WEB-based Environment for Programming and Distance Learning

Dmitry Vazhenin[1], Alexander Vazhenin[1], Ying-Hong Wang[2]

[1]*Graduate School Department of Information Systems*
The University of Aizu, Aizu-Wakamatsu, Fukushima, 965-8580, Japan
E-mail: {d8052102,vazhenin}@u-aizu.ac.jp

[2]*Department of Computer Science and Information Engineering*
Tamkang University, Tamshui, 25137, Taiwan, R.O.C.
E-mail: inhon@mail.tku.edu.tw

Abstract

The paper presents the main elements of the WWW-based Environment supporting development and execution of a big variety of user's applications. The system has universal interface allowing comfortable and unified manipulations with different project data. The system supports different types of the user's projects distinguished according to programming platforms and languages, type of a target machine, user's specialization, etc. We show also Distance Learning Projects considering as teacher/student programming workplaces as well as supporting non-programming courses.

1. Introduction

The transition towards digitalization in many application areas and the explosive growth of the Internet has had a huge impact in the modern computer science and engineering and will continue to do so into the future. During the past years, the Internet have an increasing impact on our day to day lives including the proliferation of WWW-devices, new WEB-services, an increasing number of employees and students working/studying from home, the development of Virtual Enterprises, WEB-based Collaboration Projects, Grid Computing [1], etc. A number of factors have recently contributed to bring the use of the Internet as a general programming environment closer to maturity. That is why the WEB technology is integrated with traditional methods and approaches in improving the remote accessibility of distributed applications, cooperating to solve some problems as well as offering new services.

Computational Portal [2] represents a proxy server placed between cluster and the user's workstation. It controls remote data processing operations by installing special software on the cluster nodes. WebSubmit is a Web-based utility providing access to applications on a collection of heterogeneous computing resources [3]. Its goal is to make it easy to use computing resources via the Web without requiring knowledge of the specifics of unfamiliar operating systems and dynamic application environments. WebSubmit does not allow implementing on-line analysis of results and debugging programs.

The innovative Environment for Learning to Program (ELP) [4] provides an interactive web-based environment for teaching Java programming to the first year Information Technology students. System does not support applets as well as other programming platforms. Interactive web exercises and a development environment [5] were developed to facilitate language acquisition in a beginning course in C++. System has well-designed forms to input problem solution sources in spite of absence of the program design issues.

An important direction in the modern computer science and engineering is also to engage academics and the information technology industry in shining the spotlight on the next generation of computing professionals and scholars. Accordingly, the educational process should have the stable environment for the most of educational courses oriented for both programming and so-called non-programming subjects.

The W4AP was initially designed to support WEB-based applied programming and Grid computing [6], [7]. The presented paper is devoted to adaptation of the W4AP for the Distance Learning purposes in order to keep the same environment for programming and education. In Section 2, we discuss the W4AP concepts as well as its Project Management Environment. Section 3 and 4 describe the W4AP programming techniques and different projects types supported including a variety of distance learning projects. Summary of results is presented in Section 5.

2. System Concepts

2.1. Architecture and Main Elements

Figure 1 shows the main elements W4AP-server kernel, which controls Internet access, multi-user support as well as security issues.

