

# Extensão de Ganglia para Monitoração de Programas Paralelos\*

Marcelo Veiga Neves, Tiago Scheid,  
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  
{veiga,scheid,andrea}@inf.ufsm.br

## Introdução

Os aglomerados (*clusters*) de computadores estão entre as arquiteturas paralelas mais utilizadas atualmente, pois podem prover um alto poder computacional com uma boa relação custo-desempenho. A fim de facilitar o gerenciamento de *clusters*, e também das aplicações que neles são executadas, é comum a utilização de ferramentas de monitoração.

A monitoração de *clusters* implica na coleta e apresentação de informações que representem o estado do aglomerado ao longo do tempo. Informações de monitoração podem ser úteis para administradores, programadores e usuários finais. De fato, um administrador pode analisar o estado do *cluster* a fim de localizar possíveis falhas ou gargalos no sistema. Já programadores podem usar as informações de monitoração para identificar e corrigir eventuais problemas de lógica e desempenho em seus programas. Além disso, os usuários podem acompanhar a execução de suas aplicação observando a utilização dos recursos alocados.

Atualmente, existem diversas ferramentas de monitoração de *clusters*. Dentre as mais difundidas pode-se citar Ganglia [MAS 2003], Parmon [BUY 2000] e SCMS/RMS [UTH 99]. Nota-se, no entanto, uma carência de ferramentas que monitorem recursos em diferentes níveis de abstração, servindo desde aos usuários até aos administradores de um *cluster*.

Este trabalho se propõe a acrescentar um suporte à monitoração de aplicações paralelas à ferramenta Ganglia. Com a extensão desta ferramenta, permite-se que a mesma seja utilizada para um acompanhamento *on-line* da execução de uma aplicação, auxiliando não só na administração de um *cluster*, mas também nas atividades desenvolvidas por seus usuários e programadores.

## A Ferramenta Ganglia

Ganglia [MAS 2003] é um sistema de monitoramento distribuído e escalável para sistemas de computação de alto desempenho, como *clusters* e *grids*. O sistema é baseado

---

\*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

em uma arquitetura hierárquica focada em federações de *clusters*. A implementação de Ganglia é robusta, sendo que a ferramenta já foi portada para vários sistemas operacionais e arquiteturas.

A implementação de Ganglia consiste em dois tipos de processos (*gmond* e *gmetad*), um programa de linha de comando (*gmetric*) e uma biblioteca para programação de clientes. O processo *gmond* (*Ganglia Monitoring Daemon*) é responsável pela monitoração de diferentes métricas de um *cluster*. Este processo utiliza um protocolo baseado em *multicast* para difusão dos dados e responde requisições de dados para clientes, que podem ser tanto um sistema de visualização ou uma aplicação que utilize os dados de monitoração. O processo *gmetad* (*Ganglia Meta Daemon*) é responsável pela centralização hierárquica dos dados de múltiplos *clusters*. O programa *gmetric*, por sua vez, permite o envio de dados diretamente ao *multicast*, possibilitando a publicação de métricas definidas pelo usuário.

Ganglia monitora dois tipos de métricas: as *built-in* e as definidas pelo usuário. O primeiro tipo é formado pelas métricas coletadas pela ferramenta, como por exemplo porcentagem de CPU, carga média, uso de memória, rede e outras informações relativas ao *cluster*. Já as métricas definidas pelo usuário são informações coletadas por uma aplicação externa e publicadas através de Ganglia. Assim, é possível utilizar Ganglia para monitorar qualquer informação que possa ser coletada.

O fato de Ganglia permitir a definição de novas métricas, aliado à independência em relação aos clientes confere um caráter extensível à ferramenta. Isso facilita a modificação ou integração de Ganglia a outros sistemas. Outra característica importante de Ganglia é sua escalabilidade, garantida por um baixo *overhead* no monitoramento. Para isso, Ganglia utiliza *thresholds* altos e intervalos aleatórios para coleta e difusão dos dados.

Para armazenamento e visualização dos dados monitorados, Ganglia utiliza o sistema RRDtool (*Round Robin Database*). Este sistema permite o armazenamento de seqüências temporais de dados de forma compacta em um banco de dados circular, de tamanho constante. A partir dos dados armazenados, RRDtool gera gráficos que mostram a evolução de uma ou mais métricas ao longo do tempo.

Ganglia também possui uma interface Web para visualização das métricas monitoradas através de gráficos gerados com RRDtool. Nessa interface é possível fazer a seleção de métricas, nós e granularidade de tempo dos gráficos.

## Implementação do Suporte à Monitoração de Aplicações

Primeiramente, foram implementadas rotinas para coleta de informações relativas a um dado processo, como a quantidade de recursos utilizados. Essas rotinas foram usadas na construção de um programa monitor que é lançado em todos os nós do *cluster*, juntamente com a aplicação paralela. Cada monitor localiza os processos da aplicação no nó local e acompanha sua execução, coletando métricas e enviando-as diretamente ao *multicast* de Ganglia, através do programa *gmetric*.

Na monitoração de programas paralelos, o intervalo de coleta de dados é um ponto crítico, uma vez que a dinamicidade das aplicações requer um acompanhamento com uma pequena granularidade de tempo. Para tornar a visualização mais significativa, foi necessário eliminar a aleatoriedade e reduzir o intervalo de publicação de Ganglia. Para

manter a escalabilidade, que é uma das principais características da ferramenta, optou-se por reduzir apenas os intervalos de publicação das métricas definidas pelo usuário, já que o intervalo reduzido só é necessário enquanto a aplicação estiver em execução.

