

Servlet Basics

Vittayasak Rujivorakul

Lecture 1

Servlet Basics

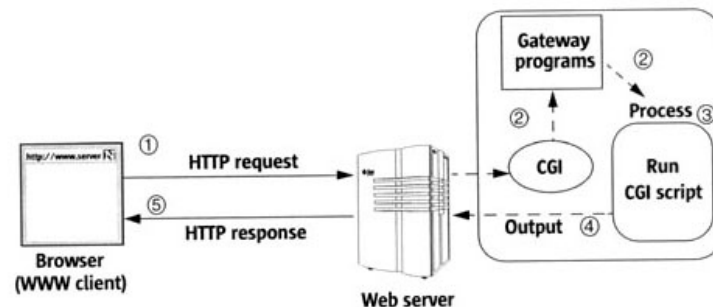
Vittayasak Rujivorakul

Lecture 1

Common Gateway Interface (CGI)

- เป็น mechanism ที่ช่วยให้ client สามารถ execute programs บนฝั่ง server ได้
- โปรแกรมเหล่านี้ถูกเรียกว่า CGI script โดยสามารถเขียนจากภาษา programming ใด ๆ เช่น
 - C, C++
 - Visual Basic
 - Perl
 - Tel / Tk
- นิยมนำมาสร้าง web utility, finger

CGI Request and Response



ขั้นตอนการทำงาน CGI Request and Response

1. Web browser สร้าง request จาก HTTP Header เพื่อเรียกใช้ CGI Program
2. HTTP Server รับ request และทำการ start shell process
3. Server ทำการ run script และ General output HTML
4. Output ถูกส่งไปให้ HTTP Server
5. Server ทำการ response output ไปยัง web server

CGI Script Code

```
#!/usr/local/bin/perl
# Perl 5.05
# HelloWorld.pl
# Print out a content-type for HTTP/1.0
print "Content-type: text/html\n\n";

&print_results;

sub print_results{
    print "<HTML><HEAD><TITLE>Hello World</TITLE></HEAD>";
    print "<BODY><h1>Hello World</h1></BODY></HTML>";
}
```

CGI Script Code (ต่อ)

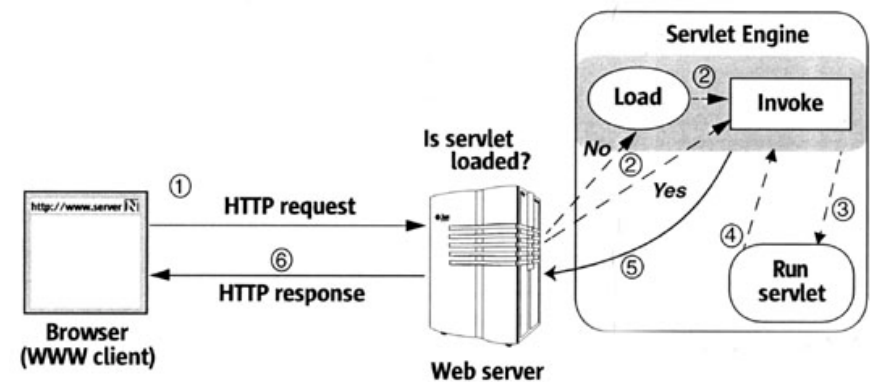
- การ set Header Type ทำได้โดยการเขียน response Content-type : text/html
- Perl ใช้ “print” command ในการเขียน output
- Script จะถูกวางไว้ใน Directory server-root/cgi-bin/ ซึ่งอาจเป็น alias Directory
- เวลาเรียกใช้ จะถูกเรียกลักษณะนี้

http://host-name:port/cgi-bin/HelloWorld.pl

Servlet

- เป็น module ที่ใช้รองรับ request/response oriented Services
- สามารถทำงานกับ หลาย ๆ protocol เช่น FTP ,SMTP, HTTP , DHCP , SNMP, etc...
- สามารถเข้าถึง remote object เช่น RMI , CORBA
- สามารถใช้ในการติดตามการไหลของ Information จึงสามารถใช้กับระบบ shopping chat ขนาดใหญ่ได้
- สามารถทำงานในลักษณะของ multi user ได้

Servlet Request and Response

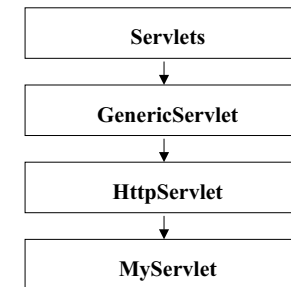


Servlet Request and Response

1. Web browser สร้าง request จาก HTTP header เพื่อเรียกใช้ servlet program
2. Web server รับ request แล้วทำการตรวจสอบว่ามี servlet ถูกโหลดเข้ามายัง Invoke หรือยัง ถ้ายังให้ทำการ load ก่อน
3. Invoker ทำการ เรียก servlet service method เพื่อทำการแตก thread และทำการ generate output
4. Output ถูกส่งกลับไปยัง servlet engine
5. Servlet Engine ส่ง output กลับไปให้ web Server
6. Web server ทำการ response output กลับไปให้ web server

Servlet Package

- จะอยู่ใน javax.servlet
- Servlet ใน Servlet API คือ Servlet interface ทุก ๆ Servlets ต้อง implement interface นี้ ไม่ว่าจะโดยการ extends หรือ implement



HttpServletRequest

Servlet Request อนุญาตให้ servlet ติดต่อได้ถึง

- ข้อมูล เช่น ชื่อของ parameter, protocol, remote host, ชื่อ Server
- input stream: Servlet Input Stream Servlet ใช้ input stream ในการรับข้อมูลจาก client ซึ่งใช้ protocol ของ application เช่น HTTP POST และ PUT

HttpServletResponse

Interface นี้ จะให้ Servlet methods สำหรับการ reply ไปยัง client

- อนุญาตให้ servlet กำหนดความยาวของ content และ MIME type ในการ reply
- ให้ output stream: Servlet Output Stream และ Writer ซึ่ง servlet สามารถส่ง reply data ให้กับ client ได้

Hello World Servlet

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class HelloWorld extends HttpServlet{
    public void doGet(HttpServletRequest req, HttpServletResponse res) throws ServletException, IOException{
        res.setContentType("text/html");
        try{
            PrintWriter out = res.getWriter();
            out.println("<HTML>");
            out.println("<HEAD><TITLE>Hello World</TITLE></HEAD>");
            out.println("<BODY>");
            out.println("<h1>Hello World</h1>");
            out.println("</BODY></HTML>");
            out.close();
        }catch(IOException ioe) {
            getServletContext().log(ioe,"Error in HelloWorld");
        }
    }
}
```

Servlet Code (ต่อ)

- response.setContentType ใช้ในการ set รูปแบบของการ response output
- PrintWriter (response.getWriter) ใช้ในการ return ข้อมูลที่เป็น text data
- ServletOutputStream (response.getOutputStream) ใช้สำหรับ return ข้อมูลที่เป็น Binary data
- เมื่อใช้งาน writer เสร็จแล้วจะต้องทำการ close เสมอ ใช้ out.close

Invoking the Servlet

- เราจะวาง servlet ไว้ใน path ต่อไป
server_root /servlets/ หรือ
server_root /classes/
- Servlet ที่เขียนขึ้นมาเป็น .java จะต้องถูก compile ให้เป็น .class ก่อนจึงจะนำมาใช้งานได้
- เวลาเรียกใช้จาก Browser เรียกได้ดังนี้
http://server:port/servlet/HelloWorld