Computer Assisted Problem Solving Environment (PSE)

C

Shigeo Kawata
Utsunomiya University, Japan

INTRODUCTION

Problem Solving Environment (PSE) concept provides integrated human-friendly innovative computational services and facilities for easy incorporation of novel solution methods to solve a target class of problems. For example, a PSE generates a computer program automatically to solve differential equations (Boonmee et al., 1998a; Boonmee et al., 1998b; Fujio et al., 1998; Fujita et al., 2000; Gallopoulos, et al., 1991; Gallopoulos et al., 1994; Hirayama et al., 1988; Houstis et al., 1992; Kawata et al., 2000; Okochi et al., 1994; Umetani, 1985; Rice et al., 1984). PSE is an innovative concept to enrich science, human life, engineering, production and our society toward a programming-free environment in computing science. In the PSE concept, human concentrates on target problems themselves and works on solutions, and a part of application of solution, which can be solved mechanically, is performed by computers or machines or software.

Even today human power still contributes greatly to develop and write a new software. For example, in scientific researches scientific discoveries would be supported by theory, experiments and computer simulations. New researches tend to require new computer programs to simulate phenomena concerned. In another example, in developing new products engineers would also need new computer programs to develop the products cost-effectively. They may have to develop the new programs or learn how to use the programs for the product development. So researchers and engineers may write or develop the new computer programs or learn how to use the programs by themselves. They like to devote themselves to solve their target problems, but not to develop or learn the computer programs.

In addition, computer simulations became the third important method after theoretical and experimental methods in science and engineering. Computer assisted problem solving is one of key methods to promote innovations in science and engineering, and contributes to enrich our society and our life from scientific and engineering sides. PSE research activities had started to enhance the programming power in 1970's.

The PSE-relating studies were started in 1970's to provide a higher-level programming language than Fortran, COBOL, ALGOL, PL/I and others in scientific computations. The trend at the time was natural to deliver more human-friendly programming environment beyond the higher-level languages shown above. Then the PSE research activity was started as well as Computer Assisted Engineering (CAE) and software library. After intensive developments in computer hardware and also of computer algorithms, researchers and engineers had expected an innovation in program writing power. However, the enhancement in programming power was relatively weak compared with the enormous evolutions in hardware and algorithm power enhancements.

At present PSE covers a rather wide area, for example, program generation support PSE, education support PSE, CAE software learning support PSE, grid/cloud computing support PSE, job execution support PSE, learning support PSE, uncertainty management in scientific computing, PSE for PSE generation support (PSE for PSE), etc. The article includes a brief history of PSE, example PSE study activities and a future of PSE, including uncertainty management in scientific computing.

BACKGROUND OF PSE (PROBLEM SOLVING ENVIRONMENT) FOR SCIENTIFIC COMPUTING SUPPORT

PSE is defined as follows: "A system that provides all the computational facilities necessary to solve a target class of problems. It uses the language of the target

DOI: 10.4018/978-1-4666-5888-2.ch119

class and users need not have specialized knowledge of the underlying hardware or software" (Gallopoulos, Houstis, & Rice, 1994). PSE provides integrated human-friendly innovative computational services and facilities to enrich science, life, engineering, production, commerce and our society. Based on the PSEs, human concentrates on target problems themselves, and a part of solution is performed mechanically by computers or machines or software.

In computing sciences, we need computer power, excellent algorithms and programming power in order to solve scientific problems leading to scientific discoveries and development of innovative new products. So far, the computer power and the computing algorithms have been developed incredibly, and have provided enormous contributions to sciences and productions. On the other hand, the programming power has not been developed well, compared with the computer power and the algorithm power. The concept of PSE was proposed to support the programming power in science and engineering, and has been explored for decades.

In 1985, IFIP (International Federation for Information Processing) WG2.5 (Numerical Software) (IFIP WG2.5, 2012) organized a working conference on PSE and published the proceedings (Ford & Chatelin, 1987). In 1991, a working conference on Programming Environments for High-Level Scientific Problem Solving was held (Gaffney, & Houstis, 1992). In addition, a book on PSE was also published (Houstis, Rice, Gallopoulos & Bramley, 2000). In 2007, a working conference on Grid-Based Problem Solving Environments was held (Gaffney & Pool, 2007). A working conference on Uncertainty Quantification in Scientific Computing was held in 2012 (Dienstfrey & Boisvert, 2012). The PSE activity including scientific computing environments is one of research projects in IFIP WG2.5 (IFIP WG2.5, 2012). In these decades, international conferences tend to include the topic of PSE as one of standard topics in scientific computing. It has been recognized that PSE is an important research area in scientific computation and high-performance computing.

