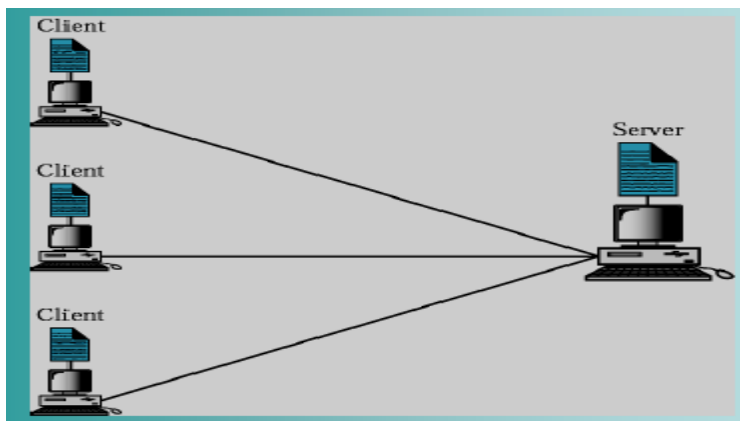


DHMTL & LẬP TRÌNH WEB CHẠY Ở CLIENT

CHƯƠNG IV

GIỚI THIỆU HỆ THỐNG WEB

- Mô hình hoạt động: Web browser/Web server



GIỚI THIỆU HỆ THỐNG WEB

- **Web server:** chương trình quản lý một cây thứ bậc các trang Web và phục vụ yêu cầu truy xuất chúng từ các client từ xa.
- **Web Browser:** chương trình giao tiếp với người dùng, nhận yêu cầu từ user rồi truy xuất trang Web ở server tương ứng để phân giải và hiển thị nội dung lên màn hình.
- **Trang Web:** đơn vị gởi nhận giữa Web server và Web client, nó là 1 file văn bản được viết bằng ngôn ngữ HTML
- Web browser và web server sẽ tạo 1 cầu nối TCP để trao đổi thông tin, server lắng nghe ở port 80 (port mặc định) và dùng giao thức HTTP (HyperText Transfer Protocol).

CẤU TRÚC MỘT GIAO DỊCH HTTP 1.0

- Bước 1: HTTP client mở kết nối đến HTTP server
- Bước 2: HTTP client gửi **request** message đến HTTP server
- Bước 3: HTTP server trả **response** message về cho HTTP client, chứa resource mà HTTP client yêu cầu
- Bước 4: HTTP server đóng kết nối
- Ví dụ
Client: **GET /path/file.html HTTP/1.0**
From: someuser@jmarshall.com
User-Agent: HTTPTool/1.0
[blank line here]
Server: **HTTP/1.0 200 OK**
Date: Fri, 31 Dec 1999 23:59:59 GMT
Content-Type: text/html
Content-Length: 1354
[blank line here]
<html>
...
</html>

GIỚI THIỆU VỀ LẬP TRÌNH WEB

- Các đặc điểm của ứng dụng web
 - Không phải là một ứng dụng độc lập, là ứng dụng chạy trên nền web-server.
 - Người lập trình có thể thực hiện các thao tác trên ứng dụng web như với một ứng dụng độc lập.
 - Có thể lấy thông tin từ user.

HTML – HyperText Markup Language

- Là một ngôn ngữ đánh dấu siêu văn bản.
- Sử dụng các thẻ (tag) để định dạng: văn bản, đoạn văn, hình ảnh, file âm thanh, Java applet, đối tượng như file Word, Excel, JavaBean...
- Cho phép chèn các đoạn mã script như JavaScript thực thi tại trình duyệt (browser).
- Là định dạng chuẩn cho trang web.
 - Mở rộng: XML, WSDL/SOAP...

CÁC CÔNG CỤ TẠO TRANG WEB

- **Microsoft Visual InterDev 6**
 - HTML, ASP...
- **Microsoft Visual Studio.NET 2003**
 - HTML, ASP.NET, ASP.NET Webservice, C#, C++...
- **Microsoft Frontpage**
 - HTML...
- **Jbuilder**
 - HTML, JSP, Java Servlet, Java EJB...
- **Macromedia Dreamweaver MX 2004**
 - HTML, ASP, JSP, ASP.NET...
- **EditPlus**
- **Notepad/Wordpad/Unix VIM**

