

Web Services

UDDI e Outras normas

Tecnologias de Middleware 2004/2005

Rui Lopes

18 de Novembro de 2004

Revisão

UDDI

Motivação

O que é?

Registo

Estruturas de dados

tModel

API

Pesquisa

Armazenamento

Ligação dinâmica

Coordenação

Outras normas

Endereçamento

Segurança

Políticas

Invocação

Conclusões

Referências

Revisão

UDDI

Motivação

O que é?

Registo

Estruturas de dados

tModel

API

Pesquisa

Armazenamento

Ligação dinâmica

Coordenação

Outras normas

Endereçamento

Segurança

Políticas

Invocação

Conclusões

Referências

- ▶ Conjunto de especificações baseadas em XML e outros standards abertos
- ▶ Potencia a interoperabilidade entre várias aplicações
- ▶ Permite que serviços sejam expostos na Web
- ▶ Ao princípio, apenas para integração ponto-a-ponto em aplicações heterogéneas
- ▶ Melhor aceite que RPC, CORBA e DCOM devido à simplificação dos protocolos (> 90 implementações)

Revisão

UDDI

Motivação

O que é?

Registo

Estruturas de dados

tModel

API

Pesquisa

Armazenamento

Ligação dinâmica

Coordenação

Outras normas

Endereçamento

Segurança

Políticas

Invocação

Conclusões

Referências

- ▶ Simple Object Access Protocol
- ▶ Protocolo de comunicação para troca de informação
- ▶ Funcionamento sobre HTTP (não normativo, mas quase ubíquo)
- ▶ Divide-se em 3 partes:
 - ▶ Envelope de mensagens
 - ▶ Regras de instanciação de dados
 - ▶ Representação de RPCs e respostas

Revisão

UDDI

Motivação

O que é?

Registo

Estruturas de dados

tModel

API

Pesquisa

Armazenamento

Ligação dinâmica

Coordenação

Outras normas

Endereçamento

Segurança

Políticas

Invocação

Conclusões

Referências

- ▶ Web Services Description Language
- ▶ Interface que descreve conjunto de mensagens SOAP e como estas são trocadas
- ▶ Equivalente ao IDL do CORBA
- ▶ Baseado em XML Schema
- ▶ Neutralidade para com linguagens de programação e plataformas
- ▶ Maior transparência pois WSDL e stubs gerados a partir directamente do código

- ▶ Gerir reutilização e manutenção de código:
 - ▶ Partilha de interfaces de serviços
 - ▶ Partilha de instruções de utilização/invocação
 - ▶ Notificação de alterações
 - ▶ Problemas de alteração de *backend versus frontend*
- ▶ Como publicitar disponibilidade de serviço?

UDDI - O que é?

- ▶ Universal Description, Discovery and Integration
- ▶ É um conjunto de APIs para efectuar:
 - ▶ Registo de descrições de serviços Web
 - ▶ Pesquisa de serviços Web
- ▶ UDDI é um serviço Web, logo ele próprio pode estar registado e ser acedido como outro qualquer
- ▶ WSDL e ligação SOAP disponível para uso por qualquer cliente
- ▶ Objectivos:
 - ▶ Ajudar programadores a utilizar serviços
 - ▶ Providenciar ligações dinâmicas

Revisão

UDDI

Motivação

O que é?

Registo

Estruturas de dados

tModel

API

Pesquisa

Armazenamento

Ligação dinâmica

Coordenação

Outras normas

Endereçamento

Segurança

Políticas

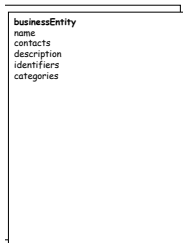
Invocação

Conclusões

Referências

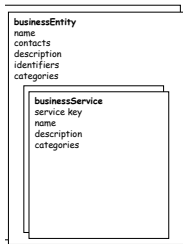
- ▶ UDDI define noção de *business registry*: serviço de nomes e directório
- ▶ *Universal Business Registry* e registos privados
- ▶ Categorização de registos:
 - ▶ *White Pages*: Listas de organizações, os seus contactos e os seus serviços
 - ▶ *Yellow Pages*: Classificações de empresas e serviços de acordo com taxonomias (pré-definidas ou não)
 - ▶ *Green Pages*: Informação de como invocar um serviço (apontadores para descritores fora do registo)