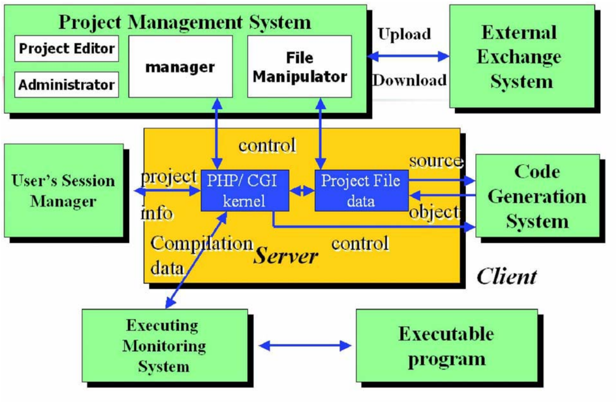
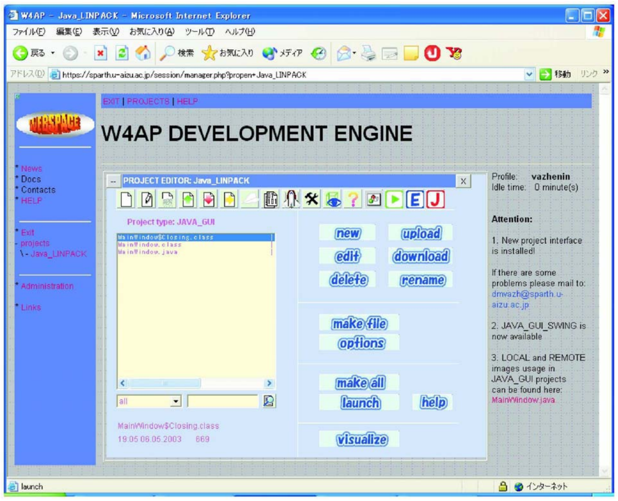


Figure 1. W4AP: System Architecture

The user's program implementation is under control of the *Project Management System* storing, manipulating and serving with the user's project files and data. The *User's Session Manager* is responsible for the user's authorization as well as services with the user's accounts. The *Code Generation System* is the multi-platform system generating the executable code. The *External Exchange System* provides data/files exchange between the W4AP server and client computer. Data can be also uploaded/downloaded from other remote systems as well as WWW-sites. It can also be a visual programming shell build on the server kernel. The *Monitoring System* allows the user to check compilation errors as well as execute and watch computational processes in real time.

2.2. Project Management System

To access the W4AP-system, it is only necessary to have any standard WWW browser like Netscape or MS Explorer by inputting the W4AP-server URL. After authorization, the user can begin or continue his/her work using a special *Project Management Interface* appearing as the special applet window with several areas of manipulation (Figure 2). The Project Management System includes components shown in Figure 1. *The Manager* provides calls to the W4AP kernel as well as to other management components. *The Project Editor* is to change existing projects or design new project types, *The Administrator* controls the system and security related settings. *The File Manipulator* implements different operations on project data (files, sources, etc.)


W4AP – Java_LINPACK – Microsoft Internet Explorer

https://sparth.u-aizu.ac.jp/session/manager.php?propen=Java_LINPACK

W4AP DEVELOPMENT ENGINE

PROJECT EDITOR: Java_LINPACK

Project type: JAVA_GUI

MainWindow$Closing.class
MainWindow.class
MainWindow.java

all

MainWindow$Closing.class
19:05 06.05.2003 669

Profile: vazhenin
Idle time: 0 minute(s)

Attention:

