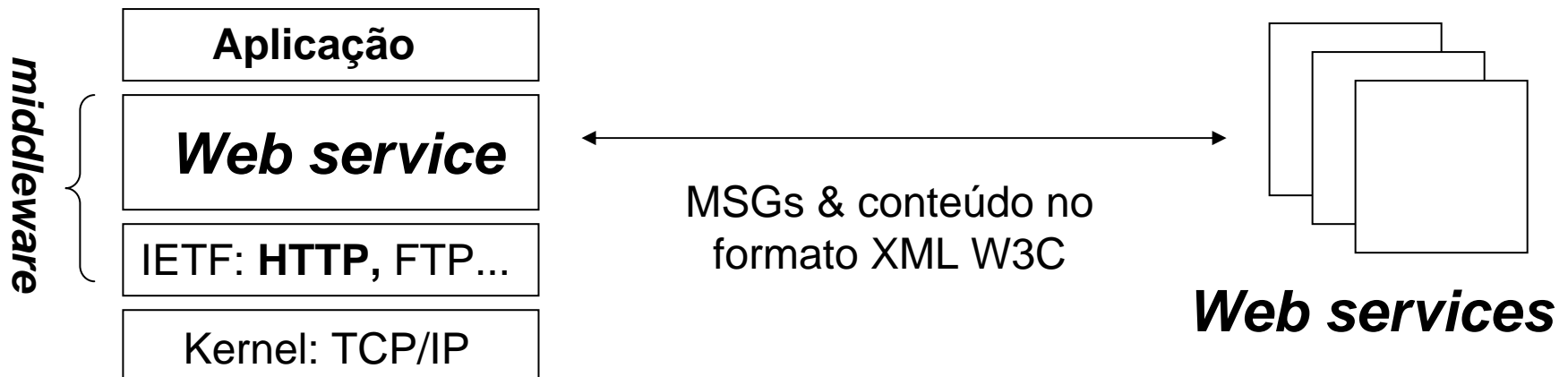


Serviços Web

O que é um *Web service*?

*A Web service is a **software system** designed to support **interoperable machine-to-machine interaction over a network**. It has an interface described in a machine-processable format (specifically **WSDL**). Other systems interact with the Web service in a manner prescribed by its description using **SOAP** messages, typically conveyed using **HTTP** with an **XML serialization** in conjunction with other Web-related standards (www.w3c.org).*



Service Oriented Architecture (SOA)

Um Serviço em SOA:

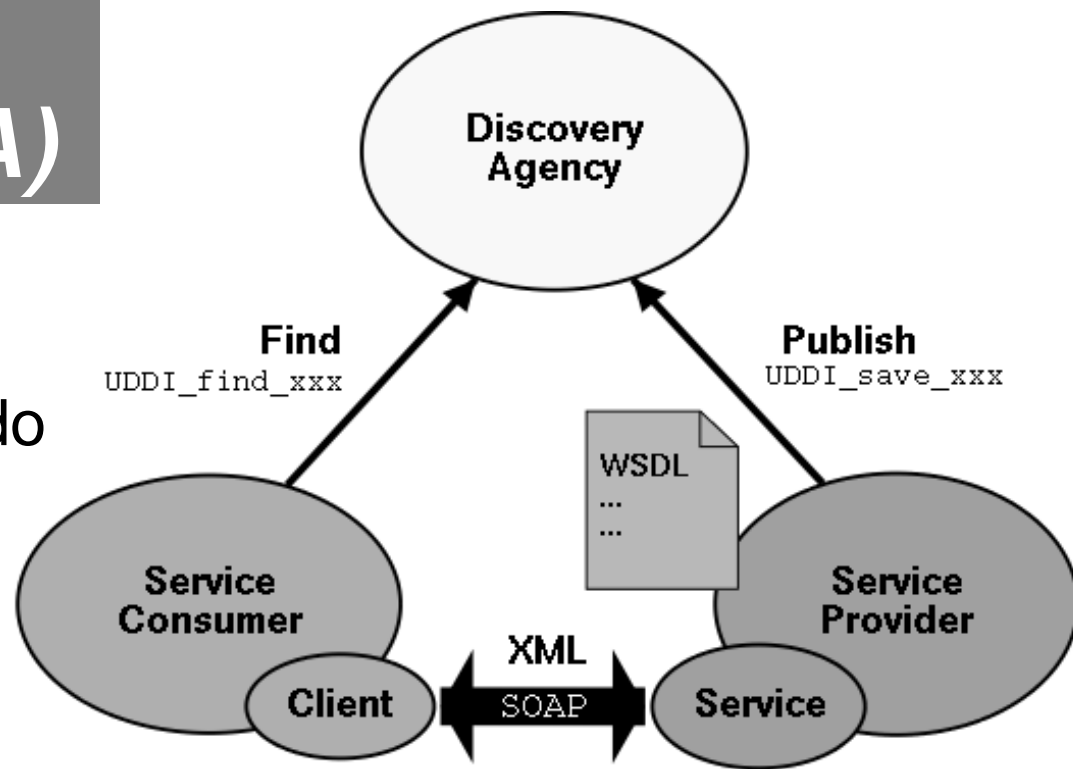
1. Um componente distribuído
(Transparência de localização)
2. Altocontido = modular e independente
3. Interoperável = independente da plataforma e **genérico**

