2006-1640: INFORMATION TECHNOLOGY COURSE DEVELOPMENT: SCRIPT PROGRAMMING

Yu Cai, Michigan Technological University
Dr. Yu Cai is an assistant professor at School of Technology in Michigan Technological University. His research interests include network protocols, distributed systems and cyber security. He received his Ph.D. in Computer Science from University of Colorado in 2005. He is a member of IEEE and ACM.

Guy Hembroff, Michigan Technological University
Mr. Guy Hembroff is an Assistant Professor within Michigan Tech University's School of Technology Department. His research interests are within the areas of cyber security, network protocols, encryption methods, health-care security, and biometrics. He has six years of industrial experience as a systems engineer and advanced network engineer. Mr. Hembroff is currently pursuing his Ph.D. degree in Computer Information Science.
Abstract

Traditional script programming courses usually concentrate on Unix Shell and Perl. However, with the fast development of Information Technology, there are some new factors that we need to take into consideration. In Computer Network & System Administration (CNSA) program at Michigan Technological University, we design a new script programming course to better prepare students for the needs of industry.

In this paper, we present the design of the new script programming course. We incorporate the web-related scripting technologies as well as Windows-related scripting technologies in the new course. Compared with traditional scripting courses, the new course covers a wider and more diverse range of topics. In the new course, the lecture notes are carefully designed so as to illustrate scripting languages through examples. We train students to learn to search resources and solutions using Internet and Google. We utilize Virtual Machines to set up lab environment. The lab assignments are designed to further students’ problem solving skills and learn “hands on” programming. The new script programming course aims to provide students the opportunity to learn multiple scripting technologies, and gain hands-on experience on scripting.

1. Introduction

A computer-related program named Computer Network & System Administration (CNSA) was introduced at Michigan Technological University in 2002 [1]. The purpose of this program was to serve the needs of industry by educating and training students to succeed in the field areas of systems administration and network engineering. An integral part of the CNSA curriculum is the three-credit hour course entitled Script Programming – SAT3400.

Traditional script programming course usually concentrates on Unix Shell scripting and / or Perl scripting. However, with the fast development of Information Technology, there are several factors that we need to take into consideration when designing a new script programming course.

First, the popularity of the Internet makes web-related scripting technologies become more important. It is necessary to include web scripting technologies like CGI (Perl), ASP.NET and PHP.

Second, the increasing market share of Microsoft Windows Servers makes Windows scripting technologies become more popular. It is helpful to include Windows scripting technologies like Windows batch file and VBScript.

Third, in the “real-world” environment, the system administrators are likely to be exposed to multiple scripting languages. It is important to let students know the links
and differences between the scripting languages. It is also vital to train students to learn how to search resources and answers using Internet and Google.

Compared with the traditional scripting course, our new course covers a wider and more diverse range of topics.

In this paper, we present the design and implementation of the new script programming course at Michigan Technological University. The new course aims to provide students the opportunity to learn multiple scripting languages, and gain hands-on experience on scripting.

2. Challenges

We address a couple of challenging issues when developing the new scripting course as follows.

The first challenge is to incorporate such a wide and diverse range of topics in one single course. To do so, we shift the focus from procedural scripting to object-based scripting because it is the trend of modern scripting technologies. We start the course with VB.NET and Object Oriented concepts.

We make comprehensive comparison between different scripting languages on a regular base so that students will not get confused with different languages. This also helps students to understand things in common between the different scripting languages. The lecture notes are carefully designed to illustrate languages through examples, instead of words. “A good example is better than a thousand words”. Through examples, students can master the language more quickly and effectively. The examples in lecture notes also serve as a good starting point for the lab assignment.

In the new course, we emphasize on the ability to read, modify and debug the existing scripts, instead of writing scripts from scratch. Students are trained to learn how to search resources and information using Internet and Google [2]. It is increasingly important to train students to know how to do self-learning and Internet search in today’s ever-changing world.

The second challenge is to set up a lab environment for students to practice script programming over multiple platforms in a network environment. We utilize Virtual Machines (VMWare [6] and VirtualPC [5]) to set up lab environment. With virtual machines, we can not only better meet the needs of the course, but also make good usage of the available hardware resources, and reduce the work load of lab management.