UDDI - Estruturas de dados



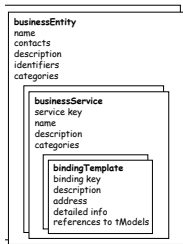
- ▶ *businessEntity*: Descrição da organização que disponibiliza o serviço

UDDI - Estruturas de dados



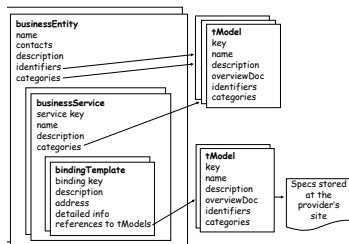
- ▶ *businessEntity*: Descrição da organização que disponibiliza o serviço
- ▶ *businessService*: Descrição de serviço e os seus *flavors*

UDDI - Estruturas de dados



- ▶ *businessEntity*: Descrição da organização que disponibiliza o serviço
- ▶ *businessService*: Descrição de serviço e os seus *flavors*
- ▶ *bindingTemplate*: Como usar um serviço específico (localização, instruções, capacidades, etc.)

UDDI - Estruturas de dados



- ▶ **businessEntity**: Descrição da organização que disponibiliza o serviço
- ▶ **businessService**: Descrição de serviço e os seus *flavors*
- ▶ **bindingTemplate**: Como usar um serviço específico (localização, instruções, capacidades, etc.)
- ▶ **tModel**: Encapsula uma especificação com metadados (WSDL, classificação, explicações textuais, etc.)

UDDI - Exemplo

```
<bindingTemplate bindingKey="uddi:..." serviceKey="uddi:...">
  <description>Hello World</description>
  <accessPoint useType="endpoint">
    http://www.exemplo.com/HelloWorld.aspx
  </accessPoint>
  <tModelInstanceDetails>
    <tModelInstanceInfo tModelKey="uddi:uddi.org:transport:http" />
    <tModelInstanceInfo tModelKey="uddi:xpto1234" />
  </tModelInstanceDetails>
</bindingTemplate>
```

- ▶ *Technical Model*
- ▶ Núcleo do UDDI, especifica uma informação chave de um serviço
- ▶ Podem (**e devem**) ser reutilizados por vários *bindingTemplate*
- ▶ Abre as portas à interoperabilidade de serviços
- ▶ Outros usos (devido à natureza de descrição de dados):
 - ▶ Definição de uso de protocolos de transporte
 - ▶ Sistemas de categorização estruturada
 - ▶ Posição geográfica
 - ▶ Endereços postais
 - ▶ ...

Revisão

UDDI

Motivação

O que é?

Registo

Estruturas de dados

tModel

API

Pesquisa

Armazenamento

Ligação dinâmica

Coordenação

Outras normas

Endereçamento

Segurança

Políticas

Invocação

Conclusões

Referências

/tModel	Cabeçalho de um <i>tModel</i>
/tModel/name	Nome do modelo
/tModel/description	Breve descrição do modelo
/tModel/overviewDoc	Referência para descrição detalhada de informação
/tModel/categoryBag	Conjunto de categorias que descrevem este modelo
categoryBag/keyedReference categoryBag/keyedReferenceGroup	Categoria específicas do modelo

Table: Descrição de um *tModel*

UDDI - Exemplo

```
<tModel tModelKey="uddi:xpto1234">
  <name>Hello World Descriptions</name>
  <description>...</description>
  <overviewDoc useType="wsdlSpec">
    <overviewURL>http://www.exemplo.com/HelloWorld.wsdl</overviewURL>
  </overviewDoc>
  <overviewDoc useType="text">
    <overviewURL>http://www.exemplo.com/HelloWorld.html</overviewURL>
  </overviewDoc>
  <categoryBag>
    <keyedReference tModelKey="uuid:C1ACF26D-9672-4404-9D70-39B756E62AB4"
      keyName="uddi-org:types"
      keyValue="wsdlSpec"/>
  </categoryBag>
</tModel>
```

Revisão

UDDI

Motivação

O que é?

