



## Formularios en SVG, varios ejemplos (5965 lectures)

Per René Mérou, [H](http://h.says.it/) (<http://h.says.it/>)

Creado el 19/08/2010 02:56 modificado el 19/08/2010 02:56

He estado unos días intentando hacer un formulario dentro de un fichero de imágenes vectoriales SVG. No es inmediato y

Básicamente os presento varias opciones que he estado trasteando.

Primero de todo yo lo he querido hacer todo con software libre y para software libre. No he usado nada que nos produzca el quedar

El segundo punto es que el objetivo es que se pueda ver bien en Firefox 3.5 que es el que yo uso. Y por lo visto el Firefox 4 traerá

La base es usar javascript dentro del SVG, lo que le ha dado unas posibilidades muy interesantes que yo no conocía. Nunca he usad

He encontrado tres posibles soluciones que os presento. Una usando librerías lgpl que no permite acentos, con cliente y servidor que Firefox. Igual que el anterior plugin para SVG supongo que acabará dentro del Firefox en futuras versiones.

La versión con librerías es esta: (primero el svg cliente)

Se trata de un pequeño cliente que pide que se rellene un recuadro con un texto que al pulsar submit se envía al servidor como si fuera una página web. Se comprueba que la comunicación funcionó. (también se recibe un email). El email y el nombre está tomado de varias formas, una so

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN" "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd" [
]>
<svg viewBox="0 0 1024 768" width="100%" height="100%" xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink">
  <script type="text/ecmascript" xlink:href="helper_functions.js"/>
  <script type="text/ecmascript" xlink:href="mapApp.js"/>
  <script type="text/ecmascript" xlink:href="timer.js"/>
  <script type="text/ecmascript" xlink:href="textbox.js"/>
  <script type="text/ecmascript">
    <![CDATA[
      var myMapApp = new mapApp(false,undefined);
      var textbox1;
      var name="probando";
      var email="mi@email.es";
      //post request
      //preparing the post request
      var currentYPost = 140;
      var postDataObj = new getData("fsmmap-tra-recv.php",receiveData,"xml","post","",undefined);

      //callBack function that receives data from the server (php files)
      function receiveData(node) {
        .documentElement.appendChild(node);
        =node.firstChild.nextSibling.firstChild.nodeValue;
        writeOutTextContent('textbox1', 'mensaje:'+name +':','release');
      }

      function writeOutTextContent(textboxId,value,changeType) {
        if (changeType == "release") {
          .getElementById("resultTextContent").firstChild.nodeValue = "Content of "+textboxId+" is \""+value+"\"";
          //writeOutTextInBox(textbox1)
        }
      }

      function initialize(evt) {
```



```

//styles
var textStyles = {"font-family":"Arial,Helvetica","font-size":15,"fill":"dimgray"};
var boxStyles = {"fill":"white","stroke":"dimgray","stroke-width":1.5};
var cursorStyles = {"stroke":"red","stroke-width":1.5};
var selBoxStyles = {"fill":"blue","opacity":0.5};
var textYOffset = 22;

//create new textboxes
    = new TextBox("textbox1", "textbox1", "", 25, 20, 50, 200, 30, textYOffset, textStyles, boxStyles, cursorStyles);
}

//send formulary
function submit() {
    currentYPost += 20;
    var postData = currentYPost + '$' + document.getElementById("textbox1Text").firstChild.nodeValue + '$' + email;
    postDataObj = postDataObj.getData();
    postDataObj.postText = postDataObj.getData();
    //writeOutTextContent('textbox1', 'mensaje:' + name + ':', 'release'); document.getElementById("textbox1").firstChild.nodeValue = postDataObj.postText;
}
]]>
</script>
<rect x="-1000" y="-1000" width="3000" height="3000" fill="white" stroke="none"/>
<g id="textbox1">
<g font-family="Arial,Helvetica" font-size="13">
<text id="resultText" x="40" y="100" xml:space="preserve">ResultText</text>
<text x="250" y="70">This textbox (textbox 1) accepts up to 25 characters [a-zA-Z ].</text>
</g>
<g >
<text x="40" y="140" onclick="submit()" cursor="pointer" >Submit.</text>
</g>
</svg>

```

Ahora el servidor en PHP que coge el contenido enviado y crea un email con el que de inmediato envía y devuelve un mensaje al cliente.

