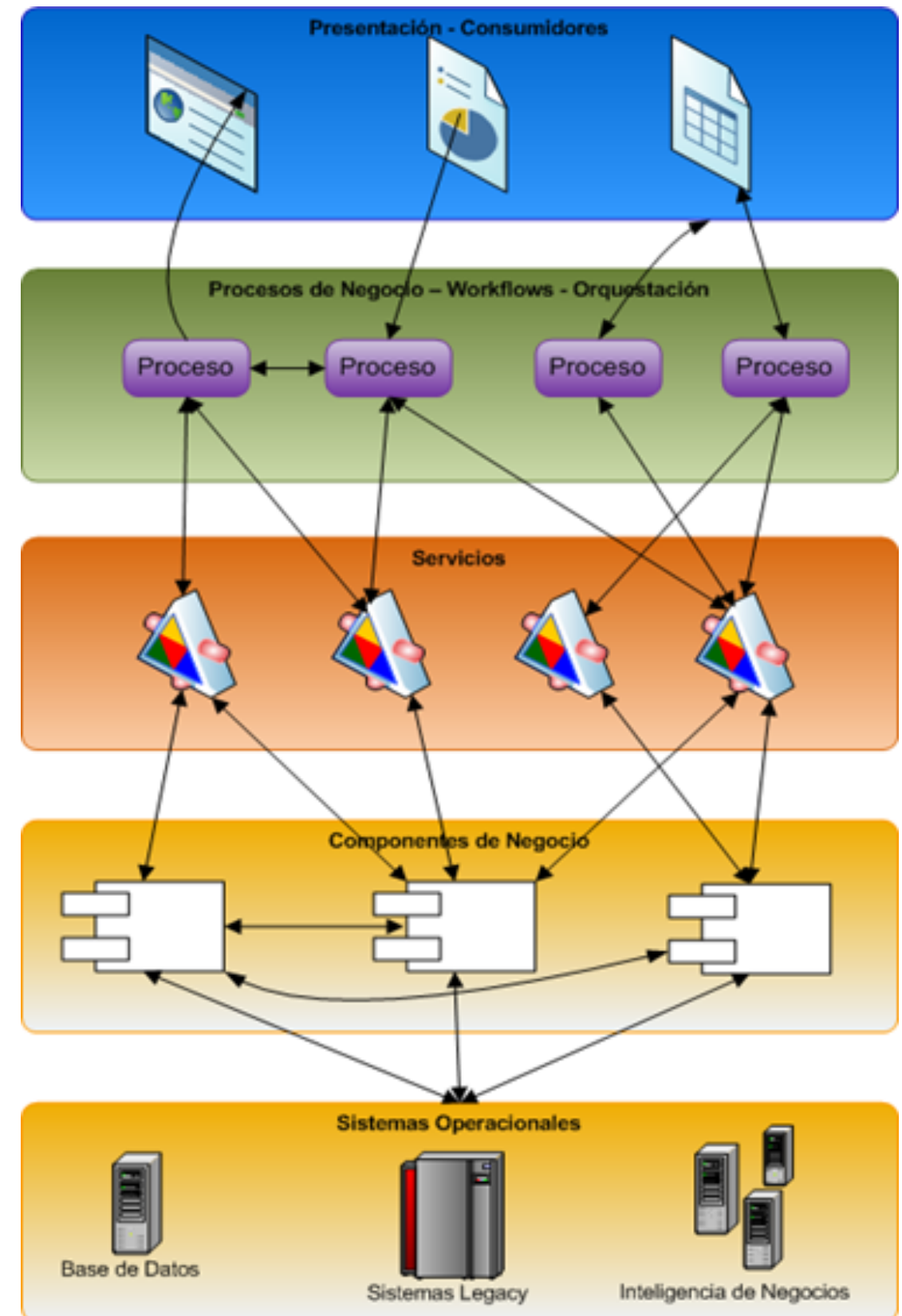