Registo

Estruturas de dados

tModel

API

Pesquisa

Armazenamento

Ligação dinâmica

Coordenação

Outras normas

Endereçamento

Segurança

Políticas

Invocação

Conclusões

Referências

- ▶ *Inquiry API*: Pesquisas através de diversos critérios
- ▶ *Publishers API*: Adicionar/modificar/apagar registos de serviços
- ▶ *Security API*: Providencia comunicação segura com os serviços de registo
- ▶ *Custody and Ownership API*: Delegação de gestão de registos entre serviços UDDI
- ▶ *Subscription API*: Monitorização de alterações a registos
- ▶ *Replication API*: Replicação de informação entre serviços UDDI

Revisão

UDDI

Motivação

O que é?

Registo

Estruturas de dados

tModel

API

Pesquisa

Armazenamento

Ligação dinâmica

Coordenação

Outras normas

Endereçamento

Segurança

Políticas

Invocação

Conclusões

Referências

- ▶ *Inquiry API*: Pesquisas através de diversos critérios
- ▶ *Publishers API*: Adicionar/modificar/apagar registos de serviços
- ▶ *Security API*: Providencia comunicação segura com os serviços de registo
- ▶ *Custody and Ownership API*: Delegação de gestão de registos entre serviços UDDI
- ▶ *Subscription API*: Monitorização de alterações a registos
- ▶ *Replication API*: Replicação de informação entre serviços UDDI
- ▶ **Todas as operações funcionam sobre SOAP**

Revisão

UDDI

Motivação

O que é?

Registo

Estruturas de dados

tModel

API

Pesquisa

Armazenamento

Ligação dinâmica

Coordenação

Outras normas

Endereçamento

Segurança

Políticas

Invocação

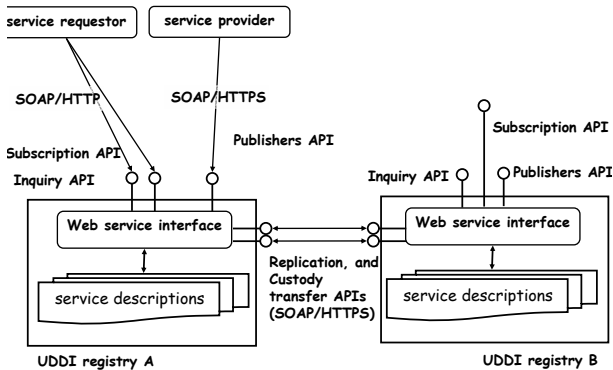
Conclusões

Referências

UDDI - API

Web Services
UDDI e Outras
normas

Rui Lopes



Revisão

UDDI

Motivação

O que é?

Registo

Estruturas de dados

tModel

API

Pesquisa

Armazenamento

Ligação dinâmica

Coordenação

Outras normas

Endereçamento

Segurança

Políticas

Invocação

Conclusões

Referências

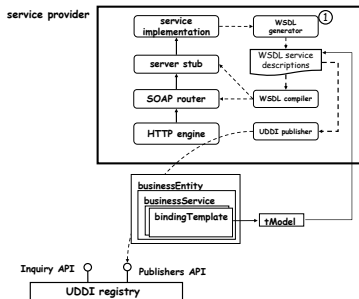
- ▶ Inquiry API
- ▶ Baseia-se na taxonomia dos *tModels*
- ▶ Disponibiliza apenas algumas directivas de pesquisa:
 - ▶ *find_business*
 - ▶ *find_service*
 - ▶ ...
 - ▶ *get_bindingDetail*
 - ▶ *get_tModelDetail*
 - ▶ ...

- ▶ Publishers API
- ▶ Principal uso do UDDI está no registo de WSDLs (como *tModels*)
- ▶ Algumas directivas:
 - ▶ *save_business*
 - ▶ *save_tModel*
 - ▶ ...
 - ▶ *delete_service*
 - ▶ *delete_binding*
 - ▶ ...

