



Revista digital edición especial

PROGRAMACIÓN

Cómo crear una marquesina dinámica
 Como crear objetos en javascript
 Cómo hacer una aplicación con MVS C
 Optimizando el Código de PHP



Seguridad en Struts
 Vulnerabilidad de enrutadores
 Nos vigilan

SEGURIDAD

DISEÑO

Crear tablas redondeadas en HTML
 Hojas de Estilo



Instalación de PostgreSQL 8 para Linux
 Comenzar con tomcat J2EE
 Instalar Tomcat 5.5.x cooperando con Apache 2

APLICACIONES

EDITORIAL

Esta es una forma de darle las gracias a todos los colaboradores que han hecho posible este proyecto y que día con día sigue creciendo, es muy motivante ver que estamos tomando fuerza como un comunidad.

Bueno haber que les parece esta idea de hacer una **revista electrónica** libre, en la cual hemos puesto algunas de las participaciones de la comunidad. La idea es que Hagamos una cada mes, hay algunos artículos que no hemos puestos ya que están pensados para la siguiente publicación la cual será mas completa.

Estamos sujetos a sugerencias y aceptamos todo tipo de ayuda para realizar este material de libre distribución...

Escríbenos a info@mygnet.com

Visítanos a <http://www.mygnet.com>

CONTENIDO

Comunidad

La comunidad mygnet	3
La unión hace la fuerza	4

Programación

Combos enlazados	6
Como crear objetos en JavaScript	8
Como crear una clase en java que lea desde el teclado.....	10
Como crear una pagina con una marquesina dinámica.....	12
Como hacer una aplicación con Visual Studio 6 y Visual C++	14
Flash Remoting y amfphp	19
Listas enlazadas.....	21
Optimizando el código PHP	23
Un piano en c usando el altavoz del pc.....	25

Seguridad

Nos vigilan?	27
Vulnerabilidad de enrutadores cisco pone en riesgo a Internet.....	28
Seguridad en struts.....	29

Diseño

Crear tablas redondeadas en html	33
Hojas de estilo [css] versión 1	35

Aplicaciones

Instalación de postgresql 8 para Linux	41
Comenzar con tmcate J2EE	45
Instalar tomcat 5.5.x cooperando con Apache 2	47

Breves

Algoritmos genéticos	49
Comandos Linux en Windows.....	50
e-learning, una efectiva forma de capacitación	52
Noticias.....	54

Trucos

Trucos	58
--------------	----

La comunidad MYGNET



myGnet
Mi Grupo Net (My Group Net).

Anteriormente **mygnet** solo estaba conformado por dos colaboradores, Gustavo Santiago y Martín R. Mondragón donde el camino a seguir era proporcionar herramientas de desarrollo para clientes y usuarios de **mygnet**, pero la visión pronto cambio a medida que varias personas nos contactaron de diferentes lugares pidiendo apoyo y otras más para apoyarnos, de las cuales quiero destacar el administrador de <http://programacionphp.net> el cuál nos hizo muy buenas sugerencias con respecto a seguridad del sistema de posibles bugs, entonces aquí surge la idea de realizar un gran proyecto donde todos los programadores, desarrolladores, diseñadores, administradores y demás personas talentosas nos pudiéramos unir para formar una comunidad donde lo único que se pretende es fomentar el conocimiento, por que sabemos que no podemos ser buenos en todas las cosas, pero más sin embargo si dominamos una parte la cuál puede ser el complemento que podemos transmitir y que le servirá a alguien, de igual manera nosotros podemos solicitar ayuda ó encontrarla.

Siempre se nos van a presentan problemas, y quizás tardemos días, semanas, meses o quizás nos rendimos por no poder resolverlo ó no encontrar a alguien que nos apoye, esa es la idea, lo que nosotros realicemos y todo el esfuerzo que nos lleve llegar a ello, lo compartamos para que a alguien más le pueda servir; lo analice y lo aprenda, por que solo así vamos a trascender, por que podemos tener la mejor idea del mundo pero sino la comunicamos se perderá como muchas que hemos tenido.

Pero se que solos no podremos lograr esta tarea, así que hagamos un gran equipo juntos, una gran comunidad ONLINE, en la cual el objetivo es aprender y saber comunicar esas buenas ideas.

Agregando a esto **mygnet** carece de fines lucrativos, no COBRA ni COBRARA por ser parte de este proyecto y además no permitiremos que personas lucren con esto mismo.

Misión

Contribuir con desarrolladores, administradores, diseñadores y demás personas interesadas del mundo de habla hispana, para formar una comunidad en la cuál compartiremos experiencias y conocimientos.

Objetivos

Construir con todos los colaboradores un acervo de códigos, artículos, manuales, tutorales y herramientas que puedan ser útiles para otros.

Apoyar en la medida y disposición que tenga cada uno de los colaboradores a compañeros que soliciten nuestra ayuda.

Contactos

Martín R. Mondragón
martin@mygnet.com

Gustavo Santiago L.
gustavo@mygnet.com

La unión hace la fuerza y ahora se puede



Por Pablo Gutiérrez
Director Técnico en el Grupo ESABE

Nada original por cierto la frase del título de este artículo, pero al buscar uno que reflejase de lo que va, sin suda este era el más adecuado, me explico.

¿Os imagináis que dentro de la Comunidad creásemos un gran proyecto de Software?, dado que ya somos más de 400 y me consta que muchos muy buenos, que además tenemos mucho desarrollado, aunque no sepamos, salvo excepciones, que tiene cada uno ó que es capaz de hacer, sólo tendríamos que tener muy claro que cosas hacer, con qué fin, como nos podría beneficiar tanto individualmente, como a todo el Colectivo. Muchos hemos pensado alguna vez, que estábamos parados ó limitados por culpa de no encontrar la herramienta ó software adecuado, además de hacernos perder muchas horas en búsquedas infinitas, ya que siempre o casi siempre tropezábamos en lo mismo "el idioma", y no es que no lo sepamos, empezamos nuestros proyectos en castellano y casi siempre tropezábamos con el idioma, porque las grandes herramientas ó tardan en salir en versiones en castellano ó simplemente nunca salen de su inglés original. Todo ello y más, nos ha llevado a pensar ¿Y por qué, no me lo hago?, Y de golpe, el jarro de agua fría ¡¡¡tropee mil horas!!!, con lo que la realidad nos colocaba en nuestro sitio ¡¡¡ Sólo no puedo, además .xxx no lo domino !!!.

Entonces piensas en Open Source y a parchear, pero no porque inmediatamente ves que ese gran proyecto en el que estás pensando, debería llevar esto y aquello y también lo otro, y ¿porque no?, algo más, en resumen multitecnología, multilinguaje, multimétodo, en definitiva multitud, y eso no había

Open Source que lo soporte, y digo había, porque ayer he leído la noticia que está más

abajo, y entonces, se hace posible y ahora pregunto ¿Y por que no nos metemos en ese gran proyecto?, hablemos de ello, haced propuestas, a ver en que acaba todo esto. Os espero... Pablo Gutiérrez

Publicación de la nueva licencia de software libre: LPO.

22-08-2005 El grupo de desarrollo del proyecto openXpertya acaba de publicar la licencia de software libre denominado Licencia Pública openXpertya.

(LPO) Esta licencia está basada en la nueva licencia de software libre CDDL ("Common Development and Distribution License"), un nuevo tipo de licencia desarrollada por SUN y presentada el 1 de Diciembre del 2.004, enviada al OSI el 14 de diciembre de 2004, y aprobada el pasado 14 de enero del 2.005)

Esta es también la primera traducción legal al español de la licencia CDDL, adaptada a la legislación al derecho español y de la Unión Europea y la Licencia Pública a openXpertya

(LPO) es la primera licencia española de software libre que permite la mayor libertad posible de utilización al usuario....

CDDL está basada en la licencia MPL 1.1 (Mozilla Public License), pero agrega ciertas condiciones no incluidas en la mayoría de las licencias open

source. Por ejemplo, CDDL permite tener cualquier tipo código dentro de la misma solución. Usando terminología de open source, CDDL no es viral, como es el caso de la vieja licencia GPL (General Public License, 1991), y por lo tanto posibilita la inclusión de cualquier otro tipo de código, sea cual sea la licencia del mismo, en la solución completa (la licencia GPL por ejemplo, está fuertemente limitada, y sólo permite el linkado con el propio código GPL, haciendo en la práctica un monopolio por parte de la organización GNU y la fundación para el software libre). Adicionalmente la licencia CDDL sólo se adjudica a la totalidad del producto al que hace referencia y no a sus partes. Dichas partes podrían estar bajo esta licencia o bajo cualquier otro tipo de licencia al extraerse del software original completo.



El proyecto openXpertya adopta esta licencia LPO para posibilitar la inclusión y reutilización de software derivado o adaptado de

otros proyectos, que de otra manera no podría ser utilizado junto.

Esto posibilita la utilización del ERP con virtualmente cualquier otro código o producto, cuya licencia no prohíba expresamente su utilización o linkado con código de terceros. Así la licencia LPO está formada por la propia licencia CDDL fielmente traducida y adaptada a la legislación española y un Addendum final haciendo referencia a las definiciones iniciales de dicha licencia en el caso del producto de software libre openXpertya.

La licencia openXpertya posibilita además expresamente el relicenciamiento bajo la licencia GNU LGPL, o en su lugar cualquier otra que respete las condiciones la propia licencia LPO; de aquellas partes del código o librerías extraídas del código nuevo del proyecto openXpertya.

La utilización de esta licencia LPO por parte del grupo de desarrollo de openXpertya es muy interesante, toda vez que la propia aplicación openXpertya está realizada en Java, y SUN planea utilizar la licencia CDDL

original para el lanzamiento de próximas versiones del entorno de desarrollo en Java, JDK; como ya hizo en el pasado liberando el sistema operativo Solaris.

La licencia LPO está abierta de manera que cualquier otro proyecto de software libre pueda utilizarla o usarla para crear su propia licencia, sabiendo en todo momento que los usuarios del código licenciado dispondrá de la máxima libertad posible en la utilización del software, actual y futura y que dicha licencia está aprobada por la OSI.

Las intenciones de la comunidad de desarrolladores de openXpertya, que publica esta licencia son la libre utilización de la misma por parte de terceros y la cesión posterior de la custodia inicial de dicha licencia a una Fundación que la promocióne y la utilice en la creación de software libre empresarial absolutamente libre, sin limitaciones de ningún tipo en su utilización.

La licencia LPO puede ser obtenida en la parte pública de la Web del proyecto openXpertya:

<http://www.openxpertya.org> en el apartado de documentación. También es posible la discusión, observaciones y opiniones en los foros del proyecto acerca de dicha licencia y obtener mayor información registrándose.

Actualizada: 22/08/2005 (Todo un descubrimiento, Pablo Gutiérrez).

Combos enlazados en ASP

Por Cosme Olvera Alonso

Voy a poner este artículo, que e visto que piden mucho, sobre como enlazar dos cuadros desplegables con el resultado de dos tablas de bd, espero les sea util, funciona con asp y javascript y solo borraré algunas partes que no afecten la ejecución del programa y que no comprometa la seguridad de la compañía para que trabajo, el código es el siguiente: (la base de datos es sql server

```
<html>
<head>
<meta http-equiv="Content-Type"
content="text/html; charset=windows-
1252">
<title>Captura</title>
<%CONSULTA PARA OBTENER LOS DATOS
Conexion="Provider=SQLOLEDB.1;Password=*
***;Persist Security Info=True;User
ID=****;Initial Catalog=****;Data
Source=*****"
set
Cnn=server.CreateObject("ADODB.Connectio
n")
Cnn.ConnectionString=Conexion
Cnn.open
Sql = "Select * from detalles "
set rs=Cnn.Execute(sql)
x=0
%>
<script language = "JavaScript">
// FUNCION DE COMBO BOX COMBINADO
function validar(formulario) {
var checkOK = "0123456789";
var checkStr = formulario.T1.value;
var allValid = true;
var decPoints = 0;
var allNum = "";
for (i = 0; i < checkStr.length; i++) {
ch = checkStr.charAt(i);
for (j = 0; j < checkOK.length; j++)
if (ch == checkOK.charAt(j))
break;
if (j == checkOK.length) {
allValid = false;
break;
}
allNum += ch;
}
if (!allValid) {
alert("Escriba sólo números.");
formulario.T1.focus();
return (false);
}
return (true);
}
function validar2(formulario) {
var checkOK = "0123456789";
var checkStr = formulario.T2.value;
var allValido = true;
var decPoints = 0;
var allNum = "";
for (i = 0; i < checkStr.length; i++) {
ch = checkStr.charAt(i);
for (j = 0; j < checkOK.length; j++)
if (ch == checkOK.charAt(j))
break;
if (j == checkOK.length) {
allValido = false;
break;
}
allNum += ch;
}
if (!allValido) {
```

```
alert("Escriba sólo números.");
formulario.T2.focus();
return (false);
}
return (true);
}

function ver1(clave)
{
Renglon="Servicios";
RengDet=document.getElementById(Renglo
n);
RengDet.style.display="";
}
function ver(clave)
{
Renglon="mes";
RengDet=document.getElementById(Renglo
n);
RengDet.style.display="";
}
function sublist(inform, selecteditem)
{
inform.subcatagory.length = 0
var option<%= trim(count) %> = new
Option('Escoje una opción',0);

inform.subcatagory.options[inform.subc
atagory.length]=option<%=
trim(count)%>;
<%
count= 0
y=0
do while not rs.eof
%>
x = <%= trim(y) %>;
subcat = new Array();
subcatagorys =
"<%= (rs("id_partida"))%>";
subcatagoryof =
"<%= (rs("detalle"))%>";
subid= "<%= (rs("id_detalle"))%>";
subcat[x,0] = subcatagorys;
subcat[x,1] = subcatagoryof;
subcat[x,2] = subid;
if (subcat[x,0] == selecteditem) {
var option<%= trim(count) %> = new
Option(subcat[x,1], subcat[x,2]);

inform.subcatagory.options[inform.subc
atagory.length]=option<%=
trim(count)%>;
}
<%
count = count + 1
y = y + 1
rs.movenext
loop
rs.close
%>
}
function esconde(form)
{
if ((form.T1.value != "") &&
(form.T2.value != ""))
{ form.Enviar.style.visibility =
"visible"; }
else {
form.Enviar.style.visibility =
"hidden"; }
}
</script>
</script>
</head>
<body>
<table border="0" width="100%"
id="table1" cellspacing="0"
```


Como crear objetos en javascript



Por Gustavo Rodríguez

Programación orientada a objetos en Javascript

A través de un ejemplo explicaré como se crea un objeto usando javascript.

Si ya tenés alguna idea de POO y javascript y quieres ver sólo como se unen, puedes ir directamente a la sección de código y [descargarte el ejemplo](#).

Se supone que el lector ya tiene alguna noción de programación orientada a objetos y de javascript, por lo que no empezaré por los fundamentos de ninguna de las dos cosas, de lo que existe muchísimo material en la red, sino que desarrollaré un ejemplo corto de como se crea un objeto personalizado en javascript.

Javascript es un lenguaje orientado a objetos y cuando se escribe por ejemplo:

```
document.write("Escribo en la
página");
estamos llamando al método write del
objeto document.
```

En este ejemplo crearemos un objeto con dos propiedades, un método propio y sobrecargaremos el método estándar *toString*.

Primero vamos a crear la clase del objeto con una función que recibe un argumento.

Dentro de una etiqueta de script escribimos el siguiente código:

```
// Esta función define la plantilla
del objeto (la clase)
function UnaClase(Argumento)
{
    //Definimos una propiedad y le
asignamos el argumento de la función
    this.Propiedad1 = Argumento;

    //Definimos otra propiedad con un
valor nulo
    this.Propiedad2 = null;

    //Definimos un método personalizado
    this.UnMetodo = _metodo;
```

```
//Sobrecargamos el método toString
this.toString = _toString;
}
```

Como vemos el constructor del objeto asigna a la primera de sus propiedades el valor pasado como argumento, pero la otra se crea con un valor null. A esta segunda propiedad podemos darle valor después.

A los métodos los define exactamente igual que las propiedades, solo que en este caso el valor asignado a la derecha del signo igual, es el nombre de una función. Tanto para el método propio, como para sobrecargar el método *toString*.

Después creamos la función que implementa el método propio. Escribimos:

```
//Función que implementa el método
personalizado
function _metodo()
{
    alert("Este es un método del
objeto");
}
```

Por último el método que sobrecarga el método *toString*. En el script ponemos:

```
//Función que implementa la
sobrecarga del método toString
function _toString()
{
    return "La primera propiedad es: "
+ this.Propiedad1 + "; " +
        "La segunda
propiedad es: " + this.Propiedad2 +
"."
}
```

Listo. La plantilla del objeto ya está completa, ahora vamos a usarla. Creamos una variable objeto de clase Una Clase:

```
//Variable que crea un objeto de la
clase creada
var MiObjeto = new
UnaClase("Gustavo");
```

Después asignamos un valor a la segunda propiedad:

```
MiObjeto.Propiedad2 = "Rodriguez";
```

Terminamos con el script. Ahora vamos al código HTML. En el cuerpo de la página creamos dos botones y al evento onclick le asignamos: a uno; directamente el método del objeto "UnMetodo()" y al otro el método alert del objeto window, al que le pasamos como argumento el método toString del objeto que acabamos de crear. El código a escribir en la página es el siguiente:

```
<p>
  <!--
  Este botón mostrará la cadena devuelta por el
  método personalizado
  -->
  <input type="button"
    value="Llamamos el metodo personalizado"
    onclick="MiObjeto.UnMetodo()" />
</p>
<p>
  <!--
  Este otro mostrará la cadena devuelta por el
  método toString sobrecargado
  -->
  <input type="button"
    value="Llamamos a toString"
    onclick="alert(MiObjeto.toString())" />
</p>
```

Ahora sólo hay que guardar el código y verlo con un navegador.

El código completo es el siguiente:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01
Transitional//EN">
<html>
<head>
<title>GAR :: Objetos personalizados en
JavaScript</title>
<script type="text/javascript">
<!--
/*****
*****
Programación Orientada a Objetos en JavaScript

Crearemos un objeto con dos propiedades, un método
personalizado y
una sobrecarga del método toString.

*****/

// Esta función define la plantilla del objeto (la
clase)
function UnaClase(Argumento1, Argumento2)
{
  //Propiedades
  this.Propiedad1 = Argumento1;
  this.Propiedad2 = Argumento2;

  //Definimos un método personalizado
  this.UnMetodo = _metodo;

  //Sobrecargamos el método toString
  this.toString = _toString;
}

//Función que implementa el método personalizado
function _metodo()
{
  alert("Este es un método del objeto");
}

//Función que implementa la sobrecarga del método
```

```
toString
function _toString()
{
  return "La primera propiedad es: " + this.Propiedad1
+ "; " +
      "La segunda propiedad es: " +
this.Propiedad2 + "."
}

//Variable que crea un objeto de la clase creada
var MiObjeto = new UnaClase("Gustavo");
MiObjeto.Propiedad2 = "Rodriguez";

// -->
</script>

</head>
<body>
  <p>
    <!--
    Este botón mostrará la cadena
    devuelta por el método personalizado
    -->
    <input type="button"
      value="Llamamos el
      metodo personalizado"
      onclick="MiObjeto.UnMetodo()" />
  </p>
  <p>
    <!--
    Este otro mostrará la cadena devuelta
    por el método
    toString sobrecargado
    -->
    <input type="button"
      value="Llamamos a
      toString"
      onclick="alert(MiObjeto.toString())" />
  </p>
</body>
</html>
```

Como crear una clase en java que lea desde el teclado



Por Fredy Ramírez Porfirio

Bueno creo que empezare por decir que para esto utilizaremos dos clases contenidas en el paquete de entrada/salida de java (java.io), estas clases son: InputStreamReader y BufferedReader, además de el flujo de entrada System.in. Bueno para no enredarlos con tanta palabrería empecemos:

1.- Primero tenemos que importar el paquete io de java de la siguiente manera:

```
import java.io.*;
```

2.- Construimos una función para la lectura de entradas de tipo String, en esta función utilizaremos las clases InputStreamReader y BufferedReader, además del objeto System.in:

```
public static String readStr() {
    String sdato = "";
    try{
        // Definir un flujo de
        caracteres de entrada: "readIn"...
        BufferedReader readIn =
        new BufferedReader(new
        InputStreamReader(System.in));
        // Leer. La entrada,
        finaliza al pulsar la tecla
        Entrar...
        sdato =
        readIn.readLine();
    } catch( IOException e) {

        System.err.println("Error: " +
        e.getMessage());
    }
    return sdato; // Devolver el
    dato tecleado..
}
```

Si construiremos nuestra clase con solo este método de lectura nos quedaría de esta manera:

```
import java.io.*;
```

```
public class readData {
    public static String readStr()
    {
        String sdato = "";
        try{
            BufferedReader
            readIn = new BufferedReader(new
            InputStreamReader(System.in));
```

```
        sdato =
        readIn.readLine();
        } catch( IOException e)
        {
            System.err.println("Error: " +
            e.getMessage());
        }
        return sdato;
    }
}
```

Así que des pues de esto compilamos, la clase para crear el *.class, una vez compilada la clase podemos utilizarla en nuestros programas, nada mas que solo nos serviría para datos de tipo String. Aquí un ejemplo de como se utilizaría:

```
import java.io.*;
class testRead01{
    public static void main(String
    args[]){
        String Nombre = "";

        System.out.println("Escribe tu
        nombre: "); Nombre =
        readData.readStr();
        System.out.println("Tu
        nombre es: " + Nombre);
    }
}
```

Compila y ejecuta este programa, tu salida deberá ser algo parecido a esto:

Escribe tu nombre: Fredy Ramirez Porfirio...
Tu nombre es: Fredy Ramirez porfirio...

3.- Bien ahora crearemos los métodos para los demás tipos de todos utilizando el método para los tipos String para capturar la entrada desde el teclado, haciendo solo simples conversiones de datos, para conseguir los demás tipos.

Para datos enteros cortos:

```
public static short readShort() {
    try {
        return
        Short.parseShort(readStr());
    } catch(NumberFormatException
    e) {
        return Short.MIN_VALUE;
    }
}
```

Para datos Enteros:

```
public static short readInt() {
    try {
        return Integer.parseInt(readStr());
    } catch (NumberFormatException e) {
        return Integer.MIN_VALUE;
    }
}
```

Para datos enteros largos:

```
public static short readLong() {
    try {
        return Long.parseLong(readStr());
    } catch (NumberFormatException e) {
        return Long.MIN_VALUE;
    }
}
```

Para datos de punto flotante:

```
public static float readFloat() {
    try {
        Float f = new Float(readStr());
        return f.floatValue();
    } catch (NumberFormatException e) {
        return Float.NaN;
    }
}
public static float readDouble() {
    try {
        Double d = new Double(readStr());
        return d.doubleValue();
    } catch (NumberFormatException e) {
        return Double.NaN;
    }
}
```

Ahora si, si unimos estos métodos con la clase anterior, nuestra clase estará completa, quedando de la siguiente manera.

