

<http://ansinet.com/itj>

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSI*net*

Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Web-based Command-line Window and its Applications for Programming in Practice

Feng Pan and Juan Han

Guizhou Key Lab of Pattern Recognition and Intelligent Control, Guizhou Minzu University, China

Abstract: At this time, many programming tools are single mode, not be shared. To solve this problem, this study presents a Web-based programming environment which users submits programs through the web interface and gets compiling or execution results in real time. The design uses a browser/server structure; provide network middleware to finish the relevant function requirement. Web server accepts HTTP requests and calls the network middleware processing business and the results be saved to a file, Web server returns URL of file to the browser. Therefore, this study describes a method that users called utility tools, as interpreter and compiler tools etc. Experimental results show that the scheme is feasible and effective, convenient for users.

Key words: Web-based programming environment, browser/server structure, command-line window

INTRODUCTION

With the continuous development of the computer networking technology and the software industry, software delivering various functions has been emerging and software has also gradually become networked. Web has turned from a pure publication medium into an interactive e-commerce medium. The extension of HTML in supporting data entry forms, the extension of HTTP in supporting security and other properties as well as the introduction of CGI (Common Gateway Interface) standard of the Web server used for inter-application communications enable data submitted by Web-based clients to be processed by the applications on the Web server. With the standardization and promotion of Web-based applications, developers can rely on relatively stable client components and focus on servers (Neil, 2004). In this network environment, users only need to use standard Web browsers on the servers for the purposes of daily work, learning and scientific research. Among the Internet services, the traffic of remote logins is second only to WWW, file transfer and network news. A lot of users rely on remote logins as part of their work environments and use remote hosts to handle matters, work and study, etc. The first three ways mainly provide users with information resource services while remote logins provide service resources, namely, a kind of work environment.

Remote logins are a kind of early service provided on the Internet. Users login remote computer via Telnet command, using utility tools and resources of the remote computer to complete their work. Through remote logins, users can utilize all resources that are public to the outside on remote computers in real time and many

commands permitted by the systems to read, write and edit files as well as compile and run programs, etc. They can also inquire databases and retrieve data or use remote computers to finish those tasks only achievable by supercomputers. In addition, many services on the Internet are realized through Telnet access. Currently, considering users' demand for remote logins as well as their preference for application, many software companies have developed remote sharing and remote control software to realize remote logins. Teamviewer provides an integrated solution for remote access and remote support through the Internet (Anonymous, 2013); Pcity enables remote users to locate, connect and control their wanted hosts (Symantec Corporation, 2013). Netman remote control software (<http://netman123.cn/>) can be used to control screens, manage files, switch on and off computers remotely, etc. Windows system remote desktops allow for similar operations but they are used at a very low rate. The above-mentioned software is suitable for professionals. Similarly, among numerous Internet applications, as the simplicity and mass-oriented characteristic of the Web enable it to become one of the most successful examples, numerous advantages of the Web-based software architectures have increasingly become choices of current software architectures. Pan *et al.* (2010) proposed a distributive compiling environment in which interpretive C/C++ computing environment is used to debug C program remotely. Wang and Yao (2006) used a shell to call Pascal compiler to compile programs and to run executable files on the basis of ASP technology in order to realize online judgment of programs. Yu *et al.* (2006) integrated Ch interpretive computing environment and the open source computer vision library Open CV to establish the web

application-online digital image processing. Zhao and Lin (2007) and Chen *et al.* (2009) gave the Web applications of MATLAB. SoftIntegration Inc. (2009) gave a comprehensive introduction of the integrated applications of Ch and Web Servers IIS, Apache, etc.

In order to make it easy for users to use remote computers, this study proposes a Web-based design scheme of the application system that simulates remote logins and gives examples of conducting common command line operations and compiling programs using remote computer systems. First, it gives descriptions of the architecture design and key technologies of the system; second, it details the design the system's client, server and key intermediate components; finally, it demonstrates the feasibility of this scheme through application of several examples.

SYSTEMS WORKING PROCESS

Design philosophy of the system: The command lines are submitted to the Web server via Web forms. The Web server then calls the programs on the server to start Shell and executes command lines in Shell. The execution results are redirected to files by standard output (STDOUT) and standard errors (STDERR) and then the content of the files are returned to the client.

Operating principle of the system: The system adopts browser/server architecture which takes the browser as the user interface, the Web server as the service broker and the network intermediate component as the processing center. The Web server receives HTTP requests and calls the intermediate components to process user input. User input is classified into common commands and programs. For the former, commands are regarded as the string parameters of system (char*) function in C so that, they can be used to transmit console command lines to the operating system and the

execution results are written to files using redirection technology; for program codes, they are written to source code files and the interpreter or compiler is called to interpret or compile the resulting in-files. At the same time, the intermediate components generate the Web pages that include the hyperlink of the resulting files which are then returned to the browser by the Web server. The operating principle of the system is shown as Fig. 1.