UDDI - Ligação dinâmica

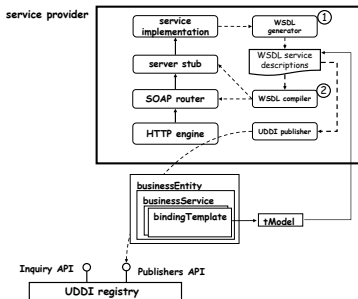
- ▶ Semelhante ao DII do CORBA
- ▶ Permite localizar interfaces em tempo de execução
- ▶ Pouco utilizado pois demasiado vago no uso numa aplicação
- ▶ Caso provável:
 - ▶ Procura de um serviço bem conhecido
 - ▶ Especificidades de classificações no uso de um serviço
 - ▶ Obter informação sobre serviços com mesma interface

Coordenação entre normas



1. Mapeamento de interface de serviço para WSDL

Coordenação entre normas



1. Mapeamento de interface de serviço para WSDL
2. Geração de stub para ligação a SOAP

Revisão

UDDI

- Motivação
- O que é?
- Registo
- Estruturas de dados
- tModel
- API
- Pesquisa
- Armazenamento
- Ligação dinâmica

Coordenação

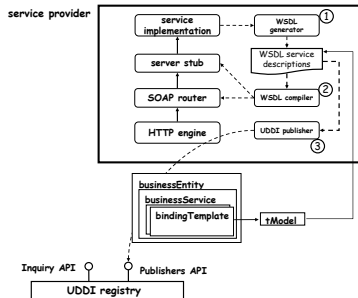
Outras normas

- Endereçamento
- Segurança
- Políticas
- Invocação

Conclusões

Referências

Coordenação entre normas



1. Mapeamento de interface de serviço para WSDL
2. Geração de stub para ligação a SOAP
3. Publicação num registo UDDI

Revisão

UDDI

- Motivação
- O que é?
- Registo
- Estruturas de dados
- tModel
- API
- Pesquisa
- Armazenamento
- Ligação dinâmica

Coordenação

Outras normas

- Endereçamento
- Segurança
- Políticas
- Invocação

Conclusões

Referências

Endereçamento - Motivação

- ▶ SOAP muitas vezes implementado sobre HTTP, mas não normativo
- ▶ Utilização extensiva do modelo browser-servidor do HTTP
- ▶ Informação de pontos envio e destino demasiado dependente
 - ▶ *Connection timeout*
 - ▶ *Forwarding* por intermediários (ex: firewalls)

- ▶ Mecanismo de integração de endereçamento nos serviços Web
- ▶ Independente do protocolo de transporte utilizado
- ▶ Não afectação por intermediários no processo de transporte
- ▶ Possibilita comunicação quer assíncrona, quer por padrões de longa duração
- ▶ Permite indirectão de encaminhamento de respostas (ao contrário do HTTP)
- ▶ Endereçamento com granularidade mais fina do que com HTTP (*EndPoints*)

- ▶ *EndPoint*
- ▶ Conjunto de *tags* que especificam dinamicamente um serviço Web
- ▶ Complementar especificação WSDL:
 - ▶ Geração e customização dinâmica de descrições
 - ▶ Flexibilizar *coupling* de serviços (através de políticas)
- ▶ Médio prazo substituir `service` do WSDL

/wsa:Address	URI identificador
/wsa:ReferenceProperties /wsa:ReferenceParameters	Conjunto de propriedades que descrevem o <i>EndPoint</i>
/wsa:PortType	<i>PortType</i> principal do <i>EndPoint</i>
/wsa:ServiceName	Identificador do wsdl:service correspondente no WSDL
/wsp:Policy	Política do <i>EndPoint</i> (WS-Policy)

Table: Descrição de um *EndPoint*

Endereçamento - Exemplo

```
<wsa:EndpointReference ...>  
  <wsa:Address> http://www.fabrikam123.example/acct </wsa:Address>  
  <wsa:PortType> fabrikam:InventoryPortType </wsa:PortType>  
</wsa:EndpointReference>
```

Revisão

UDDI

Motivação

O que é?

