

Estudo de Caso: administração e manutenção do Cluster Unifra

Iuri Fiedoruk, Ana Paula Canal

Centro Universitário Franciscano - UNIFRA
Rua dos Andradas, 1614, (55) 3220-1220
iuri.unifra@gmail.com, apc@unifra.br

Introdução

Este trabalho mostra de que forma é feita a administração e manutenção de uma arquitetura do tipo *Cluster Beowulf* existente no Centro Universitário Franciscano de Santa Maria. Um *cluster* de computadores é uma arquitetura formada por computadores pessoais, ligados através de uma rede, a fim de obter-se desempenho ao se dividir um programa em vários processos, que executam e comunicam-se por troca de mensagens.

Neste artigo, primeiramente são introduzidas as arquiteturas do tipo *cluster*, para fundamentar o item seguinte que caracteriza a arquitetura em estudo. Posteriormente, apresentada a implementação de uma das máquinas (nó) que compõem o *cluster*, sua manutenção, incluindo monitorar as contas dos usuários e programas instalados e verificar o correto funcionamento do *hardware*, e também algumas conclusões do trabalho realizado.

Arquitetura Cluster

Conforme [PAT 2005], existem várias maneiras de se implementar e obter computação paralela, uma delas é implementar uma máquina com vários processadores, mas esta técnica requer uma implementação de *hardware* específico com a construção de um barramento de comunicação entre as diversas *CPUs* e as memórias resultando em um custo elevado, o que levou à busca de alternativas quando o poder exigido de processamento elevou-se principalmente no campo militar e de pesquisa. Hoje em dia, os supercomputadores mais potentes existentes envolvem algum tipo de *cluster* entre sistemas com vários processadores, tornando essa tecnologia bastante aceita e testada nos mais variados campos de computação, como fazendas de renderização gráfica nos estúdios de Hollywood, simulação de explosões nucleares para as potências mundiais e no auxílio à descoberta a prospecção de novas bacias petrolíferas.

Existem vantagens em se utilizar *clusters*, entre as quais destacam-se: alto desempenho obtido pelo uso de vários computadores e da rede de alta velocidade; escalabilidade, onde é possível se obter maior desempenho adicionando-se mais máquinas; tolerância a falhas, onde uma máquina danificada não interrompe todo o sistema e seu papel é desempenhado por outras máquinas; independência de fornecedores, pois *software* utilizado é livre e de código aberto e qualquer computador pessoal pode ser adicionado ao cluster, independente do fabricante, desde que possua os componentes básicos; baixo custo, pelo uso de microcomputadores e *software* livre.

Uma das classes mais utilizadas para a formação de clusters é chamada *Beowulf*,

nome que remonta a um herói mitológico europeu. De acordo com [ALE 2004], ele distingue-se por uma das máquinas controlar todo o *cluster*, caracterizando o modelo mestre-escravos, utilizar sistema operacional *Linux*, utilizar *hardware* comum, como computadores *desktop* ou até servidores e não exigir equipamentos específicos para seu funcionamento, apenas o *hardware* necessário para ligar várias máquinas em uma rede, a qual pode ser *Ethernet*. Vale ressaltar que existem também outros tipos de cluster, normalmente chamados de Alta Disponibilidade, Balanceamento de Carga e Combo. Ainda de acordo com [ALE 2004], outro exemplo de sistema de cluster para *Linux* e *Unix* é o *MOSIX*, que não precisa, ao contrário do *Beowulf*, *software* específico para funcionar.

Pode-se citar que um cluster, além de poder ser formado por máquinas diferentes entre si, também pode ser composto por sistemas computacionais mais simples, ou que normalmente tem outros fins que não os de um cluster. Já foram criados sistemas interligando desde videogames *Playstation 2* [PS2C 2003] e *X-BOX* [KUB 2004], até Placas de vídeo 3D para cálculos vetoriais.

