

Servicios web en SOAP



Joan Vila

DISCA / UPV

**Departament d'Informàtica de Sistemes i Computadors
Universitat Politècnica de València**





Servicios web

● Índice



- Conceptos básicos
- SOAP
 - Estructura de SOAP
 - SOAP y WSDL
 - SOAP UDDI
- Soporte SOAP para Java
 - Interfaz dinámico de invocaciones
 - Interfaz estático de invocaciones (Axis)
- Soporte SOAP para PHP
- Un ejemplo en SOAP
 - La eurocalculadora en SOAP / Java
- XML-RPC



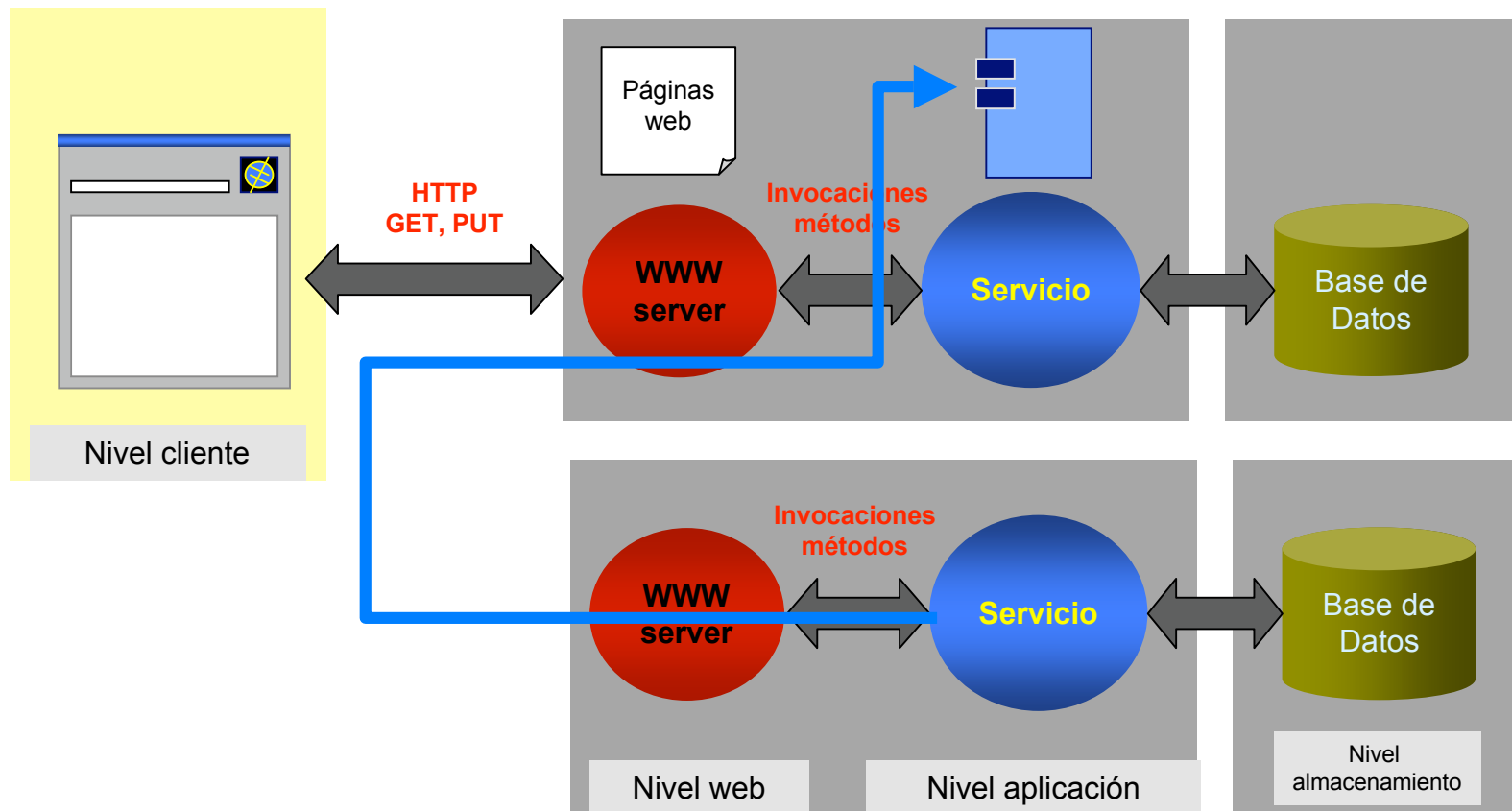
Servicios web

- ¿Qué es un servicio web?

- Es un servicio en Internet accesible con protocolos de RPC implementados sobre HTTP.
 - Basado en invocaciones RPC (invocación-respuesta) semejantes a las invocaciones a procedimientos u objetos remotos.
 - Permite utilizar el paradigma de objetos distribuidos en Internet
- **Ejemplos:**
 - Servicio de fax, servicio de SMS, motor de búsqueda de información (Google), servicio de información bursátil, servicio de traducción de texto, etc...
 - Catálogo: <http://services.xmethods.net>
- Base para computación distribuida con HTTP
 - Interfaz programática: invocable desde programas; no solo desde páginas web.
 - Permite que un servidor (programa CGI) pueda convertirse en cliente de otros servicios.

