

Computação de Alto desempenho: Clusters

Leandro Rodrigues Magalhães de Marco - RA009089
Instituto de Computação - UNICAMP
Avenida Albert Einstein, 1251
Campinas, SP - Brasil
ra009089@students.ic.unicamp.br

ABSTRACT

Existem limitações para o poder de processamento de máquinas individuais. Por mais que se procurem soluções em máquinas multiprocessadas, essas limitações sempre existirão.

O emprego de cluster, que são diversas máquinas conectadas em rede e que podem ser vistas como apenas um super-computador de fora, ajuda a resolver o problema de processamento que não poderia ser realizado em tempo hábil por uma máquina isolada.

Clusters também são aplicados para aumentar, através de redundância, a confiabilidade de sistemas que devem ser tolerantes a falhas.

Com a popularização dos PCs, tornou-se possível criar-se clusters de baixo-custo e que podem superar o poder de processamento do mais caro supercomputador.

Keywords

Computação de alta performance, Arquitetura de Computadores, Clusters

1. INTRODUÇÃO

A partir do momento em que usuários de Computadores notaram que todo o trabalho computacional não poderia ser realizado em tempo hábil ou de forma confiável por um simples computador, surge a idéia de dividir o trabalho em diversas máquinas.

Um Cluster Computacional é um conjunto de computadores que estão ligados em rede e que trabalham em conjunto. Cada computador de um cluster é denominado nó. Dependendo do objetivo do Cluster, esses nós podem processar pequenas partes de um problema maior que foi dividido ou podem servir como nós de reserva para garantir um sistema toletrante a falhas.

O objetivo de se construir um cluster é em geral, ganho de desempenho e/ou confiabilidade quando comparado ao que seria obtido em uma única máquina. Um cluster também possui um custo mais baixo do que um único computador com velocidade e confiabilidade comparáveis.

2. A HISTÓRIA DOS CLUSTERS

Uma citação de *In Search of Clusters de Greg Pfister*[4], ajuda a definir a origem dos Clusters: "Os clientes inventaram os clusters a partir do momento em que não conseguiam colocar todo o trabalho em um simples computador, ou precisavam de um backup. A data do primeiro Cluster é desconhecida, mas eu me surpreenderia se não fosse nos anos 60 ou no fim dos anos 50".

Entretanto, a base formal do Cluster computacional como meio de executar tarefas em paralelo foi criada por *Gene Amdahl* da IBM, que em 1967 publicou o artigo "*Validity of the Single Processor Approach to Achieving Large-Scale Computing Capabilities*"[1]. Neste, Amdahl apresenta a conhecida *Lei de Amdahl* que descreve matematicamente o ganho de velocidade resultante da paralelização de uma tarefa serial. Este artigo contém a base para para multiprocessadores e clusters. A diferença entre paralelizar utilizando-se multiprocessamento e cluster é que nos primeiros a comunicação é feita dentro do computador através de barramentos, etc. enquanto nos últimos a comunicação é feita fora, através de rede.

Devido a isso, a história dos clusters está intimamente ligada a história das redes de computadores. Uma das motivações do desenvolvimento de redes, era ligar computadores para poder-se somar recursos, construindo-se assim clusters. A internet pode ser encarada como um imenso cluster. Iniciativas como o projeto *SETI@home*, se valem desse imenso poder de processamento que a internet possui para executar tarefas e cálculos complicados com maior eficiência.

Outro aspecto importante que possibilita a existencia de Clusters é a definição de protocolos padrão para comunicação (TCP/IP).

O primeiro cluster comercial foi o *ARCNET* desenvolvido pela Datapoint em 1977. O *ARCNET* não foi um grande sucesso comercial e a idéia de Cluster só obteve sucesso comercial em 1984 quando a DEC lançou *VAXcluster* para o sistema operacional *VAX/VMS*. Este produto suportava além da execução de tarefas em paralelo, sistemas de ar-

quívos e dispositivos compartilhados. Este sistema proporcionava a vantagem da computação paralela mantendo a confiabilidade da integridade dos dados. Outros dois dos primeiros clusters comerciais foram o *Tandem Himalaya* e o *IBM S/390 Parallel Sysplex*.

Com o invento do *PVM: Parallel Virtual Machine*, e com a popularização dos PCS, foi possível construir supercomputadores a partir de PCs normais. Foram criados Clusters de baixo custo com poder de processamento maior que o dos melhores e mais caros supercomputadores. Em 1993 a NASA iniciou um projeto para a criação de supercomputadores a partir de PCs comuns.

3. TIPOS DE CLUSTERS

• *High-Availability (HA) Clusters*

Clusters de alta disponibilidade são construídos para aumentar a disponibilidade de serviços que os mesmos provém. Isto é feito através de nós redundantes que são usados em caso de falha de componentes. Comumente estes Clusters possuem dois nós, que é o mínimo necessário para prover redundância.