HTML – HyperText Markup Language

<HTML> ... </HTML>	Declares the Web page to be written in HTML
<HEAD> ... </HEAD>	Delimits the page's head
<TITLE> ... </TITLE>	Defines the title (not displayed on the page)
<BODY> ... </BODY>	Delimits the page's body
<Hn> ... </Hn>	Delimits a level <i>n</i> heading
 ... 	Set ... in boldface
<I> ... </I>	Set ... in italics
 ... 	Brackets an unordered (bulleted) list
 ... 	Brackets a numbered list
<MENU> ... </MENU>	Brackets a menu of items
	Start of a list item (there is no)
 	Force a break here
<P>	Start of paragraph
<HR>	Horizontal rule
<PRE> ... </PRE>	Preformatted text; do not reformat
	Load an image here
 ... 	Defines a hyperlink

HTML – HyperText Markup Language

<TABLE> </TABLE>	Bảng thông tin
<TR> </TR>	hàng thuộc bảng
<TH> ... </TH>	phần tử header
<TD> ... </TD>	phần tử dữ liệu
<OBJECT> </OBJECT>	phần tử nội rộng như ActiveX Control,...
<APPLET code=URL ...>	Applet Java
<SCRIPT> </SCRIPT>	Đoạn chương trình
.....	

HTML – HyperText Markup Language

• FORMs

- Cấu trúc cơ bản của một form

```
<FORM ACTION= "file" METHOD={GET|POST}>
  [<INPUT TYPE=" " NAME=" " VALUE=" " >]+
</FORM>
```

- Các loại (type) INPUT

- TEXT: là textbox dùng để nhập dữ liệu.

- Cú pháp:

```
<INPUT [TYPE=TEXT] NAME="text-id" [SIZE=nn]
[MAxLENGTH=nn] [VALUE="default text"]>
```

HTML – HyperText Markup Language

● FORMs(tt)

– Các loại (type) INPUT

- SUBMIT: truyền dữ liệu form đến ứng dụng web
 - Cú pháp: `<INPUT TYPE=SUBMIT [NAME="button-id"] [VALUE="text"]>`
- RESET: đưa nội dung của form trở về giá trị ban đầu
 - Cú pháp: `<INPUT TYPE=RESET [VALUE="text"]>`
- BUTTON: nút nhấn bất kỳ
 - Cú pháp: `<INPUT TYPE=BUTTON [NAME="button-id"] [VALUE="text"]>`

HTML – HyperText Markup Language

● FORMs(tt)

– Các loại (type) INPUT

- RADIO: chọn lựa một trong nhiều
 - Cú pháp: `<INPUT TYPE=RADIO NAME="radio-set-id" VALUE="choice-id" [checked]>`
- ```
<input type="radio" value="V1" checked name="R1">Option 1

<input type="radio" value="V2" name="R1">Option 2

<input type="radio" value="V3" name="R1">Option 3
```

# HTML – HyperText Markup Language

- FORMs(tt)

- Các loại (type) INPUT

- CHECKBOX: lựa chọn một hoặc nhiều

- Cú pháp: `<INPUT TYPE=CHECKBOX NAME="id" VALUE="choice-id" [CHECKED]>`

- Ví dụ:

- `<input type="checkbox" name="C1" value="ON">Check 1 <br>`

- `<input type="checkbox" name="C1" value="ON" checked>Check 2`

☐ Check 1

☒ Check 2

# HTML – HyperText Markup Language

- FORMs(tt)

- Các loại (type) INPUT

- HIDDEN: dữ liệu không hiển thị

- Cú pháp: `<INPUT TYPE=HIDDEN NAME="id" VALUE="data">`

- Nhập vùng văn bản: TEXTAREA

- `<TEXTAREA NAME="id" [COLS=nn] [ROWS=nn]>default text</TEXTAREA>`

## HTML – HyperText Markup Language

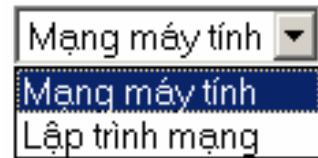
- FORMs(tt)
- FORMs(tt)
  - Lựa chọn listbox: SELECT
    - `<SELECT NAME="id" [SIZE=nn] [MULTIPLE]>`  
`[<OPTION [VALUE="value"] [SELECTED]>text ]+`  
`</SELECT>`

```
<select size="1" name="D1">
```

```
<option value="501097">Mạng máy tính</option>
```

```
<option value="501016">Lập trình mạng</option>
```

```
</select>
```



## LẬP TRÌNH WEB Ở CLIENT

- Các trang HTML có thể được lập trình để thực hiện các thao tác thực thi trên web client.
- Các phương pháp phổ biến để lập trình hiện nay:
  - Script (Javascript, Vbscript)
  - Java Applet
  - ActiveX Control



