

- Cómo funciona la Web?
 - HTTP
 - Cliente / Servidor
 - URI, URL
 - Protocolo
 - Host (Nombre de Dominio)
 - Archivo
 - Sin conexión y sin estado

- Petición del Browser al Servidor

GET /doc1.html HTTP/1.0

Accept: www/source

Accept: text/html

Accept: image/gif

User-Agent: Lynx/2.2 libwww/2.14

From: rchampre@lidi.info.unlp.edu.ar

/* esto es una linea en blanco */

- Respuesta del Servidor al Browser

HTTP/1.0 200 OK

Date: Friday, 21-Feb-05 16:30:00 GMT

Server: Apache/1.1.1

Content-type: text/html

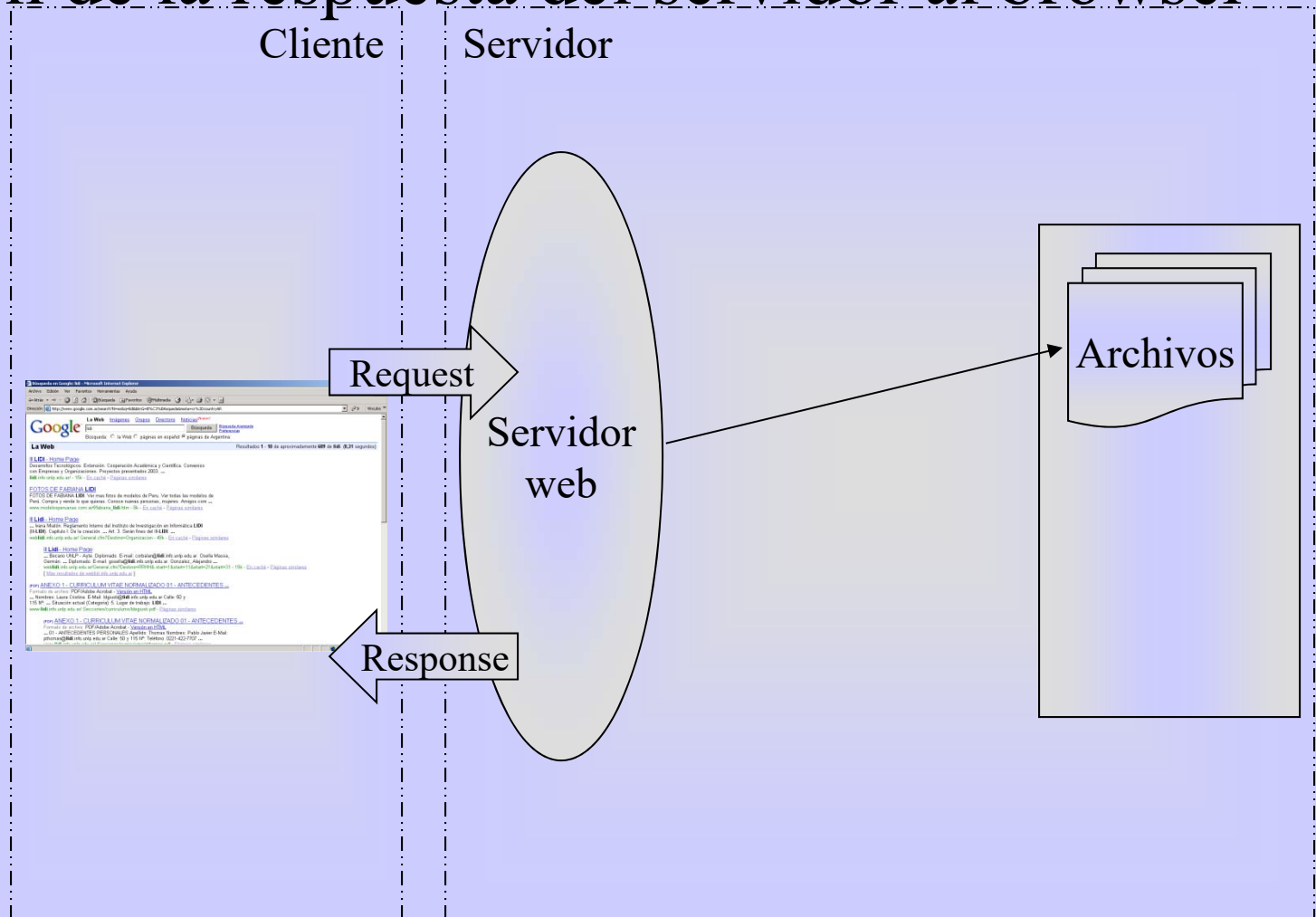
Content-length: 230

/* esto es una linea en blanco */

<HTML><HEAD><TITLE> </HTML>

Seminario de Lenguajes (opción PHP) – La Web

- Generación de la respuesta del servidor al browser
 - Estática

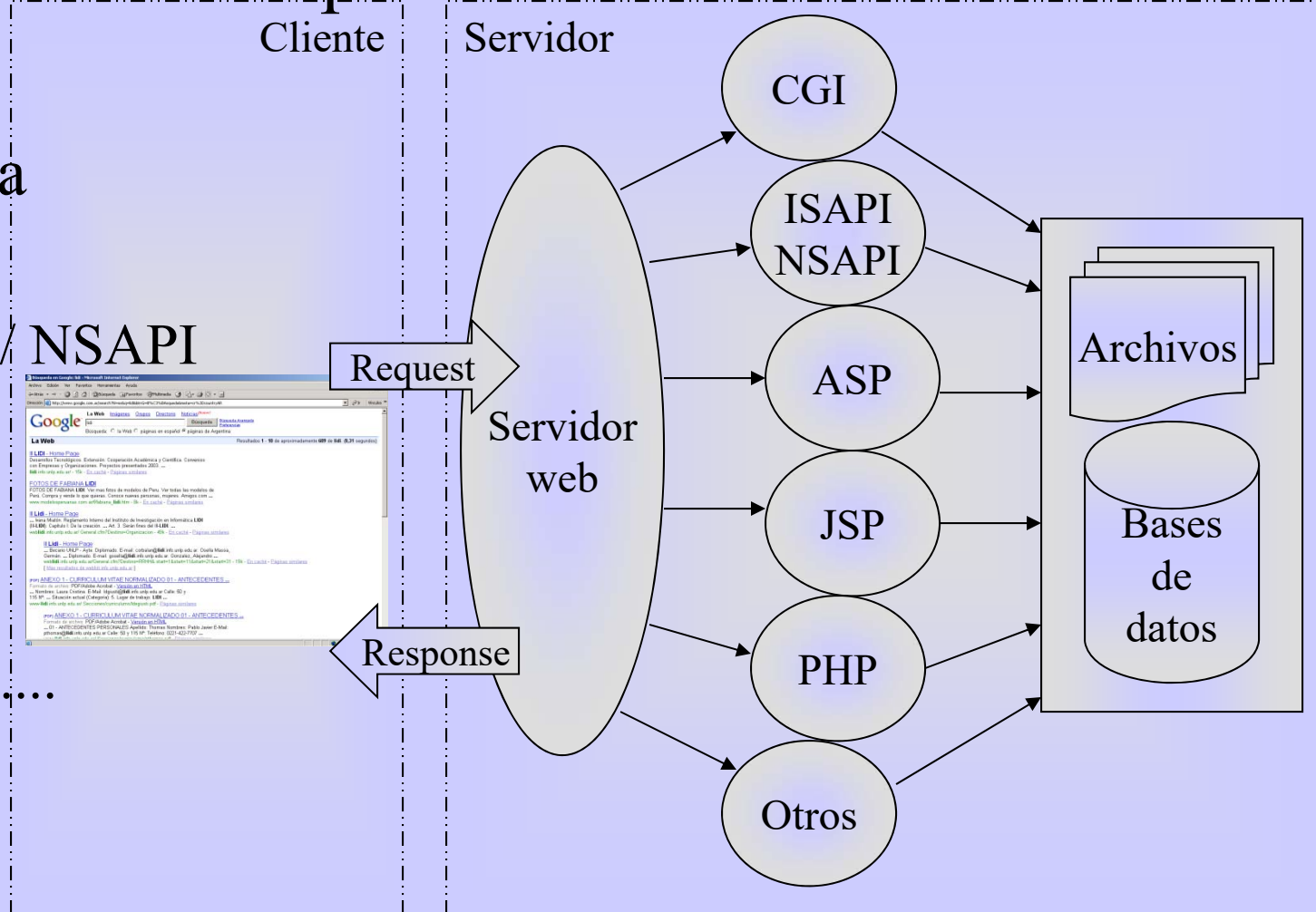


Seminario de Lenguajes (opción PHP) – La Web

• Generación de la respuesta del servidor al browser

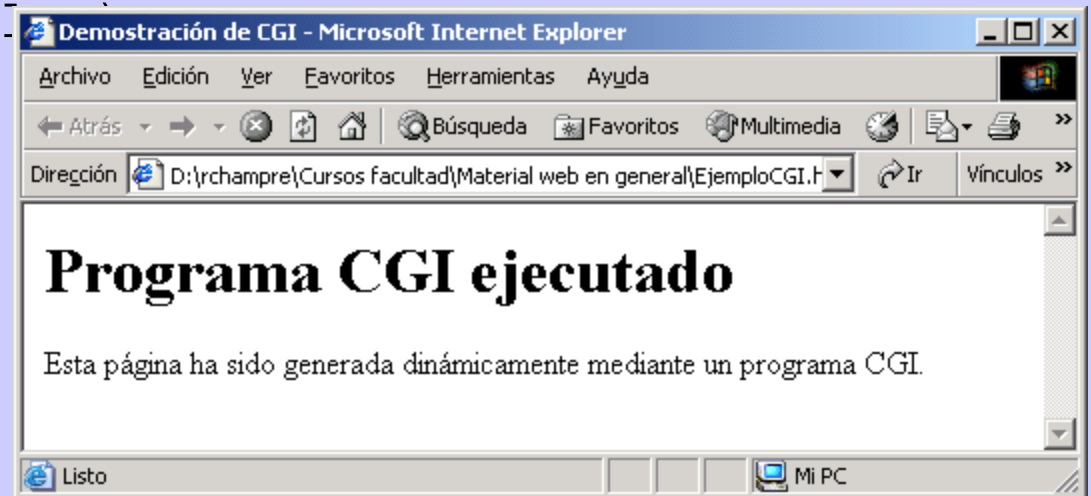
- Estática
- Dinámica

- CGI
- ISAPI / NSAPI