Servicios web

- Computación distribuida con servicios web





Servicios web

● Protocolos

- Transferencia de datos y protocolos basados en XML
 - **XML-RPC**: es la base.
 - Llamadas remotas codificadas en XML que son menos eficientes que en binario pero mucho más estándares y estructuradas.
 - <http://www.xmlrpc.com>
 - **SOAP**: extiende mejora XML-RPC. Lo que mucha gente entiende como “web services”.
 - Uso extensivo de espacios de nombres y etiquetas de marcado.
 - Sponsorizada por Microsoft como estándar y apoyada por IBM. Base de la campaña .NET.
 - <http://www.w3.org/TR/SOAP>
 - **XML-RPC vs SOAP**:
 - http://weblog.masukomi.org/writings/xml-rpc_vs_soap.htm



Servicios web

- **Indice**

- Conceptos básicos



- SOAP

- Estructura de SOAP
 - SOAP y WSDL
 - SOAP UDDI

- Soporte SOAP para Java

- Interfaz dinámico de invocaciones
 - Interfaz estático de invocaciones (Axis)

- Soporte SOAP para PHP

- Un ejemplo en SOAP

- La eurocalculadora en SOAP / Java

- XML-RPC



SOAP

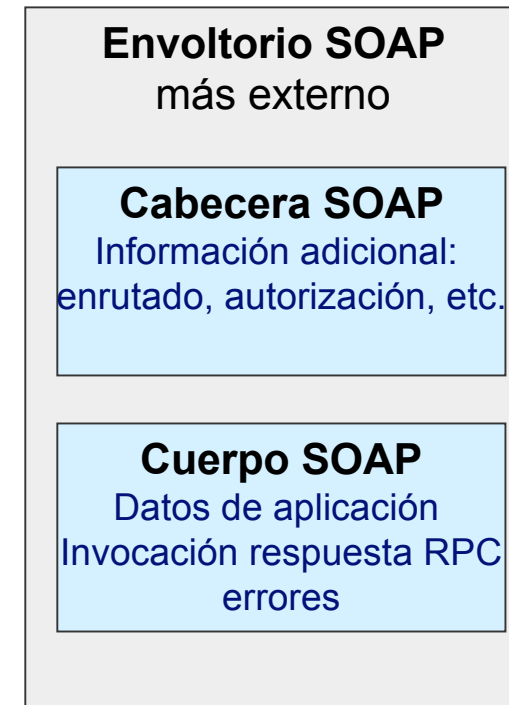
● SOAP

- Es un estándar basado en XML para la mayoría de servicios web públicos
- Usa un **lenguaje de definición de servicios** denominado WSDL (análogo a IDL), donde se definen los servicios a publicar, los tipos de datos a enviar y retornar. El WSDL es como la URL en XML-RPC.
 - **WSDL**: Web Services Definition Language
 - <http://www.w3.org/TR/2001/NOTE-wsdl-20010315>
- Existe un **directorio de servicios web** para descubrir de forma dinámica nuevos servicios:
 - **UDDI**: Universal Description Directory Integration



SOAP

- Estructura de un mensaje SOAP
 - El **envoltorio** no contiene datos de aplicación, solo empaqueta el contenido.
 - La **cabecera** opcional puede contener información de control
 - El **cuerpo** contiene los datos en XML
 - Los **datos adjuntos** (*attachments*) pueden contener otros tipos de datos (binarios, uuencoded, etc.)





SOAP

- Ejemplo simple
 - **Servicio de seguimiento de vuelos web**
 - Llamada con código de portador y número de vuelo:
 - `public Date getFlightArrival(String carrier, int number)`
 - Retorna hora estimada de llegada:
 - Arrival for AS 25 is Sat Jun 15 00:32:19 PDT 2002

<http://www.flytecomm.com/cgi-bin/trackflight>



SOAP

- Formato de la petición SOAP

```
<SOAP-ENV:Envelope SOAP-ENV:encodingStyle=
  "http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/">
  <SOAP-ENV:Body>
    <getFlightArrival>
      <op1 xsi:type="xsd:string">AS</op1>
      <op2 xsi:type="xsd:int">25</op2>
    </getFlightArrival>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```



SOAP

- Formato de la respuesta SOAP

<SOAP-ENV:Envelope

xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"

xmlns:xsd="http://www.w3.org/2001/XMLSchema"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

<SOAP-ENV:Body>

<getFlightArrivalResponse SOAP-ENV:encodingStyle=

"http://schemas.xmlsoap.org/soap/encoding/">

<getFlightArrivalResult xsi:type="xsd:dateTime">2002-06-15T07:32:19.403Z

</getFlightArrivalResult>

</getFlightArrivalResponse>

</SOAP-ENV:Body>

</SOAP-ENV:Envelope>

XML Schema: Structures specifies the XML Schema definition language, which offers facilities for describing the structure and constraining the contents of XML 1.0 documents, including those which exploit the XML Namespace facility.



WSDL

<http://www.w3.org/TR/wsdl>

- Anatomía de un documento WSDL

Si se examina cada parte de un documento WSDL se encontrará:

- **<definitions>**. El elemento **<definitions>** contiene la definición de uno o más servicios. En la mayoría de los casos, un archivo WSDL define un servicio únicamente. Seguido de la etiqueta de definición se encontrarán declaraciones de algunos atributos:

- **<message>** y **<portType>**, describe qué operaciones provee el servicio.
- **<service>**, agrupa un conjunto de ports relacionados.
- **<binding>**, endpoint, que describe detalles del protocolo a usar.
- **<documentation>**, cualquier elemento WSDL puede contener información del servicio para el usuario.

- Generando la descripción de servicio WSDL

- La mayoría de las herramientas incluyen una forma de generar WSDL a partir de un componente.
 - incluyendo el Apache Jakarta Axis, WSTK de IBM y el .NET Studio de Microsoft.



WSDL

- Ejemplo de WSDL (i)