O Cluster em estudo

A arquitetura *Cluster* Unifra é do tipo *Beowulf* e é constituída de três máquinas *Pentium* 4 de 2.4Ghz de *clock*, onde está instalado o sistema operacional *Slackware Linux 10.0*, em um modelo de mestre-escravos onde denominou-se ao mestre *master* e aos dois escravos *slave1* e *slave2*. Os três kits *Pentium* IV, foram obtidos através de doação da empresa Intel e o *cluster* construído é utilizado para fins acadêmicos e de pesquisas. Elas estão ligadas a um *switch* Intel padrão *Fast Ethernet* e as máquinas do tipo escravo utilizam como *gateway* padrão o computador configurado como mestre.

<i>Nome</i>	<i>Endereço IP</i>	<i>Memória RAM</i>
master	192.168.0.1	512 MB
slave1	192.168.0.2	256 MB
slave2	192.168.0.3	256 MB

Tabela1: Componentes do Cluster Unifra

O uso de sistema operacional *Linux*, além de ser econômico, garante estabilidade e é o lar natural para sistemas de cluster *Beowulf* desde seu surgimento, em virtude da grande instabilidade de alguns sistemas operacionais proprietários e inviabilidade econômica das instituições educacionais pagarem uma licença de sistema operacional, como um UNIX, por exemplo, para cada máquina do cluster. Conforme [LEM 2005], existem até mesmo distribuições *Linux* que rodam direito do CD e criam automaticamente um Cluster, para utilizar-se do poder computacional de sistemas *desktop* durante a noite, quando estão normalmente desligados sem nenhuma utilização.

A biblioteca utilizada para programação paralela nas máquinas é a *MPICH* 1.2. *MPI* é um padrão estabelecido por um comitê de empresas, como Intel, IBM, Shell, dentre outras, e usuários, disponível a todos que desejarem criar uma implementação [MPI 2005]. Este comitê também criou uma implementação própria e bastante utilizada, chamada *MPICH*, atualmente na versão 2. Recentemente a Microsoft anunciou que irá incorporar a versão 2 do *MPICH* na atualização do *Windows 2003*, conforme [GAL

2005] o que deverá aumentar ainda mais a popularização desta biblioteca tornando-a de fato um padrão para programação paralela.

Para que os programas tipo *MPI* que devem ser executados nas três máquinas ter acesso aos arquivos que serão compilados e executados em cada máquina, o diretório */home* da máquina *master*, que contém os arquivos das contas de usuários do sistema operacional, foi exportado para as máquinas *slave1* e *slave2* também como */home*, desta forma, quando um usuário efetua *login* em qualquer uma das máquinas, terá sempre à disposição seus arquivos e programas. Para testes foi instalado o programa *distcc* que distribui a compilação de programas escritos em linguagem C entre várias máquinas interconectadas por uma rede e permite visualizar em tempo real o quanto cada máquina está processando.

Exemplo de implementação de um nó

Cada nó do *Cluster* possui configurações bastante semelhantes, de forma que podem ser criados novos nós tomando-se como base a configuração de apenas um deles. Por ocorrência de um problema técnico no equipamento, houve a necessidade de re-configuração de um nó do *Cluster* Unifra, conforme configurações descritas a seguir.

Para iniciar, fez-se o *download* e a gravação em CD dos dois arquivos de imagem do *Linux Slackware* 10.0, versão esta, que já estava instalada anteriormente nas máquinas. Como os equipamentos possuem apenas um dispositivo leitor de *CD-ROM*, foi necessário transferir o mesmo do *master* para o *slave2* para ser feita a instalação do sistema.

A instalação do *Slackware* 10.0 é bastante simples para quem já possui algum conhecimento de sistemas *Linux*, a observar apenas que procurou-se instalar o mínimo de pacotes de programas para manter a instalação leve favorecendo o desempenho, o que gerou de certa forma uma arquitetura *cluster* heterogênea. Também instalou-se o sistema de arquivos *EXT3* no lugar do *ReiserFS*, instalado nas outras duas máquinas, como forma de testar em caso de falhas a recuperação do sistema de arquivos.