1. New project interface is installed!

If there are some problems please mail to: dmvazh@sparth.u-aizu.ac.jp

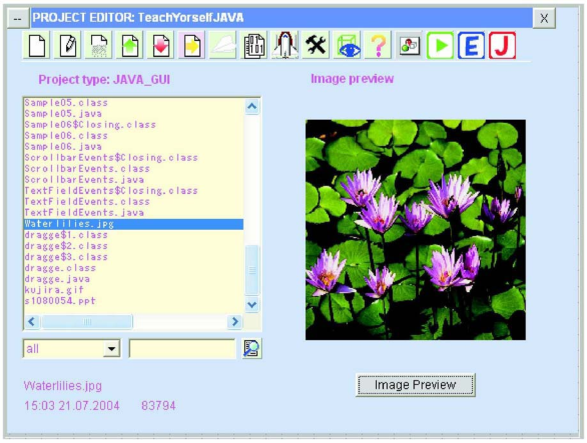
2. JAVA_GUI_SWING is now available

3. LOCAL and REMOTE images usage in JAVA_GUI projects can be found here: MainWindow.java

launch

Figure 2. The W4AP Project Management Interface

The project management window can be divided into three panels: the top toolbar, the left file list panel and the right multi mode panel (Figure 2). The file list area contains a list of project files and *filtration options* in order to separate files of the one type from all other files. The *Multi Mode Panel* is located at the right part of project management window. This panel has several view modes. **Action buttons mode** is for inexperienced users because the *Image (Toolbar) Buttons* on toolbar are duplicated on this panel. The **Image Preview Mode** is to show image files in a project (Figure 3). In the **Wide File List** mode, the space reserved for the Multi-mode Panel is used to show detailed file data. This makes possible using this system for a professional application development as well as for education.


PROJECT EDITOR: TeachYorselfJAVA

Project type: JAVA_GUI

Image preview

Sample05.class
Sample05.java
Sample06$Closing.class
Sample06.class
Sample06.java
ScrollbarEvents$Closing.class
ScrollbarEvents.class
ScrollbarEvents.java
TextFieldEvents$Closing.class
TextFieldEvents.class
TextFieldEvents.java
Waterlilies.jpg
dragge$1.class
dragge$2.class
dragge$3.class
dragge.class
dragge.java
kujira.gif
s1080054.ppt

all

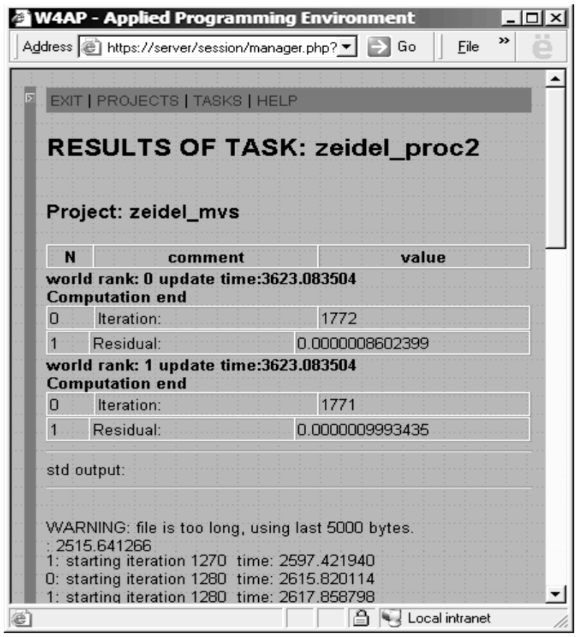
Waterlilies.jpg
15:03 21.07.2004 83794

Figure 3. Image Preview

3. Programming Environment

The W4AP programming environment supports the full circle of a program design including three stages. During the ***Source code creation stage,*** the developer should prepare files with sources by creating/modifying or uploading them from a client computer. The ***Compilation stage*** is to construct executable application from the prepared sources. If this stage was successful, it is possible to launch the ***Execution and Monitoring Stage,*** control program implementation, and collect results.

The ***Monitoring System*** (Figure 4) mentioned above allows the user to check compilation errors as well as execute and watch computational processes via the Internet in real time. *The Matrix Visualization Tool* is used to visualize matrix data in project [8].


W4AP - Applied Programming Environment
Address
https://server/session/manager.php?
Go
File
EXIT | PROJECTS | TASKS | HELP
Local intranet

RESULTS OF TASK: zeidel_proc2

Project: zeidel_mvs

N	comment	value
world rank: 0 update time:3623.083504 **Computation end**		
0	Iteration:	1772
1	Residual:	0.0000008602399
world rank: 1 update time:3623.083504 **Computation end**		
0	Iteration:	1771
1	Residual:	0.0000009993435

std output:

```
WARNING: file is too long, using last 5000 bytes.
: 2515.641266
1: starting iteration 1270  time: 2597.421940
0: starting iteration 1280  time: 2615.820114
1: starting iteration 1280  time: 2617.858798
```

Figure 4. Text-based Example of On-line Monitoring

There are special monitors to control syntax and runtime errors. The *ERROR-monitor* allows checking compilation errors as well as editing a source code. The standard *EXECUTION-monitor* shows as a HTML-document the status of program execution including errors verification during the program execution as well as execution stages like a program is **started**, **waiting**, **implementing**, and **finished**.