Figure 1 shows the setup of multiple virtual machines on one physical host machine. These virtual machines form a small-scale private network, and students can learn and practice on how to administrate, maintain and troubleshoot network with script technologies.
3. Course Design and Implementation

In the new scripting course, there are two major objectives. First, we want to enhance students’ problem solving skills using multiple scripting languages, like Unix Shell, Perl, Windows Batch, VBScript, CGI and ASP.NET. Second, we want to expose students to .NET programming and Object Oriented (OO) concepts. Because OO becomes increasingly important in modern programming.

To accomplish the course objectives, we design the course to include two lecture sessions and two lab session every week during a fourteen-week semester. Each session is fifty minutes long. The lecture session is used to present new topics and work through example problems. The lab section is used to learn “hands on” programming and to further script programming skills. Instructor is available for help during the lab session. To ensure each student gets enough attention and help from the instructor, the class (24 students) is divided into two groups. Each group has a different lab session time. Outside of scheduled meeting times, students can access the lab 24x7.

The course final grade is based on lab assignments, quizzes, mid-term exam, final exam and semester project. The mid-term exam includes one lecture exam and one lab exam, same for the final exam. In the lab exam, student is required to demonstrate his / her skills and abilities of programming, problem solving and troubleshooting within the classroom.

We utilize WebCT [3] as the teaching platform in this course. All assignments are handed out and turned in electronically via WebCT. Lecture materials are also available through WebCT and course web site.

The topics covered in the course and timeframe for each topic are listed below:

- VB.NET Introduction (3 weeks)
- Object Oriented Concepts (1 weeks)
- Windows Batch scripting (1 weeks)
- Windows VBscript (1 weeks)
There are totally thirteen lab assignments. One lab assignment for each week except the first week. Most lab assignments are designed to simulate “real-world” scenarios. We list several representative lab assignments as follows.

- A VB.NET assignment is to implement the well-known tic-tac-toe game. Students are required to design and implement the GUI interface using VB.NET. This assignment tests students’ ability to analyze the logic flow of a given problem.
- A VB.NET assignment is to implement an inventory tracking system for a bookstore. This assignment requires students to have good understanding on database-related concepts.
- A VBscript assignment is to implement a windows script to be able to join a stand-alone computer into an existing domain. This assignment requires knowledge on Active Directory Service Interface (ADSI) [4].
- An ASP.NET assignment is to implement a web authentication system. The program validates the username and password against the back-end SQL Server database, and grant or deny access to the web user accordingly.
- A Unix Shell assignment is to implement a shell script to backup data files automatically. This type of task is very common in “real-world” Unix system administration.
- A Perl assignment is to implement a perl script to process a log file. The program needs to search a key string in the log file, extract the lines containing the key string to a separate file, and then purge the original log file.
- A CGI assignment is to implement a web guestbook using perl and HTML. The Internet users can input message in the front-end HTML forms, the back-end CGI program will process and store the user input in a text log file.

4. Course Evaluation

When the semester was close to the end, students were asked to complete an anonymous evaluation of this course. 19 out of 24 students who took the course responded to the survey. Following are the summarized results of the survey.

- 95% students stated that the amount of materials covered in the course is reasonable and the pace of the course is consistent with their ability to learn. Only 5% gave negative response. This is encouraging because it has been one of our biggest concerns when we design and teach this course.
- 100% students stated that they like to use the virtual machines and think the lab equipments are adequate to support effective learning. This is a very positive response because it is the first time that we use virtual machine in this course.
- 95% students stated that they think the course handout and lecture notes are helpful and can further their learning. Only 5% gave negative response. Here
are some students’ comments, “The handout is awesome!”, “I really like the examples in the slides”.
- 90% students stated that they want to take this course. 10% gave neutral response.
- The overall student evaluation of this course is above the department average.

5. Conclusion

This paper describes the design and implementation of a script programming course, which is an integral component in the Computer Network and System Administration curriculum at Michigan Technological University. We address several challenges when design and teach this course. We also propose our answers to those challenges. The instructor evaluation and qualitative student feedback indicate that the course objectives are being met successfully.

Bibliography

1. CNSA program at MTU. http://www.tech.mtu.edu/cnsa/index.html