TECNOLOGIAS DE MIDDLEWARE

Introdução ao WSDL.
Concretização num Projecto de LBS.
INTRODUÇÃO AO WSDL:
- Introdução ao WSDL
- Descrição e Utilização

UMA CONCRETIZAÇÃO EM LBS:
- Um projecto para o GJU (ESA/CE)
- Interface Middleware - Serviços
- Especificação em WSDL

CONCLUSÕES:
- Desafios de Futuro
- .... partindo para o UDDI
WEB-SERVICES DESCRIPTION LANGUAGE

- Interface Specifications are XML documents describing Web-Services.
- These descriptions address Interfaces and Bindings.
- IDL is bound to specific middleware, WSDL is not, hence it must also describe access mechanisms.
- WSDL adopts a standard for registering services, since a common framework is not available (UDDI).
- Diverse interaction paradigms: asynchronous, synchronous, request-response & notifications.
The types element describes all the data types used between the client and server. WSDL is not tied exclusively to a specific typing system, but it uses the W3C XML Schema specification as its default choice. If the service uses only XML Schema built-in simple types, such as strings and integers, the types element is not required.

The message element describes a one-way message, whether it is a single message request or a single message response. It defines the name of the message and contains zero or more message part elements, which can refer to message parameters or message return values.

The portType element combines multiple message elements to form a complete one-way or round-trip operation. For example, a portType can combine one request and one response message into a single request/response operation, most commonly used in SOAP services. Note that a portType can (and frequently does) define multiple operations.

The binding element describes the concrete specifics of how the service will be implemented on the wire. WSDL includes built-in extensions for defining SOAP services, and SOAP-specific information therefore goes here.

The service element defines the address for invoking the specified service. Most commonly, this includes a URL for invoking the SOAP service.
WSDL – WEB SERVICES DESCRIPTION LANGUAGE

Types (type information for the document, e.g., XML Schema)

Message 1  Message 2  Message 3  Message 4  Message 5  Message 6

Operation 1  Operation 2  Operation 3

Interface (abstract service)

binding 1  binding 2  binding 3  binding 4

endpoint 1  endpoint 2  endpoint 3  endpoint 4

Service (the interface in all its available implementations)
The **types** element encloses data type definitions that are relevant for the exchanged messages. For maximum interoperability and platform neutrality, WSDL prefers the use of XSDs as canonical types.

```xml
<s:complexType name="PVT">
    <s:sequence>
        <s:element name="OBUId" type="s:string" />
        <s:element name="Latitude" type="s:double" />
        <s:element name="Longiude" type="s:double" />
        <s:element name="Altitude" type="s:double" />
        <s:element name="Velocity_Latitude" type="s:double" />
        <s:element name="Velocity_Longitude" type="s:double" />
        <s:element name="INTEGRITYFLAG" type="s:boolean" />
        <s:element name="GNSS_FLAG" type="s:boolean" />
        <s:element name="Timestamp" type="s:string" />
    </s:sequence>
</s:complexType>
```

WSDL supports the XML Schemas specification (XSD) as its canonical type system. However, since it is unreasonable to expect a single type system grammar to be used to describe all message formats present and future, WSDL allows using other type definition languages.
WSDL specification

abstract part
- types
- messages
- operations
- port types

concrete part
- bindings
- services and ports

Messages consist of one or more logical parts. Each part is associated with a type from some type system using a message-typing attribute. The set of message-typing attributes is extensible. WSDL defines several such message-typing attributes:

- element. Refers to an XSD element.
- type. Refers to an XSD simple or complex Type.

