An Evolutionary Approach to Construction of a Software Development Environment for Massively-Parallel Heterogeneous Systems

Hiroyuki TAKIZAWA, Daisuke TAHAKASHI, Reiji SUDA, and Ryusuke EGAWA
1 Tohoku University  2 University of Tsukuba  3 The university of Tokyo  4 JST CREST

How HPC system architectures may change revolutionarily, HPC applications can change only evolutionarily.

The goal of this project is to develop an extensible framework to help software evolution by achieving both high performance and high performance portability. At present, HPC applications are often optimized for their particular target systems, and hence unable to run efficiently on other systems. System-specific optimizations are essential to the performance of modern HPC systems. Thus the programming framework is designed so that it can separate those system-specific optimizations from application codes. If system-specific optimizations are written separately in external files, we can use different optimizations for different systems, resulting in high performance and high performance portability. To this end, the implementation details are hidden to application programmers but visible to expert performance tuners. System-specific optimizations are abstracted in various ways such as libraries and domain-specific tools. As a result, system-specific optimizations are hierarchically abstracted and hence their abstractions are provided at several abstraction levels.

Separation of system-specific optimizations from HPC application codes.

For an application to fully exploit the potential of a particular system, there is no silver-bullet to completely hide everything specific to the system. So there are quite a huge number of approaches for abstraction of system-specific optimizations. Yet, more or less, some code modifications are still required for various system-specific reasons. For example, abstractions such as numerical libraries are not necessarily transparent to an application code, and hence the code might need to be modified so as to use the abstractions. Such code modification is likely to be specific to a particular environment and/or a particular application. So they could severely degrade the readability, maintainability and performance portability of the application. This means that we need another abstraction layer to express such special code modifications.

In this project, we are developing an extensible programming framework, Xevolver, which enables users to define their own code transformations for special demands of individual systems and individual applications. Instead of conventionally-required code modifications, the users can use Xevolver to optimize and specialize an application code for a particular system. In the Xevolver framework, translation rules can be defined in an external file. Thus, Xevolver separates system-specific and/or application-specific code optimizations from application codes. We are also developing numerical libraries and domain-specific tools with Xevolver to help enhance performance portability of HPC applications.

What we are exploring in this project.

A proof-of-concept implementation of the Xevolver framework was developed as a component of the ROSE compiler infrastructure.
- Interconnection between ROSE Sage II ASTs and XML ASTs.
- As a result, XSLT is employed to describe translation recipes.
- 100+ translations are available in a human-readable, reusable format.
- In this framework, other ROSE-related technologies are also available for translations, analysis, and customization of ASTs.

Hierarchical abstractions of post-Peta scale systems

Communication avoiding Arnoldi method for numerically stable Krylov basis generation

We also investigate how to refactor a legacy application code for the separation.

— Guidelines and tools for refactoring HPC codes to support easy and smooth adaptation to future systems —

Examples of refactoring include:
- Code Characteristics
- Aim of optimization
- Plat form
- Process of Optimization
- Effects of optimizations
- Executable codes (Kernels)

E-mail: xevolver@kev.arch.is.tohoku.ac.jp