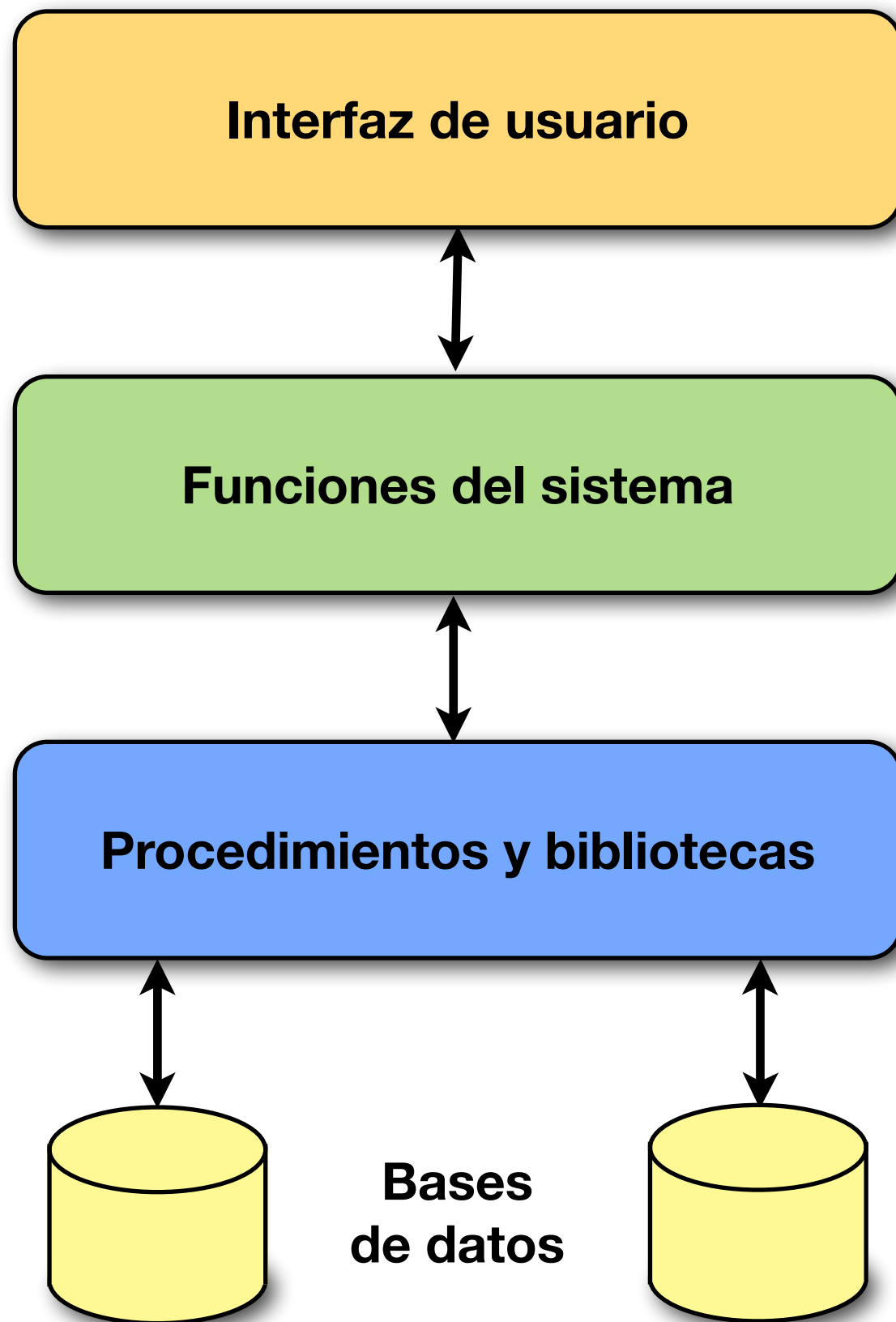