```
import java.io.*;
public class readData {
    public static String readStr() {
        String sdata = "";
        try {
            BufferedReader readIn = new
            BufferedReader(new InputStreamReader(System.in));
            sdata = readIn.readLine();
        }
    }
}
```

```
    } catch (IOException e) {
        System.err.println("Error: " + e.getMessage());
    }
    return sdata;
}
public static short readShort() {
    try {
        return
        Short.parseShort(readStr());
    } catch (NumberFormatException e) {
        return Short.MIN_VALUE;
    }
}
public static short readInt() {
    try {
        return
        Integer.parseInt(readStr());
    } catch (NumberFormatException e) {
        return Integer.MIN_VALUE;
    }
}
public static short readLong() {
    try {
        return Long.parseLong(readStr());
    } catch (NumberFormatException e) {
        return Long.MIN_VALUE;
    }
}
public static float readFloat() {
    try {
        Float f = new Float(readStr());
        return f.floatValue();
    } catch (NumberFormatException e) {
        return Float.NaN;
    }
}
public static float readDouble() {
    try {
        Double d = new Double(readStr());
        return d.doubleValue();
    } catch (NumberFormatException e) {
        return Double.NaN;
    }
}
}
```

Espero que les sirva de algo, por el momento esto es todo.

Cómo crear una página con una marquesina dinámica



Por Gustavo Alberto Rodríguez

Cómo crear una página con una marquesina dinámica

Antes que nada una aclaración. Ojo con los navegadores, porque la etiqueta Marquee no está dentro de HTML 4.01, así que esto puede no verse correctamente en navegadores que no sean Internet Explorer.

Este artículo intenta solamente, explicar como cambiar el texto dentro de la etiqueta. Por lo tanto, no entraré en detalles sobre la etiqueta en sí y sus propiedades, mas allá de las necesarias para el ejemplo. Se supone que el lector conoce algo de HTML y del lenguaje JavaScript.

1.- Crear la página

```
<html></html>
```

Esto es lo elemental de una página web. Claro esto mostrará sólo una página en blanco. Todas las demás etiquetas dentro del par de etiquetas `<html></html>`

2.- La cabecera

```
<head></head>
```

Dentro de la cabecera va información que no se muestra pero que es usada por los navegadores. Aquí colocaremos el título, el estilo visual de la página, el código JavaScript (puede ir también otro lugar), y otra información.

3.- El cuerpo

```
<body></body>
```

Dentro del cuerpo del documento va todo el resto de la página. Es fundamentalmente, lo que se vé.

4.- El set de caracteres

```
<meta http-equiv="Content-Type"
content="text/html; charset=iso-8859-1">
```

Las etiquetas meta, pueden tener varios usos. Uno de ellos es la de definir el conjunto de caracteres que usará la página. Un set de caracteres incorrecto hará que en lugar de nuestros acentos, eñes, etc., aparezcan cosas extrañas.

5.- El título

```
<title>GAR :: Una marquesina
dinámica</title>
```

Este es el título de la ventana del explorador.

6.- El estilo

```
<style type="text/css">
<!--
div {
    position: absolute;
    width: 300px;
    background-color: #FFCCCC;
    color: #CCE6FF;
    font-size: xx-large;
}
-->
</style>
```

El estilo define el aspecto visual de la página. Todo lo que tenga que ver con colores, tamaño del texto, etc., debe ir definido en una página de estilo. Puede ir dentro de la misma página como en nuestro caso o, preferentemente, en una página externa a la que se hace referencia con un link como el siguiente:

```
<link href="estilo.css"
rel="stylesheet" type="text/css"
>
```

7.- El script

```
<script type="text/javascript">
<!--
// Cadenas de texto que se irán
cambiando para cada iteración
var Textos = new Array();
Textos[0] = "Texto 1";
Textos[1] = "Texto 2";
Textos[2] = "Texto 3";
Textos[3] = "Texto 4";
Textos[4] = "Texto 5";
Textos[5] = "Texto 6";
Textos[6] = "Texto 7";
Textos[7] = "Texto 8";
Textos[8] = "Texto 9";
Textos[9] = "Texto 10";

//Función que cambia el texto de
la marquesina
var Vuelta = 0;
function CambiarTexto(Objeto)
{
    if (Vuelta==10) Vuelta = 0;

    Objeto.innerText =
Textos[Vuelta];

    Vuelta++;
}

// -->
</script>
```

En el script defino un array para contener las cadenas que se mostrarán en la marquesina. Uno por vez. También defino la función que será llamada por el evento start de la etiqueta <marquee>, pasándole el id como argumento.

Esta función antes que nada verifica que el valor de la iteración actual esté dentro de los límites del array. Si no es así vuelve a cero.

Después, cambia el contenido dentro de las etiqueta de apertura y cierre, por la cadena contenida en el array correspondiente a la vuelta actual. Por último incrementa el contador de la vuelta.

8.- Un div

```
<div></div>
```

Ya dentro del cuerpo (la parte visible de la página). Coloco una etiqueta div, con el solo objeto contener la marquesina para implementar su estilo visual.

9.-La marquesina

```
<marquee id="marquesina"
onstart="CambiarTexto(marquesina)"
>
    Aquí va el texto de la
marquesina, que será cambiado.
</marquee>
```

Por último la etiqueta <marquee> es la que define la marquesina en sí. Tiene muchas atributos que se pueden establecer.

Yo para este ejemplo solo la identifiqué poniendo "marquesina" en el atributo id y le pasé la cadena con el nombre de la función a ejecutar con su argumento en el atributo "onstart".

Código completo

```
<html>
<head>
<meta http-equiv="Content-Type"
content="text/html; charset=iso-8859-
1">
<title>GAR :: Una marquesina
dinámica</title>
<style type="text/css">
<!--
div {
    position: absolute;
width: 300px;
background-color: #FFCCCC;
color: #CCE6FF;
font-size: xx-large;
}
//-->
</style>
<script type="text/javascript">
<!--
// Cadenas de texto que se irán
cambiando para cada iteración
var Textos = new Array();
Textos[0] = "Texto 1";
Textos[1] = "Texto 2";
Textos[2] = "Texto 3";
Textos[3] = "Texto 4";
Textos[4] = "Texto 5";
Textos[5] = "Texto 6";
Textos[6] = "Texto 7";
Textos[7] = "Texto 8";
Textos[8] = "Texto 9";
Textos[9] = "Texto 10";

//Función que cambia el texto de la
marquesina
var Vuelta = 0;
function CambiarTexto(Objeto)
{
    if (Vuelta==10) Vuelta = 0;

    Objeto.innerText = Textos[Vuelta];

    Vuelta++;
}
```

```
// -->
</script>
</head>
<body>
    <div>
        <marquee
id="marquesina"
onstart="CambiarTexto(marquesina)"
>
            Aquí va el
texto de la marquesina, que será
cambiado.
        </marquee>
    </div>
</body>
</html>
```

Esto es todo. En Internet Explorer funciona perfectamente. En otros exploradores lo más probable es que nó, porque ya dije que la etiqueta <marquee> no es parte de HTML 4.01.

Cómo hacer una aplicación Win32 con Visual Studio 6 y Visual C++



Por landanohr

Seguro que estáis cansados del típico ejemplo del "Hola Mundo" cada vez que empezáis a usar un lenguaje de programación, pero eso sí, siempre con la típica y sosa ventana de comandos...

Pues bien, eso es lo que vamos a hacer aquí, pero de forma un poco más interesante: vamos a hacer una aplicación Win32 con ventanas paso a paso.

Requisitos: poca cosa, tener un mínimo conocimiento de C++ y el Visual Studio 6 a mano...

Pasos: esto es un poco más largo que el apartado de requisitos, así que vallamos por partes...

1-. Creación del proyecto

Lo primero es crear el proyecto, si queréis crear un workspace o no me trae sin cuidado, no tiene importancia...

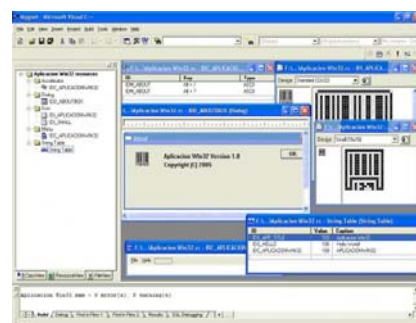
Pues eso: archivo -> nuevo -> proyecto; elegimos aplicación Win32, ponemos el nombre "Aplicacion Win32" y OK. Después seleccionamos la aplicación hola mundo y finalizamos.

2-. Revisando los recursos

Ya tenemos el proyecto creado, si damos a ejecutar vemos la ventanita con la aplicación inicial:

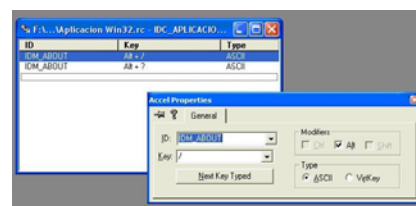


Que queda bastante curiosa, salvo por el hecho de que tenemos un menú en inglés (porque mi visual esta en inglés, sino no), así que vamos a ver cómo lo ponemos en español, y ya de camino le damos un vistazo a la los recursos:



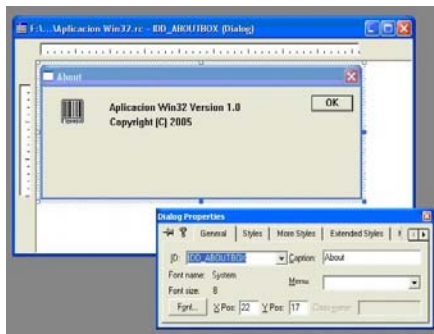
Bueno, no os asustéis que en realidad es poca cosa. Tenemos los siguientes apartados:

- Accelerator: aceleradores de teclado



Donde tenemos el id del elemento que se "acelera" (por ejemplo una opción del menú) y la combinación de teclas a usar.

- Dialog: cuadros de diálogo que tenemos en el proyecto



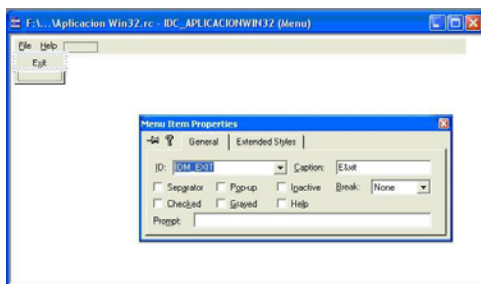
Son los cuadros de diálogo que tenemos en la aplicación. Inicialmente tenemos el cuadro de acerca de... que viene por defecto, aunque es bastante soso

-Icon: iconos de la aplicación



Iconos de la aplicación, inicialmente vienen los que tendrá el ejecutable.

-Menu: los menús de la aplicación

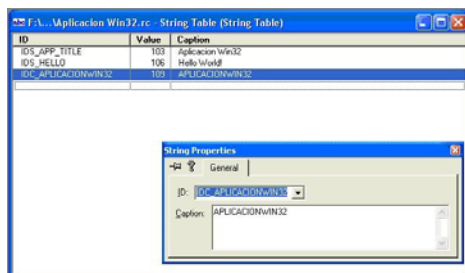


Tenemos el menú actual, en inglés, pero si vemos las propiedades (segundo botón sobre el elemento o doble clic) de cada elemento podemos cambiar el texto que aparece en el campo caption (el & indica la letra que sale subrayada).

Cambiamos los textos y tenemos nuestro menú:



-String Table: tabla de cadenas



Tenemos las cadenas que se usan por ejemplo como nombre externo de la aplicación, tipo de aplicación o cadenas simplemente para mensajes de error, mostrarlas en pantalla... lo que se os ocurra

3-. Personalizando Acerca de...

Ya hemos visto que el cuadro de Acerca de... queda bastante soso así como está, ¿por qué no le damos un toque personal? Además, ya de camino podemos ver cómo crear otros cuadros de diálogo y cómo capturar los eventos.

Para hacer esto tenemos que saber que un cuadro de diálogo se compone de una serie de elementos de distintos tipos, como pueden ser etiquetas, imágenes, botones, barras de desplazamiento, etc... que podemos insertar haciendo uso de la barra de herramientas que sale al abrir el cuadro de diálogo:



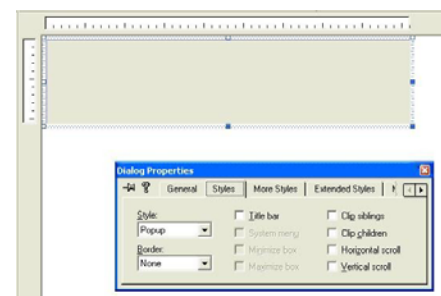
Cada uno de los elementos puede tener o no utilidad para nosotros y cada uno de los elementos tiene unas propiedades diferentes, aunque realmente todos o casi todos son los típicos elementos que tenemos en cualquier cuadro de diálogo de cualquier aplicación.

El sistema consiste en insertar un elemento y mediante las propiedades modificar la forma y estilo que tiene, la cuestión es ir probando hasta que quede a nuestro gusto, el código que pueda necesitar (por ejemplo al pulsar un botón) vendrá después.

Bien, comencemos:

- Primer paso:

Lo primero que vamos a hacer es borrar todos los controles y cambiar el estilo de visualización para que no tengamos barra de control (en la pestaña estilo de las propiedades ponemos estilo a popup borde a ninguno). Queda así:



- Segundo paso:

Si os habéis fijado, hay dos reglas en la parte superior y en la parte izquierda, valen para delimitar el espacio dentro del cuadro de diálogo en el que vamos a poder incluir controles. Vamos a delimitarlo para que en los bordes no quede nada.

También vamos a incluir un texto estático (tercer icono de la barra de controles), pinchando en el icono y dentro del cuadro de diálogo. Ponemos cualquier texto y personalizamos dentro de propiedades todo lo que nos parezca bien.

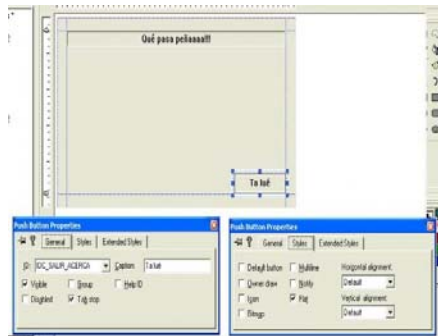


Un detalle a tener en cuenta: ya habréis visto que el control tiene un id con un valor por defecto, en este caso no importa mucho porque normalmente un texto estático no se modifica, pero tened en cuenta a la hora de insertar controles que

mediante los id's posteriormente vais a poder acceder a ellos, así que poned un id adecuado.

- Tercer paso:

Ya el cuadro está algo más original que el inicial, pero tenemos un problema, ¿cómo salimos? (ya se que está lo típico de alt+f4 o esc, pero no tenemos nada en lo que dar con el ratón). Vamos a insertar un botón (por supuesto personalizado):



Fijaos en que ya sí hemos puesto un identificador más aclarativo, que nos servirá para hacer que la ventana se cierre.

Antes de dar funcionalidad al botón veamos algo. En el archivo .cpp tenemos el método

```
LRESULT CALLBACK WndProc(HWND hWnd,
UINT message, WPARAM wParam, LPARAM lParam)
```

que es el que se encarga de controlar los eventos de la pantalla principal, es decir, las acciones que se han de realizar, como por ejemplo

Controlar los eventos del usuario, con el código que está dentro de

```
case WM_COMMAND:
```

Redibujar la pantalla, con el código que está dentro de

```
case WM_PAINT:
```

“Destruir” la aplicación, con el código que está dentro de

```
case WM_DESTROY:
```

Pues bien, si miramos el código que se encarga de controlar los eventos de usuario:

```
wmId = LOWORD(wParam);
wmEvent = HIWORD(wParam);
// Parse the menu selections:
switch (wmId)
{
case IDM_ABOUT:
    DialogBox(hInst,
(LPCTSTR)IDD_ABOUTBOX, hWnd,
(DLGPROC)About);
    break;
case IDM_EXIT:
    DestroyWindow(hWnd);
    break;
default:
    return DefWindowProc(hWnd,
message, wParam, lParam);
}
```

Veremos el identificador IDM_ABOUT y el identificador IDM_EXIT, que son ambos de acciones del menú, es decir, cuando pulsamos una acción del menú es dentro de este switch donde vemos qué acción es la pulsada y decidimos qué hacer:

- con IDM_EXIT cerramos la aplicación (el método DestroyWindow)
- con IDM_ABOUT abrimos el cuadro de diálogo acerca de...

Para abrir el cuadro de diálogo el método usado es

```
DialogBox(hInst,
(LPCTSTR)IDD_ABOUTBOX, hWnd,
(DLGPROC)About)
```

y lo que nos interesa es:

- segundo parámetro IDD_ABOUTBOX, que dice cuál es el identificador del cuadro de diálogo que se ha de abrir

- cuarto parámetro About, que dice cuál es el método que se va a encargar de controlar los eventos del cuadro de diálogo que vamos a abrir (este método está más abajo)

Visto esto, vamos a pasar a dar funcionalidad al botón.

Si miramos el código del método About:

```
// Message handler for about box.
LRESULT CALLBACK About(HWND hDlg,
UINT message, WPARAM wParam, LPARAM lParam)
{
switch (message)
{
case WM_INITDIALOG:
    return TRUE;

case WM_COMMAND:
    if (LOWORD(wParam) == IDOK ||
LOWORD(wParam) == IDCANCEL)
    {
        EndDialog(hDlg,
LOWORD(wParam));
        return TRUE;
    }
    break;
}
return FALSE;
}
```

vemos que es similar al que acabamos de ver, y que también tiene un fragmento de código dentro de:

```
case WM_COMMAND:
```

que es el que se encarga de llevar a cabo las acciones para los eventos de IDOK (botón OK que tenía el cuadro de diálogo) y de IDCANCEL (botón Cancel que tenía el cuadro de diálogo). Como los botones los hemos quitado, podemos borrar ese código, y podemos incluir el siguiente para que el cuadro de diálogo se cierre al pulsar nuestro nuevo botón:

```
if (LOWORD(wParam) ==
IDC_SALIR_ACERCA)
{
    EndDialog(hDlg,
LOWORD(wParam));
    return TRUE;
}
```

Probad y veréis como el cuadro se cierra al pulsar el botón.

- Cuarto paso

Vamos ahora a “integrar” la ventana de acerca de en la ventana principal, para ello

lo que vamos a hacer es poner la ventana transparente y centrarla en pantalla.

Pues venga, hacedlo; lo único es entrar en las propiedades y seleccionar lo apropiado. Una vez hecho probad la aplicación... a que ya queda más curioso?... el único problema es que cuando desplazamos la ventana principal hacia un lado y volvemos a dar a la ayuda el cuadro sigue estando centrado en pantalla y queda feo, pero ya lo arreglaremos.



Es el momento de solucionar este problema, y de camino vamos a ver cómo en la creación de un cuadro de diálogo se puede modificar al propio cuadro que lo ha creado, aunque resulte raro...

La idea es mover y redimensionar la ventana principal de forma que el cuadro de diálogo de acerca de... quede totalmente dentro, pero eso sí, cuando cerremos el cuadro de diálogo la ventana principal tiene que quedar como estaba.

Como queremos mantener la disposición y el tamaño necesitamos una variable global donde almacenemos temporalmente esos valores, es la siguiente:

```
WINDOWPLACEMENT localizacionPrincipal;
```

el tipo es una estructura que almacena posición y tamaño (entre otros valores), ya veremos como la usamos, ahora pasemos a redimensionar la ventana principal.

Para hacer esto nos fijamos en el manejador de eventos para el cuadro de diálogo, el mencionado método About.

. en case WM_INITDIALOG: podemos meter el código que se ejecuta antes de mostrar el cuadro de diálogo, ponemos lo siguiente:

```
GetWindowPlacement(GetParent(hDlg),
&localizacionPrincipal);
```

/* hDlg es el manejador de nuestro cuadro de diálogo; con GetParent(hDlg) obtenemos el manejador de la ventana padre, es decir, la ventana principal.

```
GetWindowPlacement(GetParent(hDlg),
&localizacionPrincipal); lo que hace es almacenar en localizacionPrincipal la localización y el tamaño de la ventana cuyo manejador le pasamos, en este caso de la ventana principal*/
```

```
WINDOWPLACEMENT localizacion;
RECT posicion;
```

/* en la variable localizacion para almacenar la localización de nuestro cuadro de diálogo acerca de; pero como hemos dicho que realmente esta estructura tiene más parámetros necesitamos otra que sólo tenga los que necesitamos (localización y tamaño), esta es RECT, así que creamos la variable posicion*/

```
GetWindowPlacement(hDlg,
&localizacion);
posicion =
localizacion.rcNormalPosition;
```

/* extraemos la información que queremos de nuestro cuadro de diálogo y colocamos la posición y el tamaño del mismo en la variable posicion*/

```
SetWindowPos(
GetParent(hDlg),
HWND_TOP,
posicion.left-50,
posicion.top-50,
posicion.right-posicion.left+100,
posicion.bottom-posicion.top+100,
SWP_SHOWWINDOW);
```

/* SetWindowPos asigna una posición y un tamaño a una ventana; hacemos la llamada con la ventana principal, y las coordenadas necesarias para que su posición sea centrada al cuadro de diálogo pero su tamaño algo mayor, para que este quede contenido dentro. Los otros dos parámetros no tienen importancia, el primero es para que la ventana quede en lo "alto" del resto de ventanas y el segundo para que la muestre*/

```
return TRUE;
```

.en case WM_COMMAND: es donde cerramos la ventana tras pulsar nuestro botón, lo modificamos de la siguiente forma:

```
if (LOWORD(wParam) ==
IDC_SALIR_ACERCA)
{
SetWindowPlacement(GetParent(hDlg),
&localizacionPrincipal);
```

/* con esto restablecemos la ventana principal a como estaba inicialmente*/

```
EndDialog(hDlg, LOWORD(wParam));
return TRUE;
}
break;
```

Ya lo tenemos... Eso sí, supongo que tanta función rara no será de mucho agrado, es normal... Os recomiendo que le deis un vistazo a la ayuda de esas funciones hasta que os quede claro lo que hacen (seleccionando el texto de la función y dando a f1 del tirón tenemos la ayuda de la función).

Si ejecutáis la aplicación y probáis veréis que la cosa funciona, salvo en un caso: si tenemos la ventana maximizada y abrimos el cuadro, al cerrarla no queda del todo bien ☹...

Pero eso tiene fácil solución: quitamos el cuadro de maximizar, que es una solución cutre (a un problema realmente de la implementación interna de las librerías, pero bueno), pero que nos sirve para ver cómo hacerlo.

Pues venga, en el método InitInstance es donde se crea por primera (y única vez) la ventana principal, y es en la creación de la ventana donde podemos quitar el botón de maximizar. Tenemos la línea siguiente, que es dicha creación:

```
hWnd = CreateWindow(szWindowClass,
szTitle, WS_OVERLAPPEDWINDOW,
CW_USEDEFAULT, 0, CW_USEDEFAULT,
0, NULL, NULL, hInstance, NULL);
```

si después de la mala impresión que causa tanta cosa rara conseguimos mirar la ayuda de la función, veremos que el parámetro

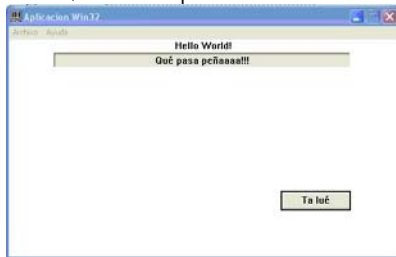
que hemos de cambiar para alcanzar nuestro fin es `WS_OVERLAPPEDWINDOW`, que literalmente, dice que es como

```
WS_OVERLAPPED + WS_CAPTION + WS_SYSMENU +  
WS_THICKFRAME + WS_MINIMIZEBOX + WS_MAXIMIZEBOX
```

siendo cada uno de estos algo más o menos raro, pero se ve claramente que `WS_MAXIMIZEBOX` es el que indica que la ventana tenga cuadro de maximizar. Pues bueno, lo ponemos todo menos ese, de la siguiente forma:

```
hWnd = CreateWindow(szWindowClass, szTitle,  
    WS_OVERLAPPED |  
    WS_CAPTION |  
    WS_SYSMENU |  
    WS_THICKFRAME |  
    WS_MINIMIZEBOX,  
    CW_USEDEFAULT, 0, CW_USEDEFAULT, 0, NULL, NULL,  
    hInstance, NULL);
```

Y listo, ahora si que todo funciona a la perfección.



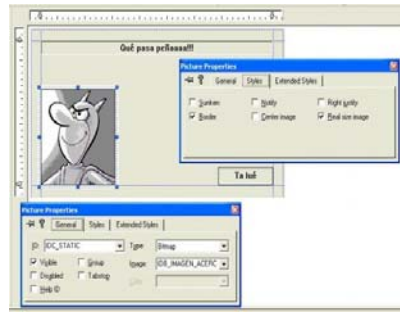
- Quinto paso

Por último vamos a insertar una imagen. Vamos a la parte de recursos, damos a nueva imagen y dibujamos cualquier cosa...

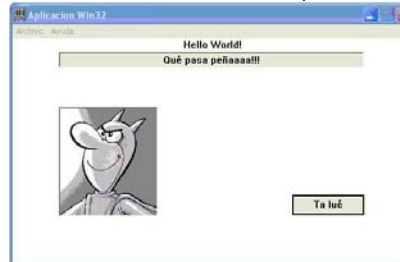


Eso mismo... joer que buen dibujante soy...

Ahora insertamos una imagen en el cuadro de diálogo y seleccionamos el bitmap que hemos creado:



Y este es el resultado de lo que hemos creado:



Que no es una gran maravilla, pero nos ha costado nuestro trabajo.

FIN???

Bueno, por ahora aquí queda la cosa, para empezar no está mal si habéis conseguido hacerlo.

De todas formas esto no tiene porqué acabar aquí... sólo hemos visto cómo empezar a hacer algo en una aplicación win32 y le hemos dado un rápido vistazo a algunos controles.

Si tenéis interés en ver otros controles o no sabéis cómo hacer alguna cosa me ponéis un comentario aquí abajo o me mandáis algún correito a landanohr@hotmail.com y veremos que se puede hacer.

Por supuesto si algo no queda claro lo mismo, que ya que hago algo que se entienda...

Bueno, pues espero que tanta letra os sirva de algo... hasta otra...

Flash Remoting y AMFPHP



Por Gustavo Santiago Lázaro

Hola colegas colaboradores he estado trabajando un poco con lo que es el flash remoting y bueno he encontrado resultados buenos y quiero compartir con ustedes un poco de lo aprendido.

Bueno obviamente antes que nada me gustaría poner algunos de los requerimientos para empezar a trabar con Flash remoting y AMFPHP:

Requerimientos:

- ❖ Flash MX 2004
- ❖ PHP 4.3.0 o superior
- ❖ AMFPHP (version 1.0)
- ❖ Flash components AS2
- ❖ Apache Web Server

Bueno la forma de instalar que es como yo lo hice es la siguiente, primero descargamos los componentes de Flash remoting AS2 http://www.macromedia.com/software/flashremoting/downloads/components/#flr_as2

Descargamos el archivo y lo ejecutamos. Cuando la instalación termine abrimos nuestro Flash MX 2004 y verificamos que en la parte de componentes aparezca el Remoting Connector

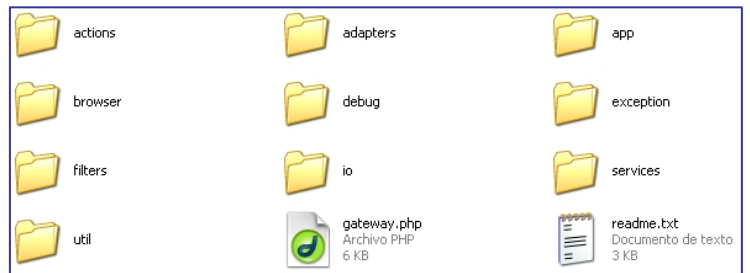


. Si esto aparece Nuestra inhalación se ha efectuado satisfactoriamente.

Ahora lo que tenemos que hacer es descargar el AMFPHP que es el gateway del flash remoting . lo descargamos: http://sourceforge.net/project/showfiles.php?group_id=72483.

Hago la aclaración que como vamos a trabar con los componentes de AS2(Action Scrip2) es muy importante descargar la versión 1.0 milestone de AMFPHP.

Una vez que lo descargamos, descomprimos el archivo y vamos a encontrar la siguiente estructura:



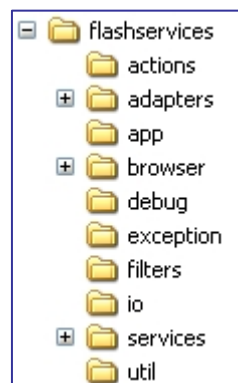
Ahora lo que tenemos que hacer es irnos a donde tenemos instalados nuestros sitios web, generalmente para el apache es el siguiente (Archivos de programa\Apache

Group\Apache2\htdocs), esto puede cambiar dependiendo la configuración de cada persona. Bueno una vez ahí creamos una carpeta llamada flashservices y copiamos ahí el contenido de nuestro archivo comprimido,

Quedando la siguiente estructura:

OK, una vez termina esto, podemos decir que hemos terminado el proceso de instalación de la aplicaciones para que el flash remoting funcione. Ahora solo nos falta hacer las correspondientes configuraciones y creaciones de servicios con los cuales vamos a trabajar.

Bueno anteriormente si se tenia que configurar el AMFPHP, pero ahora la verdad es que no requiere de modificaciones para que funcione. Como vemos en figura de arriba en la carpeta



flashservices que creamos tenemos una carpeta que se llama services, es en esta carpeta en donde se crearan cada unos de los servicios con las cuales vayamos a trabajar. Para este ejemplo crearemos un servicio muy básico.

Creamos un archivo nuevo con el nombre mygnet_component.php. Escribimos el siguiente código:

```
<?php

class mygnet_component
{ function mygnet_component()
{

$this->methodTable = array(

"suma" => array( "description"=> "suma dos numeros",

"access"=> "remote",

"arguments" => array ("uno","dos")

)

);

} //fin del arreglo de componentes

function suma($uno,$dos)

{return $uno+$dos; }

}

?>
```

Lo guardamos.

Ahora tenemos que hacer nuestra película en flash y poner la programación correspondiente.

Ok, Ahora viene lo bueno:

- Creamos una película flash.
- Nos vamos al Menús Ventana > Otros paneles > Bibliotecas comunes > Remoting
- Arrastramos los objetos RemotingClasses y RemotingDebugClasses a nuestra película.
- Ahora insertamos en siguiente código a nuestra película.

```
import mx.remoting.NetServices;
import mx.remoting.Connection;

mx.remoting.debug.NetDebug.initialize();
resultado = new Object();
resultado.suma_Result = function(data)
{trace("Total: "+data);
};

NetServices.setDefaultGatewayUrl("http://localhost/f
lashservices/gateway.php");
conn = NetServices.createGatewayConnection();
serv = conn.getService("mygnet_component",
resultado);
serv.suma(5,4);
```

Ahora si ejecutamos nuestra película y vemos el resultado.

Espero que esto les sea de ayuda para todos aquellos que van empezando con Flash Remoting

Listas enlazadas

Por Sergio

Listas enlazadas en Java:

Para empezar, habrá que explicar lo que es una lista enlazada. Es un vector de dimensión variable. Esto es que solamente utiliza la cantidad de elementos mínima necesaria.

Por ejemplo, supongamos que tenemos un programa que gestiona los datos de los alumnos de una asignatura y el programa está diseñado para una cantidad máxima de 50 plazas. Si en un año solamente hay 4 alumnos, no hay necesidad de declarar un vector de 50 elementos si solo vamos a usar 4, la memoria que necesitaría es mucho menor si se manejan las listas que si utilizamos vectores.

La forma que tiene de almacenar los valores es a través de nodos, donde guarda todos los datos del elemento más una referencia a otro nodo que representa el siguiente elemento de la lista.

En Java viene a ser algo parecido a:

```
public class Nodo{

public int dato;
public Nodo next;

public Nodo(int n){
    dato=n;
    next=null;
}
}

Public class Lista{
    private Nodo start;

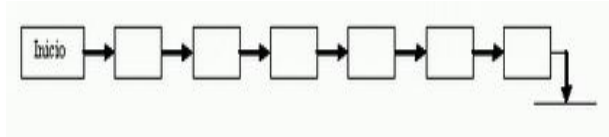
    public void add(int N){
        Nodo aux=start;
        if(aux==null){
            aux=new Nodo(N);
        }else{
            while(aux.next!=null) aux=aux.next;
            aux.next=new Nodo(N);
        }
    }

    public void remove(int N){
        Nodo aux=start;
        if(aux.dato==N){
            start=aux.next;
        }else{
            while(aux.next!=null && aux.next.dato!=N)
                aux=aux.next;
            if(aux.next!=null){
                aux.next=aux.next.next;
            }
        }
    }
}
```

Este ejemplo es una forma sencilla y simplona de manejar listas, pero tiene muchas lagunas como el acceso a un nodo, añadir en una posición distinta al final, etc. (que esperabais, no os voy a dar todo el trabajo hecho, solo quiero que entendáis como funciona esto).

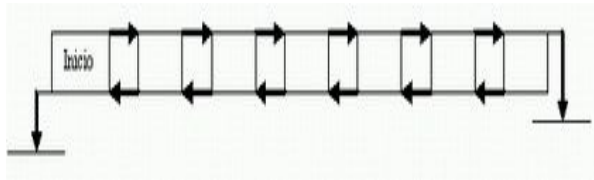
Este sistema tiene múltiples ventajas. Primero, solamente usa la cantidad de memoria estrictamente necesaria. Otra ventaja es que el límite de elementos no está definido, esto quiere decir que puede haber tantos elementos como memoria que pueda conservarlos.

Además de tener ventajas, también tiene sus desventajas. Por ejemplo, la forma de añadir o quitar elementos hay que definirla a través de métodos, al igual que acceder a un elemento para utilizar su información. A la hora de acceder a un elemento, hay que buscarlo antes de forma secuencial, es decir, de principio a fin, y esto en una lista larga con muchos datos en cada elemento puede ralentizar el programa bastante.

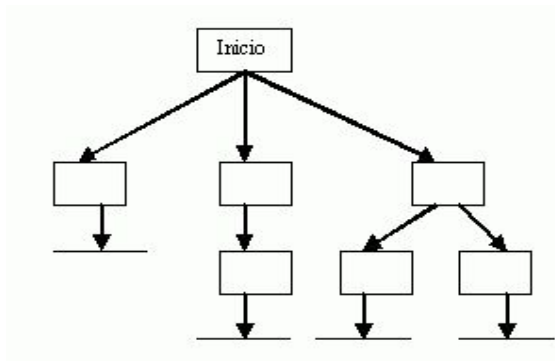


Esta es la estructura de una lista enlazada, pero hay más tipos de listas: Las doblemente enlazadas y los árboles.

Una lista doblemente enlazada tiene dos referencias a dos elementos, una al siguiente nodo o nulo si es el final de la lista, y otra al nodo anterior de la lista o nulo si es el inicio de esta.



Un árbol tiene como mínimo dos referencias a nodos, uno al nodo Padre y los otros a el/los nodo/s hijo. Esto significa que un nodo no puede tener más de un padre, pero no significa que no pueda tener más de un hijo.



Mi consejo es que si se pueden usar vectores y matrices, se usen, ya que a la hora de programar esto se puede volver un trabajo bastante tedioso, pero existen porque a veces es mejor usar listas y otras estructuras dinámicas a usar vectores.

OPTIMIZANDO EL CODIGO DE PHP



Por Rodrigo Arce

Muchas veces nos preguntamos como hacemos para bajar el tiempo del ejecución de nuestra aplicación en PHP, bueno aquí les dejo algunos tips para hacerlo.

Primero debemos preguntarnos **¿Por que necesito yo que sea mas rapida la aplicación?** Bueno si tu sitio Web tiene aproximadamente 5 visitas simultaneas y tu servidor es relativamente bueno, no creo que tengas ningún problema en cuestión de velocidad de carga. Los problemas surgen con muchos usuarios accediendo a nuestra Base de Datos y un servidor compartido donde no siempre se caracteriza por la rapidez de carga.

Ahora veamos la siguiente curiosidad.

Supongamos que nuestro servidor tiene 100 Mb de RAM disponibles para nuestra Web y no existe la fragmentación de memoria. Ahora comparemos 2scripts. El primero **ver_archivos.php** requiere 0.04 segundos de CPU y 10 Mb de RAM y el segundo **instalar_script.php** requiere 0.06 segundos de CPU y 5 Mb de RAM.

Cuando 10 scripts se ejecutan, *ver_archivos.php* se queda sin memoria ($10 \times 10 = 100$), en cambio *instalar_script.php* todavia cuenta con 50 Mb libres.

La onceava conexión de script *ver_archivos.php* necesita usar la Memoria Virtual, lo que aumenta aproximadamente el doble la velocidad de respuesta (ahora requiere 0.08 segundos). No nos olvidemos del segundo script *instalar_script.php* sigue corriendo a 0.06 segundos.

Como vimos en realidad la mejor

manera de mejorar un script no es solo hacer un código veloz. Una **performance optima** requiere un análisis del hardware, sistema operativo y software de soporte, así como un análisis de la Base de Datos y el Servidor

Bueno lo mas importante para determinar que parte del código debe ser optimizado y cual no necesitamos medir el tiempo de carga. Para esto he publicado dentro de la sección **Códigos** de esta Web un Script de una clase (**Benchmark Class**) que nos será muy útil para medir el tiempo de proceso de un script.

Conexiones	Segundos que requiere una conexión HTTP	Segundos que requiere 10 conexión HTTP	Segundos que requiere 11 conexión HTTP
ver_archivos.php	0.04	0.40	0.88 (runs out of RAM)
instalar_script.php	0.06	0.60	0.66

La clase Benchmark tiene dos tipos de mediciones:

La primera sirve para insertar dentro de un código, en la segunda ingresamos el código en una función y este nos devuelve un array con los datos, como podemos ver en **prueba.php**.

La optimización de código de PHP es algo que *debe hacerse solo en casos extremos* donde el script se haga demasiado lento. Este proceso consume muchas horas de re-escritura de código y suele hacerse al final del trabajo. Lo mejor que podemos hacer es incorporar estas practicas de mejoramiento de código para ponerlas en practica una vez que escribamos el código.

Algo muy importante que me olvidaba de decir es que a veces sacrificamos unos microsegundos de código para que nos sea mas "cómodo" el código, es decir que nos sea más legible a nosotros (los programadores) y más fácil actualizable.

Bueno ahora empezaremos a mejorar el código:

Uno de los problemas mas comunes en velocidad viene de los ciclos mal escritos ahora veremos como podemos mejorarlos con el siguiente código:

Primero declaramos un array. Lo que queremos es que nuestro código imprima en un comentario HTML (`<!-- -->`) de cada valor del array.

```
$mi_array =
array('1','2','3','4','5');
```

Bueno este es el primer ciclo (el peor de todos):

```
function ciclo1($mi_array
=null){
    for ($i =0;$i
<sizeof($mi_array);$i++){
        echo "<!--"
.$mi_array
[$i ] ." --->";
    }
}
```

Ahora veremos como mejorarlo un poco:

```
function ciclo2($mi_array
=null){
    $max =sizeof($mi_array);
    for ($i =0;$i <$max
;$i++){
        echo "<!--"
.$mi_array
[$i ] ." --->";
    }
}
```

Lo que hicimos aca fue poner el tamaño del array en una variable. Ahora veamos porque, leyendo la documentación de www.php.net mas especificamente la sección **FOR** dice

Al comienzo de cada iteración, se evalúa `expr2`. Si se evalúa como **TRUE**, el bucle continúa y las sentencias anidadas se ejecutan. Si se evalúa como **FALSE**, la ejecución del bucle finaliza.

Quiere decir que en cada pasada evaluamos si `$i` es menor al `sizeof` de `$mi_array`. Poniendo el `sizeof` en una variable mejoramos bastante el código en velocidad y en legibilidad.

```
function ciclo3($mi_array
=null){
    $max =sizeof($mi_array);
    for ($i =0;$i <$max
;$i++){
        $output
.="<!--"
.$mi_array
[$i ] ." --->";
    }
    echo $output;
```

Aquí en el Tercer Ciclo en vez de llamar a **echo** (al final de este artículo lea mas sobre **echo**) cada vez que hagamos una pasada en el array, guardamos el contenido en una variable (`$output`) y luego imprimimos todo junto. La sentencia **echo** (no es una función) demanda bastante uso de memoria,

```
function ciclo4($mi_array
=null){
    $max =sizeof($mi_array);
    $output =array();
    for ($i =0;$i <$max
;$i++){
        array_push($output,"<!--"
.$mi_array
[$i ] ." --->");
    }
    echo implode("",$output);
}
```

Ahora lo que hicimos fue en vez de ponerlas todas en un mismo `$output`, lo ponemos en un nuevo array (también llamado `$output`) que lo sumamos todo junto con la función `implode`. Esto se debe a que la concatenación (suma de cadenas) es también lenta. Bueno esto fue todo (lo que se :P) de ciclos. Ah, me olvidaba el mejor código en realidad es este que pondré abajo:

```
function ciclo5($mi_array
=null){
    echo '<!--',implode(' ---
><!--', $mi_array),' --->';
}
```

ECHO

Debemos usar la función **ob_start()**; en el comienzo del código y **ob_end_flush()**; al final para poder usar libremente el `echo`. Dichas funciones guardan el contenido de los

`echo` en una variable y la imprime al final.

También debemos tratar de usar comillas simples (`'`) en vez de dobles (`"`) puesto que las comillas dobles deben parsear su contenido para chequear las variables.

Otra implementación buena es poner comas (`,`) en vez de puntos (`.`) en la concatenación de cadenas en `echo` y hace el trabajo de la misma manera.

Un "piano" en c usando el altavoz del pc



Por landanohr

Lo de piano es por llamarlo de alguna forma... más bien sería "como hacer que suene el altavoz cuando das a una tecla" A lo que vamos, os pongo el código y lo comento sobre la marcha:

/* para no complicarnos hacemos un método principal y lo metemos todo dentro, que tampoco es tanto*/

```
main()
{
  /*declaramos un montón de
  variables... evidentemente se pueden
  declarar constantes o globales,
  pero eso es lo de menos*/
```

/*para que suene el altavoz del pc tendremos que hacer uso de entrada/salida, así que tendremos que usar las direcciones de entrada/salida o "puertos", como lo queráis llamar*/

/* puerto del teclado, para saber la tecla pulsada*/

```
int tecla=0x60;
```

/* puerto el altavoz del pc donde indicamos la frecuencia del sonido*/

```
int nota=0x42;
```

/* puerto del altavoz del pc donde hacemos que suene*/

```
int sonido=97;
```

/* vamos a elegir siete teclas, cada una una nota, así que asignamos a cada letra su código hexadecimal*/

```
int A=0x1e;
int S=0x1f;
int D=0x20;
int F=0x21;
int G=0x22;
int H=0x23;
int J=0x24;
```

/*cada nota está determinada por una frecuencia, y cada frecuencia se compone para el altavoz de 4 valores hexadecimales; definimos un par de variables con la frecuencia de cada una de las notas*/

```
int do1=0x11;
int do2=0xca;
int re1=0x0f;
int re2=0xda;
int mi1=0x0e;
int mi2=0x1f;
int fa1=0x0d;
int fa2=0x5a;
int so1=0x0b;
int so2=0xe3;
int la1=0x0a;
int la2=0x98;
int si1=0x09;
int si2=0x6f;
```

/* y esto nada, variables auxiliares*/

```
long int i;
char a;
```

/* bucle infinito para que el programa termine cuando demos control+c*/

```
for(;;)
{
```

/*vamos a lo interesante, primero veamos que vamos a usar:

byte importb(byte puerto) devuelve el valor que hay en ese puerto; por ejemplo, tras pulsar una tecla miramos en el puerto del teclado importb(tecla) y obtendremos el código de la tecla pulsada outportb(byte puerto, byte valor) coloca el valor en el puerto especificado.

Por supuesto hay puertos de sólo lectura, de escritura y de lectura escritura, pero vamos, eso ahora como que no nos importa...

Pasamos al procedimiento:*/

/* esperamos hasta que se pulse una tecla*/

```
while (!kbhit()) ;
```


/* "cogemos la tecla" para que su código esté en el puerto del teclado*/

```
getch();
```

/* ahora, dependiendo de la tecla pulsada hacemos lo siguiente: ponemos en el puerto nota la nota que queremos tocar, como la nota tiene dos bytes y sólo podemos poner uno lo tenemos que hacer con dos instrucciones outportb; si os fijáis son al mismo puerto las dos, pero no se pisan los valores, el altavoz ya se encarga de almacenarlos por separado en otro sitio después ponemos en pantalla la tecla pulsada, para hacer bonito*/

```
if (inportb(tecla)==A)
{ outportb(nota,do2);
  outportb(nota,do1);
  printf("a");
}
else if(inportb(tecla)==S)
{ outportb(nota,re2);
  outportb(nota,re1);
  printf("s");
}
else if(inportb(tecla)==D)
{ outportb(nota,mi2);
  outportb(nota,mi1);
  printf("d");
}
else if(inportb(tecla)==F)
{ outportb(nota,fa2);
  outportb(nota,fa1);
  printf("f");
}
else if(inportb(tecla)==G)
{ outportb(nota,so2);
  outportb(nota,so1);
  printf("g");
}
else if(inportb(tecla)==H)
{ outportb(nota,la2);
  outportb(nota,la1);
  printf("h");
}
else if(inportb(tecla)==J)
{ outportb(nota,si2);
  outportb(nota,si1);
  printf("j");
}
else
{
```

/*si no se pulsa ninguna de las teclas del "piano" lo que hacemos es meter silencio*/

```
outportb(nota,0xff);
outportb(nota,0xff);
}
```

/* ahora viene la parte en la que hacemos que suene. Así:*/

/*cogemos el valor que había en el puerto de sonido, que no nos importa, pero que no podemos pisar así como así(al menos no totalmente) porque son valores del altavoz*/

```
a=inportb(sonido);
```

/*aplicamos una máscara or para que los dos últimos bits se pongan a uno. La cuestión es que suena si los dos últimos bits están a uno y no suena si están a cero*/

```
a=a|0x03;
```

/*mandamos el valor al puerto de sonido, es la forma de hacer que suene el altavoz. Lo que hace es coger la frecuencia que tiene almacenada (la que le hemos pasado antes en el puerto nota) y la reproduce*/

```
outportb(sonido,a);
```

/* esperamos un tiempo para que se escuche bien*/

```
for (i=0; i<2500000; i++) ;
```

/* y ahora tenemos que hacer que pare el sonido*/

/* volvemos a coger el valor del puerto de sonido*/

```
a=inportb(sonido);
```

/* ponemos los dos últimos bits a cero con otra máscara*/

