Panorama de ferramentas para gerenciamento de clusters*

Claudio Schepke, Tiarajú A. Diverio
Programa de Pós-Graduação em Computação
Instituto de Informática, UFRGS
{cschepke,diverio}@inf.ufrgs.br

Marcelo V. Neves, Andrea S. Charão
Laboratório de Sistemas de Computação
Curso de Ciência da Computação, UFSM
{veiga,andrea}@inf.ufsm.br

Resumo

O uso de clusters *tem sido uma das alternativas mais adotadas para o desenvolvimento de sistemas computacionais paralelos. No entanto, a configuração e manutenção deste tipo de arquitetura envolve diversos fatores, sendo simplificada pela existência de ferramentas específicas para cada problema. Neste trabalho serão apresentadas algumas ferramentas que facilitam a instalação de sistemas operacionais e programas que possibilitam o gerenciamento e monitoração de todo o sistema, buscando descrever as diferentes alternativas existentes para cada caso.*

1. Introdução

Clusters são comumente utilizados em aplicações de simulação, biotecnologia, petroquímica, modelagem de mercados financeiros, mineração de dados, processamento de imagens e servidores de música e jogos para a *Internet*. Um *cluster* é um conjunto de computadores independentes conectados por rede que formam um sistema único através do uso de *software* [1]. Em geral *clusters* são classificados segundo alguns critérios. Um *cluster* é dito homogêneo quanto todos os nós da máquina possuem a mesma configuração; caso contrário eles são conhecidos como heterogêneos. Já o número de processadores existentes por máquina permite a classificação entre mono (um processador) ou multiprocessados (vários processadores), sendo freqüente neste último caso a utilização de dois processadores. Uma terceira classificação leva em conta o modo de configuração do *cluster*. Neste caso, um *cluster* pode ser formado através de um determinado número de computadores ou até de constelações (*cluster* de *clusters*) [19].

O gerenciamento de *clusters* envolve diversos fatores, desde a instalação do sistema operacional, até a definição de ferramentas para a configuração, manutenção, monitoramento e escalonamento de tarefas. Este artigo busca apresentar as características de algumas ferramentas que permitam realizar diferentes tarefas relacionadas ao gerenciamento de *clusters*. A próxima seção descreve alguns recursos que facilitam a instalação, utilização e manutenção do sistema. A seção 3 apresenta as ferramentas de escalonamento de tarefas. Na seqüência são discutidas as características de alguns programas de monitoração utilizados atualmente. Por fim são apresentadas as conclusões obtidas com a realização do trabalho.

* Este trabalho é fomentado pelo CNPq.

2. Instalação do Sistema

2.1. Mecanismos de Instalação Automática

Clusters são geralmente formados por um número bastante grande de computadores. A instalação e configuração individual de cada sistema operacional para cada máquina pode levar muito tempo. Como em geral as máquinas utilizam o mesmo sistema operacional é possível fazer uso de um mecanismo automático de instalação. Para tanto existem diversas ferramentas. A seguir serão apresentadas algumas delas, buscando descrever as suas principais características.

- *Kickstart* [16]: é um sistema desenvolvido para *RedHat Linux* que permite colocar todas as seleções que o usuário faria na instalação manual, como seleção da linguagem, partições, pacotes a serem instalados, etc, em um arquivo de configuração, eliminando toda iteração com o usuário.
- FAI (*Fully Automatic Installation*) [7]: é um conjunto de *scripts* e arquivos de configuração para instalação automatizada de sistema *Debian Linux* em um agregado com um grande número de nós. FAI é um método escalável, onde cada nó realiza a sua própria instalação a partir de um arquivo de configuração de um servidor. Para tanto, um nó cliente carrega um sistema temporário, via rede ou disquete, que começa a instalação propriamente dita.

- *Replicator* [3]: outro recurso desenvolvido exclusivamente para sistemas *Debian Linux*, funcionando como um duplicador de instalação.
- ALICE [4]: é um sistema para *SuSE Linux* que permite instalar e configurar várias máquinas automaticamente com o mínimo possível de interação humana. Baseado em interfaces como *syslinuxrc*, *YaST* e *suseconfig*, além de instalar os sistema operacional, ALICE também pode criar grupos e usuários, ativar serviços, etc.
- OSCAR (*Open Source Cluster Application Resources*) [11]: É um ambiente para a instalação, configuração e gerenciamento de *clusters*. OSCAR apresenta de forma integrada os recursos mais utilizados em *cluster*, disponibilizando a configuração automática de componentes, bem como a instalação eficiente do ambiente básico como sistema operacional e ferramentas de administração e operação. A versão corrente de OSCAR possui suporte para as distribuições *Linux Red Hat*, *Fedora* e *Mandriva*.