```
<?xml version="1.0" encoding="UTF-8"?>
  <definitions name="FlightService" targetNamespace="http://DefaultNamespace">
    <message name="getFlightArrival">
      <part name="String_1" type="xsd:string"/>
      <part name="int_2" type="xsd:int"/>
    </message>
    <message name="getFlightArrivalResponse">
      <part name="result" type="xsd:dateTime"/>
    </message>
    <portType name="FlightRemote">
      <operation name="getFlightArrival">
        <input message="tns:getFlightArrival"/>
        <output message="tns:getFlightArrivalResponse"/>
      </operation>
    </portType>
  </definitions>
(continued...)
```

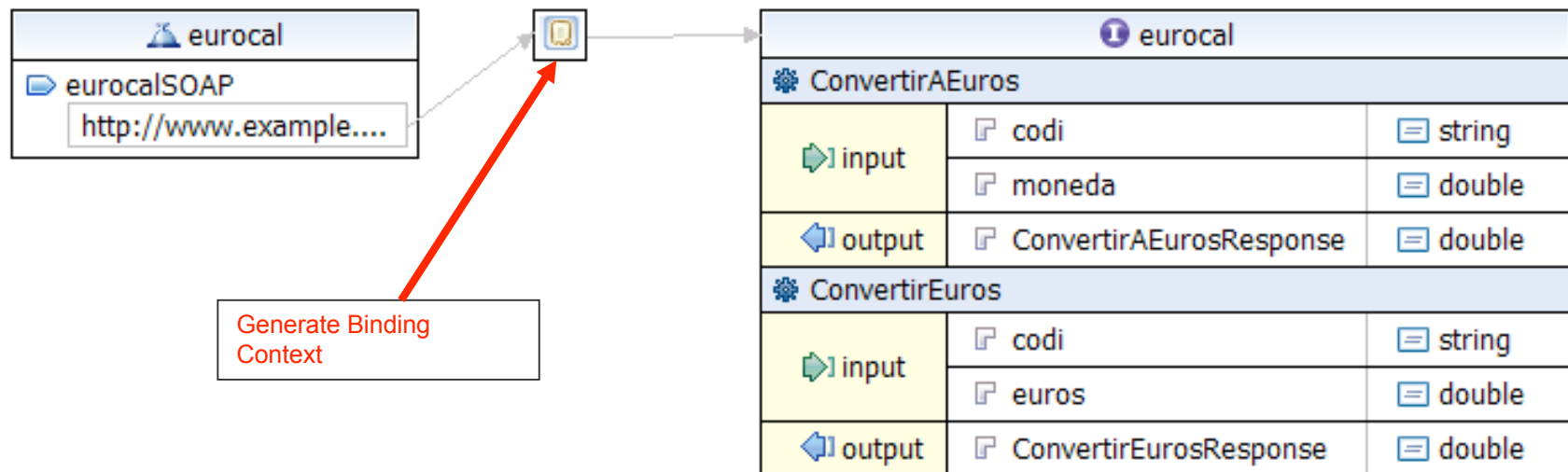


WSDL

- Ejemplo de WSDL (ii)

```
<binding name="FlightRemoteBinding" type="tns:FlightRemote">
  <operation name="getFlightArrival">
    <input><soap:body
      encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      use="encoded" namespace="http://allflighttracking.com/wsdl"/>
    </input>
    <output><soap:body
      encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      use="encoded" namespace="http://allflighttracking.com/wsdl"/>
    </output>
    <soap:binding transport=http://schemas.xmlsoap.org/soap/http style="rpc"/>
  </binding>
<service name="Flight">
  <port name="FlightRemotePort" binding="tns:FlightRemoteBinding">
    <soap:address location="http://localhost:8080/axis/..."/>
  </port>
</service>
</definitions>
```

- Otro ejemplo de WSDL (i)





WSDL

- Otro ejemplo de WSDL (i)

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:tns="http://www.example.org/eurocal/"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" name="eurocal"
  targetNamespace="http://www.example.org/eurocal/">
  <wsdl:message name="ConvertirAEurosRequest">
    <wsdl:part name="codi" type="xsd:string"></wsdl:part>
    <wsdl:part name="moneda" type="xsd:double"></wsdl:part>
  </wsdl:message>
  <wsdl:message name="ConvertirAEurosResponse">
    <wsdl:part name="ConvertirAEurosResponse" type="xsd:double"></wsdl:part>
  </wsdl:message>
  <wsdl:message name="ConvertirEurosRequest">
    <wsdl:part name="codi" type="xsd:string"></wsdl:part>
    <wsdl:part name="euros" type="xsd:double"></wsdl:part>
  </wsdl:message>
  <wsdl:message name="ConvertirEurosResponse">
    <wsdl:part name="ConvertirEurosResponse" type="xsd:double"></wsdl:part>
  </wsdl:message>
```




WSDL

- Otro ejemplo de WSDL (ii)

```
<wsdl:portType name="eurocal">
  <wsdl:operation name="ConvertirAEuros">
    <wsdl:input message="tns:ConvertirAEurosRequest"></wsdl:input>
    <wsdl:output message="tns:ConvertirAEurosResponse"></wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="ConvertirEuros">
    <wsdl:input message="tns:ConvertirEurosRequest"></wsdl:input>
    <wsdl:output message="tns:ConvertirEurosResponse"></wsdl:output>
  </wsdl:operation>
</wsdl:portType>
```



WSDL

- Otro ejemplo de WSDL (iii)

```
<wsdl:binding name="eurocalSOAP" type="tns:eurocal">
  <soap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http" />

  <wsdl:operation name="ConvertirAEuros">
    <soap:operation soapAction="http://www.example.org/eurocal/ConvertirAEuros" />
    <wsdl:input>
      <soap:body use="literal" namespace="http://www.example.org/eurocal/" />
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal" namespace="http://www.example.org/eurocal/" />
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="ConvertirEuros">
    ...
  </wsdl:operation>
</wsdl:binding>
<wsdl:service name="eurocal">
  <wsdl:port binding="tns:eurocalSOAP" name="eurocalSOAP">
    <soap:address location="http://www.example.org/" />
  </wsdl:port>
</wsdl:service>

</wsdl:definitions>
```

SOAP- UDDI

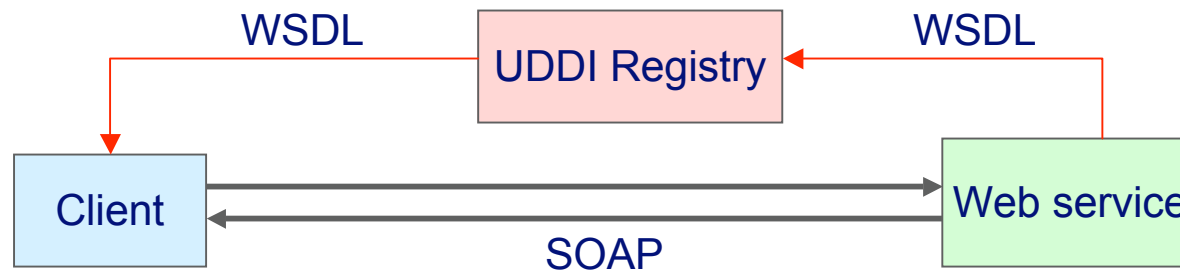
- Servicio de directorio UDDI

- UDDI (Universal Description Directory Integration)

- Servicio de directorios estandarizado
 - Automaticamente registra y busca los servicios
 - 7000 empresas registradas?