Service Oriented Architecture: Web Services

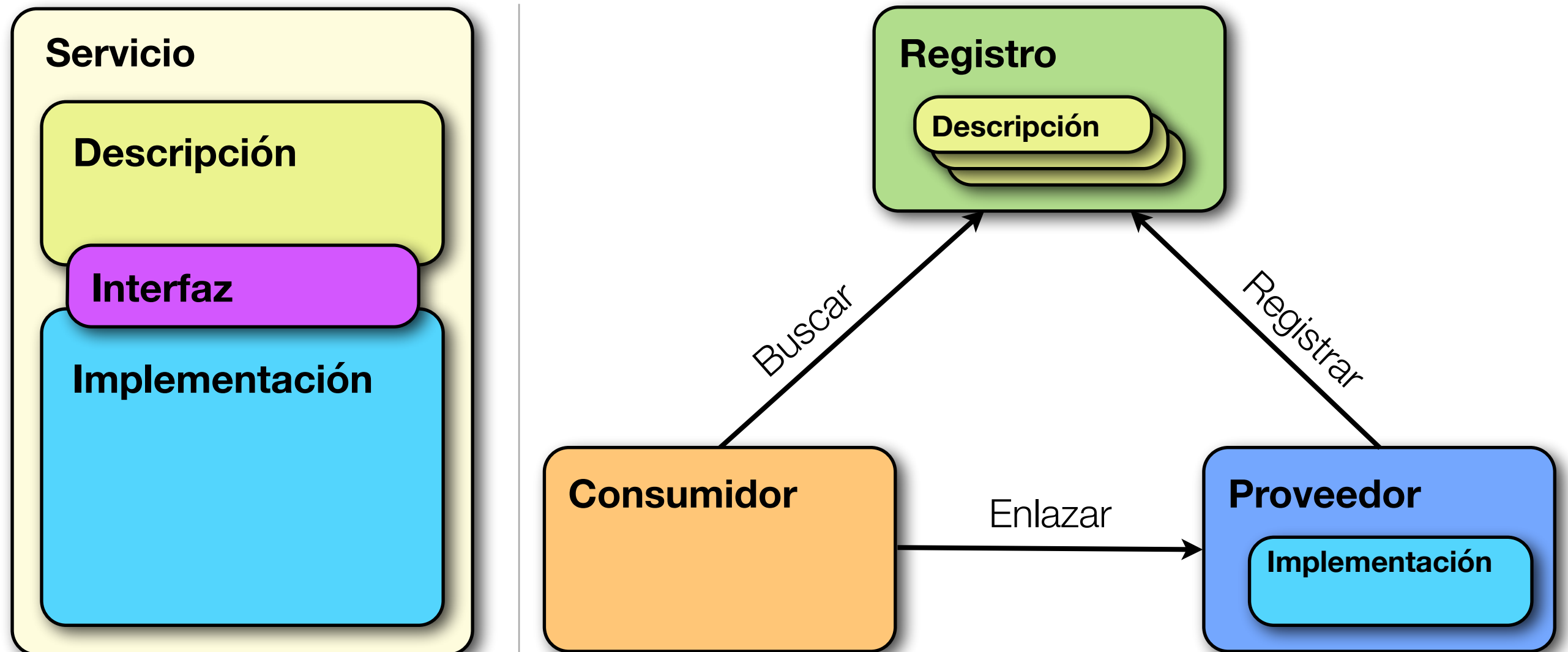
Programación Paralela y Distribuida

Pablo García Sánchez: Based on the work by Jesús González Peñalver (UGR)

Universidad de Cádiz



Aplicaciones clásicas vs. aplicaciones SOA





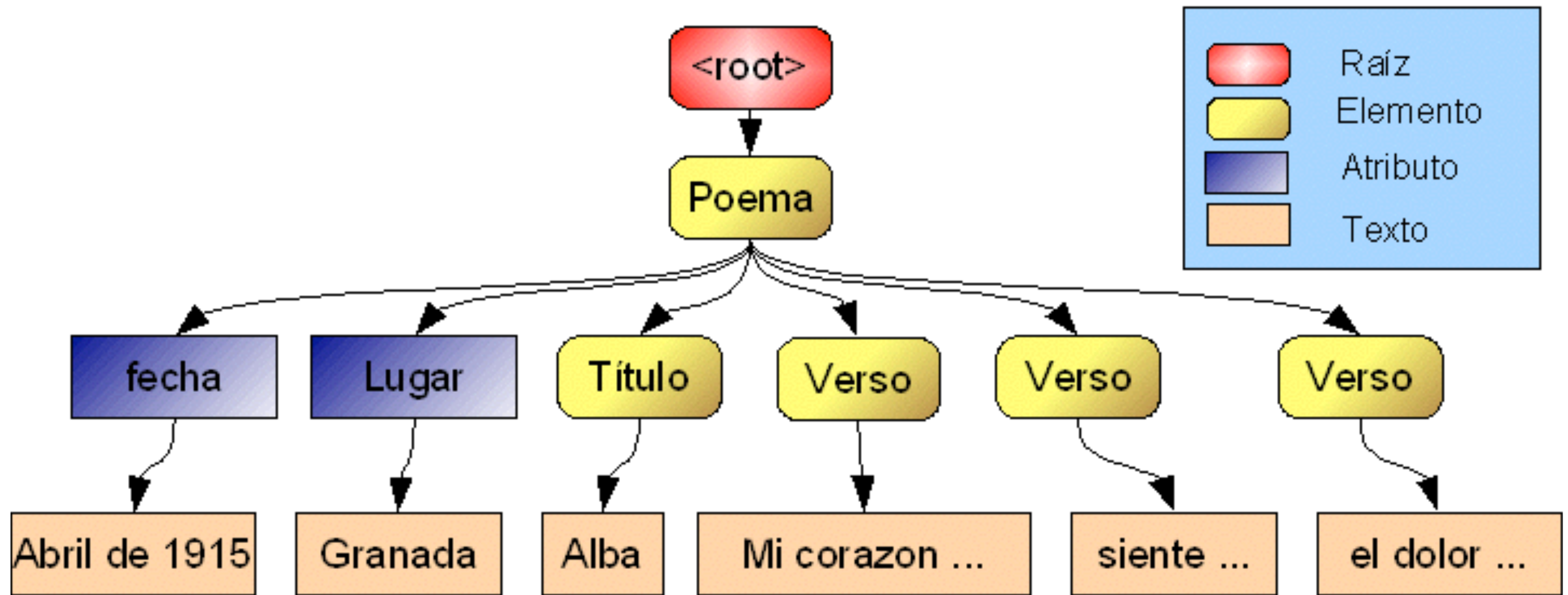
Entorno Java para el descubrimiento dinámico y el uso de servicios en red (desarrollado en los 90)



