

Crear XML con PHP (I)



Los formatos de intercambio de datos permiten pasar datos de una aplicación a otra empleando normalizaciones que son aceptadas, entendibles y manejables por cualquier lenguaje de programación moderno. Esto permite compartir información entre distintas aplicaciones, escritas, incluso, en lenguajes diferentes, agilizando muchas operaciones.

Los formatos más conocidos son, sin duda, RSS, XML y JSON. En esta serie de artículos nos vamos a centrar en el intercambio de datos en XML con PHP.

En primer lugar vamos a ver cómo crear un XML con datos procedentes de cualquier origen en nuestra aplicación (entradas de un formulario, lectura de una BBDD, etc).

CREAR XML A "LO BRUTO"

Existen dos maneras de crear un XML. A "lo bruto" y bien, mediante los recursos que PHP nos ofrece. En este primer artículo vamos a aprender a hacerlo "a lo bruto". Lo que aquí vamos a exponer es la forma en que NO debe crearse un XML, **NUNCA, JAMÁS, DE NINGUNA MANERA, Y POR NINGÚN CONCEPTO**. Y, sí es así, ¿por qué vamos a dedicarle un artículo? La respuesta breve y concisa es: porque sí. Porque, de esta forma, entenderemos la naturaleza de un archivo XML y comprenderemos cómo podemos generarlo de modo que sea entendible y procesable por cualquier aplicación.

Por lo tanto, vamos a dejar claro que el objetivo de este artículo es, digámoslo así, pre-didáctico. Será en el siguiente artículo cómo aprenderemos la forma correcta de crear un XML con PHP. Sin embargo, todos, en alguna ocasión, hemos sufrido la tentación de hacerlo cómo vamos a ver aquí. Y, si copias los códigos que incluiremos verás que, funcionar, lo que se dice funcionar, funcionan pero, si yo usara este método para crear un XML en mi trabajo, me supondría una merecida larga estancia en la cola del paro (al menos, si se enterara mi jefe, claro ;-)).

Vamos a partir de una colección de datos, en una estructura definida, que podemos tener, por ejemplo, en una matriz, cómo la que vemos a continuación:

```
array(3) {
  [0]=>{
    ["obra"]=>"Construcción de aparcamiento en el centro"
    ["fecha_de_inicio"]=>"10/02/2015"
    ["fecha_de_finalizacion"]=>"31/10/2016"
    ["contratista"]=>"Actuaciones Urbanas"
    ["miembros_tecnicos"]=>"3"
    ["personal_tecnico"]=>{
      ["arquitecto"]=>"Pedro Steven De Gloria"
      ["aparejador"]=>"Manuela Gracia Salmerón"
      ["supervisor"]=>"Andrés Garrido Fuentes"
    }
    ["presupuesto"]=>"20.000.000"
  }
  [1]=>{
    ["obra"]=>"Adaptación de estación suburbana"
    ["fecha_de_inicio"]=>"06/08/2016"
    ["fecha_de_finalizacion"]=>"01/11/2016"
    ["contratista"]=>"Obras del Norte"
    ["miembros_tecnicos"]=>"4"
    ["personal_tecnico"]=>{
```

```
["arquitecto"]=>"Manuel Alarcón Rodríguez"  
["aparejador"]=>"Carlos Torres Fuentes"  
["director_de_tunelacion"]=>"María García Pérez"  
["jefe_de_compras"]=>"Antonia Bisonette Tristán"  
}  
["presupuesto"]=>"6.500.000"  
}  
[2]=>{  
["obra"]=>"Electrificación de zona restringida"  
["fecha_de_inicio"]=>"02/02/2014"  
["fecha_de_finalizacion"]=>"26/05/2017"  
["contratista"]=>"Iluminación y Electricidad, SA"  
["miembros_tecnicos"]=>"2"  
["personal_tecnico"]=>{  
["jefe_de_electricistas"]=>"Laura De la Iglesia Cifuentes"  
["responsable_de_control"]=>"Yolanda Torres Torres"  
}  
["presupuesto"]=>"7.800.000"  
}  
}
```

Cómo puedes ver, se trata de una colección estructurada de datos que podemos querer generar con nuestra aplicación para que sea entendible por otra distinta. La salida en XML podría responder a lo siguiente:

```
<?xml version="1.0" encoding="UTF-8"?>  
<obras>  
<obra inicio="10/02/2015" final="31/10/2016" contratista="Actuaciones Urbanas" presupuesto="20.000.000">  
  Construcción de aparcamiento en el centro  
<personal_tecnico miembros="3">  
<miembro cargo="arquitecto">  
  Pedro Steven De Gloria  
</miembro>  
<miembro cargo="aparejador">  
  Manuela Gracia Salmerón  
</miembro>  
<miembro cargo="supervisor">  
  Andrés Garrido Fuentes  
</miembro>  
</personal_tecnico>  
</obra>  
<obra inicio="06/08/2016" final="01/11/2016" contratista="Obras del Norte" presupuesto="6.500.000">  
  Adaptación de estación suburbana  
<personal_tecnico miembros="4">  
<miembro cargo="arquitecto">  
  Manuel Alarcón Rodríguez  
</miembro>  
<miembro cargo="aparejador">  
  Carlos Torres Fuentes  
</miembro>  
<miembro cargo="director_de_tunelacion">  
  María García Pérez  
</miembro>
```

```
<miembro cargo="jefe_de_compras">
  Antonia Bisonette Tristán
</miembro>
</personal_tecnico>
</obra>
<obra inicio="02/02/2014" final="26/05/2017" contratista="Iluminación y Electricidad, SA" presupuesto="7.800.000">
  Electrificación de zona restringida
  <personal_tecnico miembros="2">
    <miembro cargo="jefe_de_electricistas">
      Laura De la Iglesia Cifuentes
    </miembro>
    <miembro cargo="responsable_de_control">
      Yolanda Torres Torres
    </miembro>
  </personal_tecnico>
</obra>
</obras>
```

Por supuesto, el formato de salida puede variar, en base al acuerdo al que se llegue con el fabricante de la otra aplicación, pero, para nuestro ejemplo, este está bien. En todo caso es un formato normalizado.

El PHP que, a partir de la matriz, genera el fichero XML es el siguiente:

```
<?php
/* Empezamos con una matriz de datos que puede proceder de cualquier fuente
(p.e. una lectura de base de datos) */
$matrizDeObras = array(
  array(
    "obra"=>"Construcción de aparcamiento en el centro",
    "fecha_de_inicio"=>"10/02/2015",
    "fecha_de_finalizacion"=>"31/10/2016",
    "contratista"=>"Actuaciones Urbanas",
    "miembros_tecnicos"=>"3",
    "personal_tecnico"=>array(
      "arquitecto"=>"Pedro Steven De Gloria",
      "aparejador"=>"Manuela Gracia Salmerón",
      "supervisor"=>"Andrés Garrido Fuentes"
    ),
    "presupuesto"=>"20.000.000"
  ),
  array(
    "obra"=>"Adaptación de estación suburbana",
    "fecha_de_inicio"=>"06/08/2016",
    "fecha_de_finalizacion"=>"01/11/2016",
    "contratista"=>"Obras del Norte",
    "miembros_tecnicos"=>"4",
    "personal_tecnico"=>array(
      "arquitecto"=>"Manuel Alarcón Rodríguez",
      "aparejador"=>"Carlos Torres Fuentes",
      "director_de_tunelacion"=>"María García Pérez",
      "jefe_de_compras"=>"Antonia Bisonette Tristán"
    ),
    "presupuesto"=>"6.500.000"
```

```
),  
array(  
  "obra"=>"Electrificación de zona restringida",  
  "fecha_de_inicio"=>"02/02/2014",  
  "fecha_de_finalizacion"=>"26/05/2017",  
  "contratista"=>"Iluminación y Electricidad, SA",  
  "miembros_tecnicos"=>"2",  
  "personal_tecnico"=>array(  
    "jefe_de_electricistas"=>"Laura De la Iglesia Cifuentes",  
    "responsable_de_control"=>"Yolanda Torres Torres"  
  ),  
  "presupuesto"=>"7.800.000"  
)  
);
```

```
// Convertimos la matriz a una cadena con formato XML.  
$textoXML = '<?xml version="1.0" encoding="UTF-8"?>';  
$textoXML .= "n";  
$textoXML .= '<obras>';  
$textoXML .= "n";  
foreach ($matrizDeObras as $obra){  
  $textoXML .= "t";  
  $textoXML .= '<obra inicio="'. $obra["fecha_de_inicio"]. "'>';  
  $textoXML .= 'final="'. $obra["fecha_de_finalizacion"]. "'>';  
  $textoXML .= 'contratista="'. $obra["contratista"]. "'>';  
  $textoXML .= 'presupuesto="'. $obra["presupuesto"]. "'>';  
  $textoXML .= "n";  
  $textoXML .= "tt";  
  $textoXML .= $obra["obra"];  
  $textoXML .= "n";  
  $textoXML .= "tt";  
  $textoXML .= '<personal_tecnico miembros="'. $obra["miembros_tecnicos"]. "'>';  
  $textoXML .= "n";  
  foreach ($obra["personal_tecnico"] as $keyMiembro=>$miembro){  
    $textoXML .= "t";  
    $textoXML .= '<miembro cargo="'. $keyMiembro. "'>';  
    $textoXML .= "n";  
    $textoXML .= "tt";  
    $textoXML .= $miembro;  
    $textoXML .= "n";  
    $textoXML .= "tt";  
    $textoXML .= '</miembro>';  
    $textoXML .= "n";  
  }  
  $textoXML .= "tt";  
  $textoXML .= '</personal_tecnico>';  
  $textoXML .= "n";  
  $textoXML .= "t";  
  $textoXML .= '</obra>';  
  $textoXML .= "n";  
}
```

```
}  
$textoXML .= '</obras>';  
  
// Nos aseguramos de que la cadena que contiene el XML esté en UTF-8  
$textoXML = mb_convert_encoding($textoXML, "UTF-8");  
  
// Grabamos el XML en el servidor como un fichero plano, para  
// poder ser leído por otra aplicación.  
$gestor = fopen("obras.xml", 'w');  
fwrite($gestor, $textoXML);  
fclose($gestor);  
?>
```

Y aquí es donde está la chapuza flagrante. Cómo ves, el XML se ha generado cómo una cadena de texto, pura y dura, a partir de los datos contenidos en la matriz (observa las líneas resaltadas). Si -dirás-, pero el código funciona. Es cierto, pero, en primer lugar, hiere a la vista. Además, es incómodo y difícil de mantener. Si hay que cambiar la estructura del XML, por ejemplo con los atributos o los contenidos de los elementos, o añadir espacios de nombres, sólo en "ver" cómo funciona y modificarlo se pierde mucho tiempo.

Lo cierto es que PHP nos proporciona medios más que suficientes para lograr el mismo resultado de una forma mucho más limpia, eficiente y mantenible en el tiempo. En el [próximo artículo](#) veremos la forma correcta de escribir un documento XML.