```
a=a&0xfd;
```

/* y lo mandamos al puerto de sonido: silencio*/

```
outportb(sonido,a);
}
```

Sea, pues ya tenemos un pequeño ejemplo de cómo usar la entrada/salida en c...

Pos nada, espero que os guste.

Nos vigilan?

Por Daniel Esteban Vasco Gómez

Los minúsculos puntos de algunas impresoras no son simples manchas

San Francisco, E.U.

Un grupo de defensa de la privacidad descubrió lo que significan los minúsculos puntitos que dejan algunas impresoras a color: se trata de un código que permite saber desde qué máquina y cuándo se copió el documento.

La Fundación Fronteras Electrónicas (EFF), un grupo de defensa de los derechos civiles con sede en San Francisco (California), llevaba tiempo sospechando de unos pequeñísimos puntitos que algunas impresoras láser a color esconden secretamente en cada documento.

Tras cerca de tres años de investigaciones con la ayuda de cientos de voluntarios de todo el mundo, que enviaron documentos impresos en diferentes impresoras a las oficinas de la EFF, el grupo por fin develó el misterio.

Los Servicios Secretos de E.U. admitieron la existencia de un acuerdo con varios fabricantes de impresoras para identificar productos falsificados, pero hasta ahora no se conocía la naturaleza de la información codificada en cada documento.

Se trata, en cualquier caso, de una explicación que no satisface a la Fundación. "Es extraño que te traten como a un criminal sin ni siquiera saberlo", señaló a Efe Rebecca Jeschke, portavoz del grupo.

Según dijo David Schoen, técnico de EFF, los puntitos de al menos una línea de impresoras codifican el día y la hora

en que el documento se imprimió, así como el código de serie de la impresora.

Canon y Xerox se encuentran entre la lista de fabricantes que incluyen estos códigos (la EFF dispone de una lista al completo en la página de Internet

<http://www.eff.org/Privacy/printers/list.php>).

Programa para decodificar

Las pequeñas marcas son puntos amarillos de menos de unos milímetros de diámetro, repetidos en cada página del documento y tan pequeños que no se ven a simple vista. Para observarlos, se necesita una luz azul y una lupa o un microscopio (la EFF también dispone en su página web de instrucciones para los curiosos que deseen ver las marcas por sí mismos).

Este centro comenzó el proyecto con la línea de impresoras de Xerox DocuColor, una máquina de gama alta más fácil de encontrar en oficinas o centros de fotocopiado que en las casas particulares.

Un equipo liderado por Schoen comparó diferentes documentos impresos en la misma máquina y, tras observar las similitudes y diferencias, encontró la manera de decodificarlo.

"Hasta ahora, sólo hemos roto el cifrado de las impresoras DocuColor -señala Schoen-, pero creemos que otros modelos de otras fabricantes incluyen la misma información en los puntos".

La organización ha puesto a disposición del público en su página Web un programa automático para que cualquiera puede decodificar los puntitos que deja su impresora.

Xerox admitió previamente que estaba "compinchado" con el Gobierno en este sistema de rastreo, pero aseguró que solamente los Servicios Secretos podían descifrar la información.

Contra la privacidad

Por su parte, el Servicio Secreto mantiene que sólo usa esta información para investigaciones relacionadas con falsificaciones.

Pero, según advierte el EFF, no hay una legislación específica que impida que el gobierno pueda abusar de esta información.

"Los movimientos democráticos clandestinos que publican panfletos políticos y religiosos siempre necesitarán del anonimato de una simple página", dijo Lee Tien, uno de los abogados del grupo.

"Esta tecnología se lo pone más fácil a los gobiernos a la hora de encontrar a los disidentes", dijo Tien.

El abogado del EFF señala que el descubrimiento tiene graves implicaciones, ya que estos códigos dan al gobierno y a la industria privada "más posibilidades para debilitar nuestra privacidad con un equipo que se utiliza cotidianamente, como las impresoras".

Tien cree que los técnicos del EFF tienen mucho trabajo por delante: "La próxima gran interrogante es: ¿qué otros apaños se están cociendo para asegurarse de que nuestra tecnología nos traiciona?". Buena pregunta.

Vulnerabilidad de enrutadores Cisco pone en riesgo a Internet



Por Cesar Gutiérrez

En el marco de una conferencia de seguridad informática organizada por Black Hat, un experto en seguridad IT, de nombre Michael Lynn, demostró algo que muchos creían imposible: acceder externamente a un sistema, vulnerar un enrutador de Cisco y obtener así control total de la red intervenida.

En teoría, si hackers malintencionados adquieren las mismas habilidades demostradas por Lynn, todas las redes informáticas podrían ser puestas fuera de servicio al ser posible re-dirigir arbitrariamente el tráfico digital. Una acción coordinada podría, de hecho, doblegar a toda Internet.

Según la publicación Tom's Hardware Guide, Lynn no presentó una descripción detallada del procedimiento, pero aportó los detalles suficientes que permitirían a un experto reconstruir los pasos necesarios.

Lynn aprovechó vulnerabilidades conocidas para transportar valores de memoria dentro del enrutador. La mayoría de estas vulnerabilidades han sido corregidas, pero todo indica que un gran número de enrutadores aún no han sido actualizados.

La motivación de Lynn habría sido motivar a los administradores de redes a entender, de una vez por todas, que estas vulnerabilidades pueden ser explotadas para realizar actividades hasta ahora impensables.

Hasta el miércoles 27 de julio, Lynn trabajaba en Internet Security Systems (ISS) donde formaba parte del internacionalmente reconocido grupo X-Force R&D. Justo antes de que Lynn realizar su intervención en la conferencia de Black Hat, Cisco se vio presa del pánico. Los representantes de la compañía exigieron que la presentación de Lynn fuese excluida del material impreso de la conferencia, y consiguieron que los organizadores del evento destruyeran 2000 CDs con la presentación completa. ISS dijo estar de acuerdo con Cisco en el sentido que Lynn

habría ido demasiado lejos al revelar los métodos de intrusión de los enrutadores, e instruyó al experto atender los requerimientos de Cisco.

Lynn reaccionó renunciando a su cargo en ISS. Al acercarse al podio para disertar sobre otro tema, optó lisa y llanamente por realizar su presentación del tema Cisco, tal como estaba previsto originalmente. Cisco inició inmediatamente acciones legales tendientes a impedir a Lynn realizar disertaciones similares a futuro. Parte de la argumentación presentada por Cisco apunta a que Lynn habría cometido actos ilegales al conseguir información sobre el funcionamiento de los enrutadores de la compañía. Lynn desestimó lo anterior señalando que "sólo había usado un 'disassembler' corriente para reconstruir el código fuente de Cisco".

Seguridad en Struts



Por landanohr

Antes de empezar os hago una advertencia: si queréis entender completamente el contenido del artículo sería conveniente que tuvierais conocimientos de *struts* al menos básicos. Y sería recomendable que tuvierais alguna práctica.

Aún así, a los que no sepáis de *struts* os puede servir la idea para aplicarla con otra tecnología.

Empezamos:

La información que hay por Internet referente al tema es escueta y aunque se encuentran muchos artículos al final todos tienen un enlace a alguno de los dos siguientes, que son los únicos interesantes. Por supuesto, como no podría ser de otra manera, en inglés:

<http://www.onjava.com/pub/a/onjava/2004/02/18/strutssecurity.html?page=last&x-order=date>

<http://www-128.ibm.com/developerworks/web/librariya/wa-appsec/?ca=dgr-Inxw16ServletsTaglibsStrutsSecurity#listing2>

Los dos los artículos lo primero que hacen es decir que *struts* no es una tecnología que lo haga todo, sino que pretende ser una plataforma básica abierta que sirva de base. Después te sueltan el royo de que en una aplicación que se precie se han de contemplar multitud de usuarios con diferentes roles y blablabla... y por fin pasan a dar la solución, que es similar.

Mi solución:

No me voy a poner a traducir ninguno de los artículos, sino que os voy a contar la solución que estoy adoptando, que no es que sea una cosa novedosa, es la misma que la de los artículos, sólo

que explicada por mí (de hecho la solución no es mía, sino que es la que hemos adoptado en un proyecto que estoy realizando con más personas).

Consideraciones iniciales:

Lo primero que tenemos que tener en cuenta es que necesitamos un modo de identificación de usuarios en el sistema (*login, password*). Por este motivo supondremos para la realización de las restricciones que la identificación se ha realizado y que en *session* tenemos un objeto usuario (que será un registro – u objeto que represente dicho registro – de la tabla usuarios a la que haremos referencia posteriormente).

Como a priori no hemos definido ni los diferentes roles que tendrán los usuarios de la aplicación, ni las secciones en las que se dividirá nuestra aplicación ni los diferentes tipos de acceso que tendremos; y además, lo ideal sería que en caso de tener que modificar alguna de estas cosas fuera lo más rápido posible, la solución que vamos a ver consiste en tener una base de datos (o una serie de tablas dentro de la base de datos que usemos en la aplicación, si usamos alguna) donde almacenaremos toda esta información.

Además tendremos dos niveles de restricción, área de acceso (si el usuario puede o no acceder a la página) y tipo de operación (qué operaciones puede realizar el usuario de las disponibles en la página). Por ejemplo, si tenemos una página con una serie de datos editables habrá usuarios que no puedan acceder a ella, los habrá que puedan acceder pero no cambiar los datos (modo lectura) y los habrá que puedan acceder y cambiar los datos (modo lectura/escritura).

Requisitos de información:

Vamos a ver las tablas que necesitamos incluir en la base de datos y para qué sirve cada una. Evidentemente, dependiendo de

los tipos de usuarios que se vayan a tener en la aplicación puede que tengamos que meter alguna tabla más o quitar alguna porque no sea necesaria; pero más o menos será de una forma parecida a esta.

- Usuarios

Entidad que recogerá los datos de los usuarios de la aplicación. En ella guardamos los datos personales de los usuarios, el identificador y la contraseña, y asociamos al usuario un perfil, que será el que defina los permisos.

Atributo	Tipo	BB DD	Comentario
idUsuario (PK)	numérico	not null	Código del usuario
login (UK)	texto	not null	Login en la aplicación
password	texto	Null	Contraseña del usuario
nombre	texto	Null	Nombre del usuario
apellid o	texto	Null	Apellidos del usuario
...	<!--[if !supportEmpty Paras]--> <!--[endif]-->	Null	Resto de datos personales...
idPerfil (FK)	numérico	not null	Id del Perfil que define los permisos del usuario.

PK: clave primaria
UK: clave única
FK: clave ajena

- Perfiles

Entidad que recogerá los perfiles de uso de la aplicación. Realmente la tabla es poca cosa, pero nos da un *idPerfil* que será el que usemos en los distintos permisos.

Atributo	Tipo	BD	Comentario
idPerfil (PK)	numérico	not null	Código del perfil
perfil	texto	not null	Nombre del perfil

- Tipo de acceso

Entidad que recogerá los tipos de acceso, que pueden ser de creación, edición, eliminación, consulta...

Atributo	Tipo	BBD D	Comentario
idTipoAcceso (PK)	numérico	not null	Código del tipo de Acceso
codigo	Numérico	not null	Código constante para identificar el tipo de acceso
tipoAcceso	texto	not null	Descripción del tipo de acceso

- Secciones de la aplicación

Entidad que recogerá las distintas secciones de la aplicación sobre las que queremos definir permisos de acceso. Por ejemplo en una aplicación para una tienda podemos tener una zona de para el almacén, otra para los trabajadores de la tienda y otra para los clientes.

Atributo	Tipo	BBD D	Comentario
idSeccionAplicacion (PK)	numérico	not null	Código de la Sección de la Aplicación
codigo	texto	not null	Código constante para identificar la sección
seccionAplicacion	texto	not null	Descripción de la sección

- Permisos

Entidad que recogerá los permisos que los distintos perfiles tienen sobre las diferentes secciones. En esta tabla es donde asociamos el perfil con una sección de la aplicación y lo dotamos de una serie de permisos.

Atributo	Tipo	BB DD	Comentario
idPermiso (PK)	numérico	not null	Código del permiso
idTipoAcceso(FK)	numérico	not null	Código del tipo de acceso
idSeccionAplicacion(FK)	numérico	not null	Código de la sección de la aplicación
idPerfil(FK)	numérico	not null	Código del perfil

Implementación de las restricciones sobre el sistema

Anteriormente hemos dicho que hay dos tipos de restricción, de acceso y de operación. Es ahora cuando realizaremos una diferenciación entre ellas, a la hora de implementarlas.

- Restricción de acceso

Pasando ya al lenguaje de *struts*, podemos entender una restricción de acceso como la posibilidad del usuario en cuestión de ver o no el resultado de un determinado *Action*, ya que se supone que antes de ver una página determinada se habrá de ejecutar el correspondiente *Action*, que es el que genera los datos a mostrar en la página.

Lo que queremos entonces es que si un usuario tiene permisos se ejecute el *Action* y se pase a la página jsp correspondiente; o que en el caso de no tener permisos pasemos a una página de error. Lo vamos a realizar en tres pasos:

1-. Superclase para los Action

Partiendo de que tenemos varias secciones en la aplicación y que cada sección se encuentra implementada por un conjunto de

Activo, vamos a colocar sobre los *Action* de cada sección una superclase que en su método *execute* lo primero que compruebe sean los permisos de acceso del usuario. Sería una cosa más o menos así:

```
public abstract class
RestriccionAction extends Action

{
    public ActionForward
execute(...) {
doRestriccion(...);
return _execute(...);
}
    public abstract ActionForward
_execute(...);
private void doRestriccion(...)
throws AccesoNoPermitidoException
{

/* Este metodo realiza las
comprobaciones de acceso con
el perfil del usuario que
haya en sesión */

}
}
```

De tal forma que los *Action* de la sección heredarán de este y será de la forma:

```
public class CuarquierCosaAction
extends RestriccionAction
{
    public ActionForward
_execute(...) {

/* Este es el método execute que
tenía el Action, pero le cambiamos
el nombre para que se ejecute el de
la clase padre que implementa la
restricción */

}
}
```

2-. Extender ActionMapping

Extenderemos la clase *ActionMapping* de *struts* para darle más atributos, ya que interesará tener para cada *Action* datos relativos a la sección a la que pertenece, así que le añadiremos el atributo *applicationZone*.

```
public class SecurityActionMapping
extends ActionMapping
{
    private String zone = null;
    public String
getApplicationZone() {
return zone;
}
    public void
setApplicationZone(String newZone)
{
zone = newZone;
}
}
```

A partir de ahora, el parámetro de tipo *ActionMapping* del método *execute* de los *Action* será de tipo *SecurityActionMapping* (podremos hacer un *casting* dentro del método a un objeto de esta clase con el objeto de entrada). Pero para hacer que esto sea así, antes hemos de completar el siguiente paso.

3-. Inclusión del atributo en el struts-config.xml

En el fichero *struts-config.xml* modificamos las definiciones de los objetos *Action* incluyendo el atributo *className="ruta_de_paquetes.SecurityActionMapping"* y una directiva *set-property*. De la siguiente forma

```
<action path="..."
[... ]
className="ruta_de_paquetes.Secu
rityActionMapping"
[... ] >
<set-property
property="applicationZone"
value="CODIGO_DE_SECCION" />
<forward [...] />
</action>
```

El código de sección deberá coincidir el valor del atributo *codigo* de algún registro de la base de datos de la tabla de secciones de la aplicación.

- Restricción de operación

Ya hemos dicho que la restricción de operación se puede entender cómo la necesidad de un determinado perfil para disponer de determinadas opciones o controles en las páginas JSP que se visualizan, como pueden ser permisos de edición o no de algunos datos. (por supuesto, si un usuario tiene restringido el botón editar – por poner un ejemplo – de una página jsp, también deberá tener restringida la ejecución del *Action* que realiza la edición, mediante una restricción de acceso)

Las restricciones de operación deberán tener reflejo en las páginas JSP que muestran la información y los controles para operar sobre ella; por ejemplo, si no se pueden editar los datos, los campos de edición deberán aparecer en

sólo lectura y el botón de grabar debe estar oculto. Lo haremos en dos pasos:

1-. Colocación de permisos de operación en request

En el *Action* que implementa la restricción de acceso correspondiente se establecerán en ámbito de *request* los permisos de operación que el usuario que hay en *sesión* tiene en la sección del *Action* que se invocó para llegar hasta el JSP. Los permisos posibles dependerán realmente de cómo sea nuestra aplicación y de, por supuesto, los permisos que tenga el perfil del usuario.

Por ejemplo: Si el usuario tiene permisos para modificar datos, se ha de establecer en ámbito de *request* el atributo *canEdit* a *true*

```
request.setAttribute("canEdit", new
Boolean(true));
```

2-. Restricción en las páginas JSP

En las páginas JSP se han de restringir la aparición o la funcionalidad de los diferentes controles dependiendo de los permisos que tengamos establecidos en *request*. Por ejemplo, si tenemos *canEdit* el botón de edición deberá aparecer y los campos ser editables; pero si no tenemos *canDelete*, el botón de borrar deberá permanecer oculto (por supuesto el usuario no podrá tener acceso al *Action* que realiza dicho borrado – mediante una restricción de acceso).

Anexo: De la excepción Acceso No Permitido Exception a la pantalla de error

Antes hemos usado la excepción *AccesoNoPermitidoException*, que sería de la forma:

```
public class
AccesoNoPermitidoException extends
Exception
{
    public void printStackTrace()
    {
super.printStackTrace();
}
}
```

Esto lo que provoca es un fallo en la llamada al servidor, por lo que se devuelve la típica página de error. Evidentemente queremos que la página que se muestre sea distinta,

una del estilo "no tiene permisos para realizar la acción solicitada. Contacte con el administrador...". Además, de camino vamos a tener la posibilidad de tener nuestra propia página de error para la aplicación.

¿Cómo lo haremos? De una forma fácil: vamos a crear un manejador de excepciones, lo vamos a poner como manejador de excepciones de la aplicación y vamos a crear nuestra página de error y nuestra página de acceso no permitido.

El manejador de errores va a ser una clase del estilo:

```
public class ExceptionHandler extends ExceptionHandler
{ public ActionForward execute(Exception exception, ExceptionConfig config, ActionMapping mapping, ActionForm
form, HttpServletRequest request, HttpServletResponse response) throws ServletException
{if(exception.getClass().equals(AccesoNoPermitidoException.class)) { return
mapping.findForward("permisoDenegado"); } else {
request.setAttribute("exception",exception); return mapping.findForward("error");
}
}
```

El método lo que ha de hacer es redireccionar al correspondiente *forward* (que posteriormente incluiremos en el *struts-config.xml*). Como queremos diferenciar entre un acceso no permitido y una excepción, según lo que sea (usando la condición *if(exception.getClass().equals(AccesoNoPermitidoException.class))* vemos si es un acceso no permitido) redireccionamos a un sitio o a otro. Además, si es una excepción metemos en *request* dicha excepción, por si luego la queremos mostrar en la pantalla de error.

Para establecer nuestro manejador de excepciones como manejador por defecto incluimos en el *struts-config.xml* el siguiente código detrás de la sección *form-beans*.

```
<global-exceptions>
<exception key="lang.exception" type="java.lang.Exception"
handler="ruta_de_paquetes.ExceptionHandler" />
</global-exceptions>
```

Y para los forwards de error y acceso no permitido incluimos lo siguiente a continuación:

```
<global-forwards>
<forward name="error" path="/error.jsp" contextRelative="true" />
<forward name="permisoDenegado" path="/permisoDenegado.jsp" contextRelative="true" />
</global-forwards>
```

Por último tenemos que crear las páginas *error.jsp* y *permisoDenegado.jsp*

Bueno, pues hemos llegado al final...

Espero que sirva de algo. Y si tenéis alguna duda o sugerencia, ya sabéis, poned un comentario o mandadme un correo a landanohr@hotmail.com

Crear tablas redondeadas en HTML



Por José Rico

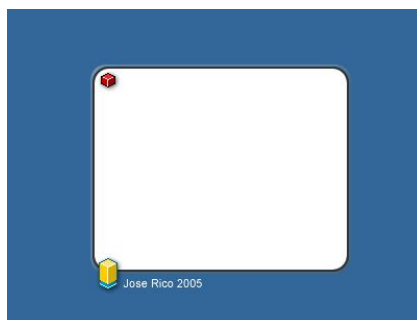
Crear una tabla con los cuadros redondos.

1º. Hay que crear una imagen en cualquier programa de diseño gráfico como pueden ser Photoshop, Fireworks...

Yo utilizaré el Fireworks MX 2004.

Se cre4 un cuadro de 400px de ancho por 300px de alto y 72ppp. (ppp recomendados para internet)

2º. Se crea la imagen, con círculos redondeados, que queramos. Obviamente deberemos dej4r un espacio para el texto que se quiera incluir. Recuerda que el fondo del lienzo deberá ser del mismo color que el fondo de la página donde lo vas a colocar.



Yo he creado este cuadro para probar.

3º. Ahora hay que cortar la imagen. Yo lo que hago es allanar la imagen convirtiéndola en un mapa de bits.

Selecciono todas las capas y pulsamos en modificar-->allanar selección. Ahora ya la tenemos en un único mapa de bits pero queremos que solo se nos quede en cuadro que hemos creado por lo que pulsamos en propiedades, deseleccionamos cualquier capa (pulsamos en un espacio vacío) y nos aparecerá la opción de 'Ajustar lienzo'. Apretamos y ahora ya tenemos solo lo que nos interesa.



4º. Ahora hay que recortar la imagen.

En Fireworks hay una opción que se llama división pero que solamente funcionará si el texto que queremos mostrar no excede del espacio que hemos dejado en el cuadrado, por lo tanto esta opción en este caso no nos interesa.

Vamos a recortar la imagen numerándolas.

Imagen 1. Es la imagen superior izquierda y debe de dejar un margen para que el texto quede centrado.

Imagen 2. La imagen superior. Esta puede ser de 1px de ancho ya que la utilizaremos como fondo de la tabla pero en este caso y para que se vea bien la haremos un poco más grande. Además deberá tener el mismo alto que tiene la imagen 1.

Imagen 3. La imagen de la izquierda. Esta debe tener el mismo ancho y alto que la imagen 1.

Imagen 4. Es igual que la imagen 2 pero con el mismo ancho que la 1 y puede ser de 1px de alto pero para este ejemplo utilizaremos 5px.

Imagen 5. Aquí no hay imagen, deberemos poner en el fondo de la tabla el mismo color que el fondo del espacio que hemos dejado para el texto.

Imagen 6. Es la misma que la 4 pero volteada horizontalmente.
 Imagen 7. En este caso como no vamos a modificar nada la dejaremos tal como está y la incluiremos en la tabla como una imagen y no como fondo.



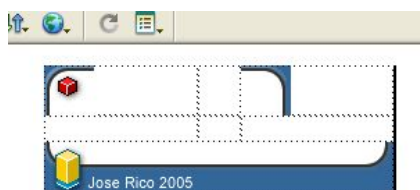
5°. Ahora que ya tenemos todos los trozos hay que montar la tabla.

Para este ejercicio yo he utilizado DreamWeaver MX pero se puede utilizar cualquier editor html.

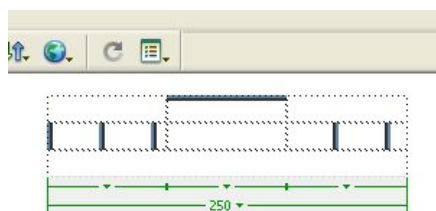
Creamos un tabla de 3 filas x tres column4s y con borde '0', relleno d3 celda '0' y espacio de celda '0'.El ancho de la tabla será el ancho de nuestro cuadro, en este caso de 250 px. La 3ª fila la partimos para que se nos quede solo una celda utilizando la opción dividir celdas o columnas situado en opciones. (Debajo de celda con el símbolo '['[')

Ahora vamos a rellenarla con las imágenes que hemos recortado.

La imagen 1, 3 y 7 se insertan como imágenes normales.



La 2, 4 y 6 como fondo de celda.



Luego cuadramos la tabla y se nos tiene que quedar algo parecido a esto.



Se rellena la tabla central de color blanco y se cambia el color de fondo de la web al color de fondo utilizado en la imagen y esto ya casi está.

6°. Ya tenemos la tabla ahora hace falta rellenarla con texto. Solo tenemos que escribir un título en el espacio 2 y el texto en el espacio donde debería ir la imagen 5 y listo...



Espero que os sirva de algo. Un saludo
 Jose Rico - yosoy@joserico.com

Hojas de Estilo (CSS)

Versión 1



Por Iván A. Figueroa Jiménez

Hojas de Estilo CSS

Este documento describe el lenguaje CSS1 (Cascade Style Sheets, level 1 - hojas de estilo en cascada, nivel 1) que es uno de los lenguajes propuestos para describir hojas de estilo en el WWW.

Las hojas de estilo en formato CSS1 funcionan con el browser Netscape Navigator 4.0 o superior, Internet Explorer 3.0 o superior o con Arena W3C Browser. Para saber cuales de las propiedades o atributos son soportados en cada browser, revise la base de datos disponible en Project Cool. Existen informes de que además por lo reciente de la tecnología hay problemas menores al momento de imprimir algunas páginas bajo algunas plataformas.

¿Qué son las hojas de estilo?

Una hoja de estilo permite aumentar el control del diseñador sobre cómo se verán sus páginas Web, asociando un conjunto de propiedades de formato físico a los elementos estándar del lenguaje HTML.

De esta forma, el resultado final que observa la persona en su browser es una mezcla entre las características predefinidas para cada comando HTML y la hoja de estilo:

Una gran virtud de esta técnica es que permite lograr una presentación muy uniforme, ya que basta definir un estilo una sola vez para que éste se aplique a todos los elementos del mismo tipo que existan en una página.

Nuestra primera página con hojas de estilo

Usar hojas de estilo es muy fácil, ya que se utilizan para ello comandos estándar de HTML. Escribamos una página muy simple:

```
<HTML>
<HEAD>
<TITLE>Hojas de estilo</TITLE>
<STYLE TYPE="text/css">
BODY { background : black}
```

```
H1 { color : yellow;
font-weight : bold;
font-size : 12pt;
line-height : 14pt;
font-family : Arial,Helvetica,sans-
serif;}
P { color : white;
font-size : 10pt;
line-height : 12pt;
text-indent : 0.5in;
font-family : Times,serif;}
</STYLE>
</HEAD>
<BODY>
<H1>Ejemplo de hojas de estilo</H1>
<P>Con hojas de estilo podemos
definir características de formato
físico para los elementos HTML
estándar.</P>
<P>; Hay muchas cosas que se pueden
hacer con hojas de estilo y no se
pueden hacer de ninguna otra forma
!</P>
</BODY> </HTML>
```

NOTA: Si aparece todo el texto de la hoja de estilo, y esta no es interpretada, probablemente el browser que usted está usando no provee de esta característica; por lo que debe buscar una versión más reciente de él.

Los comentarios dentro de la hoja de estilo (<!-- -->) sirven para esconderla de los browsers más antiguos.

Existen varias formas alternativas de asociar una hoja de estilo a un documento: Por ejemplo:

1. Se utiliza el elemento STYLE dentro del encabezamiento del documento, como en el ejemplo.
2. Se usa el elemento LINK para referenciar a una hoja de estilo puesta en un archivo aparte.
3. Se invoca a una hoja de estilo con el comando de CSS "@import ...".
4. Se agrega el atributo STYLE a un elemento dentro del cuerpo del documento.

Este último método mezcla la hoja de estilo con el contenido del documento y es por lo tanto una mala opción.

Veamos un ejemplo combinado, con varias formas de llamar a hojas de estilo:

```
<HTML>
<HEAD>
<TITLE>Demostración</TITLE>
<LINK REL=STYLESHEET
TYPE="text/css"
```