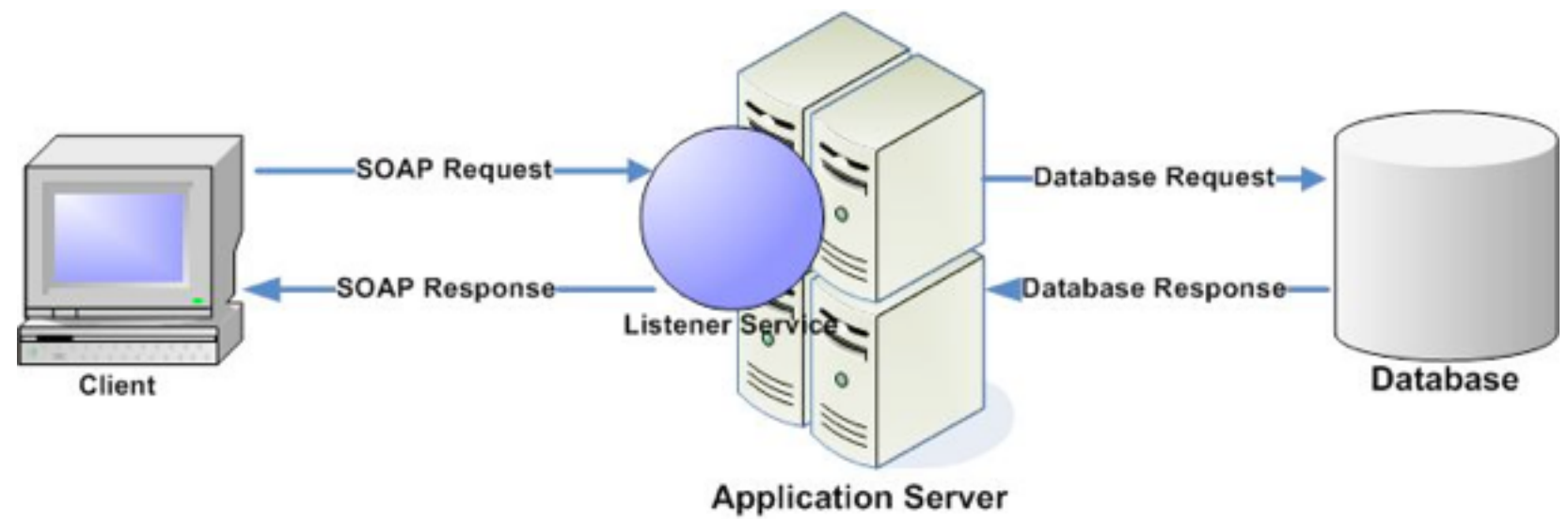
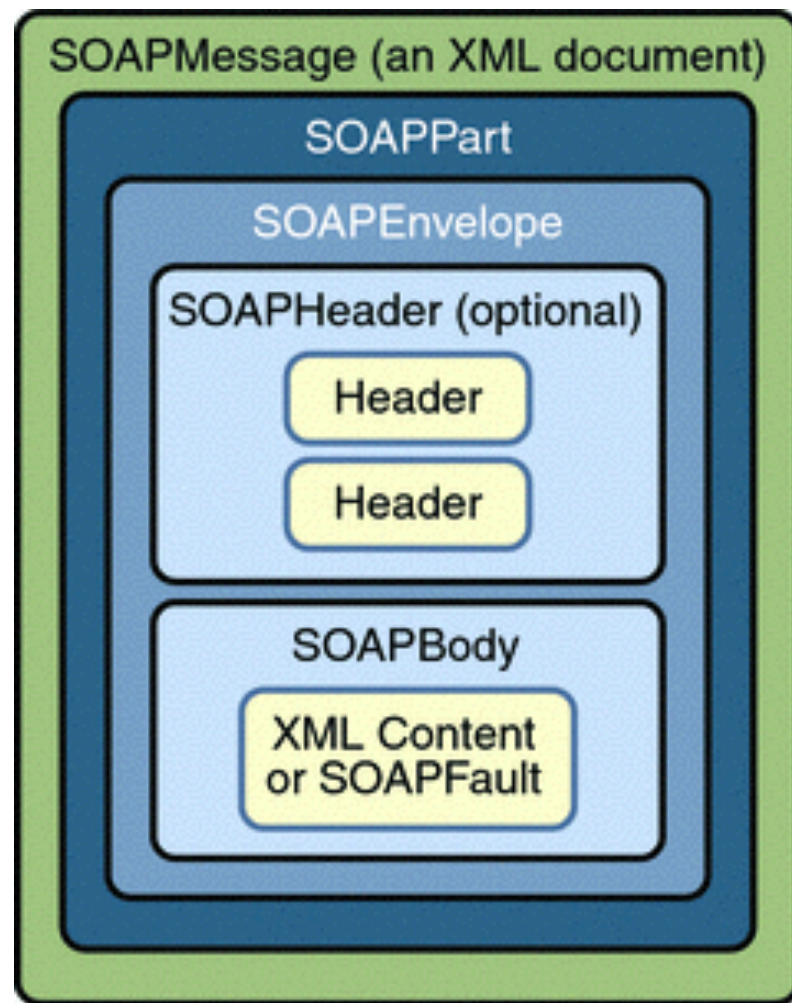
Sistema de módulos para Java que implementa un sistema dinámico de componentes



