponto que o cluster se destaca, é a segurança de funcionamento do mesmo, uma vez que se prepara o nó mestre com certa segurança de funcionamento, os outros nós, caso algum deles apresente defeito, não afeta o funcionamento do cluster, claro que haverá a perda da capacidade deste nó, porém, o cluster continuará operando.

Através dos testes ficou claro também que um cluster onde os nós têm arquitetura com processador de núcleo distribuído se torna indiferente se não houver um software que trabalhe com os mesmos princípios.

De forma geral um cluster é uma estrutura bem interessante com capacidade e arquiteturas a serem exploradas, tendo em vista que se pode montar, desde uma estrutura pequena ou até mesmo uma capaz de suprir as necessidades de grandes empresas. É um campo aberto com muito a oferecer a computação moderna.

13. BIBLIOGRAFIA

- Alecrim, Emerson. Cluster: Principais Conceitos. Info. Wester, 2004 disponível em: www.infowester.com/cluster.php. (ultimo acesso em 02/09/2011)
- BOOKMAN, Charles. Agrupamento de computadores em Linux: Aprenda a construir e manter grupos de computadores com Linux.
- BUYYA, Rajkumar: High. Performance Cluster Computing: Architectures and Systems. Volume 1.
- Dantas, Mario. Ambientes Distribuídos de alto desempenho: Clusters e grades computacionais. [S.]: Portal Brasileiro sobre computação de alto desempenho, 2006.
- Garbee, Bdale; Koptein, Hartmut; Lohner, Nils; Lowe, Will; Mitchell, Bill; Murdock, Ian; Schulze, Martin; Small, Craig. Tradução por Biscaia Filho, Israel Mendes. Uma Breve História do Debian: <http://www.debian.org/doc/manuals/project-history/index.pt.html> (acessado em 06/07/2011)
- Junior, Esli Pereira Faustino; Freitas, Reinaldo Borges. Construindo super computadores com Linux: Cluster Beowulf.
- Pasini , Rodrigo Mendes. Artigo Mosix - Instalação e configuração de um cluster de balanceamento de carga. Disponível em: <http://www.vivaolinux.com.br/artigo/Mosix-Instalacao-e-configuracao-de-um-cluster-de-balanceamento-de-carga> (ultimo acesso em 03/09/2011)
- Pitanga , Marcos. Artigo COMPUTAÇÃO EM CLUSTER . disponível em: http://www.rozero.host22.com/arquitetura/21-Cluster_Grid.pdf. (ultimo acesso em 03/05/2011)