MATERIALS AND METHODS

The OS (Operating System) of experiment server-side is Windows 7 and uses IIS (Internet Information Server) to provide Web server capabilities while setting up Ch as the running environment of Ch scripts. The seamless integration of Ch-CGI kit and Web server realizes the interaction between client-side and the server.

The system design mainly includes the design of the interactive interface and the intermediate components. The interactive interface uses the Web page and adopts frame Web page technology in order to better simulate the integrated development environment. The navigation page, the input page and the output page are separately loaded by the frame and then Web-related technologies are used to realize the interaction between the pages. The input page also uses JavaScript to ensure the validity of input data and its characteristics supporting regular expressions to format the commands in order to prevent destructive commands from being input.

CGI programs are an important technology used to expand Web servers. Ch dynamic Web page technology is used in this system. Ch dynamic Web pages are interpreted and executed by the interpretive C/C++ computing environment Ch. A Ch file is a text file using C language as the scripting language. It is executed in Ch environment and used as a CGI program. Ch is an interpreter of the C language as well as the execution engine of Ch. Scripts. The seamless integration of Ch and

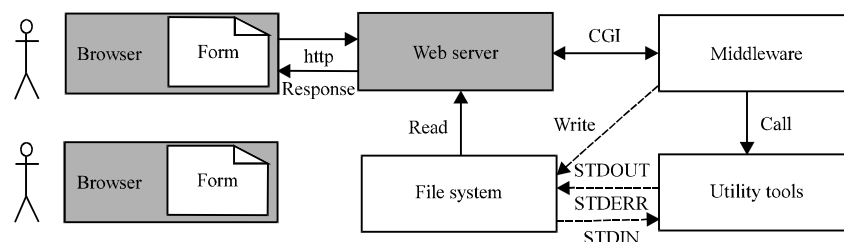


Fig. 1: Architecture of the system and mutual interactions of system components. The system is composed of browser, web server, middleware and utility tools. The command lines are submitted to the web server via web forms. A command line is executed in OS Shell when web server calls middleware. Executed result is redirected from standard output to file and then URL of the file is returned to browser

the Web server is an important technology for the construction and expansion of server applications. The standard output (STDOUT) and standard error output (STDERR) that remote computers generate by compiling programs and running commands are sent to files using redirection technology and the XHTML files dynamically generated by the intermediate components contain the hyperlinks of these files.

DESIGN AND IMPLEMENTATION

Web-based systems should consider the design of the Web layer which is the entrance to system use and the design of the business logic layer which mainly focus on the design of the intermediate.

Client design: The client uses frameset pages and can simulate an integrated development environment. Skips between pages will not change the URL of the browser's address bar and the system is transparent. The client interface of the system is shown in Fig. 2: The entire interface consists of four frames. The upper section is the title frame; the left section is the feature list frame; the middle section is the user input frame and the bottom section is the frame showing results. The four frames display different pages and the pages in the user input frame and the bottom frame change dynamically.

In the above figure, when the options in the list page are selected, the response page will be displayed in the input page; submission in the input page will respond in the output page.

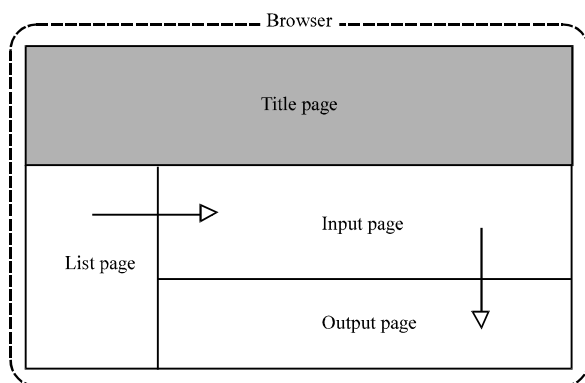


Fig. 2: Logical relation between Web pages. The interface is a frame set page which include 4 areas. Each of them displays one Web page, respectively. The middle part is the output area of the left interface; at the bottom, it is the output area of the middle part

Logical relationship of the client interface: In the feature list page, the target attribute is used in the hyperlink tag <a> and the attribute value is the name attribute value of the frameset where the input page is located. That is, click the hyperlink of the list box and request the page to display it in the user input frame; use target in the <form> tag in the user input page to indicate the output position of the processed results so that the results processed by HTTP requests are displayed in the output frame.