2.2. Carga Remota do Sistema

Para *clusters* formados por máquinas homogêneas é possível fazer a carga remota do sistema operacional. Para tanto, primeiramente é feita a instalação do sistema em uma das máquinas do agregado. A partir desta instalação é feita uma imagem que ficará armazenada em um servidor. Essa imagem é carregada automaticamente para as máquinas quando elas são iniciadas pela rede. É possível também criar várias imagens, com diferentes configurações e recursos de programação paralela, permitindo a escolha de uma delas no momento em que o sistema irá ser carregado. A seguir são apresentados alguns programas que permitem a carga remota do sistema.

- *SystemImager* [6]: um conjunto de *scripts* que simplificam os procedimentos para preparação da carga remota. Com ele também é possível atualizar as imagens já distribuídas para os nós. As atualizações são rápidas porque somente as partes modificadas são mandadas ao cliente. SystemImager também permite o armazenamento de várias imagens em um servidor, podendo estas serem associadas à nós específicos.
- *Rembo Toolkit* [22]: um inicializador remoto desenvolvido a partir da ferramenta *BpBatch* que utiliza o protocolo PXE e permite a execução de várias ações em tempo de *boot*, antes do sistema operacional ser iniciado. Assim, é possível desde particionar o disco rígido, até autenticar usuários, como também criar uma imagem de um disco rígido clonando o estrutura de partições.
- *Ka-deploy* [20]: é uma ferramenta que faz parte do *Ka Clustering Tools*, que permite replicar uma máquina Linux muitas vezes ao mesmo tempo. A carga remota em *clusters* cria uma corrente de dados entre os nós, cada nó copia os dados para seu disco local e envia para o resto da corrente.
- *ClusterWorx* [15]: é outro exemplo de ferramenta para auxiliar o processo de carga remota. Ele foi desenvolvido pela *Linux NetworX*, possuindo também um gerenciador de imagens.

2.3. Atualização e configuração do sistema

Instalar e configurar pacotes nos nós de um agregado de computadores pode ser outro grande problema, principalmente quando este é formado por algumas dezenas ou centenas de nós. A primeira alternativa é o compartilhamento dos arquivos por NFS (*Network File System*), mas isso pode congestionar a rede com o aumento do numero de nós.Assim é necessário ter um sistema que permita instalar, atualizar e configurar programas nos nós de forma automatizada e com boa escalabilidade. Alguns programas que auxiliam na configuração de *clusters* são:

- SCMS (*Scalable Cluster Management System*) [21]: é um sistema desenvolvido pela Universidade Kasetsart (Tailândia) que possui recursos úteis para configuração e atualização remota como, por exemplo, comandos UNIX paralelos, permitindo a execução da mesma tarefa em vários nós ao mesmo tempo e a instalação de pacotes RPMs em paralelo.
- xCAT [8]: desenvolvido pela IBM, é um sistema que automatiza alguns processos de instalação e configuração. Ele permite ligar e desligar as maquinas remotamente, acessar a BIOS através de um console e usar uma espécie de *shell* paralelo para executar o mesmo comando em vários nós.
- SHOC [24]: é um sistema que permite ao administrador, através de um *shell bash*, usar o agregado com se fosse uma única máquina.

3. Escalonamento

O escalonamento define como são utilizados os nós de um *cluster*, fornecendo, mediante requisição, a possibilidade de uso do mesmo por parte dos usuários. Os objetivos de um escalonador são maximizar a utilização do *cluster*, maximizar a quantidade de aplicações executadas, reduzir o tempo de resposta, mesclar requisições dos usuários com as ordens administrativas e dar a ilusão de uma máquina única e dedicada. Embora alguns pareçam ser contraditórios entre si, cabe ao escalonador definir a melhor po-

lítica de uso. Algumas ferramentas de escalonamento mais conhecidas são:

- CCS (*Computing Center Software* [10, 12]: desenvolvido pelo Centro de Computação Paralela de Paderborn (Alemanha), o objetivo de CCS é gerenciar sistemas MPP (*Massively Parallel Computing*) e *clusters* num sistema de planejamento. Para isso, a ferramenta permite o acesso aos recursos de forma exclusiva concorrentemente, processamento simultâneo do modo interativo e de fila, maximização do uso do sistema através do particionamento dinâmico e escalonamento, além de tolerância a falhas em acesso remotos. A arquitetura do CCS é modular, o que permite integrar um grande número de sistemas.
- PBS (*Portable Batch System*) [17]: Desenvolvido inicialmente pela NASA e posteriormente apresentado em uma versão comercial, PBS apresenta suporte a tarefas tanto de um único sistema como de múltiplos sistemas. Devido a flexibilidade do PBS, os sistemas podem ser agrupados de diferentes formas. O sistema de escalonamento utilizado adiciona as tarefas primeiramente a uma fila, para que, posteriormente, o escalonador analise as tarefas. Além da versão comercial existe também uma versão de código aberta conhecida como openPBS [12].
- *Condor*: é uma ferramenta que pode ser usada para gerenciar *clusters* e múltiplos *clusters*. Para a execução de tarefas é necessário primeiramente definir os recursos necessários. A seguir as tarefas são armazenadas em uma lista de espera. Algumas das características de *Condor* são a submissão distribuída de tarefas, prioridades para usuários e tarefas, suporte a múltiplos modelos de tarefas, *checkpointing* e migração, suspensão de tarefa e posterior continuação, autenticação e autorização, entre outros.
- *Maui* [9]: é um escalonador de tarefas configurável e otimizado, usado em *clusters* e supercomputadores, capaz de suportar diferentes técnicas de escalonamento, prioridades dinâmicas, reserva de recursos e compartilhamento justo. As técnicas de otimização adotadas em *Maui* permitem aumentar a utilização dos recursos e diminuir o tempo de resposta na execução de tarefas paralelas. Implementado em Java, o que permite a extensão e utilização da ferramenta em diversos ambientes, *Maui* necessita de uma JVM para ser instalado e utilizado.
- *Crono* [14]: possui como objetivo principal o gerenciamento de *clusters* pequenos e médios num sistema de planejamento, uma vez que a utilização do *cluster* ocorre por meio de de agendamentos. Ele foi desenvolvido pela PUC-RS, disponibilizando serviços necessários para compartilhar um *cluster* entre vários usuários. A arquitetura da ferramenta é composta de quatro partes, onde estas são responsáveis por realizar a interface com o usuário, gerenciar o acesso (validação das requisições), gerenciar as requisições (escalonar pedidos e preparar o ambiente de execução) e gerenciar o nó.

4. Monitoração

A monitoração é um processo que consiste em apresentar a utilização dos recursos de um *cluster* através da análise de dados recolhidos continuamente do sistema. Desta forma é possível obter informações sobre a existência de máquinas ociosas ou com problemas, utilização da rede, capacidade de processamento do processador e quantidade de memória utilizada, permitindo assim a tomada de decisões. Atualmente existem diversas ferramentas que permitem verificar o estado de um ambiente de maneira simples e intuitiva. Como exemplos de aplicações tempos *Ganglia*, SCMS e *RVision*, que serão vistas na seqüência.

- *Ganglia* [13]: é uma ferramenta de de monitoração para *clusters* e *grids* desenvolvida de forma distribuída e escalável. Um módulo centralizador coleta e atualiza as informações, enquanto que cada nó mantém uma cópia do estado corrente do sistema. Os dados coletados podem ser visualizados graficamente através de uma interface *Web*. Com *Ganglia* é possível monitorar qualquer tipo de informação, uma vez que o usuário pode definir métricas específicas através de outra aplicação, além daquelas já coletadas pelo próprio sistema.
- Parmon [2]: é uma ferramenta comercial que possui uma arquitetura centralizada, sendo dividida em duas partes: servidor, responsável por monitorar o nó, e cliente, onde é feita a centralização de todos os dados monitorados e a visualização gráfica e *on-line* ou textual das informações. Parmon permite adquirir informações dos recursos do sistema de vários nós, acompanhar processos e *logs* do sistema, além de definir eventos de alerta (*trigger*) ao administrador do *cluster*. Também é possível monitorar CPU, memória, rede e disco e executar alguns comandos paralelos.
- SCMS [21]: tem como objetivo monitorar de forma simples, eficiente e robusta *clusters* de pequeno e médio porte através de uma arquitetura centralizada organizada num módulo de monitoração e num módulo de centralização, o qual armazena os dados monitorados e atende as requisições dos clientes. A ferramenta permite monitorar o uso de CPU, memória, rede e disco, além de fornecer informações úteis sobre a configuração dos nós do *cluster*. A coleta de dados ocorre em ciclos ou por demanda, no caso das informações de con-

figuração do sistema. Já a apresentação gráfica dos dados monitorados por SCMS ocorre no cliente.

- *RVision* [5]: é uma ferramenta de monitoração desenvolvida com o objetivo de ser adaptável à diferentes *clusters*, tendo uma arquitetura aberta e configurável. Para manter essas características, a ferramenta possui uma interface para a comunicação de clientes com o núcleo da ferramenta. A arquitetura de *RVision* é centralizada, sendo composta um de programa monitor e um programa centralizador. Ao invés do monitor, é possível utilizar um agente SNMP em seu lugar, sendo possível assim a adição de novas métricas. O tipo de visualização irá depender do cliente implementado.

5. Tendências e Conclusão

De uma forma geral as ferramentas de gerenciamento tem evoluído no sentido de incluir diversas funcionalidades em um único componente, contemplando assim todas as necessidades exigidas na administração de *clusters*. Este é o caso das ferramentas ROCKS e OpenSCE [23, 18], que apresentam diferentes funcionalidades ou buscam a integração entre vários *softwares*, disponibilizando em um único recurso a possibilidade de instalação e configuração de *software*, a monitoração e gerenciamento do estado do *cluster*, o balanceamento de carga, além de ferramentas que auxiliem no desenvolvimento de aplicações.

A administração de *clusters* envolve diversas questões, muitas das quais com a existência de soluções bem definidas. A escolha de recursos e ferramentas certas para cada tipo de tarefa torna a implementação do sistema mais simples. Este trabalho apresentou diversas ferramentas que auxiliam na automatização das tarefas de instalação do sistema operacional, configuração e manutenção do sistema. Cada uma dessas ferramentas apresenta características específicas, apresentando soluções definidas para determinados tipos de problema. Já na monitoração, as ferramentas apresentam-se bastante dinâmicas, possibilitando a filtragem de novas métricas, além da possibilidade de extensão e gerenciamento de mais de um agregado.

Referências

[1] R. Buyya. *High Performance Cluster Computing: Architectures and Systems*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1999.

[2] R. Buyya. PARMON: A portable and scalable monitoring system for clusters. *Software Practice and Experience*, 30(7):723–739, jun 2000.

[3] S. Chaumat. Replicator 2.0 for Debian/GNU Linux 2.2 Manual, 2000.

[4] A. F. F. Herschel, P. Hollants. ALICE: Automatic Linux Installation and Configuration Environment, 2000. http://www.suse.de/ fabian/alice/.

[5] T. C. Ferreto, C. A. F. de Rose, and L. de Rose. Rvision: An open and high configurable tool for cluster monitoring. *2nd IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGRID'02)*, page 75, 2002.

[6] B. Finley. Systemimager, 2003. http://systemimager.org.

[7] M. Gärtner, T. Lange, and J. Rühmkorf. The fully automatic installation of a Linux cluster, 1999.

[8] M. Govindaraju, S. Krishnan, K. Chiu, A. Slominski, D. Gannon, and R. Bramley. XCAT 2.0 : A Component Based Programming Model for Grid Web Services. In *Grid 2002, 3rd International Workshop on Grid Computing*, 2002.

[9] D. Jackson, Q. Snell, and M. Clement. Core Algorithms of the Maui Scheduler. *Lecture Notes in Computer Science*, 2221:87–94, 2001.

[10] A. Keller and A. Reinefeld. CCS Resource Management in Networked HPC Systems, 1998.

[11] B. Li. OSCAR: Open Source Cluster Application Resources, 2005. http://oscar.openclustergroup.org.

[12] R. Magrin, A. Santos, R. Ávila, and P. Navaux. Gerenciamento de agregados openpbs x ccs. *Escola Regional de Alto Desempenho (ERAD)*, 2003.

[13] M.L. Massie and B.N. Chun and D.E. Culler. The Ganglia Distributed Monitoring System: Design, Implementation, and Experience. *Parallel Computing*, 30(7), July 2004.

[14] M. A. S. Netto and C. A. F. D. Rose. Crono: a configurable management system for linux clusters. In *The Third LCI International conference on linux clusters: the hpc revolution*, 2002.

[15] L. Networx. Clusterworx, 2005.

[16] J. O'Kane. Kickstart. *Sys Admin: The Journal for UNIX Systems Administrators*, 9(1):33–34, 36, Jan. 2000.

[17] OpenPBS.org. The Portable Batch System, 2003. http://www.openpbs.org.

[18] OpenSCE Project. OpenSCE. Open Scalable Cluster Environment, May 2005. http://www.opensce.org.

[19] M. Pasin and D. L. Kreutz. Arquitetura e administração de aglomerados. In *Terceira Escola Regional de Alto Desempenho*, Santa Maria, 2003. Sociedade Brasileira de Computação - UNISINOS / UFSM / UNILASSALE.

[20] Philippe Augerat and Wilfrid Billot and Simon Derr and Cyrille Martin. A scalable file distribution and operating system installation toolkit for clusters, 2003. http://katools.sourceforge.net/publications/file-distribution.pdf.

[21] Putchong Uthayopas and Arnon Rungsawang. SCMS: An Extensible Cluster Management Tool for Beowulf Cluster. In *Proceedings of Supercomputing'99 (CD-ROM)*, Portland, OR, nov 1999. ACM SIGARCH and IEEE. Department of Computer Engineering, Kasetsart University.

[22] Rembo Technology. Rembo Toolkit, 2005. http://www.rembo.com.

[23] Rocks Cluster Distribution. ROCKS, May 2005. http://www.rocksclusters.org.

[24] C. M. Tan, C. P. Tan, and W. F. Wong. Shell over a cluster (SHOC): Towards achieving single system image via the shell, Sept. 30 2002.