- Funcionalidad

- Registra y retorna **descripciones WSDL en formato XML**
 - Hay algunas críticas que dicen que no hay forma automatizada de que los programas usen la información y que se requiere intervención humana para escribir el código...
 - Llenará WSIF el hueco???





Servicios web

● Índice

- Conceptos básicos
- SOAP
 - Estructura de SOAP
 - SOAP y WSDL
 - SOAP UDDI
- ➔ – Soporte SOAP para Java
 - Interfaz dinámico de invocaciones
 - Interfaz estático de invocaciones (Axis)
- Soporte SOAP para PHP
- Un ejemplo en SOAP
 - La eurocalculadora en SOAP / Java
- XML-RPC



Soporte SOAP para Java

- Soporte SOAP para Java

- JAX: Java API for XML-based RPC

- Core technology for J2EE 1.4
 - JSR-109 define los descriptores para desarrollo de servicios web
 - Con EJB 2.1 soporta directamente servicios web tipo beans con sesiones sin estado

- Funcionalidad

- JAX-RPC mapea SOAP/WSDL a RMI:

- Porting de WSDL a interfaces Remote de RMI (interface extending java.rmi.Remote)
 - Invocaciones WSDL mapeadas a invocaciones RMI
 - Los métodos propagan la excepción java.rmi.RemoteException (Los fallos SOAP se convierten a RemoteExceptions)

- Soporta un subconjunto de RMI

- Solo soporta algunos tipos básicos - primitives and wrappers, bean-likes, String, Date, BigInteger, BigDecimal, etc.
 - Others will use JAXB or pluggable serializers



Soporte SOAP para Java

- Tipos de invocaciones SOAP
 - Estática:
 - Precisa la generación de stubs derivados de la definición del servicio en WSDL
 - Invocación del estilo:
 - referenciaObjeto->nombreMetodo(argumentos)
 - Dinámica:
 - **No** precisa generación de stubs
 - Invocación del estilo:
 - Call call = (Call) service.createCall();
 - // Construir invocación
 - call.invoke(...);



Soporte SOAP para Java

- Invocación dinámica (DII)
 - Funciona **sin stubs estáticos**: configura la información del servicio “al vuelo”:
 - Un servicio **javax.xml.rpc.Service** se crea instanciando la clase:
 - **javax.xml.rpc.ServiceFactory**
 - Una llamada se crea y se configura (endpoint, operation, parameters, return type, etc.) con la clase
 - **javax.xml.rpc.**
 - La llamada se ejecuta con:
 - **call.invoke()**
 - Adecuado para servicios simples



Soporte SOAP para Java

- Invocación dinámica (DII)

```
import org.apache.axis.client.Call;
import org.apache.axis.client.Service;
import javax.xml.namespace.QName;

public class TestClient {
    public static void main(String [] args) {
        try {
            String endpoint = "http://nagoya.apache.org:5049/axis/services/echo";
            Service service = new Service();

            Call call = (Call) service.createCall();
            call.setTargetEndpointAddress( new java.net.URL(endpoint) );
            call.setOperationName(new QName("http://soapinterop.org/", "echoString"));
            String ret = (String) call.invoke( new Object[] { "Hello!" } );
            System.out.println("Sent 'Hello!', got '" + ret + "'");
        } catch (Exception e) { System.err.println(e.toString()); }
    }
}
```




Soporte SOAP para Java

- Invocación estática

- Invocación de métodos con estilo RPC estático:
 - referenciaObjeto->nombreMetodo(argumentos)
- Proyecto **Axis**
 - <http://ws.apache.org/axis/java>
 - Successor to Apache SOAP (itself originally based on IBM SOAP4J)
- Basado en la **generación automática de stubs a partir de WSDL**
 - **Java2WSDL**: para generar WSDL y stubs a partir de las clases Java que implementan el servicio
 - **WSDL2Java**: para generar clases Java a partir de WSDL
- Más fácil de usar que JAX-RPC
 - Y compatible con JAX-RPC