Entretanto existem clusters com até dezenas de nós, dependendo do nível de confiabilidade desejada.

Algumas configurações deste tipo de Cluster estão descritas em "*Blueprints for High Availability: Designing Resilient Distributed Systems*"[3] Podemos categorizar as possíveis configurações destes cluster da seguinte maneira:

- Active/Active: o tráfego que se destinaria a um nó que falhou é ou passado a um nó existente ou balanceado entre os nós remanescentes. Usualmente exige que os nós possuam uma configuração de software homogênea.
- Active/Passive: Cada nó contém uma instância redundante que é ativada apenas em caso de falha. Esta configuração é a que requer a maior quantidade de hardware.
- N+1: Existe um nó extra que é ativado para substituir um nó que venha a falhar. Em caso de configurações heterogêneas de software, o nó extra deve ser capaz de assumir qualquer uma das tarefas dos nós principais. Esta nomenclatura se aplica para clusters com múltiplos serviços. No caso de um serviço só, este modelo é equivalente ao Active/Passive
- N+M: Em clusters que executam muitos serviços, um nó redundante apenas pode não ser suficiente. Neste caso M nós redundantes são colocados, dependendo dos requisitos de custo e confiabilidade.

• *Load Balancing Clusters*

Estes clusters operam tendo toda carga de trabalho chegando a um ou mais "front-ends" que distribuem a carga entre os demais nós (denominados "back-end servers"). Este tipo de Cluster é utilizado para prover alta-performance, entretanto ele comumente possui recursos que adicionam características de Clusters de alta-disponibilidade. São conhecidos também como "Server Farm"

• *High Performance (HPC) Clusters*

Clusters de alta-performance são construídos com o objetivo de aumentar a velocidade de tarefas dividindo-as entre inúmeros nós.

Este tipo de cluster é mais comumente utilizado para aplicações científicas. Muitas aplicações que rodam em HPC foram desenvolvidas especialmente para tirar proveito do paralelismo oferecido pelo cluster.

Clusters HPC são otimizados para rodar tarefas que executam em nós separados e comunicam-se entre si ativamente. Isso inclui problemas nos quais resultados intermediários de um nó afetam resultados de computações futuras em outros nós.

Beowulf é uma das mais populares implementações de HPC. O Beowulf é um modelo criado para obter um sistema de alta performance com baixo custo. Roda em Sistema Operacional Linux e utiliza Software livre para explorar paralelismo.

• *Grid Computing*

Grid[2] é uma tecnologia que é parecida com a tecnologia de Cluster. A principal diferença é que um Grid conecta computadores que não confiam totalmente uns nos outros. Grids suportam ambientes mais heterogêneos do que clusters.

O grid é otimizado para cargas de trabalho que consistem de muitas tarefas independentes entre si e que portanto não tem que se comunicar durante o processamento.

Assim, grids administram distribuição de tarefas para computadores que trabalharão independentemente. Alguns recursos como os de armazenamento podem ser compartilhados, mas resultados intermediários de uma tarefa em um nó não alteram a computação de tarefas em outros nós.

4. TECNOLOGIAS DE CLUSTERS

Alguns exemplos de tecnologias que estão relacionadas a Clusters

- distcc: permite a execução de compilação paralela quando se utiliza o GCC
- MPI: biblioteca que implementa um protocolo de comunicação entre processos rodando em diferentes nós de um Cluster.
- Linux Virtual Server, Linux-HA: clusters que implementam soluções de balanceamento de carga entre nós
- MOSIX, openMosix, Kerrighed, OpenSSI: cluster integrados ao Kernel do Sistema Operacional e permitem migração automática de processos entre nós homogêneos.
- MSCS: Solução da Microsoft para clusters de alta-disponibilidade em ambiente Windows.

5. CONCLUSÕES

A idéia de dividir tarefas ou mesmo recursos em computação não é nova. Muito provavelmente clusters surgiram como uma solução natural para problemas que ocorriam devido a limitações das máquinas. Limitações estas que existem até hoje e muito provavelmente irão existir no futuro. A evolução dos Clusters anda de mãos dadas com a evolução das redes de computadores e da criação de padrões de comunicação. Talvez a maior revolução trazida pela tecnologia de Cluster, seja a possibilidade de montar supercomputadores com máquinas comuns. Possuimos por exemplo na internet, um imenso poder computacional que é em sua maioria inexplorado.

6. REFERENCES

- [1] G. Amdahl. Validity of the single processor approach to achieving large-scale computing capabilities. *AFIPS Conference Proceedings*, pages 745–770, 1967.
- [2] I. Foster. What is the grid? a three point checklist.
- [3] E. Marcus and H. Stern. *Blueprints for High Availability: Designing Resilient Distributed Systems*. John Wiley Sons.
- [4] G. Pfister. *In Search of Clusters*. Prentice Hall.