```

<?php
header('Content-type: text/xml');
list($currentY,$name,$email) = explode("$",$HTTP_RAW_POST_DATA);
$cuerpo = "Formulary sended from $name<$email>\n\n";
$cuerpo .= "Name: " . $HTTP_POST_VARS["name"] . "\n";
$cuerpo .= "Email: " . $HTTP_POST_VARS["email"] . "\n";
$cuerpo .= "Comentarios: " . $HTTP_POST_VARS["coment"] . "\n"; //logicamente así por el momento
//mando el correo...primero monto las cabeceras de a donde quiero enviarlo.
$cabeceras = 'From: miotro@email.es' . "\r\n" .
'Reply-To: innecesario@email.es' . "\r\n" .
'X-Mailer: PHP/' . phpversion();
mail("h@saberlibre.net","Formulary received from:".$name.":".$cuerpo." -- ".$currentY,$currentY);
print '<g xmlns="http://www.w3.org/2000/svg">' . "\n";
print "\t" . "<text x="500" y="100" font-size="15">" . $name . "</text>" . "\n";
print "</g>\n";
?>

```

Se podría hacer un servidor SVG en vez de PHP pero eso obligaría a preparar el servidor para comprobar que todos los SVGs pedidos sean válidos.

La opción de incluir un formulario con foreignObject:

Este es un ejemplo solo a modo de introducción tomado de <http://de.wikibooks.org/wiki/SVG/Erweiterbarkeit>.<sup>(3)</sup>

Lo he puesto en utf-8 y reducido pero allá encontraréis más información.

```

<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
"http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
<svg viewBox="0 0 700 500"
xmlns="http://www.w3.org/2000/svg" version="1.1"
xmlns:xlink="http://www.w3.org/1999/xlink"
xml:lang="es">
<script type="text/ecmascript">
    <![CDATA[

```



```
//    añadid aquí- las funciones javascript que necesitéis

]]>
</script>
<title>Form into SVG</title>
<rect x="0" y="0" width="700" height="500" fill="#ffc" />
<g font-family="sans-serif" font-size="20">
<switch>
<foreignObject requiredExtensions="http://www.w3.org/1999/xhtml"
x="100" y="100" width="500" height="260">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <title>Form into SVG</title>
    <style type="text/css">
      body {display:block; font-family:sans-serif;
        height:250px; width:500px; overflow: auto}
      h1 {display:block; font-size:130%}
      p,div {display:block}
    </style>
  </head>
  <body>
    <h1>Form into SVG</h1>
    <p>Un párrafo cualquiera.</p>

    <div>
      <form action="elprogramaservervuestro.php" method="post" target="_blank">
        <p>Text:<input name="name" type="text" value="menganito?"
          size="10" maxlength="40" /></p>
      <p>
        <input type="hidden"
          name="email" value="unemailsecreto@email.es" />
        <input type="submit" value="Enviar " />
      </p>
    </form>
    </div>
  </body>
</html>
</foreignObject>
</switch>
</g>
</svg>
```

He comprobado que este servidor funciona en php devolviendo otro svg en otra pestaña o ventana. Todo eso es retocable y le falta