```

<HTML>
<HEAD>
<TITLE>Cool</TITLE>
<STYLE TYPE="text/css"> @import
url(http://style.com/basic.css);
H1 { color: blue }
</STYLE>
</HEAD>
<BODY> <H1>Los encabezamientos
son azules</H1>
<P STYLE="color: green">Y los
párrafos verdes.</P> </BODY>
</HTML>

```

Ventajas

Si bien puede parecer engorroso al principio escribir una serie de instrucciones en código para lograr nuestros propósitos, la idea que está detrás es que uno escriba este set de instrucciones una sola vez, en un archivo separado, y esta hoja de estilo sea usada por una serie de páginas (por ejemplo, por todas las páginas de un mismo servicio Web). De este modo, conseguimos una presentación uniforme con un mínimo de esfuerzo, ya que en el caso de que deseemos cambiar algo, basta editar un solo archivo y este cambio se aplicará a todas las páginas que utilicen ese estilo.

El uso de hojas de estilo evita el uso excesivo de imágenes, que además de que cuesta mucho hacerlas y mantenerlas a lo largo de las actualizaciones, ocupan bastante más tiempo de transferencia que unas cuantas líneas de código extra.

Advertencia importante

Es probable que usted detecte en su browser que soporte hojas de estilo uno o varios de los siguientes errores: La declaración de font-family debe ser la última, todas las declaraciones después de ella se ignoran (!).

Los márgenes izquierdos de BODY y P no se suman entre sí. No se puede poner estilo en un elemento tácito; si no se incluye BODY las propiedades de estilo de el cuerpo del documento no tienen ningún efecto. El espacio entre líneas también se agrega a la última línea del elemento, por eso hay tanto espacio en los encabezamientos. El espacio entre líneas va entre las líneas. Las líneas que tienen links a veces tienen espacio extra sobre ellas. No se da soporte a los seudo elementos, pero sí a las seudo clases. Esté preparado

para experimentar; sólo eso le dará una visión correcta de qué elementos vale la pena usar y cuales no, conforme vaya evolucionando el lenguaje de hojas de estilo.

Entendiendo el lenguaje de hojas de estilo

Las propiedades de estilo pueden ser aplicadas sólo a determinados tipos de elementos del lenguaje HTML: block (bloque) Tienen un quiebre de línea antes o después, como H1, BLOCKQUOTE o P text (texto) Elementos de formato físico y lógico (pertenecientes al grupo phrase) inline (en línea) Elementos que son reemplazados por otros elementos al mostrarse la página (imágenes, animaciones, etc.) Los elementos de HTML son llamados selectores para la nomenclatura de este lenguaje. Lo que hace una hoja de estilo, es asociar una declaración a cada selector, formando lo que se denomina una regla. Una declaración, a su vez, está compuesta por una propiedad a la cual se le asigna un valor.

Veámoslo de manera más clara con un ejemplo:

Las propiedades son un conjunto predefinido en el lenguaje CSS1 (35 propiedades en total), y los valores permitidos dependen de la propiedad con la que se esté trabajando.

Agrupación de reglas

Se puede asociar una misma declaración a varios selectores, usando comas entre ellos:

```
H1, H2, H3, H4, H5, H6 { color : red }
```

O varias declaraciones a un mismo selector, utilizando punto y coma:

```
H1 { color: red; text-align: center; }
```

O se pueden realizar ambas agrupaciones a la vez:

```
H1, H2, H3, H4, H5, H6 { color : red; text-align: center; }
```

Herencia de propiedades

El nombre de hojas de estilo en cascada del lenguaje no es casualidad: se refiere a que las propiedades son heredadas por los elementos más internos, por ejemplo:

```
BODY { color : blue } H1 { color : red }
```

En este caso, todo el cuerpo del documento tiene color azul, con excepción de los encabezados de nivel uno que aparecerán en color rojo.

La herencia se produce en el siguiente sentido:

Elemento <BODY>

Elementos de tipo block

Elementos de tipo text

Clases

Cuando se declara una regla en la hoja de estilo, esta regla se aplica a todos los elementos HTML descritos por su selector; por ejemplo, si se declara

```
P { font-family : italic }
```

Todos los párrafos del documento aparecerán en cursiva; en ocasiones uno quisiera destacar algunos párrafos y darles características distintas a la del resto. Para ello, se provee de un atributo extra en casi todos los elementos del lenguaje HTML, llamado CLASS (clase).

Utilizando este atributo en el elemento HTML al cual uno se refiere, se le puede aislar y uno se puede referir a él en la hoja de estilo como un subselector.

El código para describir un subselector queda muy claro mediante un ejemplo:

```

<HTML>
<HEAD>
<STYLE>
P {text-indent : 0.5in; }
P.versos {font-style : italic;
margin-left : 2in;
text-indent : 0in;
font-family : sans-serif ; }
</STYLE>
<TITLE>Subselectores en hojas de
estilo</TITLE>
</HEAD>
<BODY>
<P> En esta página tenemos unos
versos escritos por el gran poeta
chileno Pablo Neruda: </P>
<P CLASS=versos>
Me gustas cuando callas<BR> porque
estás como ausente<BR> y me oyes
desde lejos<BR> y mi voz no te
toca.
</P>
<P CLASS=versos>
Parece que los ojos<BR> se te
hubieran velado<BR> y parece que un
beso<BR> te cerrara la boca. </P>
</BODY> </HTML>

```

En el ejemplo, queríamos que los párrafos normales tuvieran una sangría (indent) de media pulgada; pero además queríamos tener una poesía, cuyas estrofas no llevan sangría, sino que están separadas con un margen adicional y un tipo de letra distinto. Para ello, inventamos una clase llamada "verso", y agrupamos bajo ella a la poesía que queríamos incluir en el documento.

También es posible definir una clase genérica para ser usada por varios selectores:

```
<STYLE>
.punk { color : green }
</STYLE>
<P class=punk> Este párrafo y
... </P> <BLOCKQUOTE class=punk>
... este bloque, están en el
mismo estilo.</P>
```

Identificadores únicos

El atributo ID de un elemento HTML permite referenciarlo en forma única como selector; claro está, el diseñador web debe tener cuidado de usarlo sólo una vez en el documento.

```
<STYLE> #z99a { letter-spacing:
0.3em } </STYLE>
<P ID=z99a> Este texto está muy
amplio </P>
```

Selectores contextuales

La herencia de propiedades en CSS permite lograr una notación más compacta para las hojas de estilo; pero tiene algunos problemas, por ejemplo; si yo pongo en la hoja de estilo:

```
H1 { color : blue } EM { color : red }
Un título como:
```

```
<H1> <EM> Primera </EM> Sección
</H1>
```

... se verá bastante extraño, pues aparecerá en verde y rojo. Para evitar este inconveniente, se puede agregar la siguiente regla:

```
H1 EM { color: blue }
```

La cual se lee: en el texto enfático dentro de un encabezamiento de nivel uno, se usará color azul.

Este tipo de selector (una lista de selectores separados por espacios en blanco) se llama selector contextual, puesto que indica el comportamiento de un elemento cuando está en un determinado contexto.

La idea que está detrás de la construcción de una hoja de estilo, desde esta perspectiva, es la de listar primero los casos normales y después las excepciones.

Seudoclasses

En CSS1, el estilo normalmente se agrega a un elemento basado simplemente en el tipo de elemento; este modelo no es suficiente para algunos efectos más complejos. El concepto de pseudoclase tiene que ver con permitir que haya más información a disposición del diseñador al momento de escribir la hoja de estilo.

Las pseudoclasses son agregadas por el browser, y referidas como "clases" por las reglas de estilo; estas pseudoclasses permiten diferenciar diferentes usos para un mismo elemento. El ejemplo típico son los hiperlinks; uno desearía que los links visitados fueran diferentes de los que no han sido visitados; esto se logra mediante una construcción como:

```
A: link { color: red } A: visited { color: blue }
A: active { color: lime }
```

Nótese el uso de los dos puntos (:) para referirse a las pseudoclasses, mientras que el punto (.) se usa para referirse a clases reales. Se pueden usar pseudoclasses con selectores contextuales:

```
A: link IMG { border: solid blue }
o con clases reales:
```

```
<STYLE> A.externo:visited { color: black }
</STYLE> <A class=externo
HREF=http://otro.host> Link a otro host </A>
```

Seudoelementos

Los pseudoelementos permiten referirse a porciones de los elementos reales.

En CSS1, se puede referenciar la primera línea de un elemento:

```
<STYLE> P: first-line { font-style:
small-caps } </STYLE> <P>Podría
usarse para reportes de prensa.</P>
O bien la primera letra:
<STYLE> P: first-letter { font-size:
220%; float: left } </STYLE> <P>Había
una vez un bosque encantado donde
jugaban hadas y duendes todo el día,
sin cesar ....</P>
```

En el ejemplo anterior, la primera letra (H) aparece del doble del tamaño normal, y el texto fluye alrededor de ella, como en los libros de cuento:

Estos dos pseudoelementos sólo pueden ser aplicados sobre elementos de tipo bloque.

Comentarios

Los comentarios dentro de las hojas de estilo comienzan con "/*" y terminan con "**"; no pueden ser anidados (uno dentro de otro) y todo su contenido es ignorado.

Ejemplo:

```
H2 { color: yellow } /* Los subtítulos van
en amarillo */
```

Valores reconocidos

Los valores que reconoce cada propiedad en una hoja de estilo dependen del tipo de propiedad. A continuación veremos algunos valores típicos admitidos para entender mejor lo que viene a continuación.

Tipos de letra

El problema de especificar tipos de letra para ser usados en el World Wide Web es un verdadero dolor de cabeza. El motivo fundamental es que sus nombres varían de una plataforma a otra; y no es poco común encontrar que bajo una misma plataforma existan tipos que estén en una versión y no en otra, etc.

Una solución es que una empresa de computación ofrezca una serie de tipos de letra y espere que todo el mundo los instale en su computador, o los incorpore en sus productos y pretenda que sean estándares; pero en el mundo de la computación esto rara vez funciona bien y en ninguna parte son buenos los monopolios.

Bueno, pero ... ¿ qué solución hay ? La solución es que cada vez que se especifica un tipo de letra, se indique una serie de alternativas: si en el computador de la persona que está leyendo la página no existe el primer tipo, se prueba con el siguiente, y así sucesivamente hasta llegar al último.

En caso de que el último tipo de letra tampoco sea reconocido, se utiliza el valor por omisión para el browser, que comúnmente es Times (esta definición, del tipo de letra por omisión no aparece en ninguna parte en la especificación, lo que pasó fue que los primeros browser usaron este tipo y para mantener compatibilidad las siguientes generaciones de browser lo siguieron usando).

Por ejemplo:

```
P { font-family: "Arial", "Helvetica", sans-serif }
```

Significa que a todos los párrafos se les asignará el tipo de letra Arial (este es un tipo de letra sin serif típico de MS Windows), en caso de no estar disponible en el sistema se utilizará Helvetica (existente bajo MacOS y X-Windows).

El último valor listado merece una explicación aparte: para evitar que el browser caiga en el tipo de letra por omisión, se utiliza como última alternativa uno de los siguientes tipos de letra genéricos:

sans-serif (sin adornos) serif (con adornos) cursive (manuscrito) fantasy (no está claro que tipo de letra sea) monospace (monoespaciado, o espacio constante)

Estos no son tipos de letra en sí, sino que son familias. Un browser decidirá al momento de la instalación cual será el tipo que usará para representar cada una de estas familias.

Colores

Los colores se indican mediante un código RGB, (ejemplo: #FFEE09) o bien mediante uno de los nombres de color estandarizados:

aqua	Agua
black	Negro
blue	Azul
fuchsia	Fucsia
gray	Gris
green	Verde
lime	Lima
maroon	Marrón
navy	Azul marino
olive	Verde oliva
purple	Púrpura
red	Rojo
silver	Plateado
teal	Té
white	Blanco
yellow	Amarillo

También existe la posibilidad de usar códigos RGB abreviados, que sólo tienen tres "cifras". Por ejemplo, se

escribe: #FOA en vez de #F000A0, #E9A en lugar de #E090A0, etc.

Los códigos RGB abreviados permiten escribir sólo 256 colores, pero eso es suficiente para la mayoría de los sistemas; esto es muy usual en hojas de estilo puesto que es difícil reconocer muchos matices en el texto, por lo que no tiene sentido ser tan específico en el color.

Longitudes

Las longitudes se indican con un número, seguido de un sufijo. Se pueden expresar en términos absolutos:

- 24mm 24 milímetros
- 2cm 2 centímetros
- 1.2in 1,2 pulgadas
- 12pt 12 puntos de impresión
- Tejedores del Web / 11
- 3pc 3 picas
- 105px 105 pixeles

o bien en términos relativos al tamaño del tipo de letra actual:

- 2em 2 veces el ancho de una letra "m"
- 3ex 3 veces el ancho de una letra "x"

Si no se especifica un sufijo, se subentiende que se está hablando de uno de los 6 tamaños definidos en el HTML.

URLs

Los URL se indican usando la palabra reservada "url", seguida de un paréntesis, luego el contenido del URL y luego un paréntesis de cierre. Veamos algunos ejemplos:

```
url ( http://www.myhost.com/ ) url ( imagen.gif ) url ( icn/globo.gif )
```

Propiedades

A continuación veremos la lista de propiedades que soporta CSS1. Información acerca de la evolución de CSS se encuentra disponible en el W3Consortium. La sintaxis con que se explica este lenguaje es un conocimiento importante, pues las especificaciones en el World Wide Web se proveen en este formato o en algún formato similar. Si bien puede parecer un poco engorroso o difícil no lo es tanto; después de un tiempo uno se puede dar cuenta de que no hay ninguna otra forma más clara de explicar la sintaxis de un lenguaje.

Sintaxis

<x> x no debe ser tomado en forma literal x y z x va primero, después y, después z, en ese orden (yuxtaposición)

x | y x o bien y (se excluyen entre sí)
x || y x o y o ambos a la vez (no se excluyen entre sí)

[x y] los corchetes se usan para agrupar en caso de haber ambigüedad

x ? x puede aparecer cero o una vez
x * x puede aparecer cero o una o más veces

x + x puede aparecer una o más veces
x {1, 4} x puede aparecer a lo menos una y a lo más cuatro veces

La yuxtaposición tiene preferencia por sobre la doble barra vertical, y la doble barra tiene preferencia sobre la barra simple.

Por ejemplo:
a b | c || d e

es equivalente a:
[a b] | [c || [d e]]

El valor por omisión para cada propiedad es mostrado en negrita, o aparece en forma separada.

Las propiedades se heredan (en cascada) a menos que se indique lo contrario. Los valores se refieren a cualquier selector que pueda adquirir la propiedad especificada, a menos que se indique lo contrario.

Características de la letra

font-size : <longitud> | <porcentaje> | <numero> | xx-small | x-small | small | medium | large | x-large | xx-large | larger | smaller

Indica el tamaño de la letra. Los valores porcentuales son en relación al tamaño del elemento del cual heredan.

Por ejemplo, si los párrafos fueron especificados con tamaño 10pt, entonces las siguientes dos líneas significan lo mismo:

```
BLOCKQUOTE { font-size: 120% }  
BLOCKQUOTE { font-size: 12pt }
```

Los valores `larger` y `smaller` aumentan o disminuyen el tamaño de la letra de manera relativa.

`font-family` : [[<nombre> | serif | sans-serif | cursive | fantasy | monospace],] * [<nombre> | serif | sans-serif | cursive | fantasy | monospace]

Indica el tipo de letra, tal como fue explicado anteriormente. Se recomienda que los tipos de letra sean encerrados entre comillas, a excepción del último.

`font-weight` : `extra-light` | `light` | `demi-light` | `medium` | `demi-bold` | `bold` | `extra-bold` | `bolder` | `lighter`

Los valores `bolder` y `lighter` se dan en relación a la opacidad actual, y lo incrementan o decrecientan (respectivamente) en dos unidades.

`font-style` : `normal` | `italic` || `small-caps` | `oblique` || `small-caps` | `small-caps`

Indica un cambio en las características de la letra. Los valores son `italic` (cursiva – o inclinada hacia adelante), `oblique` (oblicua - o inclinada hacia atrás) o `small-caps` (se coloca todo el texto como si estuviera en mayúsculas, y las letras que son verdaderamente mayúsculas se colocan más grandes). Adicionalmente, se puede usar "italic small-caps" o bien "oblique small-caps" (sin incluir las comillas).

`line-height` : <longitud> | <número> | <porcentaje>

Indican la altura de la línea de texto. Los valores porcentuales se refieren al tamaño de letra en curso.

`font` : [<font-weight> || <font-style>]? <font-size> / <line-height>]? <font-family>

Comando genérico de cambio de tipo de letra, que soporta todos los anteriores al mismo tiempo, en una notación más compacta. Ejemplo:

Propiedades del texto

`word-spacing` : `normal` | <longitud>

Distancia que separa a las palabras del texto; usualmente es asignada en forma automática de manera bastante adecuada.

`letter-spacing` : `normal` | <longitud>
Distancia que separa a dos letras en el texto; usualmente es asignada en forma automática de manera bastante adecuada.

`white-space` : `normal` | `pre` | `nowrap`
Define el tratamiento que se le dará al espacio en blanco dentro del elemento. Lo normal es que más de un espacio en blanco es tratado como un sólo espacio. Si se usa el valor `pre`, entonces los múltiples espacios en blanco producirán un espaciamiento mayor.

El valor `nowrap`, indica que los espacios en blanco serán considerados como un carácter más formando parte de una palabra, por lo que las palabras no serán cortadas.

`text-decoration` : `none` | [`underline` | `overline` | `line-through` | `blink`]+

Indica adornos o cualidades extra de la letra. Esta propiedad no es heredada. Los valores permitidos son `underline` (subrayado), `overline` (línea por encima del texto), `line-through` (tachado) y `blink` (parpadeo),

`text-transform` : `none` | `capitalize` | `uppercase` | `lowercase`

Permite cambiar el texto al cual se refiere, poniendo la primera letra en mayúscula, todo en mayúsculas o todo en minúsculas, respectivamente.

Posicionamiento

`vertical-align` : `baseline` | `sub` | `super` | `top` | `text-top` | `middle` | `bottom` | `text-bottom` | <porcentaje>

Indica la posición del selector, en cuanto a alineamiento vertical, con respecto a los límites del texto y los límites de la línea de texto.

`text-align` : `left` | `right` | `center` | `justify`

Se aplica sólo a los elementos de tipo bloque. El valor `justify` indica que se insertarán espacios extra entre las letras para que las líneas de texto ocupen todo el ancho permitido para su bloque.

`text-indent` : <longitud> | <porcentaje>
Indica la sangría en los elementos de bloque. Su valor por omisión es cero, y los valores

porcentuales se refieren a la sangría del elemento padre.

Márgenes y bordes

En este ítem, debemos hacer algunas aclaraciones previas, para definir la nomenclatura que usamos en las hojas de estilo, en lo concerniente a la diferencia entre `padding`, `margin` y `border`.

El `padding` es un espacio que forma parte del elemento, y separa la parte imprimible del selector (lo que llamamos, su contenido), de su borde. El borde corresponde al borde visible que separa al espacio de `padding` del margen. El margen es espacio que está fuera del elemento, y cuya apariencia no es controlable por él.

`margin-left` (`margin-right`, `margin-top`, `margin-bottom`): <longitud> | <porcentaje> | `auto`

Indica el margen izquierdo (respectivamente: derecho, superior e inferior) del selector.

`margin` <longitud> | <porcentaje> {1,4}
Indica todos los márgenes a la vez, en el mismo orden del ítem anterior.

`padding-left` (`padding-right`, `padding-top`, `padding-bottom`): <longitud> | <porcentaje> | `auto`

Indica el `padding` izquierdo (respectivamente: derecho, superior e inferior) del selector.

`padding`
Indica todos los espacios de `padding` a la vez, en el mismo orden del ítem anterior.

`border-left` (`border-right`, `border-top`, `border-bottom`): <ancho del borde> || <estilo del borde> || <color del borde>

Indica propiedades para el borde izquierdo (respectivamente: derecho, superior e inferior) del selector. El ancho del borde puede ser una unidad estándar de longitud, o bien una de las siguientes palabras reservadas: `thin` (pequeño), `medium` (mediano), `thick` (grueso). El estilo del borde puede ser `none` (ninguno) o alguna de las siguientes palabras reservadas: `dotted` (puntos), `dashed` (rayas), `solid` (línea continua) o `double` (doble línea continua).

También se provee de estilos de borde en 3 dimensiones: groove (borde en bajo relieve), ridge (borde en sobre relieve), inset (todo el elemento en bajo relieve) y outset (todo el elemento en sobre relieve).

border : <longitud> | <porcentaje>

Indica propiedades globales para el borde del elemento, con la misma sintaxis anterior.

Dimensiones y flujo del texto

width (height) : none | <longitud> | <porcentaje>

Indica el ancho (respectivamente: la altura) del elemento. Esta propiedad se aplica a selectores de tipo bloque, o a los que son reemplazados por otros elementos (i.e.: imágenes). Los valores porcentuales se refieren a las dimensiones del elemento del cual hereden propiedades. Si una de las dos dimensiones es fijada, y la otra está en auto, las proporciones se mantienen.

float : none | left | right

Por omisión, los elementos de tipo bloque y las imágenes son mostradas en el lugar del texto donde aparecen. Si uno indica float: left, el elemento aparecerá en el costado izquierdo de su elemento padre (el elemento en el cual se encuentra), y el texto fluirá alrededor de ella. Lo mismo ocurre si float : right. Esta declaración - como un efecto colateral - transforma cualquier elemento en un elemento de tipo bloque.

clear : none | left | right | both

Permite decidir si el elemento en cuestión admitirá imágenes flotantes, o requiere de espacio en blanco al lado izquierdo o derecho o en ambos costados.

Cuerpo del documento

color : <color>

Indica el color de todos los elementos, a menos que subsecuentes reglas asocien color explícito a algunos elementos. Los colores se indican de acuerdo a lo que se definió anteriormente.

background : transparent | <color> [/ <color>]? || <dirección de la mezcla> || <url> || <tipo de repetición> || <posición> || <scrolling>

Indica propiedades genéricas para el fondo de un elemento, o sea, la superficie en la cual se muestra el contenido (ej.: texto).

El fondo puede ser :

transparente P { background : transparent } de un sólo color BODY { background: red } de dos colores mezclados en degradación H1 { background : yellow / green } formado por una imagen BODY { background: url(nubes.gif) 50% repeat fixed }

Se puede especificar conjuntamente un fondo con color (colores) y una imagen; en ese caso, los colores se usarán para llenar las regiones transparentes de la imagen o mientras la imagen está siendo cargada.

Esta propiedad no se hereda, sino que como todos los elementos tienen por omisión fondo transparente, se puede "ver" el fondo del elemento "padre" a través del fondo del elemento "hijo".

Usando dos colores se pueden lograr interesantes efectos; la dirección de la mezcla se indica hacia donde se fusionará el primer color con el segundo, y puede tomar alguno de los valores siguientes es: N, S, E, W, NW, NE, SW y SE. El valor por omisión es S (sur). Cuando se usa una imagen para rellenar el fondo, se pueden especificar dos propiedades adicionales.

1. El tipo de repetición determina cómo se repetirá la imagen: repeat-x (repetir horizontalmente), repeat-y (repetir verticalmente), no-repeat (no repetir) y repeat (repetir horizontal y verticalmente), que es el valor por omisión.

2. La posición especifica su localización inicial a lo ancho y a lo alto del elemento, en porcentaje (si se pone un solo valor, este se refiere a ambas dimensiones a la vez). Adicionalmente, se provee de las abreviaturas left, center y right para indicar posición horizontal de 0%, 50% y 100% respectivamente, y top, middle y bottom para indicar posición vertical de 0%, 50% y 100%.

Finalmente, el scrolling indica si el fondo permanecerá fijo al fondo de la ventana (valor fixed), o se moverá junto con el elemento al cual se está aplicando (valor scrolling).

Otros

display : block | inline | list-item | none

Esta propiedad indica una transformación en el tipo de elemento al cual se refiere. Es asignada por el mismo lenguaje HTML, y no debería ser cambiada salvo situaciones especiales.

El valor block indica que el elemento produce un quiebre de línea antes y después. El valor inline indica que el elemento es reemplazado por otro elemento al momento de mostrarlo (por ejemplo, imágenes). El valor list-item indica que se trata de un ítem de lista.

El valor none indica que el elemento no será mostrado, ni los elementos que estén dentro de él ni la caja alrededor.

Por ejemplo:

```
IMG { display: none }
```

list-style : [disc | circle | square | decimal | lower-roman | upper-roman | lower-alpha | upper-alpha | none] || <url>

Esta propiedad puede ser aplicada a cualquier elemento que tenga propiedad display igual a list-item, de acuerdo a lo especificado por el usuario o por el mismo estándar HTML (el elemento LI corresponde a esta última clasificación).

Ejemplo:

```
UL UL { list-style: circle }
```

Indica que los ítems de lista no numerada, de segundo nivel, serán marcados con un círculo.

Instalación de PostgreSQL 8 para Linux



Martín R. Mondragón Sotelo
martin@mygnet.com

Introducción

PostgreSQL es un software de licencia Open Source. Desarrollado originalmente por el Departamento de Ciencias de Computación de UC Berkeley, fue pionero en muchos de los conceptos de objetos y relacionales que ahora están apareciendo en algunas bases de datos comerciales. Provee soporte para lenguajes SQL92/SQL99, transacciones, integridad referencial, procedimientos almacenados y extensibilidad de tipos.

PostgreSQL es un descendiente de código abierto de su código original de Berkeley.

Desinstalación de postgresql

En el caso de tener instalada una versión anterior hay que desinstalarla o instalar la nueva en otro directorio.

Buscar los paquetes dependientes de postgresql instalados con rpm.