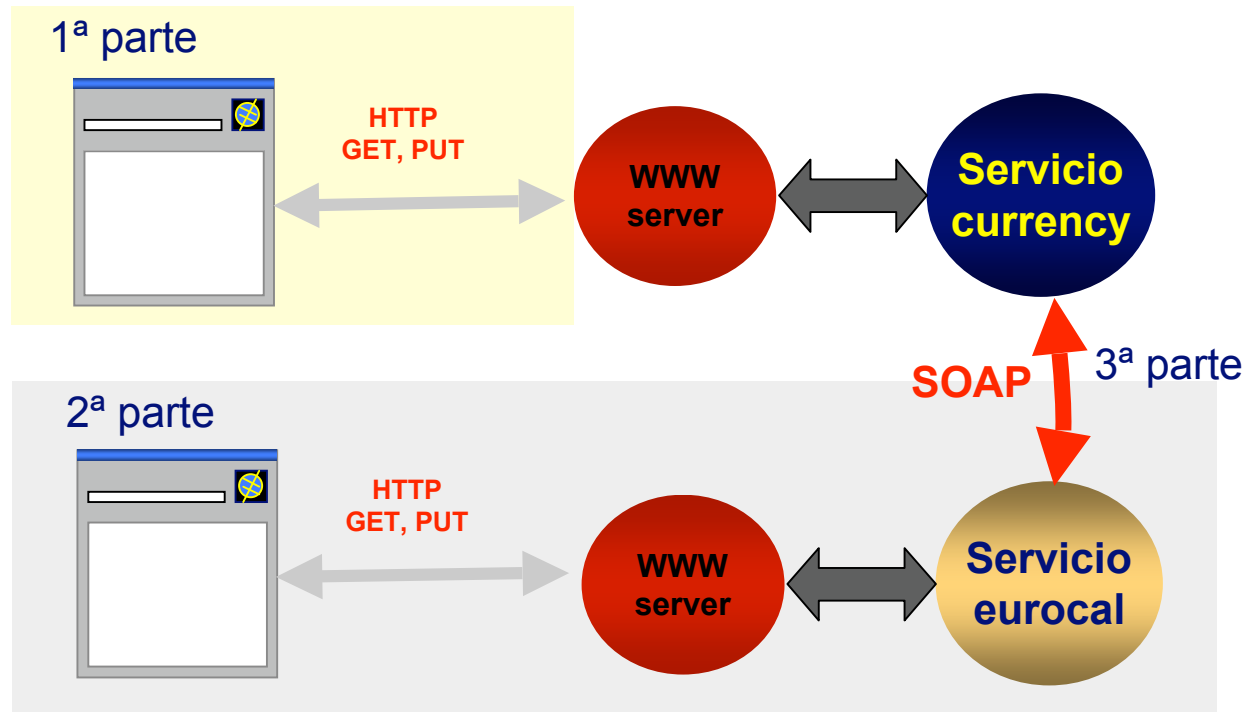


Soporte SOAP para Java

- Invocación estática
 - Desarrollo **Bottom Up**: a partir de una clase Java se genera el servicio web y la especificación WSDL.
 - Desarrollo **Top Down**: a partir de la especificación WSDL se generan las clases Java:
 - Stub del cliente
 - Skeleton del servidor

Soporte SOAP para Java

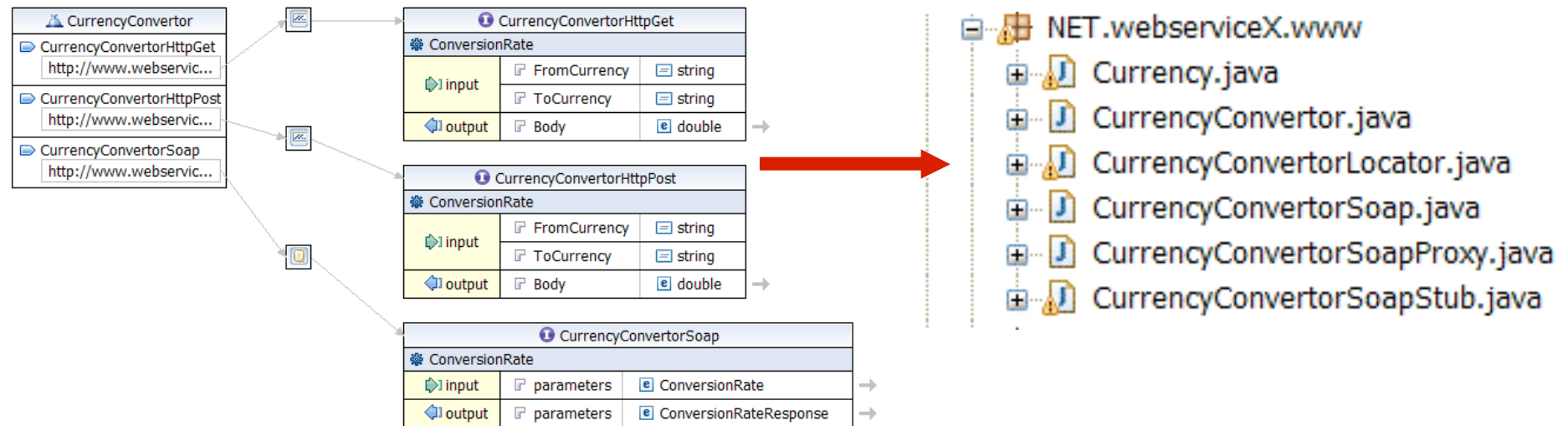
- Invocación estática: un ejemplo





Soporte SOAP para Java

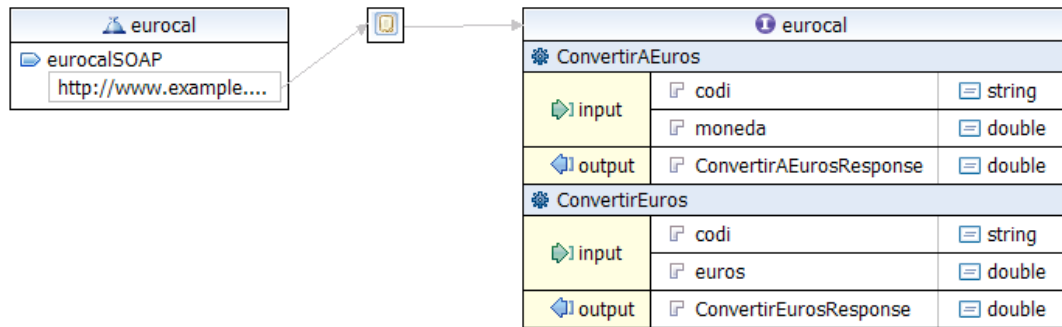
- Invocación estática: un ejemplo
 - Generación del stub del cliente





Soporte SOAP para Java

- Invocación estática: un ejemplo
 - Generación del esqueleto del servidor



org.example.www.eurocal

- + Eurocal_PortType.java
- + Eurocal_Service.java
- + Eurocal_ServiceLocator.java
- + EurocalSOAPImpl.java
- + EurocalSOAPSkeleton.java
- + EurocalSOAPStub.java