```
<?php
header('Content-type: text/xml');
$email    =$_HTTP_POST_VARS['email'];
$name     = $_HTTP_POST_VARS["name"];
$cuerpo   = "Formulary sended from $name<$email>\n\n";
$cuerpo .="Name: " . $_HTTP_POST_VARS["name"] . "\n";
$cuerpo .="Email: " . $_HTTP_POST_VARS["email"] . "\n";
$cuerpo .="Comentarios: " . $_HTTP_POST_VARS["coment"] . "\n";
//mando el correo...
$cabeceras = 'From: emaildestino@email.es' . "\r\n" .
'Reply-To: nohacefalta@email.es' . "\r\n" .
'X-Mailer: PHP/' . phpversion();
mail("recibir@email.es","Formulary received from:".$name.":",$cuerpo." -- ". $assigna);
print '<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN" "http://www.w3.org/Graphics/SVG/1.1"
]>
<svg viewBox="0 0 1024 768" width="100%" height="100%" xmlns="http://www.w3.org/2000/svg">
print '<g xmlns="http://www.w3.org/2000/svg">'. "\n";
print "\t". '<text x="20" y="55" font-size="15">Gracias '.$name.' el formulario ha sido enviado.' . "\n";
print "</g>\n</svg>\n";
?>
```

Y para finalizar un SVG con XForms que da unos resultados muy guapos pero que está mucho mejor explicado [aquí](#)<sup>(4)</sup>.



Necesitaréis tener [este plugin](#)<sup>(5)</sup>.

El código mío es LGPL lo que he tomado de esas webs, mirad allá. Lo podéis usar sin problemas.

Ah, sobre poner código de colores, dar formato html a código PHP o SVG he usado el programa que os podéis bajar con un aptitud

## Enlaces interesantes:

<http://es.php.net/><sup>(7)</sup>

Carto.net es un sitio muy interesante con muchos ejemplos útiles como:

[Manipulating SVG Documents Using ECMAScript \(Javascript\) and the DOM](#)<sup>(8)</sup> [SVG in HTML pages](#)<sup>(9)</sup> [Network Requests with g](#)

<http://www.svgopen.org/><sup>(11)</sup>

[Javascript Post Request like a Form Submit](#)<sup>(12)</sup>

<http://pilatinfo.org/><sup>(13)</sup>

[SVG - Learning by Coding](#)<sup>(14)</sup>

[An SVG Primer for Today's Browsers](#)<sup>(15)</sup>

[Form elements in SVG UI](#)<sup>(16)</sup>

[Javascript, SVG and DOM](#)<sup>(17)</sup>

[SVG Programmers' Application Resource Kit](#)<sup>(18)</sup>

[Ejemplos de proyecto pity](#)<sup>(19)</sup>

[Firefox 4 con mejoras para poder añadir formularios a un SVG](#)<sup>(20)</sup>

[Using the XML HTTP Request object](#)<sup>(21)</sup> (no lo usé)

[Formulario dinámico](#)<sup>(22)</sup> para conocer un poco el javascript.

[html5 Semantics, structure, and APIs of HTML documents](#)<sup>(23)</sup> -> dom

[Dynamically updating SVG based on serverside information.](#)<sup>(24)</sup>

[SVG/ Erweiterbarkeit](#)<sup>(25)</sup> Aleman, no se entiende, pero el código sí. :)

[Inline SVG in HTML5 and XHTML](#)<sup>(26)</sup>

[XForms 1.1](#)<sup>(4)</sup>

[Mozilla XForms](#)<sup>(5)</sup>

[SVG 1.1 working drafst](#)<sup>(27)</sup>

[Apprivoiser SVG](#)<sup>(28)</sup> muy buen manual pero en francés.

Si alguno de los lectores tiene algún otro truco o suele usar un código similar, veámoslo.

(no hay nada como que te llenen de comentarios un artículo con ideas diferentes como pasó en el de [trucos para el prompt de la she](#)

### Lista de enlaces de este artículo:

1. <http://bulma.net/body.phtml?nIdNoticia=1682>
2. <http://www.carto.net/papers/svg/gui/textbox/>
3. [http://de.wikibooks.org/wiki/SVG/\\_Erweiterbarkeit](http://de.wikibooks.org/wiki/SVG/_Erweiterbarkeit)
4. <http://www.w3.org/TR/xforms11/#concepts-xhtml>
5. <https://addons.mozilla.org/en-US/firefox/addon/824/>
6. [http://www.w3clubs.com/sp/hilite/hi\\_test1.php](http://www.w3clubs.com/sp/hilite/hi_test1.php)
7. <http://es.php.net/>
8. [http://www.carto.net/papers/svg/manipulating\\_svg\\_with\\_dom\\_ecmascript/index.shtml](http://www.carto.net/papers/svg/manipulating_svg_with_dom_ecmascript/index.shtml)
9. [http://www.carto.net/papers/svg/samples/svg\\_html.shtml](http://www.carto.net/papers/svg/samples/svg_html.shtml)
10. [http://www.carto.net/papers/svg/network\\_requests/index.shtml](http://www.carto.net/papers/svg/network_requests/index.shtml)
11. <http://www.svgopen.org/>
12. <http://stackoverflow.com/questions/133925/javascript-post-request-like-a-form-su>
13. <http://pilatinfo.org/english/index.htm>
14. <http://www.datenverdrahten.de/svglbc/?code=posturl&znr=on>
15. [http://www.w3.org/Graphics/SVG/IG/resources/svgprimer.html#text\\_methods](http://www.w3.org/Graphics/SVG/IG/resources/svgprimer.html#text_methods)
16. <http://forums.java.net/jive/thread.jspa?threadID=30594&tstart=0>
17. [http://pilatinfo.org/english/routines/js\\_dom.htm](http://pilatinfo.org/english/routines/js_dom.htm)
18. <http://spark.sourceforge.net/news/index.html#20040518>
19. [http://sourceforge.net/projects/pity/files/pity/0.1.10/pity\\_0.1.10.tgz/download](http://sourceforge.net/projects/pity/files/pity/0.1.10/pity_0.1.10.tgz/download)
20. <http://www.mozilla.com/ca/firefox/beta/#privacy>
21. <http://jibbering.com/2002/4/httprequest.html>



22. <http://soporte.miarroba.es/152498/2764058-formulario-dinamico/>
23. <http://www.w3.org/TR/html5/dom.html>
24. <http://www.jibbering.com/2002/5/dynamic-update-svg.html>
25. [http://de.wikibooks.org/wiki/SVG/\\_Erweiterbarkeit](http://de.wikibooks.org/wiki/SVG/_Erweiterbarkeit)
26. <http://dev.w3.org/SVG/proposals/svg-html/svg-html-proposal.html>
27. <http://www.w3.org/TR/SVG11/expanded-toc.html>
28. <http://pilatinfo.org/learnsvg/index.htm>
29. <http://bulma.net/body.phtml?nIdNoticia=1810>

---

E-mail del autor: ochominutosdearco\_ARROBA\_gmail.com

**Podrás encontrar este artículo e información adicional en:** <http://bulma.net/body.phtml?nIdNoticia=2589>