Quando uma nova métrica é publicada no *multicast*, o centralizador *gmetad* cria uma base de dados para armazenar as novas informações de monitoração. A visualização dessas informações é feita através de gráficos criados com RRDtool. Para facilitar a visualização foi implementada uma extensão da interface Web de Ganglia. Essa extensão agrupa as informações de uma mesma aplicação e permite a seleção do intervalo de tempo a ser visualizado.

## Avaliação da Extensão de Ganglia

A fim de avaliar o trabalho desenvolvido, utilizou-se uma aplicação que resolve as equações de Navier-Stokes em um caso clássico na área de mecânica dos fluidos (problema da cavidade), utilizando um método numérico *multigrid*. Esta aplicação é parte do pacote PETSc[BAL 2001], que é uma ferramenta amplamente utilizada no desenvolvimento de aplicações paralelas para solução de problemas modelados por equações diferenciais parciais.

O pacote PETSc utiliza a biblioteca MPI para comunicação entre processadores. Para execução da aplicação, utilizou-se um *cluster* homogêneo composto por máquinas mono-processadas Intel Pentium 4 a 2.40 GHz, com 512 Mbytes de memória RAM, 512 Kbytes de memória *cache* e adaptador de rede Gigabit Ethernet. O sistema operacional de cada máquina é GNU/Linux (distribuição Gentoo 1.4), com *kernel* versão 2.6.5.

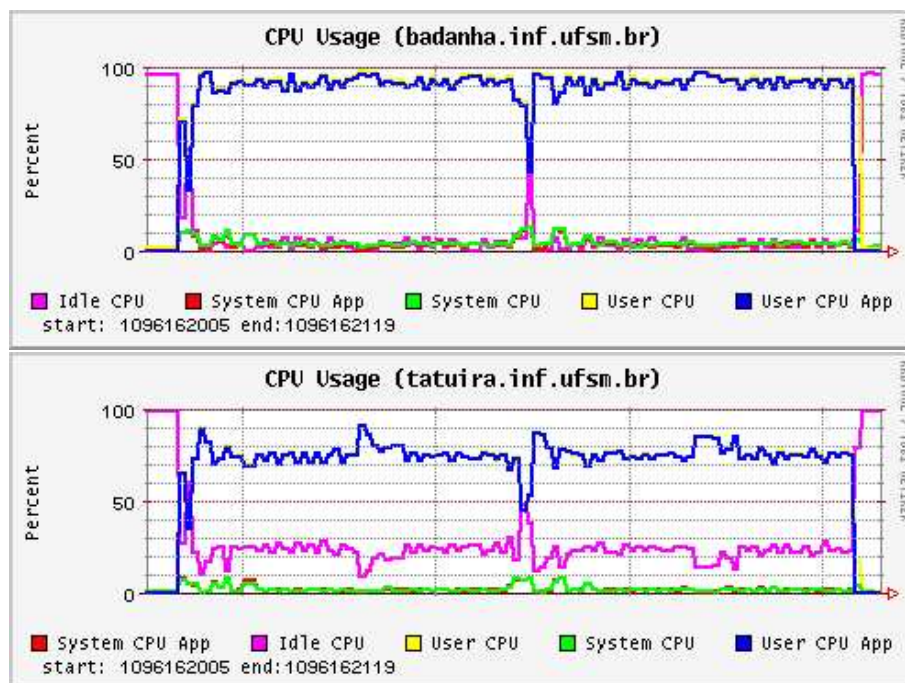


Figura 1: Monitoração do uso de CPU por uma aplicação.

Os gráficos da figura 1 apresentam informações sobre uma execução da aplicação em 2 nós do *cluster*, ao longo de aproximadamente 2 minutos. Nestes gráficos, a métrica monitorada é o uso da CPU em cada nó. No primeiro gráfico, nota-se um alto percentual de utilização da CPU (User CPU App), enquanto que no segundo observa-se uma sub-utilização de recursos, caracterizada por uma maior ociosidade da CPU (Idle CPU). Este comportamento evidencia um desequilíbrio de carga nesta execução da aplicação. É importante ressaltar que a monitoração dos tempos de CPU alocados à aplicação não seria possível com Ganglia sem a extensão implementada.

## Conclusão

A ferramenta Ganglia é uma das opções atualmente disponíveis para a monitoração de *clusters*. Neste trabalho, incorporou-se à esta ferramenta a habilidade de monitorar uma dada aplicação ao longo de sua execução, através da coleta de informações sobre os processos que a compõem. Este recurso foi utilizado para monitorar uma aplicação de computação científica, servindo para detectar um desequilíbrio de carga em suas execuções paralelas. Os resultados obtidos mostram que esta extensão de Ganglia pode ser de grande utilidade para a análise do desempenho de aplicações paralelas, complementando os recursos de Ganglia voltados à administração de *clusters*.

## Referências

- [BAL 2001] BALAY, S. et al. **PETSc Web page**. <http://www.mcs.anl.gov/petsc>.
- [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.
- [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. Department of Computer Engineering, Kasetsart University.