- ASP
- JSP
- PHP
-



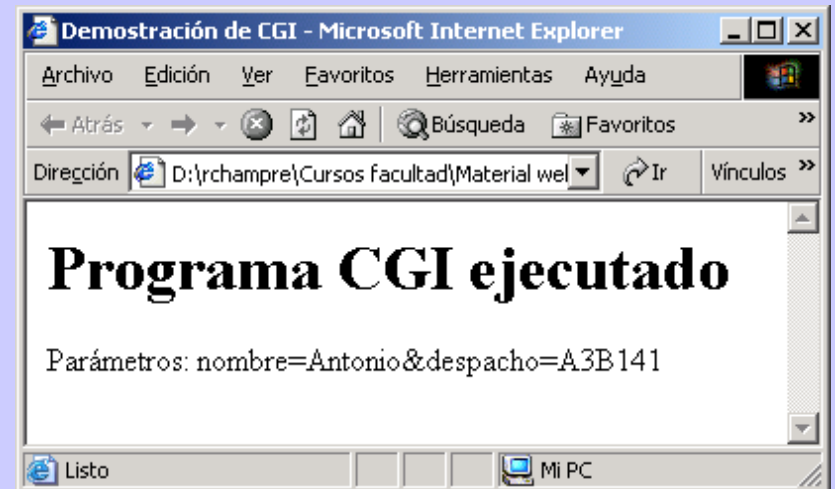
- Primer ejemplo CGI

```
#include <stdio.h>
int main (int argc, char **argv) {
    puts ("Content-type: text/html\n");
    puts ("<html><head>");
    puts ("<title>Demostración de CGI</title>");
    puts ("</head><body>");
    puts ("<h1>Programa CGI ejecutado</h1>");
    puts ("Esta página ha sido generada dinámicamente");
    puts ("mediante un programa CGI.");
    puts ("</body></html>");
    return 0;
}
```



