

Refatoração de Programas Fortran de Alto Desempenho

Bruno Batista Boniati¹, Andrea Schwertner Charão¹

¹Programa de Pós-Graduação em Informática (PPGI)
Universidade Federal de Santa Maria (UFSM)

bruno@unijui.edu.br, andrea@inf.ufsm.br

1. Introdução

A computação de alto desempenho (*High Performance Computing* - HPC) é uma área da ciência da computação que se preocupa com o emprego de metodologias e técnicas para a execução eficiente de aplicações. Isso abrange técnicas de programação, melhorias de código e distribuição ou paralelização de tarefas [Neely 2004]. A escolha de uma linguagem de programação é importante para garantir desempenho às aplicações.

Fortran foi a primeira linguagem de programação baseada em procedimentos. Sua principal utilização é na computação científica e análise numérica. Dispõe de bibliotecas matemáticas com alto grau de otimização. Considerando a HPC, Fortran é muito utilizado em função da existência de operações nativas para tratamento de vetores [Koffmann and Friedman 1996] ao mesmo tempo em que é compatível com bibliotecas para computação distribuída como MPI (*Message Passing Interface*), por exemplo. Boa parte código legado de HPC é escrito em linguagem Fortran [Overbey et al. 2005]. Em função de sua baixa utilização para fins não científicos, não dispõe de muitos ambientes para desenvolvimento, além de que, os poucos que existem possuem limitações.

Recentemente, um grupo de pesquisadores ligados à Universidade de Illinois (Urbana-Champaign) e ao Laboratório Nacional de Los Alamos (ambos nos EUA) vêm dedicando esforços para conceber um ambiente de desenvolvimento que, além de funções básicas (identificação de sintaxe, sugestão de código, etc.), forneça ferramentas avançadas, como a refatoração de código. Neste trabalho, apresenta-se o estado atual de tal ferramenta, bem como perspectivas acerca da aplicação de técnicas de refatoração na programação para HPC, em especial aplicáveis a código Fortran. Este estudo servirá de apoio à elaboração de uma proposta de dissertação de mestrado nesta área de pesquisa.

2. Refatoração

Refatoração (*refactoring*) é uma técnica de engenharia de software, que consiste basicamente em aplicar modificações internas no código do software, sem que isso afete seu comportamento [Fowler et al. 1999]. Geralmente são modificações que passam despercebidas ao usuário final, mas que representam otimizações na estrutura interna do programa fonte. O processo de refatoração em si pode ser executado de forma manual, mas o ganho de qualidade se dá quando aliamos ferramentas de software que detectam potencialidades no código-fonte que possam ser modificadas e efetuam tais alterações de forma automatizada. Algumas ferramentas de desenvolvimento já possuem em sua IDE (*Integrated Development Environment*) funcionalidades que aplicam a refatoração ao código ou sugerem trechos de código candidatos à refatoração.

3. Photran

Photran é um *plugin* para Eclipse, voltado ao desenvolvimento de aplicativos Fortran, com suporte às versões 77, 90, e 95 [Overbey et al. 2005]. Em geral, apresenta todas as características básicas de um editor visual avançado focado no apoio à codificação. É desenvolvido pela Universidade de Illinois de Urbana-Champaign, mantido sob licença GPL, e foi baseado em outro *plugin* para Eclipse, o CDT (*C/C++ Development Tooling*).

Desde a versão atual (4.0 beta 3), uma das funcionalidades mais anunciadas e esperadas foi disponibilizada: a refatoração de código. A relação de refatorações disponibilizadas ainda é pequena (apenas duas funcionalidades), mas a promessa e a indicação de novas pesquisas são motivadoras para a área de HPC.

4. Refatoração de Programas Fortran de Alto Desempenho

É comum encontrar pesquisas relacionadas à refatoração de código Java ou outras linguagens de ponta. Inclusive algumas, como .NET, já apresentam essas características desde sua concepção. Mas é importante lembrar que a maior parte do código legado, principalmente de HPC não está escrito nessas linguagens. Considerando que técnicas de refatoração são aplicadas principalmente visando aumentar o tempo de vida do código, reestruturando-o, essa é uma área ainda pouco explorada dentro da HPC. Trabalhos como [Neely 2004] e [Skjellum et al. 2004] destacam um cenário promissor de tal técnica e demonstram alguns aplicações na computação de alto desempenho.

O Photran traz duas técnicas já implementadas (*rename* e *introduce implicit none*) e faz a indicação de outras que estão em desenvolvimento [Overbey et al. 2005]. Embora tais funcionalidades ainda sejam poucas, elas representam uma iniciativa até então pouco explorada e muito importante para a HPC. A degradação natural pelo qual o código fonte de aplicações legadas sofre com o tempo, poderia ser amenizada aplicando-se técnicas de refatoração. Neste sentido, ações como essa podem representar detalhes importantes na tentativa de agregar melhoras na execução de sistemas computacionais de alto desempenho.

Referências

- Fowler, M., Beck, K., Brant, J., Opdyke, W., and Roberts, D. (1999). *Refactoring: Improving the Design of Existing Code*. Addison-Wesley, 1 edition.
- Koffmann, E. B. and Friedman, F. L. (1996). *Fortran*. Addison-Wesley, 5 edition.
- Neely, R. (2004). Pactical software quality engineering on a large multi-disciplinary hpc development team. In *International Workshop on Software Engineering for High Performance Computing System Applications*.
- Overbey, J., Xanthos, S., Johnson, R., and Foote, B. (2005). Pactical software quality engineering on a large multi-disciplinary hpc development team. In *Second International Workshop on Software Engineering for High Performance Computing System Applications*.
- Skjellum, A., Bangalore, P., Gray, J., and Bryant, B. (2004). Reinventing explicit parallel programming for improved engineering of high performance computing software. In *26th International Conference on Software Engineering*.