Soporte SOAP para Java

- Invocación estática: un ejemplo

```
import javax.xml.rpc.ServiceException;
```

```
import NET.webserviceX.www.Currency;
```

```
import NET.webserviceX.www.CurrencyConverterLocator;
```

```
import NET.webserviceX.www.CurrencyConverterSoap;
```

```
public class EurocalSOAPImpl implements org.example.www.eurocal.Eurocal_PortType{
```

```
    CurrencyConverterLocator service;
```

```
    CurrencyConverterSoap stub;
```

```
public EurocalSOAPImpl() {
```

```
    service = new CurrencyConverterLocator();
```

```
    try {
```

```
        stub = service.getCurrencyConverterSoap();
```

```
    } catch (ServiceException e) {
```

```
        e.printStackTrace();
```

```
    }
```

```
}
```



Soporte SOAP para Java

- Invocación estática: un ejemplo

```
public class EurocalSOAPImpl implements org.example.www.eurocal.Eurocal_PortType{

...

    public double convertirAEuros(java.lang.String codi, double moneda) throws java.rmi.RemoteException {
        double rate = stub.conversionRate(Currency.fromString(codi), Currency.EUR);
        return (moneda*rate);
    }

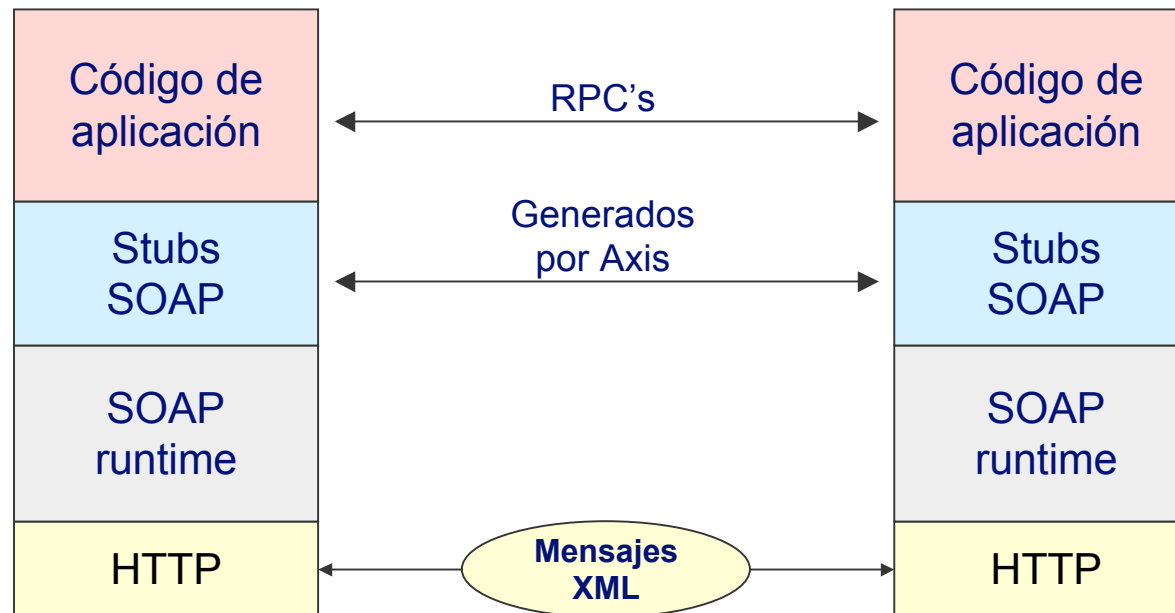
    public double convertirEuros(java.lang.String codi, double euros) throws java.rmi.RemoteException {
        double rate = stub.conversionRate(Currency.EUR, Currency.fromString(codi));
        return (euros*rate);
    }

}
```



Soporte SOAP para Java


- Soporte SOAP para Java: montaje estático
 - Estructura





Servicios web

● Índice

- Conceptos básicos
- SOAP
 - Estructura de SOAP
 - SOAP y WSDL
 - SOAP UDDI
- Soporte SOAP para Java
 - Interfaz dinámico de invocaciones
 - Interfaz estático de invocaciones (Axis)
-  - Soporte SOAP para PHP
- Un ejemplo en SOAP
 - La eurocalculadora en SOAP / Java
- XML-RPC



Soporte SOAP para PHP

- Soporte SOAP para PHP
 - Incluido a partir de PHP5
 - Obtención de referencias a servidores
 - `$client = new SoapClient("http://unhost.upv.es/php/eurocalc.wsdl");`
 - Invocación remota estática
 - `$qdst = $client->ConvertirAEuros($_POST["euros"], $_POST["codi"]);`
 - Creación de un servidor
 - `$server = new SoapServer("eurocalc.wsdl");`
 - `$server->setClass("ServeiEuroCalculadora");`



Servicios web

● Índice

- Conceptos básicos
- SOAP
 - Estructura de SOAP
 - SOAP y WSDL
 - SOAP UDDI
- Soporte SOAP para Java
 - Interfaz dinámico de invocaciones
 - Interfaz estático de invocaciones (Axis)
- Soporte SOAP para PHP
- Un ejemplo en SOAP
 - La eurocalculadora en SOAP / Java
- XML-RPC





La Eurocalculadora en SOAP

- El formulario HTML

The screenshot shows a web browser window titled "Eurocalculadora - Mozilla". The address bar displays "http://futura.dicea.upv.es/~vrd/ivla-soap/eurocalc.html". The page content includes:

Benvingut a l'exemple de l'Eurocalculadora

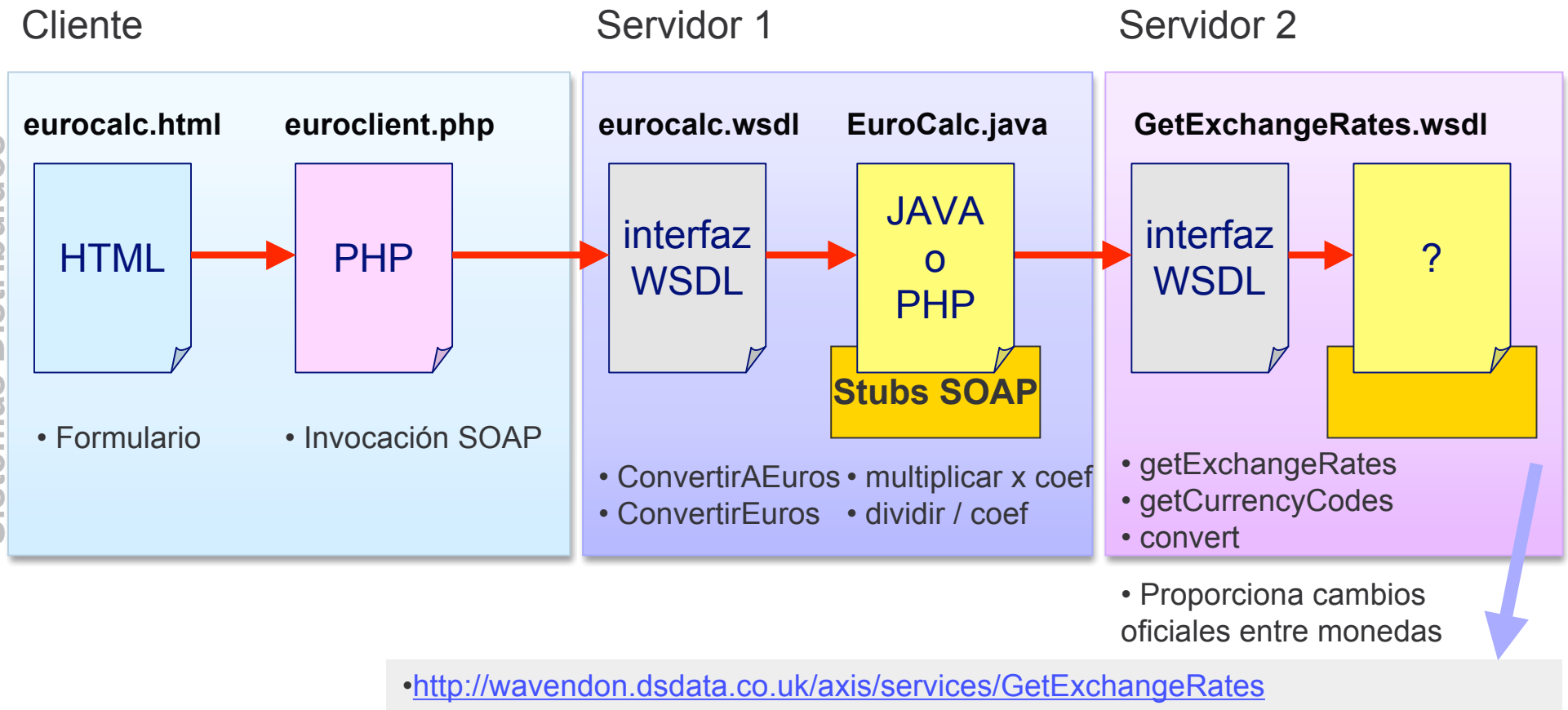
Conversió Euros -> moneda : introduïu la quantitat d'euros i la moneda objectiu a convertir en el següent formulari:

Conversió moneda -> Euros : introduïu la quantitat de moneda a convertir a euros en el següent formulari:

Consulta dels codis de moneda

La Eurocalculadora en SOAP

● Estructura de la aplicación





La Eurocalculadora en SOAP

● El cliente

```
<?php
...
cabeza("Eurocalculadora");
echo "<h1> Eurocalculadora </h1>";
$client = new SoapClient("http://unhost.upv.es/php/eurocalc.wsdl");
...
switch ( $_POST["conv"] ) {
    case 1:
        ...
        $qdst = $client->ConvertirAEuros( $_POST["euros"], $_POST["codi"]);
        ...
        break;
    case 2:
        ...
        $qdst = $client->ConvertirEuros( $_POST["moneda"], $_POST["codi"], );
        ...
        break;
}
...
?>
```

Obtener referencia al Servidor 1

Invocación SOAP

Invocación SOAP



La Eurocalculadora en SOAP

- El servidor1 en PHP (i)

```
<?php
```

```
class ServeiEuroCalculadora {
```

```
function ConvertirEuros($euros, $codi) {
```

```
    $codeclient=new SoapClient(
```

```
        "http://wavendon.dsdata.co.uk/axis/services/GetExchangeRates?wsdl );
```

```
    $rate=$codeclient->getExchangeRate("EUR",$codi);
```

```
    return $euros*$rate;
```

```
}
```

```
function ConvertirAEuros($moneda,$codi) {
```

```
    $codeclient=new SoapClient(
```

```
        "http://wavendon.dsdata.co.uk/axis/services/GetExchangeRates?wsdl");
```

```
    $rate=$codeclient->getExchangeRate($codi,"EUR");
```

```
    return $moneda*$rate;
```

```
}
```

```
}
```

Obtener referencia al Servidor 2

Invocación SOAP

Obtener referencia al Servidor 2

Invocación SOAP



La Eurocalculadora en SOAP

- El servidor1 en PHP (ii)

```
<?php
```

```
...
```

```
$server = new SoapServer("eurocalc.wsdl");
```

```
$server->setClass("ServeiEuroCalculadora");
```

```
if ( $_SERVER["REQUEST_METHOD"] == "POST") {
```

```
    $server->handle();
```

```
} else {
```

```
    echo "This SOAP server can handle following functions: ";
```

```
    $functions = $server->getFunctions();
```

```
    foreach($functions as $func) {
```

```
        echo $func . "\n";
```

```
    }
```

```
}
```

```
?>
```

Crear Servidor



La Eurocalculadora en SOAP

- El servidor2 en Java (i)

```
import uk.co.dsdata.ws.finance.forex.exchangeService.*;
```

```
public class EuroCalc {
```

```
    private GetExchangeRatesService exchangeRatesService;
```

```
    private GetExchangeRates exchangeRates;
```

```
    public float ConvertirEuros(float euros, String codi) throws java.rmi.RemoteException {
```

```
        double rate = exchangeRates.getExchangeRate("EUR", codi);
```

```
        return (float) (euros * rate);
```

```
    }
```

```
    public float ConvertirAEuros(float moneda, String codi) throws java.rmi.RemoteException {
```

```
        double rate = exchangeRates.getExchangeRate(codi, "EUR");
```

```
        return (float) (moneda * rate);
```

```
    }
```