↓ Padrões W3C

SOAP: XML para invocar *Web services*

WSDL: XML para descrever interfaces para *Web services*

UDDI: XML para manter listas de *Web services* disponíveis



Parênteses XML

HTML



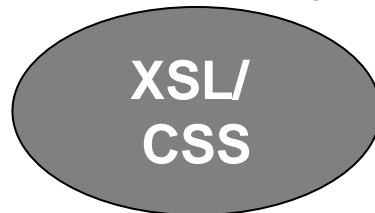
XML



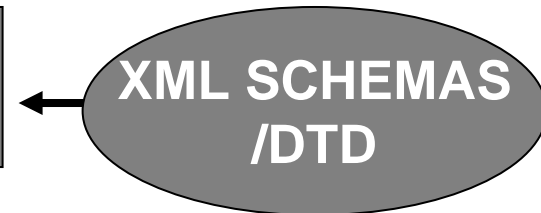
XML-RPC

```
<?xml version="1.0" >
<!DOCTYPE empregados SYSTEM "empregados.dtd">
<!-- Empregados da Empresa -->
<empregados>
  <empregado>
    <nome idade="26">Maria Aparecida</nome>
    <endereco>Rua Terra, 13</endereco>
  </empregado>
</empregados>
```

defines as regras

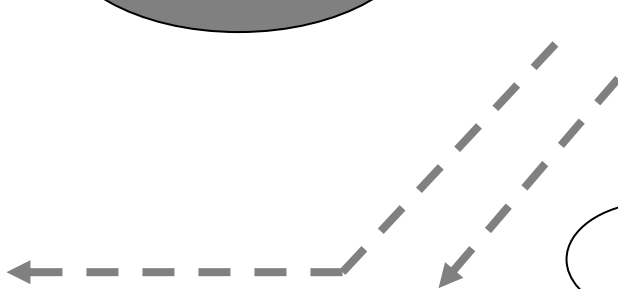


meta-dados



HTML

...ML



Sintaxe e Significado! Ah..

