

Profamaçãõ - 330  
Profamaçãõ paralela  
Depuraçãõ: Software

# Depuração de Programas Paralelos e Distribuídos

enPq 1.03.03.00-6

Mairo Pedrini<sup>1</sup>

Philippe Olivier Alexandre Navaux<sup>2</sup>

388775

## 1 Introdução

Devido ao pouco desenvolvimento de ferramentas de depuração para ambientes paralelos e distribuídos, (STRINGHINI, 2002) propôs um modelo para uma interface gráfica para depuração, que incluísse pontos considerados importantes pela autora.

O projeto previa a criação de uma ferramenta para depuração *online*, ou seja, o mesmo modelo usado em depuradores sequenciais comuns. Tal modelo não auxilia na detecção de erros inerentes ao paralelismo, como *deadlocks* ou *condições de corrida* mas, segundo (BUHR; KARSTEN; SHIH, 1996), tais erros, apesar de mais difíceis de detectar e corrigir, ocorrem com menor frequência que erros comuns.

## 2 O Protótipo

Partindo do projeto citado acima, foi criado um primeiro protótipo da ferramenta, com o objetivo de validar os mecanismos propostos e a interface.

O protótipo implementa os mecanismos de seleção e visualização, possibilitando que um número grande de processos possa ser visualizado, juntamente com um mecanismo de agrupamento de processos que, no protótipo, consistia em agrupamento do usuário, agrupamento por estado do processo ou agrupamento por nome do executável.

O projeto da interface previa dois níveis de abstração: um nível responsável pela aplicação, e outro responsável pelos processos individualmente. Na interface, tais níveis foram implementados como *Main View* (nível de aplicação) e *Process View* (nível de processos).

O protótipo foi implementado em Java, por simplificar grandemente a tarefa de criar interfaces gráficas, e o controle dos processos foi feito utilizando-se uma infraestrutura pronta, o Fiddle, um conjunto de serviços para controle de aplicações distribuídas.

<sup>1</sup>mpedrini@inf.ufrgs.br Bolsista CNPq

<sup>2</sup>navaux@inf.ufrgs.br

### 3 Avanços Alcançados

Dando continuidade ao desenvolvimento da ferramenta, os primeiros passos consistiram basicamente em atualizar a PADI para utilizar uma versão mais recente do Fiddle e, assim, eliminar problemas com alguns dos bugs existentes na versão utilizada.

Entretanto, a versão atualizada do Fiddle continuava a apresentar um bug, no qual era impossível ter dois processos executando ao mesmo tempo. Devido ao interesse no bom funcionamento do Fiddle, foi iniciado um trabalho, em cooperação com os autores do mesmo e, por fim, o problema foi resolvido, juntamente com outros problemas menores que não haviam sido detectados devido ao anterior.

Com a resolução deste problema, foi possível retomar o desenvolvimento da PADI, indo desde a correção de *bugs* até melhorias na interface (especialmente na *Process View*). Atualmente a PADI é capaz de depurar programas MPI e DECK (BARRETO, 2000), mas não possui suporte à threads.

### 4 Continuação dos trabalhos

A continuação do desenvolvimento inclui melhorias na interface, especificamente na navegação pelo código-fonte e visualização de variáveis, e também o suporte a múltiplas *threads*, aumentando grandemente suas possibilidades de uso.

No que se refere a visualização de variáveis, estão sendo feitos alguns testes na direção de implementar um mecanismo gráfico, semelhante ao existente no GNU *Data Display Debugger* (ZELLER, 2001), para variáveis locais aos processos.

### Referências

BARRETO, M. E.; NAVAU, P.; RIVIERE, M. *DECK: um ambiente para programação paralela em agregados de multiprocessadores*. Porto Alegre: [s.n.], 2000.

BUHR, P. A.; KARSTEN, M.; SHIH, J. Kdb: A multi-threaded debugger for multi-threaded applications. In: *SYMPOSIUM ON PARALLEL AND DISTRIBUTED TOOLS*. [S.l.: s.n.], 1996. p. 80–87.

STRINGHINI, D. *Depuração de Programas Paralelos - Projeto de uma interface intuitiva*. Tese (Doutorado) — Instituto de Informática, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2002.

ZELLER, A. Visual debugging with ddd. *Dr. Dobb's Journal*, Março 2001. Disponível em [www.ddj.com/articles/2001/0103](http://www.ddj.com/articles/2001/0103).