# LẬP TRÌNH WEB Ở CLIENT

- **Cú pháp cho một đoạn code Javascript**

```
<script language="JavaScript">
 //các khai báo biến
 [var variable;*]
 //các lệnh gọi hàm và các hàm
 function function-name([agrv]*){
 //các câu lệnh khai báo và thực thi
 [return [value];]
 }
</script>
```

- **Sử dụng file Javascript (\*.js) đã có**

```
<script language="JavaScript" src="*.js"></script>
```

# LẬP TRÌNH WEB Ở CLIENT

- **Các đối tượng được xây dựng sẵn trong JavaScript**

- window: chứa các thuộc tính liên quan đến cửa sổ hiện thời
- document: chứa các thuộc tính trong trang web
- location: các thuộc tính về địa chỉ trang web
- history: các thuộc tính về vị trí mà web browser đã đến

- **Mỗi phần tử trong trang HTML đều có thể coi là một đối tượng, có thể khai báo sự kiện để gọi hàm Javascript.**

- Cú pháp

```
<tags-name [properties]*
 [event-name="procedure call"]*>
```

# LẬP TRÌNH WEB Ở CLIENT

## – Các sự kiện có thể dùng

- *onClick*: click chuột vào đối tượng.
- *onFocus*: focus vào đối tượng trong form.
- *onMouseOver*: di chuyển chuột vào trên đối tượng.
- *onMouseOut*: di chuyển chuột ra khỏi đối tượng.
- *onChange*: thay đổi giá trị của các đối tượng chứa văn bản.
- *onBlur*: chuyển focus khỏi đối tượng trong form.
- *onSelect*: chọn phần tử trong listbox.
- *onLoad*: xảy ra khi một document được load.

# LẬP TRÌNH WEB Ở CLIENT

## ● Kiểm tra phần tử SELECT

```
var select_value;
function dropDownMenu() {
var myindex=document.forms[0].menu.selectedIndex;
if (myindex==0) {
alert("\nYou must make a selection from the drop-down
menu.");
document.forms[0].menu.focus();
}
else {
select_value
=document.forms[0].menu.options[myindex].value;
return true;
}
}
```

# LẬP TRÌNH WEB Ở CLIENT

- **Kiểm tra thông tin địa chỉ E-mail nhập vào**

```
function isEmail() {
 if (document.forms[0].elements[1].value == '') {
 alert ("\n The E-Mail field is blank. \n\n "+
 "Please enter your E-Mail address.")
 document.forms[0].elements[1].focus();
 return false;
 }
 if (document.forms[0].elements[1].value.indexOf('@',0) == -1 ||
 document.forms[0].elements[1].value.indexOf('.',0) == -1) {
 alert ("\n The E-Mail field requires a \"@\" and a \".\""+
 "be used. \n\nPlease re-enter your E-Mail address.")
 document.forms[0].elements[1].select();
 document.forms[0].elements[1].focus();
 return false;
 }
 return true;
}
```

## DHTML

- DHTML (Dynamic HTML) là trang HTML có chứa các đoạn mã JavaScript tạo các hiệu ứng, thay đổi các phần tử trong trang HTML.
- DHTML kết hợp giữa HTML, Javascript và Cascading Style Sheets(CSS).
- CSS là sự kết hợp giữa các tags HTML và style.
- Style có thể chứa nhiều thuộc tính.

# DHTML

- Mọi tags HTML đều có style. Javascript có thể dựa trên các biến cố để thay đổi các thuộc tính của style.
- Mỗi phần tử trong trang HTML đều có thể được gán ID (khác với name) và có các thuộc tính style.
- Có thể dùng tag `<div id="">...</div>`
- Mỗi web browser có thể thực thi một số lệnh khác nhau.