Other message-typing attributes may be defined as long as they use a namespace different from that of WSDL.

```xml
<wsdl:message name="GetPVTsSoapIn">
  <wsdl:part name="parameters" element="tns:GetPVTs" />
</wsdl:message>

<wsdl:message name="GetPVTsSoapOut">
  <wsdl:part name="parameters" element="tns:GetPVTsResponse" />
</wsdl:message>
```
WSDL specification

abstract part
- types
- messages
- operations
- port types

concrete part
- bindings
- services and ports

One-way. The endpoint receives a message.
Request-response. The endpoint receives a message, and sends a correlated message.
Solicit-response. The endpoint sends a message, and receives a correlated message.
Notification. The endpoint sends a message.

```xml
<wsdl:operation name="GetPVTs">
  <wsdl:input
  message="tns:GetPVTsSoapIn"/>
  <wsdl:output
  message="tns:GetPVTsSoapOut"/>
</wsdl:operation>

<wsdl:operation name="NotifyClient">
  <wsdl:output
  message="tns:NotifyClient"/>
</wsdl:operation>
```
A port MUST NOT specify more than one address.

A port MUST NOT specify any binding information other than address information.
WSDL em LBS

WSDL specification

abstract part

- types
- messages
- operations
- port types

concrete part

- bindings
- services and ports

WSDL - BINDINGS

```xml
<wSDL:binding name="WebServiceSoap" type="tns:AdvantisWebService">
  <soap:binding transport="http://schemas.xmlsoap.org/soap/http" style="document" />
  <wSDL:operation name="GetPVTs">
    <soap:operation soapAction="http://localhost/AdvantisWebService/GetPVTs" style="document" />
  </wSDL:operation>
  <wSDL:operation name="Subscribe">
    <soap:operation soapAction="http://localhost/AdvantisWebService/Subcribe" style="document" />
  </wSDL:operation>
  <wSDL:operation name="UnSubscribe">
    <soap:operation soapAction="http://localhost/AdvantisWebService/UnSubscribe" style="document" />
  </wSDL:operation>
</wSDL:binding>

<wSDL:binding name="NotificationService.NET" type="tns:AdvantisWebService">
  <binding transport=".NET_Remoting" style="document" />
  - <wSDL:operation name="NotifyClient">
    <soap:operation soapAction="http://localhost/AdvantisWebService1/NotifyClient" style="document" />
  </wSDL:operation>
</wSDL:binding>
```
WSDL – SERVICES

WSDL specification

abstract part

- types
- messages
- operations
- port types

concrete part

- bindings
- services and ports

<wsdl:service name="AdvantisWebService">
  <documentation xmlns="http://schemas.xmlsoap.org/wsdl/" />
  <wsdl:port name="WebServiceSoap"
    binding="tns:AdvantisWebService">
    <soap:address
      location="http://localhost/AdvantisWebService/AdvantisWebService.asmx" />
  </wsdl:port>
  <wsdl:port name="NotificationService"
    binding="tns:NotificationService.NET">
    <soap:address
      location="http://localhost/AdvantisWebService/AdvantisWebService.dll" />
  </wsdl:port>
</wsdl:service>

If a service has several ports that share a port type, but employ different bindings or addresses, the ports are alternatives. Each port provides semantically equivalent behavior (within the transport and message format limitations imposed by each binding). This allows a consumer of a WSDL document to choose particular port(s) to communicate with based on some criteria (protocol, distance, etc).
WSDL provides a mechanism to define the interface to Web services in terms of messages exchanged with that Web service:

- it allows for several forms of interaction (single message, request-response)
- it allows for several bindings (several implementations of the same interface)

WSDL plays a similar role as Interface Definition Languages in conventional middleware platforms:

- describe a service
- can be used to automatically generate code to invoke the service
- can be used by the infrastructure to enforce well formed interactions

Like other IDLs, WSDL does not contain information about:

- semantics
- business protocols and conversations
1. publish the service description  
2. find  
3. interact

the abstraction and infrastructure provided by the registry are part of the external middleware

Company C (directory service provider)
ALGUNS DESAFIOS PARA O FUTURO DO WSDL:

- Web service messaging systems (on top of SOAP but providing all the features needed in a real messaging system, namely asynchronous message notifications!) **WS-EVENTING**

- The existing standard is not yet fully encompassed by any existing middleware technologies: J2EE or .NET. Some workarounds must be implemented in order to fully comply.