```
rpm -qa | grep postgresql
```

Desinstalar postgresql

```
rpm -ev --nodeps postgresql*
```

Descargarse el código fuente de la siguiente dirección:

<http://www.postgresql.org/download/>

Descomprimir

```
tar xvfz postgresql-8.0.3.tar.gz
cd postgresql-8.0.3
```

Configuración personalizada

Esta puede variar según nuestras necesidades, en mi caso es de esta manera:

```
./configure --prefix=/usr/local/pgsql/
--exec-prefix=/usr/local/pgsql/ --
enable-nls --with-perl --with-python --
with-pam --enable-odbc --with-unixodbc
--with-openssl
```

Compilar

```
make
```

Instalar

```
make install
```

Si ocurre un error durante, la configuración o compilación entonces tenemos 2 casos que hacer: **1).** Instalar las dependencias y repetir el proceso desde la configuración. **2).** Cambiar la configuración quitando el o los parámetros del conflicto. Antes de ejecutar la nueva configuración hay que ejecutar el `make clean`

Crear la cuenta de usuario

```
adduser postgres -d /data/pgsql
su - postgres
mkdir data
/usr/local/pgsql/bin/initdb -D
/data/pgsql/data
/usr/local/pgsql/bin/postmaster -D
/data/pgsql/data >logfile 2>&1 &
```

Variables de entorno.

Agreguemos estas variables de entorno al final del archivo: `/etc/profile`

```
LD_LIBRARY_PATH=/usr/local/pgsql/lib
export LD_LIBRARY_PATH
PATH=/usr/local/pgsql/bin:$PATH
export PATH
MANPATH=/usr/local/pgsql/man:$MANPATH
export MANPATH
```

Escribir el script de arranque

Script de arranque

```

cat > /etc/rc.d/init.d/postgresql << "EOF"
#! /bin/sh
# chkconfig: - 26 34
# description: Servidor de base de datos version 8
# processname: postgresql
# config: /data/pgsql/data/postgresql.conf
# pidfile:/var/run/postmaster.pid

PGVERSION=8.0
INITD=/etc/rc.d/init.d
. $INITD/functions
TYPESET=`typeset -f|grep "declare"`
. /etc/sysconfig/network
NAME=`basename $0`
PGPORT=5432

export PGDATA=/data/pgsql/data

if [ -f $PGDATA/PG_VERSION ] && [ -d $PGDATA/base/template1 ]
then
    echo "Using old-style directory structure"
else
    export PGDATA=/data/pgsql/data
fi
prefix=/usr/local/pgsql
[ -f /etc/sysconfig/pgsql/${NAME} ] && . /etc/sysconfig/pgsql/${NAME}
export PGDATA
export PGPORT
export PGOPTS

[ "${NETWORKING}" = "no" ] && exit 0
[ -f $prefix/bin/postmaster ] || exit 0
start() {
    PSQL_START="Starting ${NAME} service: "
    if [ -f $PGDATA/PG_VERSION ] && [ -d $PGDATA/base ]
    then
        if [ `cat $PGDATA/PG_VERSION` != $PGVERSION ]
        then
            SYSDOCDIR="(Your Systems documentation directory)"
            if [ -d $prefix/doc/postgresql-$PGVERSION ]
            then
                SYSDOCDIR=$prefix/doc
            fi
            if [ -d $prefix/share/doc/postgresql-$PGVERSION ]
            then
                SYSDOCDIR=$prefix/share/doc
            fi
            if [ -d $prefix/doc/packages/postgresql-$PGVERSION ]
            then
                SYSDOCDIR=$prefix/doc/packages
            fi
            if [ -d $prefix/share/doc/packages/postgresql-$PGVERSION ]
            then
                SYSDOCDIR=$prefix/share/doc/packages
            fi
            echo
            echo $"An old version of the database format was found.\\n\\nYou need to upgrade the data format before
            using PostgreSQL.\\n\\nSee $SYSDOCDIR/postgresql-$PGVERSION/README.rpm-dist for more information."
            exit 1
        fi
    else
        echo -n $"Initializing database: "
        if [ ! -d $PGDATA ]
        then
            mkdir -p $PGDATA
            chown postgres.postgres $PGDATA
            chmod go-rwx $PGDATA
        fi
        [ -f /etc/sysconfig/i18n ] && cp /etc/sysconfig/i18n $PGDATA/./initdb.i18n
        [ ! -f /etc/sysconfig/i18n ] && echo "LANG=en_US" > $PGDATA/./initdb.i18n
        echo "export LANG LC_ALL LC_CTYPE LC_COLLATE LC_NUMERIC LC_CTYPE LC_TIME" >> $PGDATA/./initdb.i18n
        su -l postgres -s /bin/sh -c "$prefix/bin/initdb --pgdata=$PGDATA >/dev/null 2>&1" </dev/null
        [ -f $PGDATA/PG_VERSION ] && echo_success
        [ ! -f $PGDATA/PG_VERSION ] && echo_failure
        echo
    fi
    pid=`pidof -s $prefix/bin/postmaster`
    if [ $pid ] && $prefix/bin/pg_ctl status -D $PGDATA >/dev/null 2>&1

```

```

then
echo "$Postmaster already running."
else
rm -f /tmp/.s.PGSQL.${PGPORT} > /dev/null
echo -n "$PSQL_START"
su -l postgres -s /bin/sh -c "$prefix/bin/postmaster -D $PGDATA >logfile 2>&1 &"
sleep 1
pid=`pidof -s $prefix/bin/postmaster`
if [ $pid ]
then
success "$PSQL_START"
touch /var/lock/subsys/${NAME}
echo $pid > /var/run/postmaster.${PGPORT}.pid
echo
else
failure "$PSQL_START"
echo
fi
fi
}

stop(){
echo -n "$Stopping ${NAME} service: "
su -l postgres -s /bin/sh -c "$prefix/bin/pg_ctl stop -D $PGDATA -s -m fast" > /dev/null 2>&1
ret=$?
if [ $ret -eq 0 ]
then
echo_success
else
echo_failure
fi
echo
rm -f /var/run/postmaster.${PGPORT}.pid
rm -f /var/lock/subsys/${NAME}
}

restart(){
stop
start
}

condrestart(){
[ -e /var/lock/subsys/${NAME} ] && restart
}

reload(){
su -l postgres -s /bin/sh -c "$prefix/bin/pg_ctl reload -D $PGDATA -s" > /dev/null 2>&1
}

# See how we were called.
case "$1" in
start)
start
;;

stop)
stop
;;

status)
status postmaster
;;

restart)
restart
;;

condrestart)
condrestart
;;

reload|force-reload)
reload
;;

*)
echo "$Usage: $0 {start|stop|status|restart|condrestart|reload|force-reload}"
exit 1
esac
exit 0
EOF

```


Le damos permiso de ejecución al script:

```
Hmod 755 /etc/rc.d/init.d/postgresql
```

Últimos enlaces fuertes

```
cd /etc/rc.d/init.d &&
ln -sf ../init.d/postgresql ../rc0.d/K26postgresql &&
ln -sf ../init.d/postgresql ../rc1.d/K26postgresql &&
ln -sf ../init.d/postgresql ../rc2.d/K26postgresql &&
ln -sf ../init.d/postgresql ../rc3.d/S34postgresql &&
ln -sf ../init.d/postgresql ../rc4.d/S34postgresql &&
ln -sf ../init.d/postgresql ../rc5.d/S34postgresql &&
ln -sf ../init.d/postgresql ../rc6.d/K26postgresql
```

Arrancar, parar o reiniciar

```
/etc/rc.d/init.d/postgresql start
/etc/rc.d/init.d/postgresql stop
/etc/rc.d/init.d/postgresql restart
```

O también:

```
/sbin/service postgresql start
/sbin/service postgresql stop
/sbin/service postgresql restart
```

Configuración

Hay que hacer algunas modificaciones a la configuración de PostgreSQL para que pueda escuchar la red, por que así como esta solo escucha nada más peticiones locales "localhost" postgresql.conf

Dentro de postgresql.conf que esta /data/pgsql/dada/ que fue la ruta donde decidimos instalar la base de datos:

```
listen_addresses = '192.168.19.46'
password_encryption = trae
```

En esta misma ruta habilitamos los permisos de rangos o que ip's que se van conectar remotamente pg_hba.conf

```
local all all password
host all all 127.0.0.1 255.255.255.255 password
host all all 192.168.19.50 255.255.255.255 password
```

Todavía no hay que reiniciar el servidor postgresql para que levante esta nueva configuración si no asta que demos de lata todos los usuarios de base d datos con sus contraseña correspondientes.

Crear usuarios y bases de datos

Tenemos que iniciar sesión con el usuario postgres

```
su postgres
```

Crear los usuarios

Para crea usuarios de postgresql con el comando createuser:

Que se encuentra dentro de de: /usr/local/pgsql/bin/

createuser [OPCIÓN]... [USUARIO]

-a, --adduser	podrá crear otros usuarios
-A, --no-adduser	no podrá crear otros usuarios
-d, --createdb	podrá crear bases de datos
-D, --no-createdb	no podrá crear bases de datos
-P, --pwprompt	asignar una contraseña
-E, --encrypted	contraseña cifrada
-N, --unencrypted	almacenar la contraseña sin cifrar
-i, --sysid=SYSID	escoger un ID para el nuevo usuario
-e, --echo	mostrar los comandos
-q, --quiet	no escribir ningún mensaje
--help	desplegar esta ayuda y salir
--version	desplegar información de versión

Opciones de conexión:

-h, --host=SERVIDOR	servidor o directorio del socket
-p, --port	puerto del servidor
-U, --username=NOMBRE	nombre de usuario
-W, --password	pedir contraseña para conectarse

```
createdb -P nuevo_usuario
```

Crear base de datos

createdb crea una base de datos PostgreSQL.

createdb [OPCIÓN]... [NOMBRE] [DESCRIPCIÓN]

-D, --location=RUTA	lugar donde ubicar la base de datos
-E, --encoding=CODIFICACIÓN	codificación
-O, --owner=DUEÑO	usuario que será dueño
-T, --template=PATRÓN	base de datos patrón a copiar
-e, --echo	mostrar los comandos
-q, --quiet	no escribir ningún mensaje
--help	desplegar esta ayuda y salir
--version	desplegar información de versión

Opciones de conexión:

-h, --host=SERVIDOR	servidor o directorio del socket
-p, --port	puerto del servidor
-U, --username=USUARIO	nombre de usuario
-W, --password	preguntar la contraseña

```
createdb nueva_base
```

Ahora solo resta por reiniciar el servidor de postgresql para que cargue la nueva configuración.

Comenzar con tomcat J2EE



Por kitty
kittynancy@gmail.com

Para todos los que no saben por que 'mi aplicación' J2EE en tomcat no funciona

Requerimos

Java development kit de SUN lo puedes descargar de la página de SUN en download =P

- Apache tomcat lo puedes obtener de la pagina y descargar la versión release que gusten en:
<http://jakarta.apache.org/tomcat/> para instalar pues siguiente ->nexty mas next-> no deberían tener problemas en la instalación. Esta pagina <http://www.mysql-hispano.org/page.php?id=13&pag=4>
- Sería muy útil un IDE como netbeans, eclipse, jcreator, etc.

Variables de entorno

Una vez que tengamos instalado el tomcat tenemos que declarar las variables de entorno entonces clic derecho sobre el icono de MIPC->propiedades->pestaña 'opciones avanzadas' -> botón 'variables de entorno' -> botón nueva 'variables del sistema'

- Variable de sistema `JAVA_HOME` en `C:\j2sdk1.4.2_09` o donde lo tengas instalado.
- Variable de sistema `CATALINA_HOME` en `C:\jakarta-tomcat-5.0.28`
- Variable de sistema `CLASSPATH` en `C:\jakarta-tomcat-5.0.28\common\lib\servlet-api.jar`
- Variable de sistema `PATH` en `C:\j2sdk1.4.2_09\bin`

Arrancar el servidor

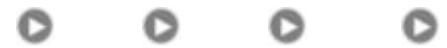
Ahora si todo va bien hasta aquí, si arrancas el Server ubicado en `C:\CATALINA_HOME\bin\tomcat5.exe` dependiendo de la versión y en tu

browser pones la siguiente dirección <http://localhost:8080> te debe aparecer el inicio de apache tomcat y un mensaje como siguiente:

```
if you're seeing this page via a web browser, it means you've setup Tomcat successfully. Congratulations!
```

y podrás probar los ejemplo que trae tanto de servlets como de paginas JSP

Ahora para probar uno nuestro...



Un JSP para empezar

1. crear la carpeta 'test'
2. dentro de 'test' WEB-INF
3. dentro de 'WEB-INF' las carpetas 'lib' y 'classes'



Dentro de `WEB-INF` debe estar el archivo "`web.xml`" puedes crearlo o copiarlo de alguno ya hecho

luego el archivo JSP

```
<< CodigopaginaJSP >>
index.jsp
```

```
<html>
<head>
<title>My First JSP</title>
</head>
<body>
Hello World!<br>
The current date and time is <%= new
java.util.Date()
%>
</body>
</html>
```

Modificar el Archivo 'server.xml'

Este archivo esta ubicado en la siguiente ruta: CATALINA_HOME/conf en C:\jakarta-tomcat-5.0.28/conf

Agregar casi al final del archivo justo antes del cierre del TAG </Host> la siguiente línea:

```
<Context path="/test" docBase="test"
debug="0" reloadable="true"
crossContext="true" />
```

Si tu carpeta esta dentro de webapps si no y esta en por ejemplo c:/midirectorio/otrodirectorio/test

Seria:

```
<Context path="/test"
docBase="c:/midirectorio/otrodirectorio/test"
" debug="0" reloadable="true"
crossContext="true" />
```

reinicias el Server y abres tu navegador con la dirección

<http://localhost:8080/test> y listo

se supone que debería de decirte

```
Hello World!
The current date and time is ' la
hora '
```

digo se 'supone' por que en estas cosas de la informática ...

Ahora un servlet

Es casi lo mismo casi...

Con la diferencia que este se compila por ti mismo.. Tienes que crear el punto class '.class'

1. Crear la carpeta 'hola'
2. Dentro de 'hola' WEB-INF
3. Dentro de 'WEB-INF' las carpetas 'lib' y 'classes'

```
Hola
└─ WEB-INF
   └─ lib
      └─ classes
```

Dentro de WEB-INF debe estar el archivo "web.xml"

```
<< web.xml >>
<?xml version="1.0"
encoding="ISO-8859-1"?>
<!DOCTYPE web-app
PUBLIC "-//Sun Microsystems,
Inc.//DTD Web Application
2.3//EN"
"http://java.sun.com/dtd/web-
app_2_3.dtd">
<web-app>
  <display-name>
    Hola
  </display-name>
  <servlet>
    <servlet-
name>hola</servlet-name>
    <servlet-
class>hola</servlet-class>
  </servlet>
  <servlet-mapping>
    <servlet-
name>hola</servlet-name>
    <url-pattern>/hola</url-
pattern>
  </servlet-mapping>
</web-app>
```

```
<< finweb.xml >>
```

En la carpeta classes debe ir el archivo .class

```
<< hola.java >>
import java.io.*;
import javax.servlet.http.*;
import javax.servlet.*;

public class hola extends
HttpServlet
{
  public void doGet
(HttpServletRequest req,

    HttpServletResponse res)
    throws ServletException,
IOException
  {
    PrintWriter out =
res.getWriter();

    out.println("Hola Mundo");
    out.close();
  }
}
<< fin hola.java >>
```

para compilar no debería haber problemas si usamos un IDE que si le agregamos al perfil JDK la API que esta en CATALINA_HOME\common\lib\servlet-api.jar

o desde MS-DOS

```
inicio- > ejecutar -> cmd
cd
c:\CATALINA_HOME\webapps\hola\WEB-
-INF\classes
```

Comprobar que todo vaya como debería

ejecutar el comando java

Deben aparecer una serie de opciones ahora dentro de nuestro directorio compilamos el archivo 'hola.java'

```
javac hola.java
```

```
if (te marca algun error ) {
c:\CATALINA_HOME\webapps\hola\WEB-
INF\classes> set
CLASSPATH=C:\CATALINA_HOME\common\li
b\servlet.jar;
}
```

Ahora ya reconoce el package 'javax.servlet.http' y no nos marcara ningun error al compilarlo

y si vemos el contenido del directorio tenemos nuestra clase 'hola.class'

Modificar el archivo 'server.xml'

Este archivo esta ubicado en CATALINA_HOME/conf en C:\jakarta-tomcat-5.0.28/conf

Agregar casi al final del archivo justo antes del cierre del TAG</Host> la siguiente línea:

```
<Context path="/hola" docBase="hola"
debug="0"reloadable="true"
crossContext="true" />
```

Si es que tu carpeta esta dentro de webapps si no y esta en por ejemplo c:/midirectorio/otrodirectorio/test

Seria:

```
<Context
path="/hola"docBase="c:/midirectorio
/otrodirectorio/hola"
debug="0"reloadable="true"
crossContext="true" />
```

Reinicias el server y abres tu navegador con la dirección

<http://localhost:8080/hola/hola> y listo se supone que debería de aparecer

```
Hola Mundo
```

El ruido de un beso no es tan retumbante como el de un cañón, pero su eco dura mucho más. =* :*

(Oliver Wendell Holmes)

Instalar Tomcat 5.5.x cooperando con Apache 2



Martín R. Mondragón Sotelo
martin@mygnet.com

Introducción

Tomcat (Jakarta Tomcat) Es un servidor de aplicaciones que sirve como contenedor de Servlets y JavaServer Pages (JSP) desarrollado bajo en proyecto Jakarta en Apache Software Foundation.

Tomcat 5.5.x necesita la versión 1.5 del (J2SE) del JDK, Java Development Kit"(JDK),"Standard Development Kit" (SDK) y "Java 2 Standard Edition" (J2SE) son nombres para el mismo componente e incluyen: El API de Java, el JRE (JVM), compilador de Java y otras funcionalidades definidas por Sun.

Descargar el JDK 1.5

<http://java.sun.com/j2se/1.5.0/download.jsp>

el Download JDK 5.0 Update 4

Y descargamos esta versión para Linux en binario:

`jdk-1_5_0_04-linux-i586.bin`

Lo copiamos dentro de Linux en `/tmp/`

Damos permisos de ejecución:

```
chmod +x jdk-1_5_0_04-linux-i586.bin
```

Instalación del JDK

```
./jdk-1_5_0_04-linux-i586.bin
```

Al mostrar la licencia solo tendremos que aceptar algunos términos de uso del jdk.

Se crea toda un estructura de nombre `jdk1.5.0_04` la cual podemos moverla donde queremos que este.

```
mv jdk1.5.0_04/ /usr/local/jdk1.5
```

Variables de entorno

Configuración de variables de entorno:

Agregar las variables de entorno dentro del archivo `/etc/profile`:

```
export JAVA_HOME=/usr/local/jdk1.5
export JRE_HOME=${JAVA_HOME}/jre
export PATH=$PATH:${JAVA_HOME}/bin
```

Instalar Tomcat

Ahora hay que descargar tomcat desde:

<http://archive.apache.org/dist/jakarta/tomcat-5/v5.5.11/bin/>

la versión `jakarta-tomcat-5.5.11.tar.gz`

La copiamos dentro de `/tmp/`

Descomprimos la estructura

```
tar xvfz jakarta-tomcat-5.5.11.tar.gz
```

podemos moverla dentro de `/usr/local/` como el caso de `jdk`

```
mv jakarta-tomcat-5.5.11
/usr/local/tomcat5.5.11
```

Ejecutamos tomcat con el comando:

```
/usr/local/tomcat5.5.11/bin/startup.sh
```

Dará como salida:

```
Using CATALINA_BASE:
/usr/local/tomcat5.5.11
Using CATALINA_HOME:
/usr/local/tomcat5.5.11
Using CATALINA_TMPDIR:
/usr/local/tomcat5.5.11/temp
Using JRE_HOME: /usr/local/jdk1.5/jre
```

Listo ya podemos abrir el navegador con nuestra dirección del servidor por el puerto 8080 como se muestra en la imagen 1.

Tomcat cooperando con Apache

Para que Apache sea el que reciba las peticiones de JSP y Servlet hay que agregar un modulo dentro de Apache el cual se va encargar de realizar la comunicación con el Tomcat para procesar las solicitudes de paginas echas en Servlet y JSP.

En la página:

<http://archive.apache.org/dist/jakarta/tomcat-connectors/jk2/source/>

Descargar la versión 2.0.4

jakarta-tomcat-connectors-jk2-2.0.4-src.tar.gz

Lo copiamos dentro de /tmp/
Descomprimos

```
tar xvfz jakarta-tomcat-connectors-jk2-2.0.4-src
cd jakarta-tomcat-connectors-jk2-2.0.4-src/
cd jk/
cd native2/
```

Configuración

Ejecutamos la siguiente configuración:

```
LD_FLAGS=-lc ./configure --with-
apache2=/usr/local/http --with-
apxs2=/usr/local/http/bin/apxs --with-java-
home=/usr/local/jdk1.5
```

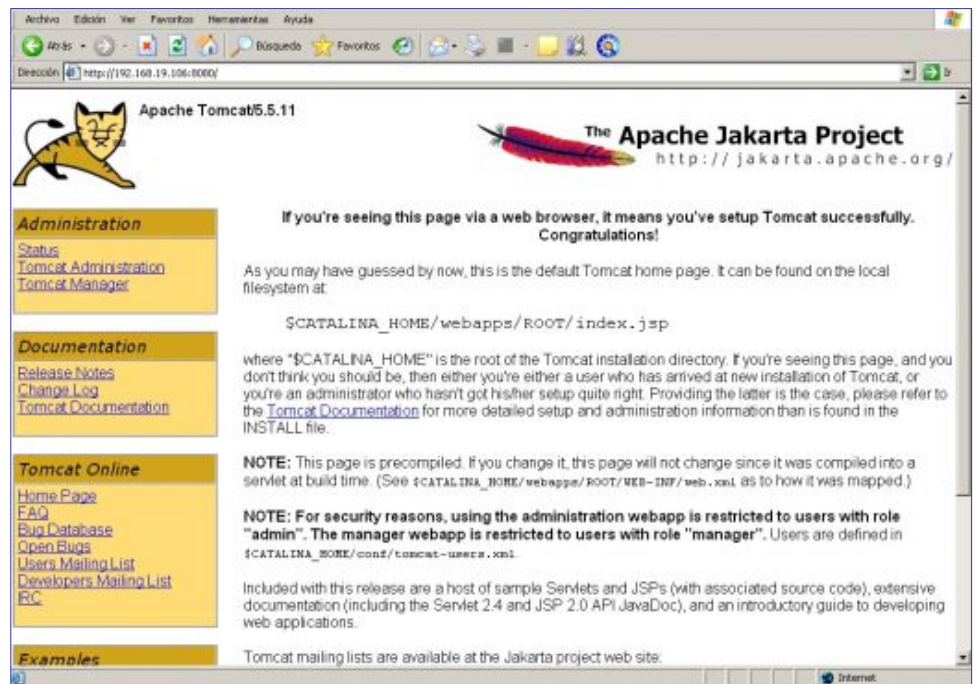
/usr/local/http aquí tenemos instalado el Apache
/usr/local/jdk1.5 Aquí instalamos el JDK

Compilación

```
make
```

Luego hay que copiar el modulo que esta en esta ruta de la cual es el resultado de nuestra compilación:

```
jakarta-tomcat-connectors-jk2-2.0.4-src/jk/ build/jk2
/apache2/mod_jk2.so
en carpeta de módulos de Apache.
cp mod_jk2.so /usr/local/http/modules/
```



Listo solo hay que configurar el `httpd.conf` en la última línea agregamos la siguiente:

```
LoadModule jk2_module modules/mod_jk2.so

<IfModule mod_jk2.c>
    JkSet2 channel.socket:localhost:8009 info "Socket de
    AJP13"
    JkSet2 status: info "Status"
    JkSet2 workerEnv: logger logger.file:0
</IfModule>

<Location "/jsp-examples">
    JkUriSet group lb:lb
    JkUriSet info "Ejemplos de jsp"
</Location>

<Location "/tomcat-docs">
    JkUriSet group lb:lb
    JkUriSet info "Documnetación de tomcat"
</Location>
```

Guardamos la configuración y solo hay que reiniciar el servidor Apache

```
/sbin/service/httpd restart
```

Probamos la dirección de nuestro servidor en nuestro navegador

<http://127.0.0.1/jsp-examples/>

Algoritmos genéticos

Por: Alfredo de Jesús Gutiérrez Gómez
neojag@hotmail.com



Introducción

¿Qué son los algoritmos genéticos?

Simplemente una técnica de resolución de problemas inspirada en el proceso natural de evolución y adaptación de los seres vivos. En otras palabras, son programas basados en algoritmos matemáticos que simulan la capacidad de los seres vivos de aprender, cambiar, adaptarse y evolucionar. Se basan en la teoría evolutiva de Darwin, es decir, en la selección de acuerdo con la aptitud (supervivencia del más apto) y la reproducción entre las entidades (las distintas soluciones), con procesos de recombinación y mutación del "material genético".

En este tipo de algoritmos, se codifica cada una de las posibles soluciones a un problema dado en forma de hilas de caracteres llamados "genes". Se genera normalmente al azar una "población" de posibles soluciones de prueba, a las cuales se evalúan posteriormente según un criterio de desempeño fijado con anterioridad (objetivo). En cada ciclo (cada "generación") se seleccionan las soluciones que más se acercan al objetivo buscado, eliminando el resto de las soluciones. Las soluciones seleccionadas ("las más aptas") se combinan

("reproducen") entre sí para producir nuevas soluciones (su "descendencia"), permitiendo de vez en cuando introducir alguna modificación al azar (una "mutación") durante la reproducción. El ciclo se repite mejorando progresivamente las soluciones hasta llegar a aquella considerada aceptable.

Desde el punto de vista ingenieril, la utilidad de los algoritmos genéticos está en su habilidad de adquirir comportamientos no previstos para hacer frente a entornos dinámicos complejos, que no pueden especificarse completamente con anterioridad y que, además, pueden cambiar con el transcurso del tiempo. Lo importante es que las soluciones a los problemas no se programan sino que "se cultivan".
¿Circuitos que evolucionan?

A través de la aplicación de un proceso adaptable (como los algoritmos genéticos) sobre una clase especial de circuitos lógicos electrónicos (como los dispositivos reconfigurables), se logra una nueva tecnología denominada hardware evolutivo, capaz de modificar su propia configuración interna para adaptarse mejor al entorno en donde debe trabajar o a los errores en el hardware.

"Los circuitos evolucionados pueden ser extremadamente complejos en su estructura y dinámica, pudiendo alcanzar mayores niveles de desempeño que las que son posibles con las técnicas tradicionales de diseño humanas", afirma el Dr. Hugo de Garis, del Instituto de Investigaciones Avanzadas de Telecomunicaciones de Kyoto, Japón. El hardware evolutivo puede también tener considerables consecuencias prácticas en el futuro: siempre se soñó con un sistema electrónico que trabaje con independencia del entorno, de sus defectos de fabricación o de los daños sufridos en el transporte. Alcanzar este objetivo clave es algo extremadamente difícil para un diseñador humano. Sin

embargo, es probable que la evolución produzca (¿o sería mejor decir "engendre"?), chips extraordinariamente eficientes, flexibles y tolerantes a las fallas. Además de ser capaces de operar en cualquier parte, estos chips también se deberían adaptar inmediatamente si la aplicación a la cual están asignados llegase a cambiar a lo largo del tiempo. Las conexiones internas entre los elementos del chip tendrían que evolucionar a fin de resolver una nueva versión del mismo problema, ofreciendo la posibilidad de realizar hardware evolutivo directamente sobre el chip, a máximas velocidades. "Si tal cosa es posible, se revolucionaría la industria de microelectrónica actual", comenta el especialista.

Muy bien, pero ¿qué es un circuito reconfigurable? Son chips capaces de modificar dinámicamente sus propios circuitos internos, enviando señales al chip desde el exterior por el usuario. Aunque se diseñan para hacer sumamente bien una reducida cantidad de tareas específicas, pueden también ser reconfigurados para ejecutar otras tareas igualmente específicas, mucho después de que éste ha dejado la fábrica. Su repertorio, por lo tanto, es variable. Para el inicio del próximo siglo se espera que alcancen el millón de compuertas y una velocidad de configuración de menos de una décima de milisegundo. ¿Qué se puede hacer con ellos? Se los puede utilizar, por ejemplo, para construir electrodomésticos reconfigurables a fin de poder aprovechar los avances tecnológicos o las preferencias del consumidor. Si uno compra un teléfono celular, por ejemplo, y a los tres meses sale un nuevo chip mucho más rápido, podría reconfigurarlo y tener una vez más la tecnología más avanzada.

Esto es una introducción de lo que es los AG's

Comandos Linux en Windows



Por Gustavo Alberto Rodríguez

A alguien le pasa que cuando tiene que usar la consola de Windows, escribe comando Linux, como **ls** en lugar de **dir**, **cat** en vez de **type**, etc.

A mí sí. Así que me hice algunos de los comandos que mas uso. En Linux, es bastante sabido que los comandos se pueden personalizar, simplemente escribiendo un archivo de texto y dándole permisos de ejecución. Pero en Windows, no es tan conocido. Pero para cosas sencillas también se puede hacer. Sólo que al nombre de los archivos hay que agregarle la extensión **.bat**. ¡Los viejos archivos **.bat**!

En este artículo sólo voy a emplear algunas de las posibilidades del lenguaje batch, para un explicación mas extensa se puede consultar un manual en <http://www.elhacker.net/ProgramacionBat.htm>.

Empezando

Voy a empezar explicando algunas cosas y luego pondré los ejemplos.

a) Los archivos **.bat** Simplemente son archivos de comandos que se van ejecutando uno a uno como si se los escribiera en la consola. Con ellos se puede hacer todo lo mismo que en la consola de comandos;

b) Ocultar los mensajes del sistema operativo Para que los comandos se ejecuten en forma silenciosa y no se muestren hay que ejecutar la orden **@ECHO OFF**;

c) Mostrar un mensaje Para enviar una salida a la pantalla hay que usar el comando **echo**. Por ejemplo **ECHO Mensaje**;

d) Argumentos Si al archivo se le pasan argumentos cuando se lo llama, en el programa se pueden recuperar usando variables que anteponen al número de orden

del argumento el signo **%** (porcentaje). Así el primer argumento será **%1**, el segundo **%2**, etc.;

f) Etiquetas Para controlar el flujo del programa se pueden usar etiquetas. Los nombres de las etiquetas empiezan con **:** (dos puntos). Pero cuando se las usa los dos puntos se omiten;

g) Alterar el flujo Para que el flujo de ejecución de un programa se usa **GOTO**, en conjunto con una etiqueta. Por ejemplo si escribimos lo siguiente:

```
:etiqueta
COMANDO1
GOTO etiqueta
```

Ejecutará el **COMANDO1** infinitamente hasta que oprimamos **^C** (ctrl c);

h) Ejecución condicional Se puede usar la sentencia condicional **IF** para determinar si determinado comando se ejecutará o no, según se cumpla una condición. También se puede usar **FOR**;

i) Operadores lógicos El operador igual se escribe **==** (dos signos igual, como en c), el operador de negación es **not**. Por ejemplo **A==B**, **not (A==B)**;

j) Operador de asignación El operador de asignación es el signo **=** (igual);

k) Variables Se pueden declarar y asignar variables y usar las variables de entorno del sistema operativo. Para declarar y asignar una variable de texto se usa:

```
SET variabledetexto = cadena.
```

Si se quiere declarar y asignar una variable numérica hay que usar:

`SET /n variablenumerica = numero`

Para recuperar el valor de la variable hay que encerrar el nombre entre signos %. Por ejemplo: `ECHO %VARIABLE%`

El comando Linux ls

Es el comando que yo más uso por eso es el que primero hice. Claro que sin los colores que tiene Linux, para eso hace falta más que un simple bat.

Ahora bien el comando ls de Linux es el comando dir de DOS. Pero tienen algunas diferencias, que vamos a ir analizando.

Empezamos por crear un archivo de texto llamado ls.bat (cuidado que el editor no le agregue la extensión txt). En la primera línea escribimos `@ECHO OFF`, para que los comandos se ejecuten sin eco en la pantalla.

Ahora veamos las opciones.

a) Sin argumentos

En Linux el comando ls sin ningún argumento, lista los nombres todos los archivos de la carpeta actual, en columnas.

Para que en DOS el comando dir muestre sólo los nombres y en columnas (a lo ancho) hay que agregarle el parámetro `/w`. Escribimos entonces en una nueva línea de nuestro archivo: `DIR /w`.

Si ahora lo guardamos y desde la consola ejecutamos ls nos desplegará ls lista de archivos en columnas.

b) Con filtros

Pero que pasa si queremos que liste sólo los archivos que cumplen una condición, por ejemplo todo los que tienen la extensión .bat. Si ejecutamos `ls *.bat` vemos que el resultado es el mismo que el anterior. Hay que tener en cuenta pasarle parámetros.

Al final de nuestra última línea le agregamos `%1`, de manera que quedaría `DIR /w %1`.

Ahora sí. Ya podemos filtrar los archivos a listar.

c) Listado detallado

Pero, ¿Cómo hacemos que el listado nos muestre el detalle de los archivos? En Linux hay que ejecutar `ls -l`, pero tal como está nuestro archivo si pasamos `-l`

como argumento, lo tomará como un nombre de archivo. Nuestro comando debe ser capaz de distinguir cuando el primer argumento es una opción o un nombre de archivo.

Tal como lo escribimos será válido siempre que no se quiera un listado detallado. Así que le pondremos una etiqueta, para ejecutarlo sólo en tal caso. Al principio de la línea escribimos nuestra etiqueta `:NOHAY`

Apretamos la tecla [ENTER] para cortar la línea.

Ahora determinemos si hay un argumento o no. Para eso, declaramos una variable y le asignamos un valor conocido concatenado al primer argumento. Antes de la etiqueta escribimos:

```
SET HAY=NO%1
```

Analicemos un poco. Si no se pasó ningún parámetro la variable que declaramos sólo contendrá el valor que nosotros le dimos, así que podemos compararlos.

En la línea siguiente escribimos:

```
IF %HAY%==NO GOTO NOHAY
```

Pero también tenemos que ir a ejecutar las órdenes posteriores a nuestra etiqueta, si el primer argumento es distinto a la opción `-l`. Así que escribimos:

```
IF NOT %1==-l GOTO NOHAY
```

Después viene la orden a ejecutar si se deben detallar los archivos. Si el primer argumento es `-l`, el filtro para los nombres de archivos será el segundo. Escribimos:

```
DIR %2
```

Ahora solo tenemos que saltar las órdenes que siguen para ir a terminar la ejecución de comandos. En la siguiente línea escribimos:

```
GOTO SALIR
```

Y cómo última línea del archivo escribimos la siguiente etiqueta:

```
:SALIR
```

El código completo deberá haber quedado como sigue:

```
@echo off
SET HAY=NO%1
IF %HAY%==NO GOTO NOHAY
IF NOT %1==-l GOTO NOHAY
DIR %2
GOTO SALIR

:NOHAY
DIR /w %1

:SALIR
```

El comando Linux cat

En Linux el comando cat es el equivalente al comando DOS type, sólo que de manera predeterminada muestra el archivo una página por vez.

Para este comando no voy a explicar paso por paso cada uno de los comandos sino que vamos directamente al archivo. Creamos un archivo de texto llamado cat.bat, con el siguiente código:

```
@ECHO OFF
SET HAY=NO%1
IF % HAY%==NO GOTO NOHAY
TYPE %1 | MORE
GOTO SALIR

:NOHAY
ECHO No ha especificado un fichero
para mostrar.
:SALIR
```

Final

Por supuesto que esto no pretende reemplazar los comandos de Windows por los de Linux, sino solamente mostrar como se los puede emular usando el casi desaparecido y olvidado lenguaje batch que todavía puede ser muy útil.

e-Learning, una efectiva forma de capacitación



Por Ramón Mondragón Sotelo

Sin duda, uno de los grandes inventos de nuestra época ha sido la Internet, donde a través de un simple "click", nos hemos hecho capaces de acceder a una cantidad inimaginable de información. Y como era de esperarse, un medio de comunicación tan eficiente como éste, ha traído consigo necesariamente el desarrollo de nuevas alternativas en todo tipo de ámbitos incluyendo el de la educación. Una de estas alternativas ha sido la creación del concepto e-Learning, término que se comienza a escuchar con mayor frecuencia en nuestros días.

Pero... ¿y qué es e-Learning?

Seguramente en algún momento ha escuchado mencionar el término "educación a distancia", especialmente en referencia a postgrados y maestrías que pueden cursarse en centros educativos que se encuentran incluso en otros países; pues bien, una de las herramientas que hace posible esto es e-Learning, que consiste en una metodología de enseñanza asíncrona que utiliza a la Internet como medio de comunicación, permitiendo el acceso a los alumnos sin la necesidad de que éstos se muevan de sus lugares y sin importar la hora ni ubicación geográfica que les corresponda.

Esta metodología de enseñanza, se basa en información diseñada específicamente con fines de formación y capacitación, que es publicada en Internet a través de una plataforma de administración del aprendizaje (LMS), donde el alumno accede de manera autónoma a ella, permitiendo evaluar su desempeño a través del seguimiento puntual de sus actividades.

Algunas ventajas de la implementación de e-Learning como metodología de aprendizaje se listan a continuación:

- Administración de cursos, usuarios, reportes y avances en tiempo real vía Internet.

- Acceso al entrenamiento a cualquier hora y desde cualquier parte del mundo, con el solo hecho de estar conectado a Internet
- El entrenamiento se puede extender al cliente, al usuario final, empleados y distribuidores. NO HAY LIMITE
- No se necesita conocimiento del usuario para acceder al entrenamiento en línea.
- No se requiere ningún software específico en el cliente para tener acceso a la plataforma de entrenamiento.
- Reducción de costos en entrega, capacitación, logística de capacitación y viáticos de participantes.
- Liberación de tiempo para el personal involucrado.

Autonomía e-Learning

Aún con todos estos beneficios, la implementación de esta tecnología como alternativa de capacitación en las empresas, se enfrenta a una gran barrera en el campo empresarial, siendo ésta la dependencia en producción de contenidos y administración del sistema que los diferentes proveedores del mercado imponen a sus clientes, ocasionando que muchos de ellos prefieran no adquirir esta tecnología debido al alto costo que representa el servicio permanente de un tercero para operarla.

Para evitar este sometimiento de clientes con proveedores y lograr que el área de e-Learning opere adecuadamente, se debe adoptar un esquema que les permita a éstos ser completamente independientes. En medida que las empresas e instituciones sean capaces de ser totalmente autónomas para la producción, publicación y acceso a los contenidos e-Learning, mayor será la aceptación de esta metodología de enseñanza.

Cabe mencionar que, la autonomía e-Learning es posible gracias a la existencia de herramientas amigables para la creación de contenidos y que pueden ser utilizadas incluso por personas que no tienen conocimientos informáticos.

Implementar e-Learning y lograr la autonomía, requiere de los siguientes elementos:

1. Sistema Administrador del Aprendizaje, mejor conocido como LMS (Learning Manager System), que consiste en una plataforma electrónica donde se publican y administran contenidos educativos en la Web, a los que los alumnos pueden acceder vía Internet, teniendo simultáneamente la posibilidad de dar un seguimiento puntual de sus actividades. Es importante que esta plataforma soporte al menos el estándar de comunicación SCORM (el más ampliamente utilizado) ya que esto, garantizará la apertura del LMS para soportar toda clase de cursos desarrollados por cualquier despacho que respete sus lineamientos.

Al utilizar este estándar, se asegura el acceso a una educación de alta calidad y a materiales de entrenamiento y aprendizaje que se puedan adaptar al estudiante y estén disponibles en el momento y lugar que se requieran.

2. Una herramienta de generación de contenidos rápidos, que permita que usuarios sin experiencia informática puedan desarrollar sus materiales e-Learning y que éstos cuenten con el soporte del estándar.

3. Entrenamiento en la plataforma de administración del aprendizaje y en la herramienta de generación de contenidos rápidos, que garantice la autonomía en e-Learning que la empresa requiere en el proyecto, olvidándose por completo de los proveedores de estos servicios que anteriormente tenía que contratar para dicho fin.

Así, la autonomía e-Learning es ya una realidad que puede ser implementada en todo tipo de organizaciones, logrando importantes mejoras en cuanto a costos y eficiencia en la distribución de capacitación y entrenamiento.

De este modo, es importante fomentar que las empresas den el primer paso en el uso de e-Learning como metodología de enseñanza, entendiendo con precisión la forma de generar contenidos eficaces que adopten modelos instruccionales y estándares para el aprendizaje.

Ramón Mondragón Sotelo es Coordinador e-Learning – CREC y Presidente de la Rama de Desarrolladores de Software Sector Industrial de Tecnología Informática y Comunicaciones. Cámara Nacional de la Industria de la transformación (CANACINTRA)

Comentarios:

elarning@crec.com.mx

www.crec.com.mx

Noticias

Yahoo Publisher Network

Por Ivan A. Figueroa Jimenez

Después del éxito de AdSense de Google, Yahoo, uno de sus eternos competidores, ha presentado Publisher Network, un lugar donde pequeños y medianos sitios web pueden suscribirse para presentar en su página anuncios de texto contextuales.

La idea es prácticamente la misma que con AdSense: mostrar anuncios en las páginas web que estén relacionados con el contenido. Los anuncios son generalmente de texto y se obtiene ingresos únicamente por los clics realizados.

Como Yahoo conoce las palabras clave de nuestra página web, puede mostrar anuncios que tengan mucha relación con el contenido y así mejorar la efectividad de las campañas.

El servicio está todavía en fase Beta. Se puede obtener más información en <http://publisher.yahoo.com/>

Cuando se sepa más sobre este nuevo programa lo comentaremos puntualmente.

 <http://www.desarrolloweb.com/articulos/2233.php>

Liberado Maven 2.0

Ivan A. Figueroa Jimenez

Apache ha anunciado la versión definitiva de [Maven 2.0](#).

Esta versión es una reescritura de la herramienta anterior, conteniendo una gran cantidad de cambios respecto a las versiones 1.x. La nueva es más rápida y más pequeña ya que su núcleo no emplea ni Ant, ni Jelly ni Xerces; cuenta con un ciclo de vida definido, soporta múltiples proyectos, desaparece el archivo maven.xml y aparece un nuevo archivo pom.xml, que permite configurar todos los plugins nuevos.

Estos cambios pretenden obtener una herramienta más fácil de usar, más eficiente tanto en términos de consumo de memoria como de CPU, y el dar soporte a aquellas características demandadas por los usuarios que no eran posibles en la arquitectura de Maven 1.0.

Más información:

<http://maven.apache.org/maven2/>

Fuente: [JavaHispano](#)

 <http://www.programacion.com/noticia/1411/>

Apple presentó el nuevo vídeo 'iPod' y un acuerdo con Disney TV

Por Ilandanohr

El gigante tecnológico Apple Computer Inc dio a conocer un nuevo lector de vídeo 'iPod' y un acuerdo con Walt Disney Co. que le permitirá a los usuarios ver, mientras están en movimiento, sus programas televisivos favoritos.

 <http://iblnews.com/story.php?id=4713>

JFreeReport 0.8.6 released

Por Gerardo Angeles Nava

JFreeReport is an embedded report generator written in 100% pure Java. XML-based templates provide flexible reporting and printing functionality using data from multiple sources. JFreeReport supports output to display devices, printers, PDF, Excel, HTML, XHTML, PlainText, XML and CSV files.

JFreeReport 0.8.6 adds the ability to distribute wide pages over multiple physical pages, much like spreadsheet applications like Excel

print overly large tables. The new StackedLayoutManager simplifies the usage of dynamic elements and improvements in the XML parser implementations allow the definition of global stylesheets for all available report definition formats.

Along with the many bug-fixes, the new features give numerous reasons to upgrade to this new version.

A complete list of all changes can be found in the ChangeLog.

https://sourceforge.net/project/shownotes.php?group_id=51669&release_id=358104

<https://sourceforge.net/projects/jfreereport>

 http://sourceforge.net/forum/forum.php?forum_id=497645

Presentacion Macromedia Studio 8 en Madrid

Pro Santiago Martin

Presentación Macromedia Studio 8

22 Septiembre 2005

Centro de Convenciones MAPFRE

Madrid

Ni mucho menos se llenó el auditorio. Calculo que cuando empezamos a eso de las 10:15 seríamos aproximadamente unos cien asistentes al acto, conducido por Enrique Duvos y Jordi Masfarné.

Supongo que sería para abrir boca, Enrique nos proyectó un video de introducción sobre la nueva Suite. Nada de características, nada de pantallazos...solamente imágenes de portales desarrollados con la versión anterior de Flash, la MX 2004.

Comienza la presentación con lo clásico...el tedioso estudio que demuestra que 'el 98% de los PC conectados a Internet tienen instalado Flash Player', el aumento de facturación de Macromedia en el último año, qué soft compone el Studio 8 (se ha incluido el Contribute y Flash Paper en detrimento de Freehand) y características básicas de cada uno de ellos (Dreamweaver, Flash, Fireworks y los mencionados antes). También se toca la fusión (o mejor dicho la compra) de Macromedia por parte de Adobe y la fecha en la que Studio 8 en castellano va a estar disponible: 10 de octubre

Macromedia Flash Professional 8

Ayudándose de una pequeña interfaz, Jordi nos enseña varias de las características nuevas como la posibilidad de incluir efectos gráficos a los movieclips...y que éstos tengan los mismos comportamientos que los mc. Blur, glow, bevel, gradient glow, gradient bevel, adjust color.

El nuevo 'Anti-aliasing'

con los distintos métodos, incluido uno personalizado. En un ejemplo nos mostraba un texto visto en Flash Player 7 y el mismo texto en Flash Player 8.

El video.

Todos sabíamos ya que disponíamos de un códec de video Sorenson Spark, pues bien, se ha añadido un nuevo códec de vídeo, 'On2VP6'. La particularidad de este códec es que nos permite hacer efectos de alfa a los videos. Es decir que podemos tener un video situado encima de un objeto y ver el video y el objeto al mismo tiempo. Incluso un vídeo sobre otro vídeo. Veamos la imagen. Se puede ver una chica como si estuviera mirando hacia ese centro comercial. En realidad el centro comercial es un video y la chica desplazándose es otro video con un croma. Con los Alpha Channel se ha 'quitado' el croma y se le ha añadido el fondo que es un movieclip con un movimiento continuo.

Dispositivos

Algo que ya estaba presente en la versión MX 2004 eran las plantillas de móviles para simulación de contenido desarrollado con Flash y Flash Lite 1.1. En Flash 8, podemos decir que tipo de aplicación va a ser en el móvil: alarma, un menú de iconos, una animación, un explorador...todo esto desde la ventana de Configuración de Dispositivos 'Device Settings'. De hecho veremos que dependiendo de la 'aplicación' que le vayamos a dar, nos aparecerán disponibles unos u otros dispositivos móviles y que con los botones de éstos podremos interactuar con nuestra aplicación.

Móviles.

Macromedia Dreamweaver 8

Video

Inclusión de video flv directamente. Mediante el boton de Insertar Media. También tenemos la posibilidad de cambiar el skin del video, e incluso crear el nuestro con Flash.

Búsqueda Google

A mi particularmente me parece una completa chorrada y es que podemos insertar una casilla de búsqueda de Google con tan solo un par de clics

A parte de esto, lo que sí destacaría sería una mejora de todo lo relacionado con las CSS, nuevo soporte para PHP5, y los archivos de estilos XML (XSLT), y para los amantes del código –como yo- la ‘contracción de código’. Esto lo cierto es que era algo que se echaba de menos en un editor tan potente como Dreamweaver y que ya incorporaban otras herramientas como Visual Studio. Esto hizo que en el auditorio se oyera un tremendo: ‘ohhhhhh’. Si he de ser sincero me sorprendí....pero de que la gente no supiera esa nueva característica. Y me preocupo.....esto quiere decir que soy muy friki y el único que había visto la web de macromedia antes de la presentación.

Importante vulnerabilidad en Java

Pro Gerardo Angeles Nava

Importante vulnerabilidad en Java

de Oxygen3 24h-365d

por [Panda Software](#)

Madrid, 25 de noviembre de 2004 - Sun Microsystems [ha informado](#) de la existencia de un grave problema de seguridad en la tecnología Java Plug-in de Software Developers' Kit (SDK) y de Java Run-time Environment (JRE), que podría permitir ejecutar código arbitrario en los sistemas afectados.

La vulnerabilidad tiene su origen en la ausencia de controles en el intercambio de datos entre el código _JavaScript y [Java](#), en los navegadores que utilizan la tecnología Java Plug-in. En la práctica, permite a un atacante ejecutar un applet que puede llevar a cabo acciones —entre las que se incluyen la lectura y escritura de archivos en el sistema—, con los mismos privilegios que el usuario del equipo afectado.

El mencionado problema de seguridad se ha detectado en las versiones 1.3.1_12 y anteriores, 1.4.0, 1.4.1, 1.4.2_05 y anteriores de SDK y JRE, en plataformas Linux, Solaris y Windows. Sun ha publicado las actualizaciones 1.3.1_13 y 1.4.12_06, en las que ya se encuentra resuelta la vulnerabilidad.

 [Panda Software](#)

Ejecución remota de código en Solaris 10 a través de cliente DHCP

Pro Martin R. Mondragón Sotelo

Se ha anunciado la existencia de un problema en el cliente DHCP de Sun Solaris 10, que podrá ser empleado por un usuario remoto para conseguir el control de los sistemas vulnerables.

El problema reside en el script "/lib/svc/method/net-svc", que se ve afectado por una vulnerabilidad sobre la que no se han facilitado los detalles. Un atacante remoto podrá enviar múltiples nombres DNS en respuesta a una consulta enviada desde el cliente objetivo. De esta forma logrará la ejecución de código con privilegios de root en el sistema del cliente DHCP afectado.

Este problema sólo afecta a Solaris 10, por lo que las versiones 8 y 9 de Solaris se ven libres de la vulnerabilidad.

Sun ha publicado las siguientes actualizaciones:

Para plataforma SPARC:

<http://sunsolve.sun.com/pub-cgi/pdownload.pl?target=119593-01&method=h>

Para plataforma x86:

<http://sunsolve.sun.com/pub-cgi/pdownload.pl?target=119594-01&method=h>

 <http://noticiaspc.redtotalonline.com/modules.php?op=modload&name=News&file=article&sid=472>

ASM y C embebido en VB (Ahora es una realidad con ThunderVB)

Por Heriberto Mantilla Santamaría

[ThunderVB First Release](#)

Primero dejo el comentario original de sus autores:

ThunderVB is an add-in for VB6. ThunderVB let's you embed Asm and C instructions directly in you VB source programs. Furthermore ThunVB let's you create a true StdCall DLLs. Moreover ThunVB can extend VB IDE to a basic Asm/C code editor. That means it can do syntax-highlighting of Asm/C keywords and supports intelli-sense for InlineAsm code as well. See the screenshots bellow. --- ThunderVB

Development Team [drkII Raziél & Libor] ---

Con estas dll, podemos embeber código C y ASM sin problema en nuestros programas de VB (quien dijo que VB no lo podía hacer), usted puede generar luego las dll's.

Visiten este proyecto, es de código abierto GNU.

Development Status: [5 - Production/Stable](#)

Intended Audience: [Developers](#)

License: [GNU Library or Lesser General Public License \(LGPL\)](#)

Operating System: [32-bit MS Windows \(NT/2000/XP\)](#)

Programming Language: [Visual Basic](#)

Topic: [Build Tools](#), [Compilers](#)

User Interface: [Win32 \(MS Windows\)](#)

 <http://thundervb.sourceforge.net/>

Yahoo! permite hacer llamadas de teléfono

Por Gustavo Santiago L.

El portal de Internet Yahoo! ha incorporado a su sistema de mensajes instantáneos (IM) la posibilidad de hacer llamadas telefónicas. Este nuevo servicio, denominado "Yahoo! Messenger with Voice", incrementa la calidad de las capacidades de llamadas, permitiendo a los usuarios entablar comunicación telefónica mediante el ordenador y comunicarse así con amigos y familiares de cualquier parte del mundo, asegura la compañía.

Incorpora además una versión del sistema de búsquedas de Yahoo! a las conversaciones de mensajes instantáneos. Así, los internautas pueden destacar una palabra en una conversación de IM y rápidamente ver los resultados de su búsqueda, o noticias en las que aparezca el término.

Además, "Yahoo! Messenger with Voice", que está disponible en la dirección <http://messenger.yahoo.com/>, incluye un sistema de correo de voz gratuito, intercambio de fotografías y una mayor protección contra el "spam" o correo basura.

Aparte de en EE.UU., "Yahoo! Messenger with Voice" está disponible en Argentina, Australia, Brasil, Canadá, China, Francia, Hong Kong, Italia, Alemania, Corea del Sur, México, Rusia, España, sur de Asia, Taiwán, Reino Unido y América Latina.

 <http://www.consumer.es/web/es/tecnologia/2005/08/10/144389.php>

MSN abre su código

Por Jesus Nava Estrada

Microsoft buscará la creación de un "ecosistema" de desarrolladores independientes

La empresa Microsoft propiciará la creación de una plataforma de desarrollo para varios de los servicios de su división MSN. Según el sitio NewsFactor, esta nueva plataforma permitirá que los desarrolladores independientes puedan crear sus propias aplicaciones relacionadas con MSN Search, MSN Virtual Earth y MSN Messenger.

La política de abrir el código a terceros, y la creación de un "ecosistema" de desarrolladores es algo habitual en otras empresas como Google y Yahoo, pero hasta ahora Microsoft había sido reacio a este tipo de iniciativas. Según los expertos, el cambio de enfoque obedece a la necesidad de la compañía de aumentar su competitividad, lograr una mayor exposición en el mercado y mostrar una mayor apertura.

A partir de este nuevo nivel de acceso a los códigos y APIs, los desarrolladores podrán utilizar las interfases de búsqueda de MSN en sus propias aplicaciones no comerciales que utilicen mapas e imágenes satelitales. Las empresas y programadores independientes también podrán acceder a las APIs de Virtual Earth para usos comerciales, así como utilizar el servicio MapPoint Web como una plataforma de mapas. Además, se podrán crear aplicaciones para ser usadas como extensiones de MSN Messenger.

 <http://www.tectimes.com/secciones/notas.asp?codnota=18249>

Trucos

Formateo a bajo nivel (para discos arruinaditos)

Por Javier Calderón jav_esm@hotmail.com
Colaborador de mygnet

Seguramente cuando sientes que tu disco duro ya va lento, que después de tanta formateada ya no cala más... entonces puedes hacer un formateo a bajo nivel. Un formateo a bajo nivel es la forma por la cual "limpias" realmente el disco duro, poniéndolo a cero, cosa que no logra en un 100% el formateo normal.

¿Cómo puedo formatear mi disco duro a bajo nivel?
Arranca tu PC desde un disco que tenga una versión de MS-DOS (o en su defecto Windows 98, me), después de arrancar la PC en modo DOS, introduce los siguientes comandos (presiona enter después de cada línea)...