Intermediate component design: The focus of the system is on the design of the intermediate component which is the key component that processes the programs and commands submitted by users. If a program is submitted to the server to be compiled and runs on the server, then the process can be described as follows:

- Step 1:** Read in program code and write it to a text file
- Step 2:** Call the compiler to compile source code files and redirect the results to the text file A using the standard error stream output
- Step 3:** Call the compiler to compile source code files and redirect the results to the text file B using the standard output stream
- Step 4:** Load executable files and redirect the results to the text file C using the standard output stream
- Step 5:** Append the text file C to the text file A
- Step 6:** Return the URL containing the text file A

The above process realizes the remote use of compilation tools. If you want to use some utility tools provided by the operating system, you just need to submit the used command format via forms. The intermediate components are loaded and executed by calling System function. The standard output is redirected to files. The files will finally return to the client.

Examples

Design of remote command-line interface: The focus of the system is on the design of the intermediate component which is the key component that processes the programs and commands submitted by users. The process of processing programs submitted by the client is described as follows: Use Web forms to input command lines which will then be executed on remote hosts after being submitted. These commands include the internal and external commands of the system and they are generally executed in the command line window. For example, the system's internal command dir can be used to view current contents and files the del command can be used to delete some files. This system uses the interface in Fig. 3a to input command lines.

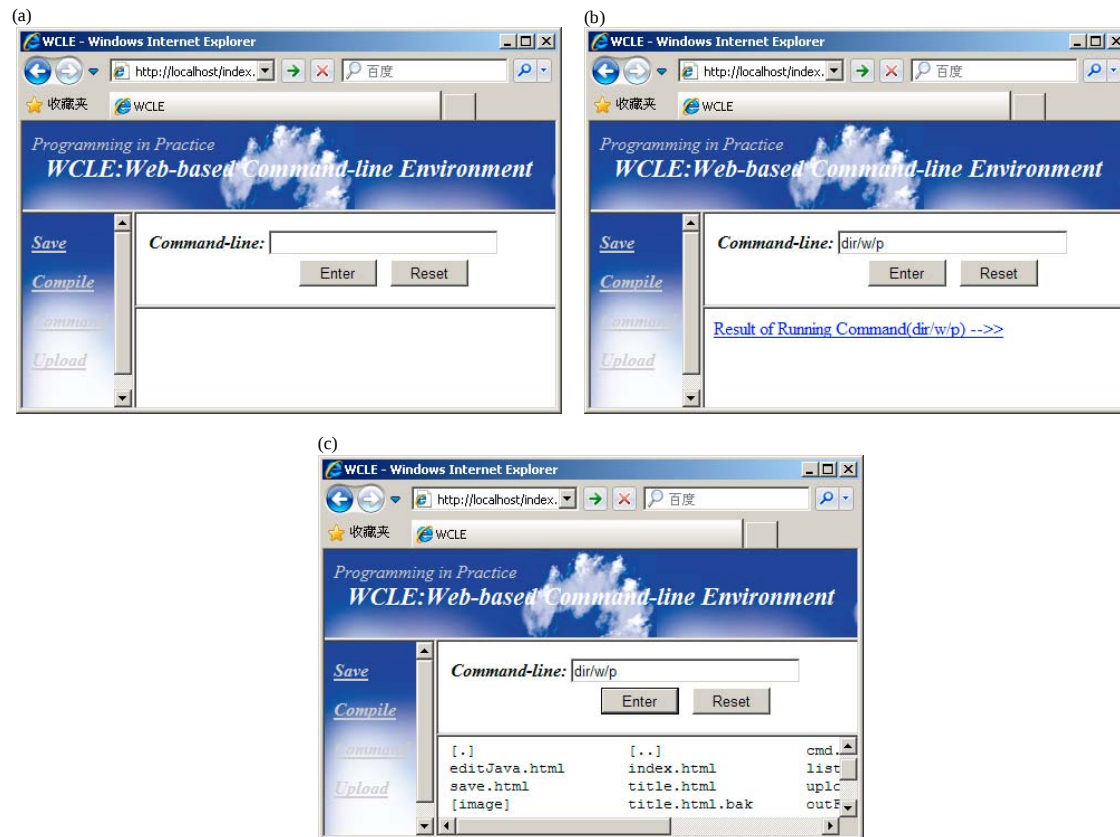


Fig. 3(a-c): (a) Web interface of inputting command line with Web form, (b) Hyper-results after submission of command line: dir/w/p and (c) Results of Execution commands after clicking hyperlink

The key codes of the intermediate components used to process command lines are as follows:

```
class CRequest Req;
class CResponse Res;
char* value = Req.getForm("DOSCMD");
char cmd[30];
strcpy(cmd,value);
strcat(cmd," 1>cmdresult.txt");
system(cmd);
Res.setContentType("text/html");
Res.begin();
Res.title("Windows Command");
printf("<a
href='cmdresult.txt'>Command:%s</a>",value);
Res.end();
```

If the command line input to the Web form is dir/w, then the intermediate component executes the statement system ("dir/w 1>cmdresult.txt"); That is, view current contents and the execution results of the command lines will be redirected to the text file cmdresult.txt by standard output. The intermediate component returns a XHTML

file to the browser on the client and the file contains the hyperlinks of the resulting files. The XHTML file is displayed in the bottom frame of the frameset which is shown in Fig. 3b.