- Segundo ejemplo CGI

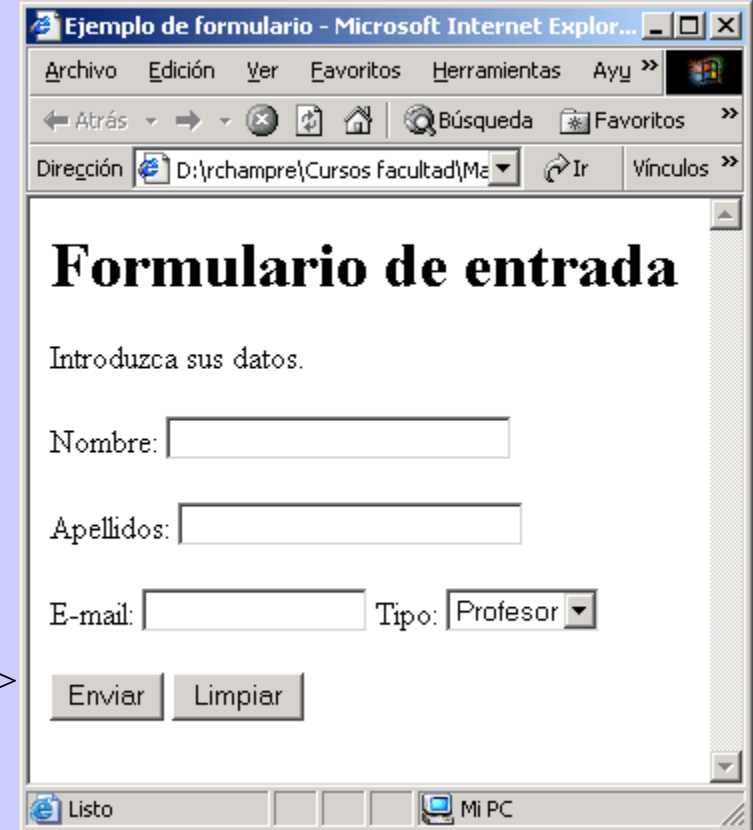
```
#include <stdio.h>
#include <stdlib.h>
int main (int argc, char **argv) {
    puts ("Content-type: text/html\n");
    puts ("<html><head>");
    puts ("<title>Demostración de CGI</title>");
    puts ("</head><body>");
    puts ("<h1>Programa CGI ejecutado</h1>");
    printf ("Parámetros: %s",
getenv("QUERY_STRING"));
    puts ("</body></html>");
    return 0;
}
```



Seminario de Lenguajes (opción PHP) – La Web

- Tercer ejemplo CGI

```
<html>
<head>
<title>Ejemplo de formulario</title>
</head>
<body>
  <h1>Formulario de entrada</h1>
  Introduzca sus datos.<p>
  <form action="cgi-bin/pruebacgi3" method="get">
    <label for="nombre">Nombre: </label>
    <input type="text" name="nombre" size=25><br><br>
    <label for="apellidos">Apellidos: </label>
    <input type="text" name="apellidos" size=25><br><br>
    <label for="email">E-mail: </label>
    <input type="text" name="email" size=15>
    <label for="tipo">Tipo: </label>
    <select name="tipo">
      <option>Profesor</option><option>Alumno</option>
    </select><br><br>
    <input type="submit" value="Enviar"> <input type="reset" value="Limpiar">
  </form>
</body>
</html>
```