- Integration between WS and CORBA - danger of over-simplification. WS as middleware atop CORBA, and CORBA atop WS as in the SIP. Hence, CORBA&WS are not mutually exclusive but complementary: automatic mapping between CORBA-IDL and WSDL is needed.
WSDL em LBS

UM PROJECTO ESPECÍFICO EM LBS para o GJU (ESA/CE)...
...BASEADO NUM SISTEMA (muito) DISTRIBUÍDO
PROTOTYPE APPLICATIONS GUI

PSS
WSDL em LBS

PROTOTYPE CLIENTS

PCSS
SOAP/XML REQUESTS .NET PORT: 20

PSS
SOAP/XML REQUESTS .NET PORT: 30

CLIENT #1 IP: 192.168.A.B

NOTIFICATION SERVICE

GCLSI

ADVANTIS XML WEB-SERVICE

PCSS / PSS .NET REMOTING SERVICES

PCSS .NETPORT: 20

ALARM !!!

OBU#1 ENTERED CSA
OBU#2 LEFT CSA

PSS .NETPORT: 30

ALARM !!!

OBU#5 LEFT CSA
OBU#8 ENTERED CSA

ALARMING
WSDL specification

abstract part
- types
- messages
- operations
- port types

concrete part
- bindings
- services and ports

WSDL - STACK

WSDL em LBS
JUSTIFICAÇÃO TECNICA (WS):
- Firewall-transparent Traffic (port:80)
- Definição standard de operações diversas
- Integrabilidade com aplicações 3rd-party

JUSTIFICAÇÃO COMERCIAL (.NET/ArcGIS):
- GUIs GIS evoluídos (PCSS-2D ; PSS-3D)
- OLE em Office. (alguém usa o StarOffice?)
- SkySoft é parceiro da MS ; ESRI.

NOTA: se alguém comenta isto eu sou despedido e tenho de ir fazer data-warehousing para a NovaBase ou SysAdmin para um banco ou, pior ainda, bilhética para a Transtejo.
PLANEAMENTO RUP – MODELO ITERATIVO

ADVANTIS PROTOTYPE APPLICATIONS DEVELOPMENT & SYSTEM VALIDATION SKYSOFT’s - ROADMAP

1st September | OCTOBER  | 29th NOVEMBER | ADR T+10 | DECEMBER  | 15th JANUARY | CDR-2w | FEBRUARY  | 28th MARCH | APRIL  | 26th MAY | SAR | JUNE | JULY | ORR

**PHASE B1**
- **WSDL to UDDI**
- **PLANEAMENTO RUP – MODELO ITERATIVO**
- **ADVANTIS PROTOTYPE APPLICATIONS DEVELOPMENT & SYSTEM VALIDATION SKYSOFT’s - ROADMAP**

**INHOUSE RELEASE**
- **PCSS/PSS**
- **WebGIS**

**INTRANET SKY**
- **PCSS/PSS**
- **INTRANET**

**PHASE B2**
- **FUNCTIONAL UNIT ACCEPTANCE REVIEW**
- **SUBSYSTEM ACCEPTANCE REVIEW**

**RELEASE VERSION**
- **PROJECT RELEASE**

**LEGEND**
- **MILESTONES**
- **VALIDATION REPORTS**
- **PROTOTYPE RELEASE**
- **PMP**
- **WPs & DOCS**

18/11/04
E para quem pensa que sabe algo acerca disto......

XML SPECIFICATIONS AND STANDARDS
SOLUÇÃO TECNICA:
- Interoperabilidade/Flexibilidade
- Notificações e Alarmes (.Net Remoting)
- Data-Providing;Subscription (SOAP/HTTP)

PROJECTO:
- Complexidade (HW/MW/SW/Nav/Com)
- Agility: RUP - Iterative Development.

NOTA: Não fiquei suficientemente aborrecido com esta apresentação, e quero saber mais:
www.galileo-advantis.com
Muito Obrigado.

FIM de APRESENTAÇÃO

ângulo inverso:

andre.barbosa@skysoft.pt