Protocolos



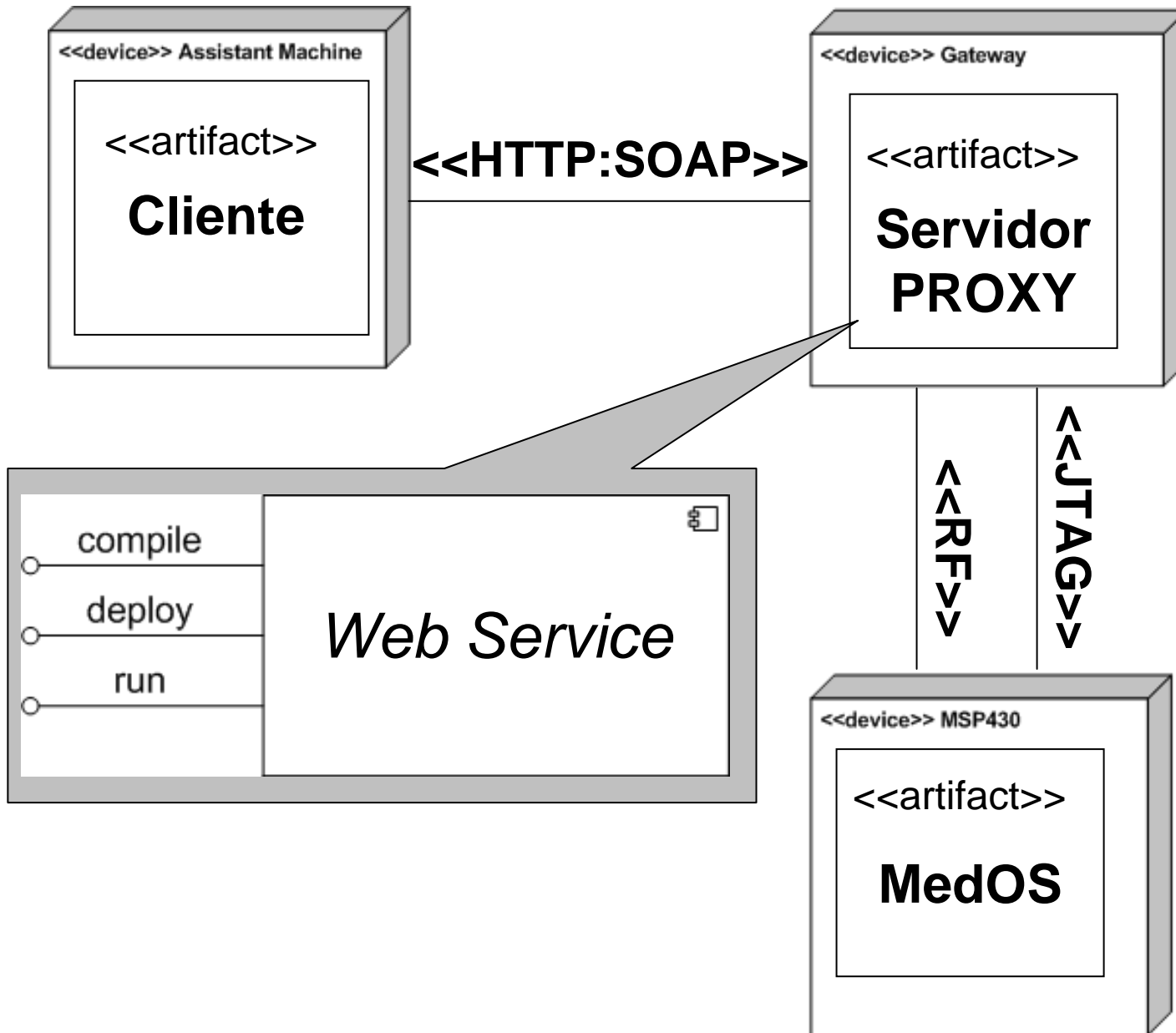
SOAP: *Simple Object Access Protocol*

- 1. Conversational Message Exchange**
- 2. Remote Procedure Call (RPC)**

```
POST /teste HTTP/1.1
Host: localhost
Content-Type: application/soap+xml; charset="utf-8"
Content-Length: nnnn
```

```
<?xml version='1.0' ?>
<env:Envelope xmlns:env="http://www.w3.org/2001/12/soap-envelope">
  <env:Header>
    ....
  </env:Header>
  <env:Body>
    <m: mensagem xmlns:m="http://localhost/teste"
      MENSAGEM ECHO
    </m: message>
  </env:Body>
</env:Envelope>
```

Exemplo



Exemplo

msg compile (ecgdata ecgxml)

errorCompile: erro ao compilar

okCompile: compilação bem sucedida

msg deploy (void)