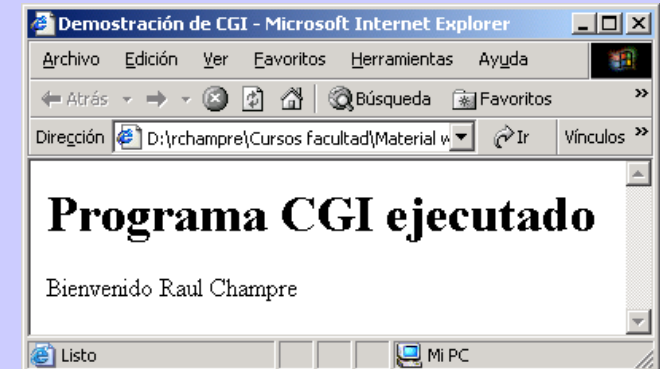
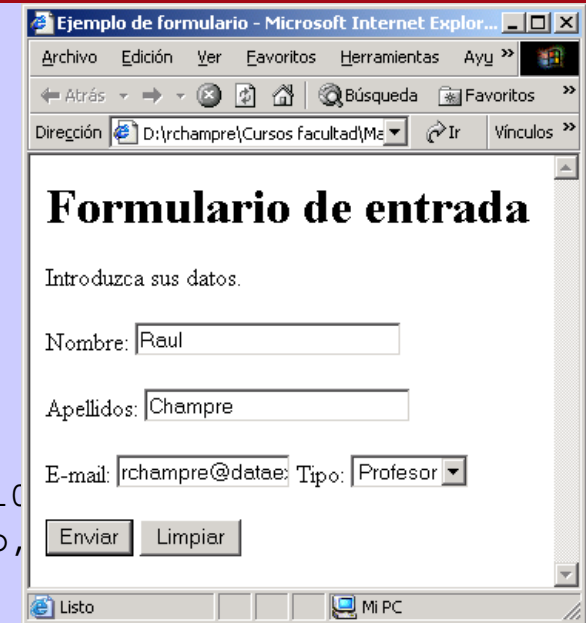


Seminario de Lenguajes (opción PHP) – La Web

• Tercer ejemplo CGI (cont.)

```
int main (int argc, char **argv) {
    char nom[1024], ap[1024];
    puts ("Content-type: text/html\n");
    puts ("<html><head>");
    puts ("<meta content=\"text/html; charset=utf-16\">");
    puts ("<title>Demostración de CGI</title>");
    puts ("</head><body>");
    puts ("<h1>Programa CGI ejecutado</h1>");
    if ((leerparam(getenv ("QUERY_STRING"), "nombre", nom, 1024),
        (leerparam(getenv ("QUERY_STRING"), "apellido", ap, 1024),
        printf("Bienvenido, %s %s\n", nombre, apellido);
    else
        printf("No sé tu nombre.\n");
    puts ("</body></html>");
    return 0;
}
```

`http://.../pruebacgi3?nombre=Raul&apellid
os=Champre&email=rchampre%40lidi.info.unlp
.edu.ar&tipo=Profesor`



- Métodos de petición
 - Get
 - Head
 - Post
 - otros

- Método de Petición GET

- Request

- GET /index.html HTTP/1.0

- Connection: Keep-Alive

- User-Agent: Mozilla/2.02Gold (WinNT; I)

- Host: www.ora.com

- Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, */*

- Response

- HTTP/1.0 200 Document follows

- Date: Fri, 20 Sep 1996 08:17:58 GMT

- Server: NCSA/1.5.2

- Last-modified: Mon, 17 Jun 1996 21:53:08 GMT

- Content-type: text/html

- Content-length: 2482

- (body of document here)*

- Parámetros en QUERY_STRING

- Método de Petición HEAD

- Request

- HEAD /index.html HTTP/1.0

- Connection: Keep-Alive

- User-Agent: Mozilla/2.02Gold (WinNT; I)