Registo

Estruturas de dados

tModel

API

Pesquisa

Armazenamento

Ligação dinâmica

Coordenação

Outras normas

Endereçamento

Segurança

Políticas

Invocação

Conclusões

Referências

/wsa:MessageID	URI identificador da mensagem
/wsa:RelatesTo	Relação com mensagem anterior
/wsa:To	URI de envio da mensagem
/wsa:Action	Semântica implícita da mensagem (recomendado URI de PortType de WSDL)
/wsa:ReplyTo	<i>EndPoint</i> para onde enviar resposta
/wsa:From	<i>EndPoint</i> de onde veio a mensagem
/wsa:FaultTo	<i>EndPoint</i> para onde enviar respostas de erro

Table: Cabeçalho de informação de mensagens

Endereçamento - Exemplo

```
<S:Envelope ...>
  <S:Header>
    <wsa:MessageID>
      uuid:6B29FC40-CA47-1067-B31D-00DD010662DA
    </wsa:MessageID>
    <wsa:ReplyTo>
      <wsa:Address>http://business456.example/client1</wsa:Address>
    </wsa:ReplyTo>
    <wsa:To>http://fabrikam123.example/Purchasing</wsa:To>
    <wsa:Action>http://fabrikam123.example/SubmitPO</wsa:Action>
  </S:Header>
  <S:Body>
    ...
  </S:Body>
</S:Envelope>
```

Revisão

UDDI

Motivação

O que é?

Registo

Estruturas de dados

tModel

API

Pesquisa

Armazenamento

Ligação dinâmica

Coordenação

Outras normas

Endereçamento

Segurança

Políticas

Invocação

Conclusões

Referências

- ▶ Web Services muito baseados em HTTP, apesar de não normativo
- ▶ Outros protocolos de transporte: SMTP, Message Queues, etc.
- ▶ Alguns não suportam níveis de segurança exigidos (ex. SMTP)
- ▶ Necessidade de adicionar segurança de um modo *standard*

- ▶ Web Services Security Language
- ▶ Fornecer um mecanismo de suporte à construção de protocolos SOAP com segurança
- ▶ Suportar toda variedade de modelos de segurança
- ▶ Requisitos:
 - ▶ Múltiplas tecnologias de autenticação
 - ▶ Múltiplos domínios de confiança
 - ▶ Múltiplas tecnologias de cifra
 - ▶ Segurança no nível de mensagens e não no nível de transporte
- ▶ Qualidade de protecção:
 - ▶ Prevenir leitura/modificação por antagonistas
 - ▶ Prevenir envio de mensagens não autorizadas

Revisão

UDDI

Motivação

O que é?

Registo

Estruturas de dados

tModel

API

Pesquisa

Armazenamento

Ligação dinâmica

Coordenação

Outras normas

Endereçamento

Segurança

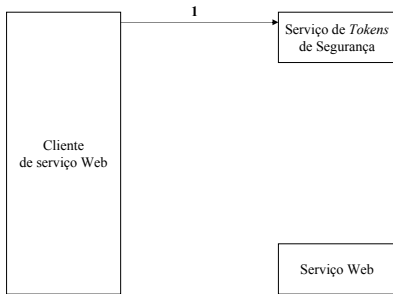
Políticas

Invocação

Conclusões

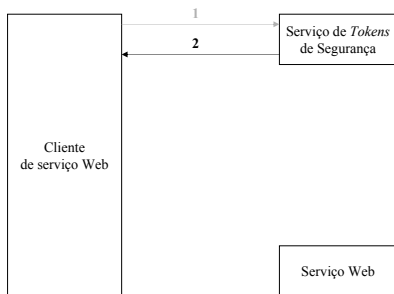
Referências

Segurança - WS-Security



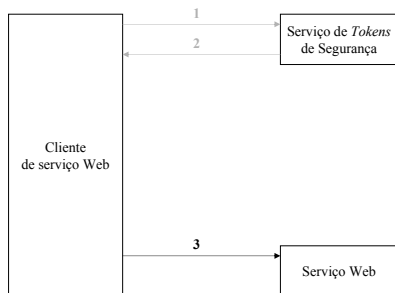
1. Envio de pedido de *tokens* de segurança (pode não ser serviço Web)