Sistemas de interacción máquina-máquina en red basados en estándares XML abiertos como WSDL, SOAP y UDDI



```
<poema fecha="Abril 1915" lugar="Granada">
  <titulo>Alba</titulo>
  <verso>Mi corazón oprimido</verso>
  <verso>siente junto a la alborada</verso>
  <verso>el dolor de sus amores...</verso>
</poema>
```



SOAP (Simple Object Access Protocol)

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

  <soap:Body>
    <getProductDetails
      xmlns="http://warehouse.example.com/ws">
      <productId>827635</productId>
    </getProductDetails>
  </soap:Body>

</soap:Envelope>
```

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

  <soap:Body>
    <getProductDetailsResponse
      xmlns="http://warehouse.example.com/ws">
      <getProductDetailsResult>
        <productName>Toptimate 3-Piece Set</productName>
        <productId>827635</productId>
        <description>3-Piece luggage set</description>
        <price>96.50</price>
        <inStock>true</inStock>
      </getProductDetailsResult>
    </getProductDetailsResponse>
  </soap:Body>

</soap:Envelope>
```

Ejemplo: Petición de información de un producto a un proveedor

`<definitions>`: Raíz del documento XML

`<types>`: Tipos de datos se intercambiarán

`<message>`: Formato de cada mensaje, tanto los parámetros como el resultado

`<portType>`: Operaciones (funciones) soportadas por el servicio

`<binding>`: Protocolo con el que se transmitirán los mensajes (SOAP, HTTP, MIME, etc.)

`<service>`: Dirección (URL) del servicio

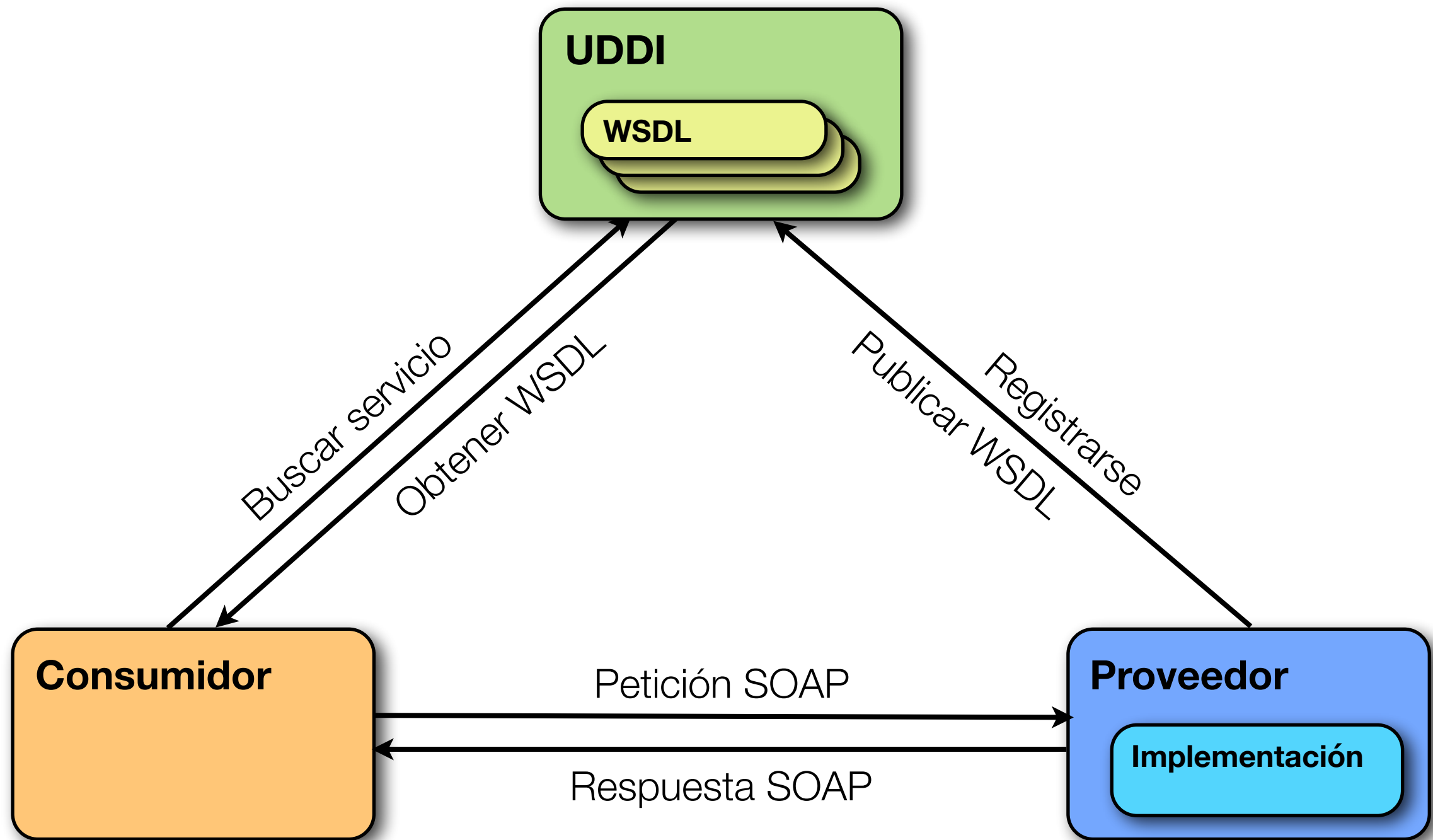

```
<?xml version="1.0" encoding="UTF-8"?>
<definitions name="HelloService"
  targetNamespace="http://www.ecerami.com/wsdl/HelloService.wsdl"
  xmlns="http://schemas.xmlsoap.org/wsdl/"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:tns="http://www.ecerami.com/wsdl/HelloService.wsdl"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
```

```
<message name="SayHelloRequest">
  <part name="firstName" type="xsd:string"/>
</message>
<message name="SayHelloResponse">
  <part name="greeting" type="xsd:string"/>
</message>
```

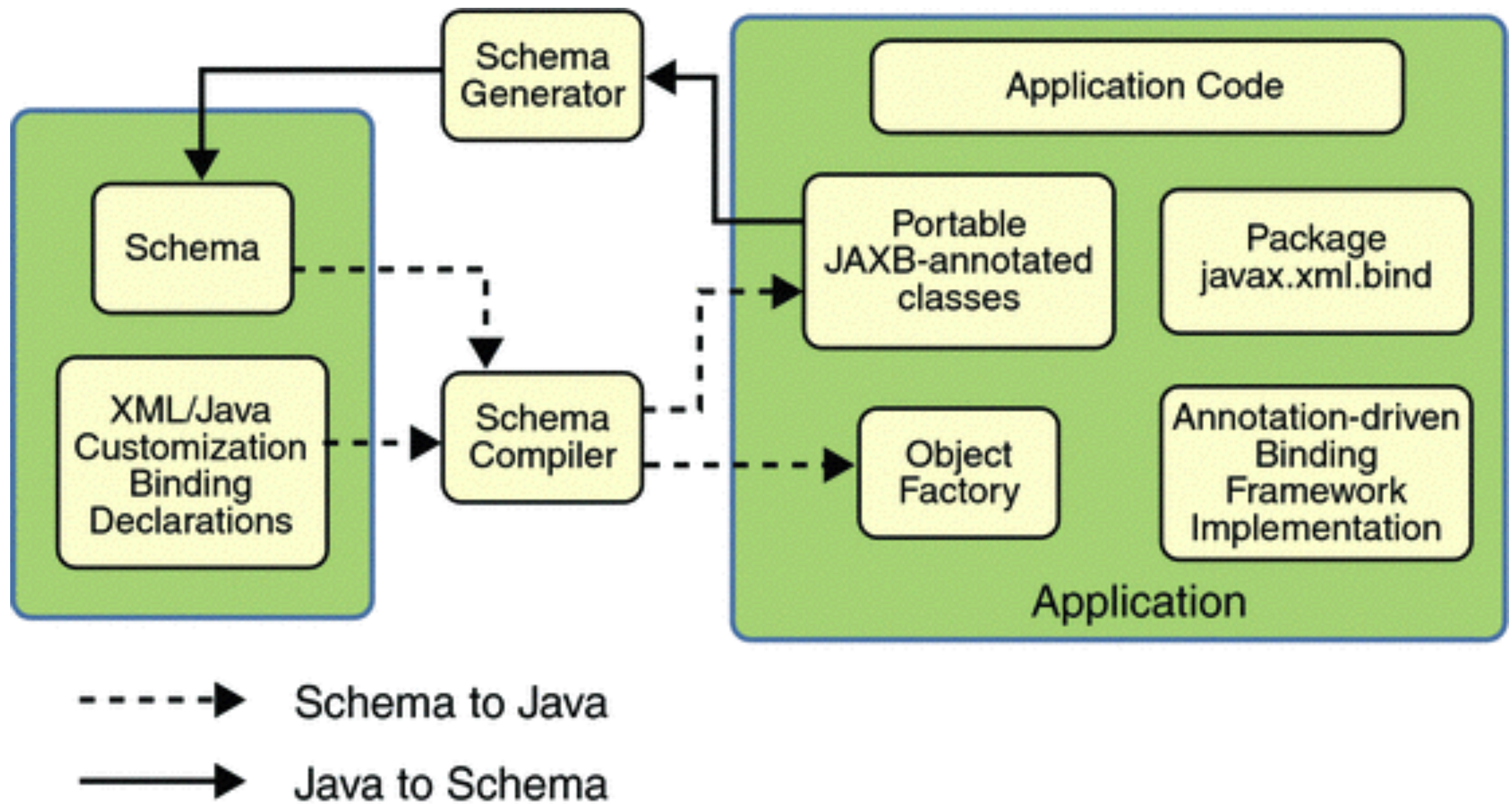
```
<portType name="Hello_PortType">
  <operation name="sayHello">
    <input message="tns:SayHelloRequest"/>
    <output message="tns:SayHelloResponse"/>
  </operation>
</portType>
```