- Host: www.ora.com

- Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, */*

- Response

- HTTP/1.0 200 Document follows

- Date: Fri, 20 Sep 1996 08:17:58 GMT

- Server: NCSA/1.5.2

- Last-modified: Mon, 17 Jun 1996 21:53:08 GMT

- Content-type: text/html

- Content-length: 2482

- Parámetros en QUERY_STRING

- Método de Petición POST

- Request

- POST /cgi-bin/birthday.pl HTTP/1.0

- User-Agent: Mozilla/2.02Gold (WinNT; I)

- Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, */*

- Host: www.ora.com

- Content-type: application/x-www-form-urlencoded

- Content-length: 20

- month=august&date=24

- Response

- HTTP/1.0 200 Document follows

- Date: Fri, 20 Sep 1996 08:17:58 GMT

- Server: NCSA/1.5.2

- Last-modified: Mon, 17 Jun 1996 21:53:08 GMT

- Content-type: text/html

- Content-length: 2482

- (body of document here)*

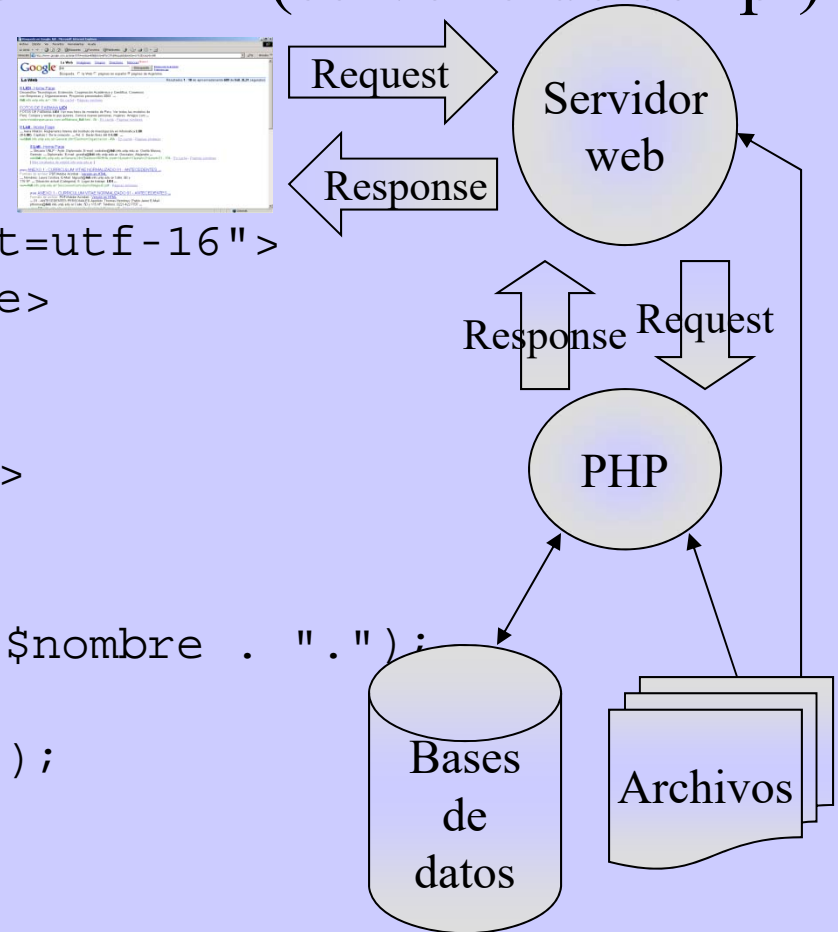
- Parámetros en CONTENT_STRING

- ISAPI / NSAPI
 - No es un programa
 - Es una DLL
 - Entonces
 - No requiere una creación por requerimiento
 - Varios hilos de ejecución
 - Mayor facilidad para mantenimiento de estados
 - Especificación
 - Funciones
 - GetExtensionVersion()
 - HttpExtensionProc()
 - Estructura de datos
 - EXTENSION_CONTROL_BLOCK

- PHP, ASP, ...