errorDeploy: erro no *deployment*

okDeploy: *deployment* bem sucedido

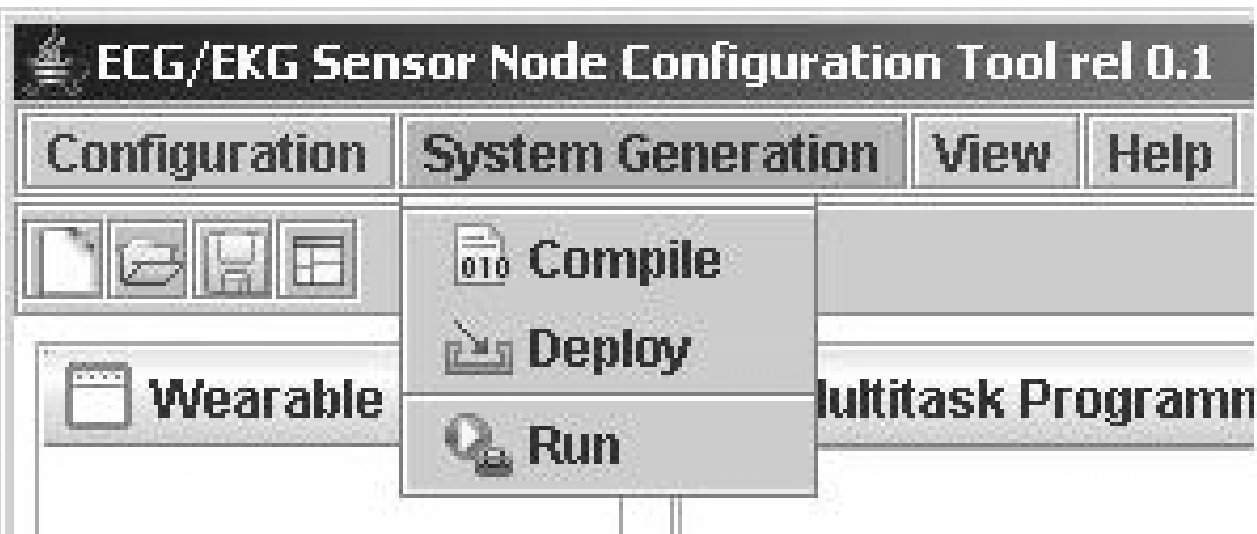
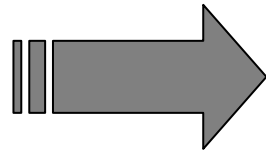
msg run (void)

errorRun: erro no *start* do microcontrolador

okRun: *start* bem sucedido

Web service:

- 1) Compile
- 2) Deploy
- 3) Run



```
<?xml version="1.0" encoding="utf-8" ?>
<description
  xmlns="http://www.w3.org/2006/01/wsdl"
  targetNamespace="http://localhost:8080/wsdl/ecg"
  xmlns:tns="http://localhost:8080/wsdl/ecg"
  xmlns:ghns="http://localhost:8080/wsdl/ecg"
  xmlns:wsoap="http://www.w3.org/2006/01/wsdl/soap"
  xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
  xmlns:wsdlx="http://www.w3.org/2006/01/wsdl-extensions">

  <documentation> This document describes .... </documentation>
```

cabeçalho

```
<types>
  <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
    targetNamespace="http://localhost:8080/wsdl/ecg"
    xmlns="http://localhost:8080/wsdl/ecg">
    <xs:element name="inoperativeError" type="xs:string"/>
    <xs:element name="errorCompile" type="xs:string"/>
    <xs:element name="okCompile" type="xs:string"/>
    <xs:element name="errorDeploy" type="xs:string"/>
    <xs:element name="okDeploy" type="xs:string"/>
    <xs:element name="errorRun" type="xs:string"/>
    <xs:element name="okRun" type="xs:string"/>
```

Declaração de tipos

```
<xs:complexType name="ecgdata ">
  <xs:sequence>
    <xs:element name=" BWSNET " type="xs:string"/>
    <xs:element name=" DSV" type="xs:string"/>
    <xs:element name=" ECG1" type="xs:string"/>
    <xs:element name=" ECG3" type="xs:string"/>
    <xs:element name=" ECG3I" type="xs:string"/>
    <xs:element name=" ECG3II" type="xs:string"/>
    <xs:element name=" ECG3III" type="xs:string"/>
    ....
  </xs:sequence>
<xs:element name="ecgxml" type="xs:ecgdata"/>
</xs:complexType>
</xs:schema>
</types>
```