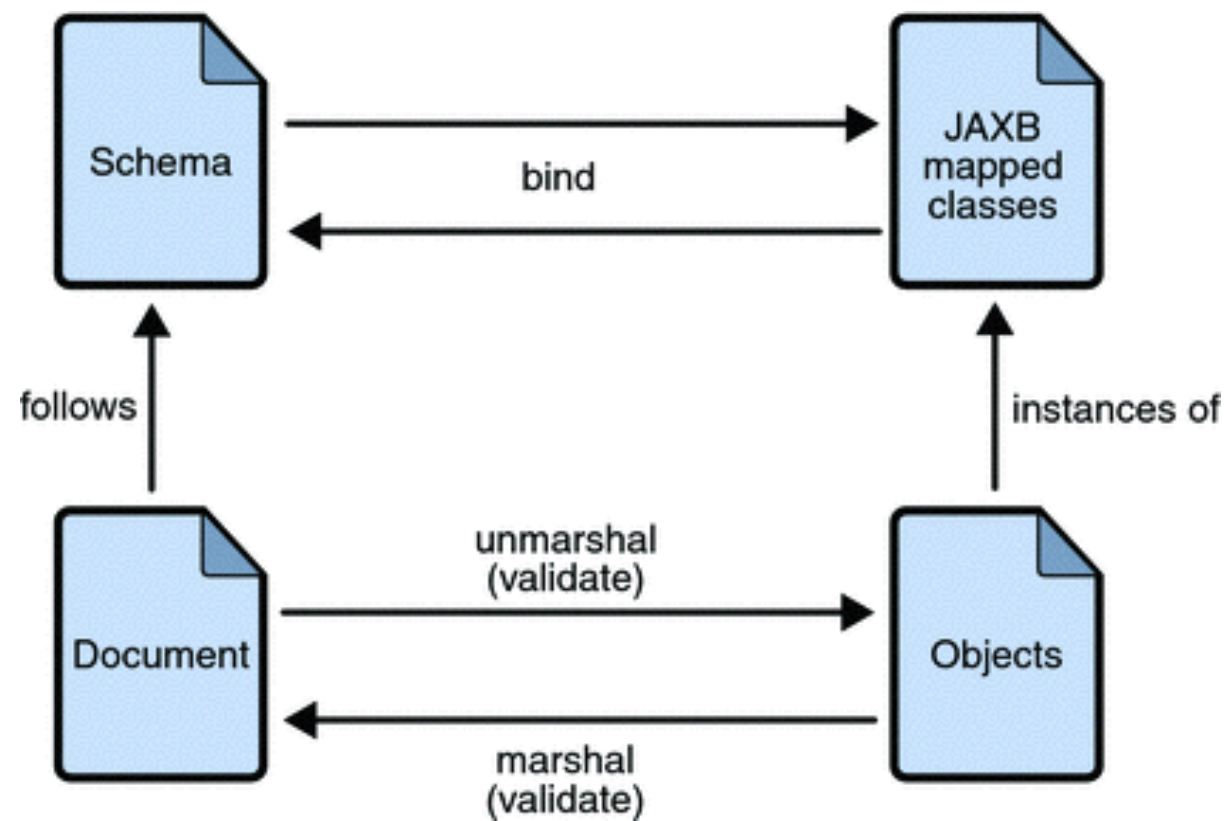
```
<binding name="Hello_Binding" type="tns:Hello_PortType">
  <soap:binding style="rpc"
    transport="http://schemas.xmlsoap.org/soap/http"/>
  <operation name="sayHello">
    <soap:operation soapAction="sayHello"/>
    <input>
      <soap:body
        encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
        namespace="urn:examples:helloservice"
        use="encoded"/>
    </input>
    <output>
      <soap:body
        encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
        namespace="urn:examples:helloservice"
        use="encoded"/>
    </output>
  </operation>
</binding>
```

```
<service name="Hello_Service">
  <documentation>WSDL for HelloService</documentation>
  <port binding="tns:Hello_Binding" name="Hello_Port">
    <soap:address
      location="http://localhost/rpcrouter"/>
  </port>
</service>
</definitions>
```

