

Comparação entre Ferramentas de Monitoração de *Clusters**

Tiago Scheid, Marcelo Veiga Neves,
Andrea Schwertner Charão

Laboratório de Sistemas de Computação - LSC
Curso de Ciência da Computação – Universidade Federal de Santa Maria - UFSM
Informática/CT - UFSM Campus - 97105-900, Santa Maria, RS
{scheid, veiga, andrea}@inf.ufsm.br

Introdução

Atualmente há uma ampla utilização de aglomerados de computadores (*clusters*) no processamento de alto desempenho. Para facilitar o gerenciamento de um *cluster*, é geralmente necessário monitorar a utilização de seus recursos. A monitoração consiste em observar o estado do *cluster* em um dado momento ou em momentos passados, através de um histórico. Hoje em dia existem várias ferramentas para monitorar-se um *cluster*, permitindo obter as mais diferentes informações e visualizá-las de diversas maneiras. Nesse contexto, este trabalho busca analisar algumas ferramentas de maneira aprofundada e, finalmente, traçar um comparativo entre elas.

Ferramentas de Monitoração

Para a realização deste trabalho, escolheu-se algumas ferramentas representativas do estado-da-arte em monitoração de *clusters*. As ferramentas escolhidas foram Ganglia [MAS 2003], Parmon [BUY 2000], SCMS [UTH 99] e RVision[FER 2002]. Estas ferramentas foram instaladas, configuradas e analisadas. Segue uma descrição de cada ferramenta e na próxima seção a comparação entre elas.

Ganglia

Ganglia [MAS 2003] é um sistema de monitoração distribuído e escalável desenvolvido para monitorar *clusters* e *grids*. Este sistema é baseado em uma arquitetura hierárquica focada em federações de *clusters*. A monitoração de *clusters* é feita de maneira distribuída, na qual cada nó de monitoração (*gmond*) possui uma cópia do estado atual do *cluster*. Também há um módulo centralizador (*gmetad*) que solicita, ciclicamente, os dados de cada nó monitorado, a fim de manter o histórico dos estados do *cluster*.

A monitoração com Ganglia se dá através de dois tipos de métricas: as *built-in* e as definidas pelo usuário. O primeiro tipo é formado pelas métricas coletadas pelo próprio

*Este trabalho recebeu apoio do Conselho Nacional de Desenvolvimento Científico e Tecnológico - CNPq e da Fundação de Amparo à Pesquisa do Estado do Rio Grande do Sul - FAPERGS

sistema, como por exemplo porcentagem de CPU, carga média, uso de memória, rede, etc. Já as métricas definidas pelo usuário são informações colhidas por uma outra aplicação e publicadas através do Ganglia. Assim, é possível usar Ganglia para monitorar qualquer informação que possa ser coletada. Para armazenamento dos dados monitorados por *gmond*, Ganglia (*gmetad*) utiliza o sistema RRDtool (*Round Robin Database*). RRDtool é um sistema que permite o armazenamento de seqüências temporais de dados em um banco de dados circular de forma compacta.

Ganglia possui uma interface Web onde é possível visualizar graficamente os dados armazenados por RRDTool, incluindo as métricas monitoradas em todos nós ativos e inativos do *cluster*. Além dessa interface, Ganglia também possui uma biblioteca (*lib-ganglia*) que auxilia na criação de clientes, facilitando a sua adaptação às necessidades do administrador do *cluster*. A monitoração é facilmente gerenciável, sendo que, para adicionar algum nó ao *cluster* monitorado, basta executar *gmond* no novo nó, que este passará automaticamente a ser monitorado e agregado ao *cluster*. Ganglia tem RRDTool como um pré-requisito para a instalação.

Parmon

Parmon [BUY 2000] é uma ferramenta comercial desenvolvida com o propósito de monitorar *clusters*. Essa ferramenta possui uma arquitetura centralizada e é dividida em duas partes: servidor (*parmond*) e cliente. O servidor é responsável por monitorar o nó e já vem compilado para a arquitetura da máquina. Já o cliente, por ser em Java, pode executar em qualquer computador, independente de arquitetura, mas que pertença a mesma rede.

Parmon permite agregar informações dos recursos do sistema de vários nós, acompanhar processos, acompanhar *logs* do sistema, definir eventos que se atingidos disparam e-mail (*trigger*) ao administrador do *cluster*. Parmon permite monitorar CPU, memória, rede e disco, bem como executar alguns comandos paralelos. A centralização de todos os dados monitorados ocorre no cliente, o qual converte os dados e os mostra de forma gráfica e *on-line*, podendo também apresentar algumas informações de configuração do sistema de forma textual.

A monitoração se dá por demanda cíclica, ou seja, no momento em que o cliente faz uma requisição, ele envia a métrica a ser monitorada e o tamanho do período, que pode ser ajustado. O servidor então começa a monitorar a métrica até que o cliente deixe de requerer a métrica. Essas informações são apresentadas graficamente em forma de linha, barras ou torta. Parmon não é extensível, pois o servidor atende somente a requisições do cliente, não sendo possível a utilização de seus dados por outro programa. Além disso, Parmon não permite monitorar outras métricas. Para a instalação de Parmon é necessário a distribuição de GNU/Linux RedHat 8.0, ou superior, ou Solaris e Java 1.4.2, para rodar o cliente. Já o servidor pode executar em qualquer distribuição de GNU/Linux ou Solaris.

Neste trabalho, utilizou-se uma versão de demonstração de Parmon, por isso algumas de suas principais características (monitorar processos, CPU de usuário e outras) não puderam ser comprovadas.

SCMS

SCMS [UTH 99] foi desenvolvido para monitorar *clusters* de pequeno e médio porte de forma simples, robusta e eficiente. Para tanto, SCMS possui uma arquitetura centralizada organizada em dois módulos: um módulo de monitoração e um módulo de centralização, o qual atende a requisições de um cliente.

Essa ferramenta permite monitorar 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 configuração do sistema. Os dados monitorados são armazenados pelo módulo centralizador em um repositório próprio.

A apresentação dos dados monitorado é na forma de gráficos no cliente. Esse cliente permite selecionar as métricas e os nós que devem ser monitorados, desde que estes estejam listados num arquivo de configuração da ferramenta. SCMS disponibiliza uma API com rotinas em C, TCL/TK e Java, para permitir que outros programas utilizem os dados monitorados. Para instalar SCMS é necessário ter Python versão 2.1 ou superior e algumas bibliotecas e ferramentas de base.

RVision

RVision [FER 2002] é uma ferramenta de monitoração desenvolvida para que possa ser adaptada a diferentes *clusters*, com uma arquitetura aberta e configurável. Para manter essas características, RVision possui uma interface para a comunicação de clientes com o núcleo da ferramenta (*rvcore*). A arquitetura de RVision é centralizada, sendo composta um de programa monitor (*rvspy*) e um programa centralizador (*rvcore*).

RVision é capaz de monitorar o sistema através de seu programa monitor (*rvspy*) ou de um agente SNMP. A fonte de dados pode ser selecionada através de um arquivo de configuração de seção, onde também pode-se selecionar o tipo de análise (*on-line* ou *post-mortem*), tipo de obtenção dos dados (demanda, cíclico ou por alteração, ou seja, o módulo monitor só envia quando uma condição for satisfeita), nós a serem monitorados e outras configurações.

A forma de visualização em RVision é variável, dependente do cliente implementado. Em caso de monitoração *post-mortem*, o núcleo é responsável por armazenar os dados em arquivos. Para a instalação do RVision precisa-se apenas de *gcc* e *make* instalados.

Comparação

O estudo das ferramentas foi feito através da leitura da documentação existente, da instalação e configuração das ferramentas, seguida da experimentação e análise das ferramentas. A comparação das ferramentas é baseada nos seguintes critérios: arquitetura, visualização, análise, recolhimento, extensibilidade, gerenciabilidade e pré-requisitos de instalação. A Tabela 1 mostra uma comparação das ferramentas testadas à luz dos vários critérios considerados.

Tabela 1 - Comparação entre diferentes ferramentas de monitoração

	Ganglia	Parmon	SCMS	RVision
Arquitetura	distribuída	centralizada	centralizada	centralizada
Visualização	on-line e post-mortem	on-line	on-line e post-mortem	on-line e post-mortem
Apresentação	gráfica e textual	gráfica e textual	gráfica e textual	depende do cliente
Forma de Coleta	cíclica	demanda cíclica	cíclica	cíclica, demanda ou alteração
Extensibilidade	possui	não possui	possui	possui
Gerenciabilidade	automática	interface	manual	manual

Conclusão

Neste trabalho apresentou-se uma comparação entre as ferramentas de monitoração Ganglia, Parmon, SCMS e RVision. Através desta, comparação pôde-se observar algumas particularidades relevantes. Em Ganglia, observou-se que cada núcleo de monitoração possui o estado do *cluster*, o que garante uma tolerância à falhas no sentido em que pode-se obter os dados de qualquer nó. Em Ganglia e RVision observou-se a possibilidade de adicionar novas métricas à monitoração, o que confere uma grande extensibilidade a estas ferramentas. Em SCMS percebeu-se a existência de API em algumas linguagens, o que facilita a utilização dos dados monitorados por outros programas. RVision permite o recolhimento por alteração, o que diminui o tráfego na rede. Em RVision notou-se que não existe um cliente funcional e intuitivo disponível, o que pode ser uma desvantagem à sua utilização. Em Parmon percebeu-se a exigência da distribuição RedHat de GNU/Linux como um pré-requisito, o que diminui a portabilidade da ferramenta.

Esta comparação pode ser útil para os administradores de *clusters*, pois diminui o tempo de procura pela ferramenta mais adequada ao seu uso. Também facilita a escolha de uma ferramenta, pois apresenta alguns dados provenientes da instalação e configuração das ferramentas, o que, muitas vezes, não é encontrado em artigos e manuais.

Referências

- [BUY 2000] BUYYA, R. PARMON: a portable and scalable monitoring system for clusters. **Software Practice and Experience**, v.30, n.7, p.723–739, June 2000.
- [FER 2002] FERRETO, T. C.; ROSE, C. A. F. de; ROSE, L. de. Rvision: an open and high configurable tool for cluster monitoring. **CCGRID'02**, p.75, 2002.
- [MAS 2003] MASSIE, M.; CHUN, B.; CULLER, D. **The ganglia distributed monitoring system: design, implementation, and experience**. [S.l.]: University of California, Berkeley Technical Report, 2003.
- [UTH 99] UTHAYOPAS, P.; RUNGSAWANG, A. SCMS: an extensible cluster management tool for beowulf cluster. In: SUPERCOMPUTING'99 (CD-ROM), 1999, Portland, OR. **Proceedings...** ACM SIGARCH and IEEE, 1999.