# DHTML

## • Phát hiện loại web browser

- Dùng các thuộc tính *appName* và *appVersion* của đối tượng *navigator*

```
var browser_name= navigator.appName;
var browser_version = navigator.appName;
alert("You use browser "+browser_name+" version "+
 browser_version);
```

- Phân biệt hai loại chính

```
if(document.all){ //IE
 the_div=eval("window.document.all."+div_name+".style");
}else if(document.layers){ //Netscape
 the_div=eval("window.document.all."+div_name);
}
```

# DHTML

```
<HTML>
<HEAD>
<TITLE>Javascript example about timer</TITLE>
<SCRIPT language="JavaScript">
<!--
var second_counter=0, counter=0;
function my_timer(){
D = new Date();
seconds= D.getSeconds();
if(seconds!=second_counter) counter++;
second_counter=seconds;
if(counter>=1000) counter=0;
document.images[0].src= get_images(counter/100);
document.images[1].src= get_images((counter%100)/10);
document.images[2].src= get_images(counter%10);
SecondText.innerText=counter;
setTimeout('my_timer()',1000);
}
```

# DHTML

```
<HTML>
<HEAD>
<TITLE>Javascript example about timer</TITLE>
<SCRIPT language="JavaScript">
<!--
var second_counter=0, counter=0;
function my_timer(){
D = new Date();
seconds= D.getSeconds();
if(seconds!=second_counter) counter++;
second_counter=seconds;
if(counter>=1000) counter=0;
document.images[0].src= get_images(counter/100);
document.images[1].src= get_images((counter%100)/10);
document.images[2].src= get_images(counter%10);
SecondText.innerText=counter;
setTimeout('my_timer()',1000);
}
```

# DHTML

```
function get_images(number){
var newNum= Math.floor(number);
return "../images/"+newNum+".jpg";
}
-->
</SCRIPT>
</HEAD>
<BODY onload='my_timer()'>
<P>Timer :

000
seconds</P>
</BODY>
</HTML>
```

# LẬP TRÌNH WEB CHẠY Ở CLIENT

## • Java Applet

- Là một ứng dụng được viết bằng Java, nhúng trong trang HTML.
- Khi trang HTML có chứa tag applet được gọi, class applet được tải về máy client và thực thi trên máy client.
- Một ứng dụng applet phải thừa kế class java.applet.Applet
- Có thể thực hiện các thao tác như một chương trình Java, tuy nhiên bị hạn chế một số chức năng vì tính bảo mật cho máy client.

# LẬP TRÌNH WEB CHẠY Ở CLIENT

## ● Hoạt động của một Applet

- Được web browser tải mã(byte code) về máy client.
- Method init() sẽ được gọi để khởi động các thông số.
- Method start() được gọi để thực thi.
- Method stop() được gọi khi người dùng thoát khỏi applet
- Method destroy() được thực thi khi applet kết thúc.

## ● ActiveX Control

- Tương tự như Java Applet, được viết bởi các công nghệ của Microsoft.

# LẬP TRÌNH WEB CHẠY Ở CLIENT

**//file SampleApplet.java**

```
import java.applet.*; import java.awt.*;
public class SampleApplet extends Applet {
String text = "error"; int x = 0; int y = 20;
 public void init() {
 text = getParameter("text");
 try { x = Integer.parseInt(getParameter("x"));
 y = Integer.parseInt(getParameter("y"));
 } catch (NumberFormatException ex) { }
 }
 public void paint(Graphics g) {
 g.setFont(new Font("TimesRoman", Font.BOLD+
 Font.ITALIC, 36));
 g.drawString(text, x, y);
 }
}
```

# LẬP TRÌNH WEB CHẠY Ở CLIENT

```
<HTML> <HEAD> <TITLE>Using the Applet Tag
</TITLE> </HEAD>
<BODY>
<H1>An Applet that Displays Text at a Designated
 Location</H1>
<APPLET CODE="SampleApplet.class" HEIGHT=300
 WIDTH=300>
<PARAM NAME="text" VALUE="Applets are fun!">
<PARAM NAME="x" VALUE="50">
<PARAM NAME="y" VALUE="50">
Text displayed by browsers that are not Java-
 enabled.
</APPLET>
</BODY>
</HTML>
```

## TÀI LIỆU THAM KHẢO THÊM

- **HTML Homepage on W3C**
  - <http://www.w3.org/MarkUp/>
- **HTML ver 4.0**
  - <http://www.w3.org/TR/REC-html40/>
- **HTML Form**
  - <http://www.w3.org/TR/REC-html40/interact/forms.html>
- **Dave Raggett's Introduction to HTML**
  - <http://www.w3.org/MarkUp/Guide/Overview.html>
  - <http://www.w3.org/MarkUp/Guide/Advanced>
  - <http://www.w3.org/MarkUp/Guide/Style.html>
- **Javascript & DHTML**
  - <http://javascript.internet.com/>
  - <http://www.dynamicdrive.com/>



# TỔNG KẾT

- Giới thiệu hệ thống Web
- Giới thiệu lập trình Web
- Lập trình Web phía Client
  - Javascript
  - DHTML