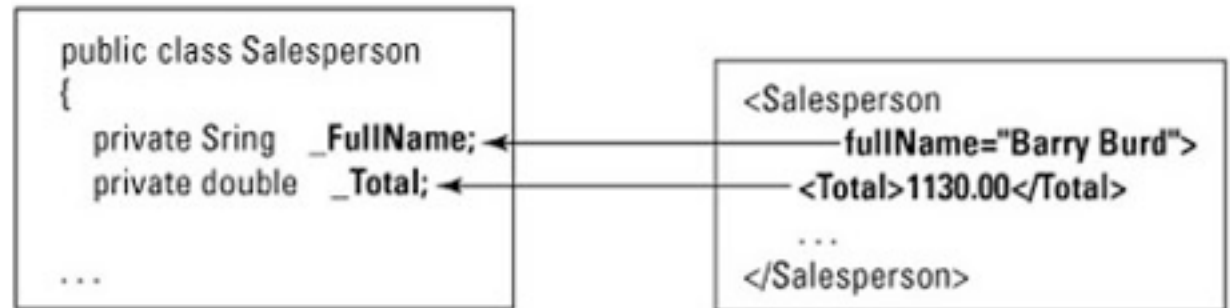


SOAP + WDSL + UDDI = Web Services

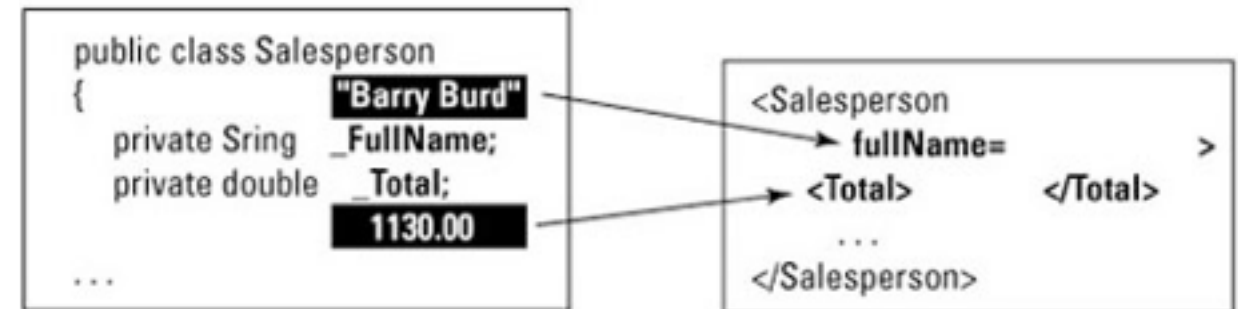




unmarshal

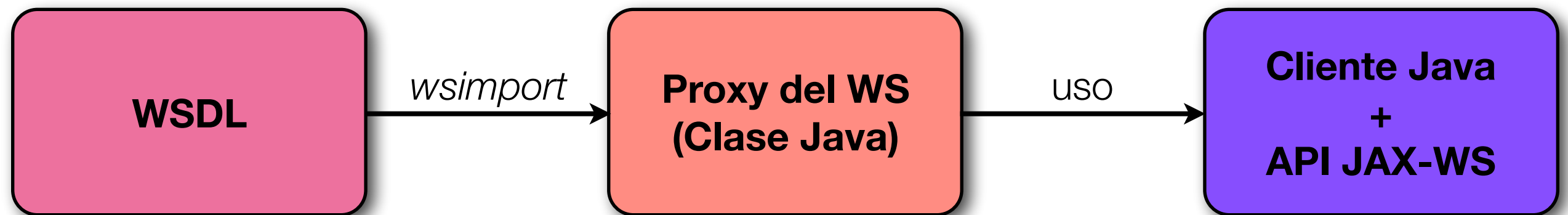


marshal





Creación de un servicio Web en Java



Creación de un cliente de un servicio Web en Java