Após o sistema ser instalado, partiu-se para a sua configuração. Inicialmente é preciso editar o arquivo */etc/hosts.equiv*, incluindo todos os nomes de máquinas que compõe o cluster para permiti-las executar comandos remotos através do programa *rsh* (*remote shell*). Em seguida deve-se importar o diretório *NFS /home* do *master* incluindo-o no arquivo */etc/fstab*, como no exemplo abaixo:

master:/home	/home	nfs	defaults	0	0
--------------	-------	-----	----------	---	---

Tabela2: Configuração do NFS

Além disso, foi preciso certificar-se que todos podem acessar de maneira simples este novo nó e vice-versa, para isto utiliza-se uma conveniência do *Linux* que permite a criação de um apelido relacionado a um endereço *IP*. A seguir é feita a configuração dos serviços de rede do nó. Esta é uma etapa importante para que a troca de mensagens realizadas pelos programas paralelos que rodam utilizando a biblioteca *MPI* possa ser efetuada através da rede; para tanto devemos nos certificar que os serviços de *shell* e *login* remotos serão executados no nó, incluindo ou substituindo, se já houverem linhas desses serviços com outras opções que não as desejadas, no arquivo */etc/inetd* conforme a tabela 3.

shell	stream	tcp	nowait	root	/usr/sbin/tcpd	in.rshd -h
login	stream	tcp	nowait	root	/usr/sbin/tcpd	in.rlogind -h
exec	stream	tcp	nowait	root	/usr/sbin/tcpd	in.rexecd -h

Tabela3: Serviços de rede

Conclusão

A arquitetura de um cluster exige de seu administrador estudo sobre vários campos da computação e área técnica. Os problemas que surgem podem ser dos mais variados, desde falha em um dos equipamentos exigindo realização de cópias de segurança dos arquivos, até um cabo de rede defeituoso. Percebe-se a importância de ser criada documentação a fim de manter o conhecimento adquirido, evitando que problemas futuros se tornem de difícil resolução.

Como trabalhos futuros planejam-se investigar maneiras das máquinas agregarem-se ao *cluster*, de forma automática utilizando-se de serviços de rede. Também deseja-se substituir o sistema de senhas pelo serviço oferecido pelo NIS - *Network Information Service*.

Referências

- [MPI 2005] ANL Mathematics and Computer Science. **The Message Passing Interface standard**. <http://www-unix.mcs.anl.gov/mpi/> (acesso em 09 de setembro de 2005)
- [PS2C 2003] NCSA. **PlayStation 2: Computational Cluster**. <http://arrakis.ncsa.uiuc.edu/ps2/cluster.php> (acesso em 5 de outubro de 2005)
- [ALE 2004] ALECRIM, Emerson. **Cluster: principais conceitos**. <http://www.infowester.com/cluster.php> (acesso em 10 de setembro de 2005)
- [GAL 2005] GALLI, Peter. **Open Source Code Finds Way into Microsoft Product**. <http://www.eweek.com/article2/0,1895,1859439,00.asp> (acesso em 15 de setembro de 2005)
- [LEM 2005] LEMAY, Renai. **Linux distro turns PCs into supercomputers**. <http://www.zdnet.com.au/news/software/0,2000061733,39187301,00.htm> (acesso em 15 de setembro de 2005)
- [KUB 2004] KUBICKI, Kristopher. **The AnandTech Linux XBOX PC Experiment**. <http://www.anandtech.com/linux/showdoc.aspx?i=2271> (acesso em 18 de setembro de 2005)
- [PAT 2005] PATTERSON, David A. e HENNSSY, Jonh L. **Organização & Projeto de Computadores: a interface Hardware / Software**. Rio de Janeiro, RJ; Editora Campus, LTC, 2000