The *On-line Monitoring* Environment is a set of algorithms and tools for serving remote applications supporting optimal delivering and integrating results of remote applications to the client computer, and multimedia representation of intermediate and result data on a client computer. This information will be immediately open to the user. Figure 4 shows an example of parallel iterative computations with periodical refreshing the content of observable variables *Iteration* and *Residual.* On-line Monitoring is extremely useful for large-scale jobs, which may require about several hours, days or more of processors time.

To support a big variety of possible applications, the W4AP has several types of the user's projects distinguished according to programming platforms and languages, type of a target machine (sequential or parallel), user's specialization, etc. We distinguish three types of the target machines and corresponding projects. *Sequential project* is to implement sequential programs on a local server. *Parallel project* is to provide calculations on a remote parallel cluster. *Parallel GRID project* is to execute parallel programs using the different types of clusters.

The W4AP system supports different programming platforms, compilers and libraries. It is possible to apply CNU C, Java, etc. Parallel and GRID applications are based on Message Passing Interface (MPI). Special project allows designing and implementing of MPI-programs on the University of Aizu network.

Any multimedia and GUI application represents a Java applet implementing on the client. This applet generation is implemented automatically using avoidance of cashing effects in the WEB-browsers. To support multimedia programming in the W4AP environment, the special *Java-projects* and library were designed. This library is a collection of classes providing comfortable manipulations of multimedia objects such as images, image buttons, audio and video files.

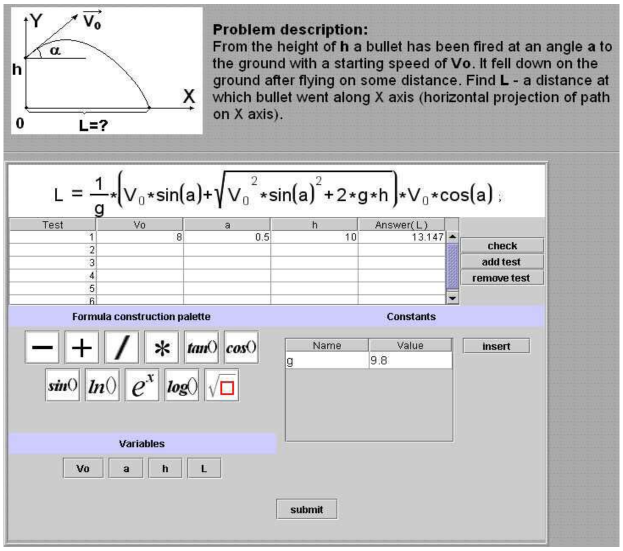
Projects can also support private and/or collaborative work. *Collaborative projects* include message exchange subsystem, file/data sharing and CVS-mechanisms.

4. Distance Learning Environment

The Distance Learning Environment includes a set of hierarchical collaborative projects. The learning project architecture consists of two special clients: the Student Workplaces and the Teacher Area. Each student team occupied the one workplace, and should implement a set of exercises or problems. The Teacher Area has special interface to prepare problems as well as tools to verify/estimate solutions. After job solution, a Student Team should submit via the W4AP solutions the Teacher Area for verification and counting.

Programming Learning Projects are to help users in increasing their programming skill as well as studying modern Software Engineering Topics like Competitive and Extreme Programming. To support these technologies, the Teacher Area was extended by special tools for the running and testing programs. These tools provide an automatic verification of the program implementation and returns to teams and teachers result of testing. In this case, a run-able code should be submitted for checking. There exist several scenes of examinations between students, teachers and the W4AP system. Simple exercises allow getting some information to solve the problems. Exams are supporting traditional on-line examination procedures.