**Declaração
de tipos**


```
<interface name = "staticECGInterface" >
```

```
  <fault name = "inoperativeFault" element = "ghns:inoperativeError"/>
```

```
    <operation name="compile"
```

```
      pattern="http://www.w3.org/2006/01/wsdl/in-out"
```

```
      style="http://www.w3.org/2006/01/wsdl/style/iri" wsdlx:safe = "true">
```

```
      <input messageLabel="In" element="ghns:ecgxml" />
```

```
      <output messageLabel="Out" element="ghns:okCompile" />
```

```
      <outfault ref="tns:errorCompile" messageLabel="Out"/>
```

```
    </operation>
```

```
    <operation name="deploy"
```

```
      pattern="http://www.w3.org/2006/01/wsdl/in-out"
```

```
      style="http://www.w3.org/2006/01/wsdl/style/iri" wsdlx:safe = "true">
```

```
      <output messageLabel="Out" element="ghns:okDeploy" />
```

```
      <outfault ref="tns:errorDeploy" messageLabel="Out"/>
```

```
    </operation>
```

```
    <operation name="run"
```

```
      pattern="http://www.w3.org/2006/01/wsdl/in-out"
```

```
      style="http://www.w3.org/2006/01/wsdl/style/iri" wsdlx:safe = "true">
```

```
      <output messageLabel="Out" element="ghns:okRun" />
```

```
      <outfault ref="tns:errorRun" messageLabel="Out"/>
```

```
    </operation>
```

```
</interface>
```

```
<binding name=" staticECGSOAPBinding "  
  interface="tns:staticECGInterface"  
  type="http://www.w3.org/2006/01/wsdl/soap"  
  wsoap:protocol="http://www.w3.org/2003/05/soap/bindings/HTTP">  
  <fault ref="tns: inoperativeFault" wsoap:code="soap:Sender"/>  
  <operation ref="tns:compile"  
  wsoap:mep="http://www.w3.org/2003/05/soap/mep/soap-response"/>  
  <operation ref="tns:deploy"  
  wsoap:mep="http://www.w3.org/2003/05/soap/mep/soap-response"/>  
  <operation ref="tns:run"  
  wsoap:mep="http://www.w3.org/2003/05/soap/mep/soap-response"/>  
</binding>
```

```
<service name="staticECGService" interface="tns:staticECGInterface">  
  <endpoint name="staticECGEndpoint" binding="tns:staticECGSOAPBinding"  
    address = " http://localhost:8080/ecg "/>  
</service>
```

```
</description>
```

...
<env:Body>

<**compile** env:encodingStyle="http://www.w3.org/2003/05/soap-encoding"
 xmlns: **compile** = " http://localhost:8080/ECG/">
 <**ecgdata** xmlns: **ecgdata** = " http://localhost:8080/ECG/"> ">

```
<?xml version="1.0" encoding="UTF-8" ?>
<java version="1.5.0_06" class="java.beans.XMLDecoder">
<object class="XMLECGData">
  <void property="BWSNET">
    <string>inactive</string>
  </void>
  <void property="DSV">
    <string>inactive</string>
  </void>
  <void property="ECG1">
    <string>100Hz</string>
  </void>
```

?

....

</ecgdata>
 </compile>
 </env:Body>

...

Requisição

<?xml version='1.0' ?>

<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope" >

<env:Header>

...

</env:Header>

<env:Body>

<compileResponse

env:encodingStyle="http://www.w3.org/2003/05/soap-encoding"

xmlns: **compileResponse** =" http://localhost:8080/ECG/">

<m:statuscompile xmlns: **statuscompile** =" http://localhost:8080/ECG/">

okCompile

</m:statuscompile>

</compileResponse>

</env:Body>

</env:Envelope>

Resposta

<?xml version='1.0' ?>

<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope"
xmlns:rpc='http://www.w3.org/2003/05/soap-rpc'>

<env:Body>

<env:Fault>

<env:Code>

<env:Value>env:Sender</env:Value>

</env:Code>

<env:Reason>

<env:Text xml:lang="en-US">**errorCompile**</env:Text>

</env:Reason>

</env:Fault>

</env:Body>

</env:Envelope>

Falha

Conclusões

- Na prática existem várias ferramentas e APIs para desenvolvimento de *Web services*
- *Web services* são interessantes?
 - Vantagens?
 - Desvantagens?
- Exercícios: