

1 Curso POO PHP Servizos web: WSDL e SOAP

1.1 Servizos web: WSDL e SOAP

O **protocolo RPC** creouse para permitir a un sistema acceder de forma remota a funcións ou procedementos que se encontren noutro sistema. O cliente conéctase co servidor, e indícalle que función debe executar. O servidor execútaa e devólvelle o resultado obtido. Así, por exemplo, podemos crear no servidor RPC unha función que reciba un código de produto e devolva o seu PVP. RPC usa o seu propio porto, pero normalmente só a xeito de directorio. Os clientes conéctanse a el para obter o porto real do servizo que lles interesa. Este porto non é fixo; asígnase de forma dinámica.

Os **servizos web** créanse para permitir o intercambio de información ao igual que RPC, pero en lugar de definir o seu propio protocolo para transportar as peticións de información, utilizan HTTP para este fin. A resposta obtida non será unha páxina web, senón a información que se solicitou. Desta forma poden funcionar sobre calquera servidor web; e, o que é aínda máis importante, utilizando o porto 80 reservado para este protocolo. Polo tanto, calquera ordenador que poida consultar unha páxina web, poderá tamén solicitar información dun servizo web. Se existe algún firewall na rede, tratará a petición de información igual que o faría coa solicitude dunha páxina web.

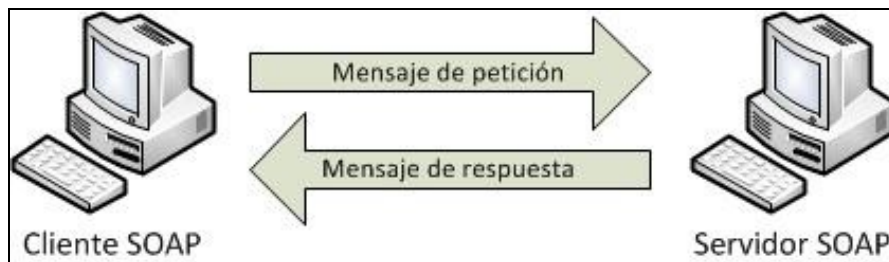
Un servizo web baséase normalmente en:

- O protocolo **SOAP**, que utiliza a linguaxe XML para intercambiar información entre o cliente e o servidor.
- A linguaxe **WSDL**, que tamén está baseada en XML e foi creada para describir servizos web, é dicir, indicar como se debe acceder a un servizo e utilizalo.

1.1.1 SOAP

SOAP é un protocolo que indica como deben ser as mensaxes que se intercambien o servidor e o cliente, como deben procesarse estas, e como se relacionan co protocolo que se utiliza para transportalas dun extremo ao outro da comunicación (no caso dos servizos web, este protocolo será HTTP).

O nome SOAP xurdiu como acrónimo de *Simple Object Access Protocol*, pero, a partir da versión 1.2 do protocolo, o nome SOAP xa non se refire a nada en concreto. Ao igual que o seu antecesor, XML-RPC, SOAP utiliza XML para compoñer as mensaxes que se transmiten entre o cliente (que xera unha petición) e o servidor (que envía unha resposta) do servizo web.



Nunha mensaxe SOAP, como mínimo debe figurar un elemento **Envelope**, que é o que identifica ao documento XML como unha mensaxe SOAP, e onde se deben declarar polo menos os seguintes espazos de nomes:

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  soap:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
```

Como primeiro membro do elemento *Envelope*, pode haber de forma opcional un elemento **Header**. Se existe, pode conter varios elementos con información adicional sobre como procesar a mensaxe SOAP. A continuación debe figurar obrigatoriamente un elemento **Body**, que é onde se inclúe, dependendo do tipo de mensaxe, a petición ou a resposta.

1.1.2 WSDL (*Web Services Description Language*)

WSDL é unha linguaxe baseada en XML que utiliza unhas regras determinadas para xerar o documento de descrición dun servizo web. Unha vez xerado, ese documento adóitase poñer a disposición dos posibles usuarios do servizo (normalmente accédese ao documento WSDL engadindo **?wsdl** á URL do servizo). O espazo de nomes dun documento WSDL é <http://schemas.xmlsoap.org/wsdl/>, aínda que nun documento WSDL se adoitan utilizar tamén outros espazos de nomes. A estrutura dun documento WSDL é a seguinte.

```
<definitions
  name="">
```

```

    targetNamespace="http://?"
    xmlns:tns="http://?"
    xmlns="http://schemas.xmlsoap.org/wsdl/"
  ?
>
  <types>
    ?
  </types>

  <message>
    ?
  </message>

  <portType>
    ?
  </portType>

  <binding>
    ?
  </binding>

  <service>
    ?
  </service>
</definitions>

```

O obxectivo de cada unha das seccións do documento é o seguinte:

- **types**. Inclúe as definicións dos tipos de datos que se usan no servizo.
- **message**. Define conxuntos de datos, como a lista de parámetros que recibe unha función ou os valores que devolve.
- **portType**. Cada portType é un grupo de funcións que implementan o servizo web. Cada función defínese dentro do seu portType como unha operación (**operation**).
- **binding**. Define como vai transmitirse a información de cada *portType*.
- **service**. Contén unha lista de elementos de tipo **port**. Cada port indica onde (en que URL) se pode acceder ao servizo web.

--Víctor Lourido 01:14 26 jul 2013 (CEST)