We are also designing a set of so-called **Non-programming Projects.** An example of these problems can be physics and mathematics problems and exercises. Special tools allowing to input/verify/calculate formulas support these projects. Importantly, these formulas can include multimedia/visual objects (Figure 5).



Problem description:
From the height of **h** a bullet has been fired at an angle **a** to the ground with a starting speed of **Vo**. It fell down on the ground after flying on some distance. Find **L** - a distance at which bullet went along X axis (horizontal projection of path on X axis).

$$L = \frac{1}{g} * \left(V_0 * \sin(a) + \sqrt{V_0^{\;2} * \sin(a)^2 + 2 * g * h}\right) * V_0 * \cos(a) ;$$

Test	Vo	a	h	Answer(L)
1	8	0.5	10	13.147
2				
3				
4				
5				
6				

Name	Value
g	9.8

Figure 5. Student's interface for problems in physics

The teacher should make and solve the problem, find an answer formula and prepare other necessary information: a set of variables, constants and verification tests. Verification tests are necessary to check correctness of formula. Each test consists of a set of values for variables and result for these values. Result is calculated automatically using answer formula.

Students should input result formula (answer) using a special student's window (see Figure 5.) It is also possible to check the answer by creating student verification tests. After problem solution, the student should submit his/her answer to the teacher. After receiving the student's answer, the verification process is started. A result of the student's answer formula is calculated using teacher's verification tests. Then, corresponding results of student's formula and teacher's formula are compared. If all verifications are successful then the solution will be considered as correct.

5. Conclusion

We have presented main elements and features of the WWW-based environment supporting the development of a big variety of user's applications not only for program design but also for the Distance Learning Technologies. This hides hardware specifics and reduces working conditions to the standard HTML-environment. The software presented is installed in the University of Aizu and used in student research projects as well as in the Software Engineering Courses. The practical experiments with the system confirm its good adaptability to the user's profile because of the same environment for different programming platforms.

Our further investigations include development of the multilingual interfaces as well as convenient Assessment Tools for testing and grading of students.

6. References

[1] Foster, C. Kesselman (Eds.), *The GRID Blueprint for a New Computing Infrastructure*, Morgan Kaufman Publ., 1999.

[2] V. V. Prohorov, "Computational Portal: Remote Access to the High-Performance Computing," *LNCS,* Vol. 2127, pp. 308-313, 2001.

[3] *Websubmit: A web-based Interface to High-Performance Computing Resources.* (http://math.nist.gov/mcsd/savg/websubmit)

[4] N. Truong, P. Bancroft, P. Roe, "A Web Based Environment for Learning to Program", *Proc. of the Twenty-Sixth Australasian Computer Science Conference (ACSC2003), Adelaide, South Australia, February 2003,* CRPIT, Australian Computer Society, pp. 255-264, 2003.

[5] B. S. Elenbogen, B. R. Maxim, Ch. McDonald, Yet, "More Web Exercises for Learning C++", *Proceedings of the thirty-first SIGCSE technical symposium on Computer science education, Austin, Texas, USA,* pp. 290 – 294, 2000.

[6] A.P. Vazhenin, D.A. Vazhenin, "Applied Programming on WEB-based Environment", *Topics in Applied and Theoretical Mathematics and Computer Science,* WSEAS Press, pp. 287-292, 2001.

[7] A. Vazhenin, Y.-H. Wang, D. Vazhenin, "WEB-based Multimedia Platform for Programming," *Proc. of the 8-th International Conference on Distributed Multimedia Systems,* San Francisco, California, USA, pp. 76-83, 2002.

[8] A.P. Vazhenin, N.N. Mirenkov, D.A. Vazhenin, "Multimedia Representation of Matrix Computations and Data", *Information Sciences,* Vol. 141, pp. 97-122, 2002.