EXAMPLE PSES IN SCIENTIFIC COMPUTING

The PSE studies have been extensively explored over the past few decades. The explorations have been supported by the reinforced computer power and algorithm power. PSE has boosted the programming power, and have enriched problem solving methods in science and engineering to bring us innovations.

PSE studies (Boonmee et al., 1998a; Boonmee et al., 1998b; Fujio et al., 1998; Fujita et al., 2000; Gallopoulos, et al., 1991; Gallopoulos et al., 1994; Hirayama et al., 1988; Houstis et al., 1992; Kawata et al., 2000; Okochi et al., 1994; Umetani, 1985; Rice et al., 1984) have been extensively explored in order to support engineers and scientists to compute or solve their own problems based on for partial differential equations (PDEs). One of the major objectives in PSE researches is to help users compute or solve their problems without heavy tasks, for example, without complete knowledge for computations (Rice & Boisvert, 1984) and/or the programs used. In this sense, the PSE provides an infrastructure for software for computational engineering and sciences.

One of typical PSEs for PDEs-based problems is ELLPACK (Rice & Boisvert, 1984; Houstis et al., 1992). ELLPACK is a high level system for solving elliptic boundary value problems. One can solve routine problems by simply writing them down and naming the methods to be used. The ELLPACK high-level language can reduce the programming effort for a "routine" elliptic problem.

Another typical PSE for PDEs-based problems is DEQSOL (Umetani, 1985; Hirayama et al., 1988; Okochi et al., 1994). DEQSOL was designed to develop an easy-to-use system for problem solving of PDE-based problems by finite difference method and finite element method. DEQSOL creates optimal Fortran codes oriented to the Hitachi vector processors (Figure 1).

Another PSE system of NCAS (Boonmee et al., 1998a; Boonmee et al., 1998b; Kawata et al., 2000; Fujita et al., 2000) inputs a problem information including PDEs, initial and boundary conditions, and discretization and computation schemes, and outputs a program flow graph, a C-language source code for the problem and also a document for the program and for the problem. On a host computer a user inputs his/her problem, and NCAS guides the user to solve the problem. The distributed PSE for PDEs consists of several modules: problem description, discretization, equation manipulation, program design, program generation, documentation support, module liaison and job execution service. Each module is distributed on distributed computers. Each distributed module communicates with the host module, so that outputs

Related Content

Movie Analytics for Effective Recommendation System using Pig with Hadoop
Arushi Jain and Vishal Bhatnagar (2016). *International Journal of Rough Sets and Data Analysis (pp. 82-100).*
www.irma-international.org/article/movie-analytics-for-effective-recommendation-system-using-pig-with-hadoop/150466

Identification of Chronic Wound Status under Tele-Wound Network through Smartphone
Chinmay Chakraborty, Bharat Gupta and Soumya K. Ghosh (2015). *International Journal of Rough Sets and Data Analysis (pp. 58-77).*
www.irma-international.org/article/identification-of-chronic-wound-status-under-tele-wound-network-through-smartphone/133533

The Use of Structural Equation Modeling in IS Research: Review and Recommendations
Kun S. Im and Varun Grover (2004). *The Handbook of Information Systems Research (pp. 44-65).*
www.irma-international.org/chapter/use-structural-equation-modeling-research/30342

Hindi Text Document Classification System Using SVM and Fuzzy: A Survey
Shalini Puri and Satya Prakash Singh (2018). *International Journal of Rough Sets and Data Analysis (pp. 1-31).*
www.irma-international.org/article/hindi-text-document-classification-system-using-svm-and-fuzzy/214966

Design of a Migrating Crawler Based on a Novel URL Scheduling Mechanism using AHP
Deepika Punj and Ashutosh Dixit (2017). *International Journal of Rough Sets and Data Analysis (pp. 95-110).*
www.irma-international.org/article/design-of-a-migrating-crawler-based-on-a-novel-url-scheduling-mechanism-using-ahp/169176