La Eurocalculadora en SOAP

- El servidor2 en Java (ii)

```
public class EuroCalc {
```

```
...
```

```
public boolean CodiValid(String codi) throws java.rmi.RemoteException {
```

```
    CurrencyData[] array = exchangeRates.getCurrencyCodes();
```

```
    boolean existeix = false;
```

```
    for (int i = 0; i < array.length; i++) {
```

```
        if (array[i].getCode().matches(codi)) { existeix = true; break;}
```

```
    }
```

```
    return existeix;
```

```
}
```

```
public EuroCalc() throws javax.xml.rpc.ServiceException {
```

```
    exchangeRatesService = new GetExchangeRatesServiceLocator();
```

```
    exchangeRates = exchangeRatesService.getGetExchangeRates(),
```

```
}
```

```
};
```

Constructor

Inicializar referencia



Servicios web

● Índice

- Conceptos básicos
- SOAP
 - Estructura de SOAP
 - SOAP y WSDL
 - SOAP UDDI
- Soporte SOAP para Java
 - Interfaz dinámico de invocaciones
 - Interfaz estático de invocaciones (Axis)
- Soporte SOAP para PHP
- Un ejemplo en SOAP
 - La eurocalculadora en SOAP / Java
- XML-RPC





La Eurocalculadora en XML-RPC

- La eurocalculadora como servicio web en XML-RPC
 - Se va a adecuar el código de la eurocalculadora en PHP para poderla utilizar como servicio web en XML-RPC
 - XML-RPC como servidor se incorpora a PHP a través de una herramienta GPL “XML-RPC Class Server”
 - <http://www.webkreator.com/download/xcs-1.2.zip>



La Eurocalculadora en XML-RPC

```
<?php
...
// ----- Servicios Web -----
Class Test {
    // Función de pruebas, que nos permite ver el código XML Intercambiado
    function Test($secret) {
        if ($secret != '42') {
            trigger_error('Secret does not match.', E_USER_ERROR);
        }
    }
    function pts2euro($quant) {
        escribelP();
        return $qorg /167;
    }
    function euro2pts($quant) {
        escribelP();
        return $qorg *167;
    }
}
```



XML-RPC

- Invocación de un servicio web en XML-RPC
 - Para invocarlo habría que enviar, en principio, una solicitud XML y la respuesta es una página web cuyo código es XML.
 - La función Test permite ejecutar el Servicio Web, sin tener que enviar la solicitud en XML sino simplemente, una solicitud GET corriente:
 - http://www.midominio.com/eurocal.php/test/42/_method=pts2euro&quant=3546

- Respuesta del servicio web

```
<xml version='1.0' encoding="iso-8859-1" ?>
<methodResponse>
<params>
  <param>
    <value><string>21,234</string></value>
  </param>
</params>
</methodResponse>
```



XML-RPC

- Invocación de un servicio web desde otros servicios
 - Se trata de una RPC pues se pueden invocar rutinas PHP **remotas** desde un servidor escrito en PHP. Esta es la potencia de este esquema.

- En PHP

```
<?
include_once("../class.remotetest.php");
RemoteClassRegistry::addURL('DEFAULT_URL','http://www.midominio.com/eurocal.php');
$test = new Test(42);
print $test->pts2euro('3546');
?>
```

- En Perl

```
#!/vol/perl-5.6/bin/perl
Use RPC::XML::Client;
my $client = RPC::XML::Client->new('http://www.midominio.com/eurocal.php');
my $response = $client->send_request('Test.pts2euro', '3546');
print $response->value,"\n";
```



XML-RPC

- Petición XML de un servicio web
 - Las anteriores invocaciones en PHP o Perl generarían automáticamente una petición XML del siguiente estilo:

POST /ec

UserAgent: SupaDupa XML-RPC Client/0.1

Host: unhost.upv.es

Content-Type: text/xml

Content-Length: 201

```
<xml version='1.0' encoding="iso-8859-1" ?>
<methodCall>
  <methodName>Test.pts2euro</methodName>
  <params>
    <param>
      <value><string>3546</string></value>
    </param>
  </params>
</methodCall>
```




Enlaces

● Enlaces

- Web Services & Java home -
 - <http://java.sun.com/j2ee/webservices/index.html>
- Java Web Services tutorial -
 - <http://java.sun.com/xml/docs.html#tutorials>
- Apache Axis
 - <http://xml.apache.org/axis/index.html>
- SOAP
 - <http://www.w3.org/TR/SOAP>
- WSDL
 - <http://www.w3.org/TR/wsdl>
- JAX-RPC home -
 - <http://java.sun.com/xml/jaxrpc/index.html>
- Tutorials:
 - <http://www.sosnoski.com>



XML namespaces

- [Definition:]

- An XML namespace is a collection of names, identified by a URI reference [RFC2396], which are used in XML documents as element types and attribute names.

- `<termdef id="dt-dog" term="dog">`

- XML namespaces differ from the "namespaces" conventionally used in computing disciplines in that the XML version has internal structure and is not, mathematically speaking, a set. These issues are discussed in "A. The Internal Structure of XML Namespaces".



XML namespaces

- An example namespace declaration, which associates the namespace prefix edi with the namespace name `http://ecommerce.org/schema`:
 - `<x xmlns:edi='http://ecommerce.org/schema'>`
 - `<!-- the "edi" prefix is bound to http://ecommerce.org/schema`
 - `for the "x" element and contents -->`
 - `</x>`
- An example of a qualified name serving as an element type:
 - `<x xmlns:edi='http://ecommerce.org/schema'>`
 - `<!-- the 'price' element's namespace is http://ecommerce.org/schema -->`
 - `<edi:price units='Euro'>32.18</edi:price>`
 - `</x>`