Compiling and running of C++ programs: The interface for compiling programs is similar to that for running command lines. You simply need to set the text entry box in the Web form to multiple lines and change the programs used to process forms to corresponding intermediate components. When the Web form submits a C++ program, the corresponding intermediate component will write source codes to a.cpp text file, call a C++ compiler to compile them and redirect the standard output or standard error during compiling and running to a text file. The Web server returns the Web page containing the hyperlink of the text file to the browser and displays the results in the bottom output box shown in Fig. 3b. Click the hyperlink and you will be able to see the content of the text file at the bottom of the frame set which is shown

in Fig. 3c. If input data is needed when programs are running, it is enough to add a text input box in the original interface for compiling programs and each data is separated by spaces. The network intermediate component will be written to a text file after being read. During the running of the programs, redirection of standard output is used to transmit the data of the data files to the running programs.

Key statements executed by the intermediate component that compiles and runs programs are:

```
system("g++ main.cpp>CResult.txt 2>&1");
if(isEmpty("CResult.txt")){
system("a.exe>RResult.txt 2>&1");
returnURL("RResult.txt");
}
```

The above example uses GUN compiler g++ to compile program main.cpp. A web form is used to submit a C++ program and the corresponding intermediate component writes source codes to the main.cpp text file. A C++ compiler is called for compiling and the results of compiling and running are written to a text file. The Web server returns the hyperlink of the text file to the browser so that users can view the final results by clicking the hyperlink.

DISCUSSION AND ANALYSIS

Web-based use has the following characteristics for users:

Instant online use: Web-based software needs no installation and deployment and it is in the charge of software providers. Users can use this software as long as their computers can be networked.

Users' remote maintenance: All running environments of traditional software are in the charge of users themselves, so the operation and maintenance of software largely rely on users. With this application pattern, users can maintain systems remotely using the Web.

Calculation everywhere: Data of traditional software are stored in a fixed place and are related to physical location. When users want to handle businesses during their business trips or vacations, they will have no ideas. This system can save files directly on the server and users can also upload files and save them on the server, so that, it is very convenient for users to use commands such as dir to locate.

Low costs: Web-based software is not a kind of goods but a kind of service. This software needs no purchase, so there is no purchase cost, neither are there upgrade or maintenance costs. It is charged according to the amount of service.

Easy extension of systems: For example, by adding intermediate components and designing the client user interface, we can make JDK networked to provide users with the remote development environments of Java.

CONCLUSION

This system submits commands to the server through Web forms and transmits the execution results to the browser, so that, the WWW services of Browser/Server structure and other programs are effectively integrated and extended to a networked work environment. It frees from installation of client tool software and saves users a lot of troubles. Through design and deployment of intermediate components that have different functions, the functions of the server are extended freely which makes it easy to extend functions according to users' needs. The system is flexible. In short, with this system, users can use the command lines of the operating system just like on local computers and call compilation tools to debug programs at will.

ACKNOWLEDGMENTS

This study is sponsored by the Science and Technology Fund Project of Guizhou Province of China (No. LKM[2011]18, No. [2009]4002). I would like to extend my sincere thanks to all members of the research team.

REFERENCES

- Anonymous, 2013. TeamViewer 8 manual: Remote control. pp: 12-18. <http://www.teamviewer.com/en/index.aspx>
- Chen, W.X., J.X. Xiang and J.G. Chen, 2009. Design and implementation of Web-based remote MATLAB command window. J. Comput. Appl. Software, 2: 138-141.
- Neil, G., 2004. Web Server Programming. Tsinghua University Press, Beijing, China, pp: 1-20.
- Pan, F., H. Xi and R.L. Zhang, 2010. A design and implementation of distributed compiling platform. J. Southwest China Normal Univ., 4: 141-144.
- SoftIntegration Inc., 2009. Ch CGI user's guide. <http://www.softintegration.com/>

- Symantec Corporation, 2013. Fact sheet: Symantec PC anywhere 12. <http://www.symantec.com/zh/cn/>
- Wang, T. and D.L. Yao, 2006. Online judge system design and development. J. Comput. Appl. Software, 12: 129-137.
- Yu, Q.C., X.J. Weng and Y. Zhao, 2006. Interactive open architecture computer vision platform: Ch OpenCV. J. Comput. Eng. Appl., 23: 78-81.
- Zhao, J.D. and J. Lin, 2007. Web-based MATLAB decision support system research. J. Comput. Appl., 12: 266-272.