```
DEBUG
A100
INT13
RAX
0301
RBX
0200
F 200 L 200 0
RCX
0001
RDX
0080
P
Q
```

OJO esto formateará el disco duro PRIMARIO que tengas conectado, si en un dado caso deseas formatear el esclavo, cambia la línea: 0080 por 0081, ten cuidado antes de ejecutar...

Ahora reinicia tu PC y formatea normalmente

Cómo montar una partición NTFS en Linux

Por Alberto Sánchez García alberto.sanchez.garcia@gmail.com
Colaborador de mygnet

Lo primero que hay q hacer es comprobar nuestra versión de Linux, para ello realizamos un "ls vmlinuz*" en la carpeta /boot. En mi caso la version es vmlinuz-2.4.20-8; una vez hecho esto buscamos en la página:
<http://linux-ntfs.sourceforge.net/rpm/downloads.html> el fichero correspondiente a nuestra versión y lo descargamos.

Una vez descargado se puede instalar haciendo doble click o utilizando el siguiente comando:

```
"rpm -ihv kernel..."
```

Para probar si funciona correctamente utilizamos la siguiente línea de comando:

```
"/sbin/modprobe ntfs"
```

Para recompilar los paquetes utilizamos

```
"/sbin/depmod -a"
```

Utilizamos **fdisk** y pulsamos **"p"** para obtener la lista de particiones disponibles.

Una vez sepamos cual es la partición que nos interesa la montamos utilizando el comando **mount**

Por ejemplo:

```
"mount /dev/hda1 /mnt/winxp"
"mount /dev/hda5 /mnt/datos"
```

Para realizar auto montajes, es decir, si queremos que cada vez q arranquemos Linux la partición esté ya montada, editamos el fichero **"/etc/fstab"** y escribimos en este orden:

```
1.-Particion origen (/dev/hda1)
2.-Directorio destino (/mnt/winxp)
3.-Sistema de archivos (ntfs ro [ro es read only])
```

Espero que os resulte útil. ;-)

Convertir fechas de formato DATETIME a formato NORMAL por medio de una función en PHP

Por Santiago Martin info@martinpulido.com
Colaborador de mygnet

No habrán sido pocas las veces que hemos tenido que mostrar una fecha extraída de una base de datos **MySQL** y nos encontramos con que tenemos que reordenarla del todo ya que el formato **DATETIME** propio de SQL nos guarda la fecha de forma:

```
'YYYY-MM-DD HH:mm:ss'
```

y nosotros deseamos mostrarla en el formato típico no-anglosajón, es decir:

```
'DD-MM-YYYY HH:mm:ss'
```

Pues bien, esta función lo que hace es precisamente eso, convertir una fecha **DATETIME** que se le pasa como parámetro al tipo **'DD-MM-YYYY HH:mm:ss'**.

Lo primero de todo es separar la cadena a través del guión '-'

```
$fecha = split("-", $fecha_datetime);
//Donde $fecha_datetime es la fecha que traemos de la BBDD
```

Aquí habremos obtenido ya claramente el año y el mes, pero en el

dia \$fecha2 esta tendrá un valor que será el día más lo que sigue a continuación.

Es decir, la hora, los minutos y los segundos.

```
$hora = split(":", $fecha[2]);
```

Ahora en \$hora[0] tendremos el día más la hora, mientras que \$hora[1] serán los minutos y \$hora[2] los segundos. ¿Qué nos queda? Pues por último dividir a partir del espacio.

```
$fecha_hora = split(" ", $hora[0]);
```

Y esto nos da como resultado que \$fecha_hora[0] es el día y \$fecha_hora[1] son las horas.

Ya solo nos queda armar la fecha y guardarla en una variable.

```
$fecha_convertida = $fecha_hora[0].'-'. $fecha[1].'-'.
'$fecha[0].'.
'. $fecha_hora[1].':'. $hora[1].':'. $hora[2];
```

Y hacer un return de esa variable

```
return $fecha_convertida;
```

```
función convertir_fecha
function convertir_fecha($fecha_datetime){
//Esta función convierte la fecha del formato
DATETIME de SQL
//a formato DD-MM-YYYY HH:mm:ss
$fecha = split("-", $fecha_datetime);
$hora = split(":", $fecha[2]);
$fecha_hora=split(" ", $hora[0]);
$fecha_convertida=$fecha_hora[0].'-'. $fecha[1].'-'.
'. $fecha[0].'.
'. $fecha_hora[1].':'. $hora[1].':'. $hora[2];
return $fecha_convertida;
}
```

Abrir 2 vínculos con javascript

Por Santiago Martin info@martinpulido.com
Colaborador de mygnet

En este tutorial rápido vamos a ver la forma en que podemos abrir 2 vínculos al hacer clic en el mismo enlace.

Debemos de crear una función en javascript que será la que se encargue de abrir los dos vínculos.

Para ello vamos al código de nuestra página y antes de la etiqueta de cierre insertamos:

```
<script language="javascript">
function abrirvinculos() {
window.open('http://www.lawebdelprogramador.com',
'_blank');
window.open('http://www.flash-db.com', '_blank');
}
</script>
```

Una vez que tenemos el código javascript que nos permite abrir las páginas, solamente nos queda asignar al texto o imagen (según proceda) nuestra función. Para ello seleccionamos el texto objeto el link y en la etiqueta '<a>' escribimos:

```
<a href="#" onClick="abrirvinculos();" >Abrir 2 vinculos</a>
```

Por supuesto este script se puede cambiar y en lugar de que abra 2 vínculos que abra más. Tan solo hay que añadir una nueva línea de `window.open('nombrepagina', 'destino')` y listo. Además si se desea abrir en un frame, tan solo hay que indicar en el destino el nombre del frame.

How Generate Random Carácter con Javascript

Por Gerardo Ángeles Nava gangeles@informatique.com.mx
Colaborador de mygnet

```
<script language="JavaScript">
!-
//Global Function
function getRandomNum(lbound, ubound) {
return (Math.floor(Math.random() * (ubound -
lbound)) + lbound);
}
//-->
</script>
```

Referencias:

<http://javascript.internet.com/messages/random-number.html>
http://www.free-solutions.de/js/zeichensatz_ascii-ansi_code.htm
<http://web.mit.edu/cjoye/www/links/ascii.html>

Objective : How Generate Random Character?

By Gerardo Ángeles. The next example is a result of searching on Google.

1. Random function
 - 1.1 Generate a range of random numbers function
2. All character have a code
 - 2.1 Range code for upper char
 - 2.2 Range code for lower char

```
Javascript Random function : Math.random()
```

Ejemplo:

```
<script language="JavaScript">
<!--
document.write(Math.random());
//-->
</script>
```

Generate a range of random numbers javascript function:

```
function getRandomNum(lbound, ubound) {
return (Math.floor(Math.random() * (ubound -
lbound)) + lbound);
}
```

Testing the `<code>getRandomNum(lbound, ubound)</code>` function:

```
<script language="JavaScript">
<!--
for(var i = 0; i <= 9; i++){
document.write('<b>');
document.write(getRandomNum(65, 90));
document.write('</b>');
document.write(' [65-90] ');
}
for(var i = 0; i <= 9; i++){
```

```
document.write('<b>');
document.write(getRandomNum(97, 122));
document.write('</b>');
document.write(' [97-122] ');
}
//-->
</script>
```

Code number for HTML char representation (0-255):

```
<script language="JavaScript">
<!--
var c = 0;
for(var iCodeNumber = 0; iCodeNumber <= 255;
iCodeNumber++){
    document.write(' [' + c + ' = ' + '&#;' +
iCodeNumber + ' ] ');
    c++;
}
document.write('<hr>');
//-->
</script>
```

The code for upper case letter are:

```
<script language="JavaScript">
<!--
for(var iCodeNumber = 65; iCodeNumber <= 90;
iCodeNumber++){
    document.write('&#' + iCodeNumber + ' (' +
iCodeNumber + ') ' + '<br>');
}
document.write('<hr>');
//-->
</script>
```

The code for lower case letter are:

```
<script language="JavaScript">
<!--
for(var iCodeNumber = 97; iCodeNumber <= 122;
iCodeNumber++){
    document.write('&#' + iCodeNumber + ' (' +
iCodeNumber + ') ' + '<br>');
}
//-->
</script>
```

So, have two range, one for upper case letters (65-90) and another for lower case letters (97-122), and have a function to get a random number range.

Test to get only upper case letters:

```
<script language="JavaScript">
<!--
for(var test = 0; test <= 25; test++){
    document.write('&#' + getRandomNum(65, 90) +
'<br>');
}
document.write('<hr>');
//-->
</script>
```

Test to get only lower case letters:

```
<script language="JavaScript">
<!--
for(var test = 0; test <= 25; test++){
    document.write('&#' + getRandomNum(97, 122) +
'<br>');
}
document.write('<hr>');
//-->
</script>
```

Sobre como obtener la CPU del sistema en uso a través de la API de Windows

Por Aitor Calderón Martínez

Como obtener el porcentaje del CPU en uso en uso de un sistema con ayuda de la API de Windows.

Antes de entrar en detalles, voy a explicarla función de API de windows q vamos a usar, se trata de GetSystemTimes que se define como (fuente: microsoft msdn library):

```
BOOL GetSystemTimes(
    LPFILETIME lpIdleTime,
    LPFILETIME lpKernelTime,
    LPFILETIME lpUserTime
);
```

Que en Delphi sería algo así:

```
function GetSystemTimes(var lpIdleTime:FileTime; var
lpKernelTime: FileTime; var lpUserTime: FileTime):
Boolean; stdcall;external 'Kernel32.dll';
```

La función a un no está implementada en las librerías de Delphi, por lo que habrá que hay que importarla (por eso la sentencia "extern" al final de la declaración de la función, como la importamos directamente de la librería Kernel32, tenemos que declarar que se trata de una función con convenio de llamada Stdcall, que es el usado por las librerías de Windows). Esta función devuelve verdadero si no se ha producido ningún error, y falso en caso contrario. Las variables pasadas como parámetros son modificadas por la función, la cual asigna los siguientes valores:

- lpIdleTime: La cantidad de tiempo que el sistema se encontraba en Idle.
- lpKernelTime: Cantidad de tiempo que el sistema ha usado para el modo Kernel
- lpUserTime: Cantidad de tiempo que el sistema ha usado para el modo Usuario.

Nota: lpKernelTime y lpUserTime incluyen los hilos en todos los procesos de todos los procesadores.

Usaremos esta función para calcular el porcentaje de uso. Para ello tenemos que tener en cuenta que lo que vamos a medir son variaciones de tiempo, con lo que la primera vez tendremos que llamar a la función dos veces. Sin embargo, tenemos un pequeño problema, dado que los parámetros son del tipo FileTime (que en el fondo es igual que ULarge_Integer y se puede interpretar como tal), no se pueden restar (para medir la variación) directamente. Para ello podemos convertirlo en una variable del tipo Double (64 bits, lo convertimos a Double porque el tipo FileTime es básicamente una estructura de 64 bits con 2 componentes de 32 bits cada uno), para convertirlo en Double creamos una función:

```
function FileTime2Double(FileT: FileTime): Double;
begin
    Result := FileT.dwHighDateTime* 4.294967296E9 +
FileT.dwLowDateTime;
end;
```

Una vez convertido a un formato manejable con operaciones

aritméticas convencionales, las variaciones se pueden obtener con algo como esto:

```
GetSystemTimes(NewIdle,NewKern,NewUsr);//Obtenemos
los tiempos actuales
Idle := Li2Double(NewIdle) -
Li2Double(OldIdle);//Variación de Idle
Kern := Li2Double(NewKern) -
Li2Double(OldKern);//Variación de Kernel
Usr := Li2Double(NewUsr) -
Li2Double(OldUsr);//Variación de User
OldIdle:= NewIdle; // Una variable que puede ser
global, usada para calcular lasiguiente variación
OldKern:= NewKern; // Igual que el anterior
OldUsr :=NewUsr; // Igual que el anterior
```

Una vez calculadas las variaciones solo nos queda una última operación convertir estos valores en porcentaje. Dado que el proceso "Idle" ocupa el 100% de la CPU, se lo restamos a las variaciones de Kernel y User, al final la ecuación es:

Porcentaje:= (Kern + Usr – Idle)*100 / (Kern + Usr); //Donde Porcentaje es del tipo Real.

Y ya tenemos el uso del CPU en porcentaje. Se aprecian comentarios e informes sobre cualquier errata o error. Un saludo Aitor.

Efecto Easing. Redimensionamiento con desaceleración

*Por Santiago Martin
Colaborador de mygnet*

Todos habremos visto alguna vez un efecto interesante deredimensionamiento con deceleración. (En el caso raro de no haberlo visto, aquí podréis tenerlo).

Podéis ver un ejemplo de este redimensionamiento, si entráis en la Web de ManuAlvarez [www.notacube.com].

Bien, empecemos pues como lo haríamos. Creamos un cuadrado de color sin borde (en nuestro caso, es blanco porque le hemos puesto en fondo negro a la película). Convertimos el cuadrado en un clip de película y establecemos el origen de coordenadas a la esquina izquierda superior. Le damos como nombre de instancia marco_mc

Creamos 2 cuadros de introducción de texto (que van a ser donde vamos a introducir el ancho y el alto que le vamos a dar a nuestro marco_mc y les damos como nombres de instancia: miAncho y miAlto)

Por último creamos un botón. Una vez hecho esto y situado en la misma capa, creamos una capa nueva que va ser la de las acciones y en las mismas del fotograma en el que estamos escribimos:

```
onEnterFrame = function(){
    this.marco_mc._width += (_root.ancho -
this.marco_mc._width)*0.3;
    this.marco_mc._height += (_root.alto -
this.marco_mc._height)*0.3;
}
```

Esto lo que hace es que continuamente está asignando al valor del ancho y el alto del marco la operación que hay entre paréntesis; que es al contenido de la variable ancho y alto -más adelante veremos como rellenamos esas variables- le restamos el valor que tenía antes el marco y lo multiplicamos por 0.3 para que haga el efecto de la desaceleración.

NOTA: Podéis hacer una prueba para ver a lo que me refiero sien un nuevo fla escribís esto:

```
a = 100;
b += (a-b)*0.3;
trace(b);
```

El 0.3 por tanto indica la velocidad de deceleración. Cuanto más se acerque este número a 1, la velocidad será mayor, con lo que el movimiento lo apreciaremos menos.

Bien, ahora vamos al botón, lo seleccionamos y escribimos lo siguiente:

```
on(release){
    _root.ancho = this.miAncho.text;
    _root.alto = this.miAlto.text;
}
```

Al pulsar el ratón, éste guarda en las variables ancho y alto (las de antes) el valor correspondiente que hemos introducido en los cuadros de texto miAncho y miAlto.

Y eso es todo, solo queda darle a Ctrl+Enter y probarlo.

Podéis ver un ejemplo funcionando en mi blog http://www.martinpulido.com/tutoriales/actionscript/efecto_easing.html

Desinstalar windows messenger de windows xp

*Por Javier Calderón
Colaborador de mygnet*

Hello!! soy yo again!!
Bueno aquí les explico como eliminar una cuenta de Windows Messenger de Windows XP (es que a veces se vuelve molesto que te cargue)... Para poder desinstalarlo tienes dos opciones:

1. Inicio > Panel de control >Agregar o quitar programas
2. Dale click en "Agregar o quitar componentes de windows"
3. Se abrirá una ventana donde debes deseleccionar la casilla "Windows messenger" Luego le das siguiente hasta que finalice la desinstalación (es posible que reinicies la pc)

OPCION B:
Ve a Inicio > Ejecutar
Copia la siguiente línea y pégala allí

```
RunDll32advpack.dll,LaunchINFSection
%windir%\INF\mmsgs.inf,BLC.Remove
```

Ahora dale aceptar y sigue el proceso de la desinstalación
Reinicia el PC

60(en algunos casos las actualizaciones desde Windows update reinstalan el windows messenger y se vuelve tedioso estar repitiendo la opción B)

Eliminar inicio de sesión automático de MSN y Yahoo messenger en windows xp

Por Javier Calderón

Seguramente has ido a un café Internet o a casa de un amigo y de repente por error de click le das "iniciar automáticamente"..¿Y ahora como hago? !!!

Para poder eliminar el inicio de sesión automático de la cuenta con la iniciaste sesión (OJO solo win xp) sigue las siguientes instrucciones:

- 1) Ve a inicio > Panel de control
- 2) Da click en "Cuentas de usuario"
- 3) Elige la cuenta en la que iniciaste sesión
- 4) En la ventana de tu cuenta de usuario busca "Administrar tus contraseñas de red"
- 5) Se abrirá una nueva ventana con: "Nombres de usuarios y contraseñas almacenados"
- 6) Busca la cuenta con la que iniciaste sesión e inmediatamente dale click en "Quitar" a veces es necesario también quitar la cuenta "net/passport*" si es MSN
- 7) Cierra las ventanas que abriste y ahora abre el messenger
- 8) Y ahora tu cuenta no iniciará más automáticamente!!

Si me equivoque en algo, por favor corrijan...