- Scripts insertado en el código HTML (server side script)

```
<html>
<head>
<meta content="text/html; charset=utf-16">
<title>Demostración de PHP</title>
</head>
<body>
  <h1>Programa PHP ejecutado</h1>
  <?php
    if ($nombre != "")
      echo ("Bienvenido, " . $nombre . ".");
    else
      echo ("No sé tu nombre");
  ?>
</body>
</html>
```



- Servlets
 - Clase Java
 - Extiende a la clase `javax.servlet.http.HttpServlet`
 - Redefine las operaciones
 - `doGet` (para método GET)
 - `doPost` (para método POST)
 - Ambas operaciones reciben como parámetros a los objetos
 - `HttpServletRequest`
 - `HttpServletResponse`
 - Contenido en un “Contenedor” (Tomcat)
 - La primera vez que se utiliza, se crea dentro del contenedor y allí permanece.
 - Cada acceso crea un nuevo hilo de ejecución (posibles problemas con recursos compartidos)

- Ejemplo de Servlet

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

public class PruebaServlet extends HttpServlet {
    protected void procesarPetición(HttpServletRequest request,
        HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html"); // Establecer tipo de
        salida
        PrintWriter out = response.getWriter ();
        out.println("<html><head>");
        out.println("<title>Demostración de servlet</title>");
        out.println("<meta content=\"text/html; charset=utf-17\">");
        out.println("</head><body>");
        String nombre = request.getParameter ("nombre");
        if (nombre != null) out.println("Bienvenido, " + nombre +
        "<p>");
        else out.println ("No sé tu nombre.<p>");
        out.println("</body></html>");
        out.close();
    }
}
```

- Ejemplo de Servlet (cont.)

```
protected void doGet(HttpServletRequest request,
HttpServletRequest response)
    throws ServletException, IOException {
    procesarPeticion(request, response);
}
protected void doPost(HttpServletRequest request,
HttpServletRequest response)
    throws ServletException, IOException {
    procesarPeticion(request, response);
}
}
```

Seminario de Lenguajes (opción PHP) – La Web

- JSP (Java Server Pages)
 - Scripts insertado en el código HTML (server side script)
 - El primer acceso genera un servlet que atiende el requerimiento
 - Los siguientes accesos se envían directamente al servlet

- Por ejemplo

```
<!-- Página prueba.jsp -->
<html>
<head><title>Prueba de JSP</title></head>
<body>
<%
    String s = request.getParameter ("nombre");
    if (s != null)
        out.println ("Bienvenido, " + s + ".");
    else
        out.println ("No sé tu nombre");
%>
</body>
</html>
```

- El ejemplo anterior genera

```
// Fichero prueba_jsp.java
public class prueba_jsp extends HttpServlet {
    protected void processRequest(HttpServletRequest
        request,

        HttpServletResponse response)
        throws ServletException, IOException {
        PrintWriter out = response.getWriter();
        out.println("<html>")
        out.println("<head><title>Prueba de
JSP</title></head>");
        out.println("<body>");
        String s = request.getParameter("nombre");
        if (s != null)
            out.println("Bienvenido, " + s + ".");
        .....
    }
}
```

Seminario de Lenguajes (opción PHP) – La Web

- Se vio que:
 - URL
 - Protocolo
 - Nombre de dominio
 - Path
 - Recurso
 - Parámetros
 - Request
 - Método
 - Recurso

```
http://www.yyy.com/doc1.html
```

```
GET /doc1.html HTTP/1.0
Accept: www/source
Accept: text/html
Accept: image/gif
User-Agent: Lynx/2.2 libwww/2.14
From: rchampre@lidi.info.unlp.edu.ar
/* esto es una linea en blanco */
```

- Pero ...
 - El servidor web se identifica por su IP
 - Cómo se establece la relación entre el nombre de dominio y el número de IP?
 - Cómo se asegura que no existan nombres de dominio repetidos?
 - => administración de nombres de dominio
- En Argentina administra Cancillería por medio de <http://www.nic.ar>