

NASA SBIR/STTR Technologies

Fortran Testing and Refactoring Infrastructure

PI: Stefan Muszala / Tech-X Corp. – Boulder, CO

Proposal No: S6.05-8885



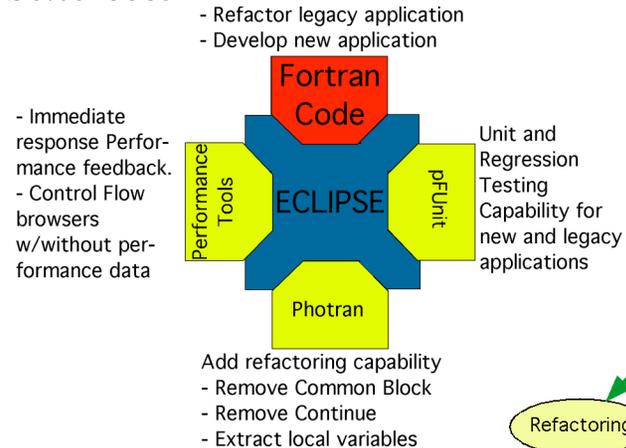
Identification and Significance of Innovation

Identification:

- Unit and regression testing are proven methods to develop robust, portable and extensible codes.
- Refactoring legacy codes is a difficult problem that requires modern tools used often for C++ and Java but have not been readily adopted by the Fortran community.
- Testing, refactoring and performance are used in concert to develop stable and usable codes.
- Testing and refactoring within an Integrated Developer Environment (IDE) may additionally enhance Fortran code development.

Significance:

- Providing a testing and refactoring infrastructure usable from within an Integrated Developer Environment can greatly increase developer efficiency and code robustness.
- The combination of testing and refactoring with immediate performance feedback increases productivity and code efficiency for developers in many programming languages.



The proposed innovation allows Fortran developers to accurately test and refactor codes with an emphasis on single- and multi-processor performance

Technical Objectives

- Provide extensive unit and regression testing and refactoring capability within an Integrated Developer Environment (IDE) for Fortran developers (Phases 1 and 2).
 - Demonstrate the feasibility of extending pFUnit capability to satisfy testing functionality (Phase 1).
 - Demonstrate the feasibility of adding refactoring capability to Photran (Phase 1).
- Show complete Fortran IDE capability of testing and refactoring with integrated performance measurement, feedback and visualization (Phase 2).

Work Plan

- Extend pFUnit functionality to legacy applications by implementing a prototype parser that will scan procedure and function signatures.
- Implement a prototype graphical user interface (GUI) for pFUnit as an Eclipse plugin.
- Add additional refactoring capability to Photran and test key prototype examples.

Phase 1 development work will use the NASA GISS code modelE as a test case and we will work closely with NCAR CCSM/CAM developers.

NASA Applications

- NASA MAP and HEC provide support to and use many Fortran applications (modelE, GEOS-4 and 5).
- The Fortran applications must be continually tested and refactored as new algorithms and ideas are implemented; as computer architectures advance, there is a greater emphasis on speeding up application performance.

Non-NASA Applications

- Fortran developers outside of NASA have many similar concerns and many Fortran codes are in use for computational fluid dynamics, climate and weather modeling (CCSM/CAM, WRF), combustion and geophysics.
- Providing testing and refactoring support to on the level of C++ and Java utility will be highly advantageous to the Fortran community.

Firm Contacts

Tech-X Corporation
5621 Arapahoe Avenue, Suite A
Boulder, CO 80303

www.txcorp.com
info@txcorp.com
(303) 448 - 0727