Segurança - WS-Security

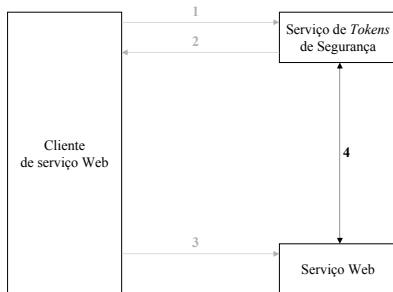


1. Envio de pedido de *tokens* de segurança (pode não ser serviço Web)
2. Recebe *tokens* para adicionar a mensagem SOAP

Segurança - WS-Security

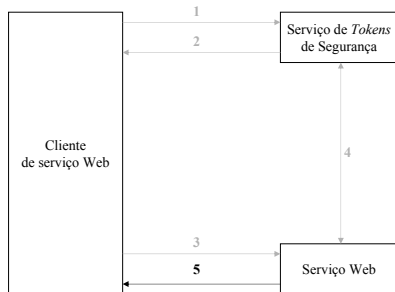


1. Envio de pedido de *tokens* de segurança (pode não ser serviço Web)
2. Recebe *tokens* para adicionar a mensagem SOAP
3. Assina e envia mensagem para serviço Web



1. Envio de pedido de *tokens* de segurança (pode não ser serviço Web)
2. Recebe *tokens* para adicionar a mensagem SOAP
3. Assina e envia mensagem para serviço Web
4. Validação de *tokens*

Segurança - WS-Security



1. Envio de pedido de *tokens* de segurança (pode não ser serviço Web)
2. Recebe *tokens* para adicionar a mensagem SOAP
3. Assina e envia mensagem para serviço Web
4. Validação de *tokens*
5. Recepção de resposta

- ▶ Declaração num cabeçalho SOAP: `/wsse:Security`

<code>wsse:UsernameToken</code>	Identificação de utilizador
<code>wsse:BinarySecurityToken</code>	Codificação binária (certificados X509, <i>tokens</i> Kerberos, etc.)
<code>wsse:SecurityTokenReference</code>	Referência para <i>token</i> de segurança
<code>ds:*</code>	Assinaturas (XML Signature)

- ▶ Cifração feita com XML Encryption: `xenc:ReferenceList`,
`xenc:EncryptedKey` e `xenc:EncryptedData`
- ▶ Estampilhas temporais: `wsu:Timestamp/wsui:Created` e
`wsu:Timestamp/wsui:Expires`

Segurança - Exemplo (User Name Token)

```
<S11:Envelope ...>
  <S11:Header>
    ...
    <wsse:Security>
      <wsse:UsernameToken>
        <wsse:Username>Zoe</wsse:Username>
      </wsse:UsernameToken>
    </wsse:Security>
    ...
  </S11:Header>
  ...
</S11:Envelope>
```

Revisão

UDDI

Motivação

O que é?

Registo

Estruturas de dados

tModel

API

Pesquisa

Armazenamento

Ligação dinâmica

Coordenação

Outras normas

Endereçamento

Segurança

Políticas

Invocação

Conclusões

Referências

Segurança - Exemplo (Binary Security Token)

```
<S11:Envelope ...>
  <S11:Header>
    ...
    <wsse:Security>
      <wsse:BinarySecurityToken ValueType="...#X509v3"
        wsu:Id="X509Token"
        EncodingType="...#Base64Binary">
        MIIEZzCCA9CgAwIBAgIQEmtJZc0rqrKh5i...
      </wsse:BinarySecurityToken>
    </wsse:Security>
    ...
  </S11:Header>
  ...
</S11:Envelope>
```

Segurança - Exemplo (Digital Signature)

```
<S11:Envelope ...>
  <S11:Header>
    ...
    <wsse:Security>
      <ds:Signature ...>
        <ds:SignedInfo>
          <ds:Reference URI="#body">
            <ds:DigestMethod Algorithm="...#sha1" />
            <ds:DigestValue>LyLsF094hPi4wPU...</ds:DigestValue>
          </ds:Reference>
        </ds:SignedInfo>
        <ds:SignatureValue>Hp1ZkmFZ/2kQLXDJbchm5gK...</ds:SignatureValue>
      </ds:Signature>
    </wsse:Security>
    ...
  </S11:Header>
  ...
</S11:Envelope>
```

Segurança - Exemplo (Encryption)

```
<S11:Envelope>
  <S11:Body wsu:Id="body">
    <xenc:EncryptedData ...>
      <xenc:EncryptionMethod Algorithm="...#tripleDES-cbc" />
      <xenc:CipherData>
        <xenc:CipherValue>d2FpbmdvbGRfE0lm4byV0...</xenc:CipherValue>
      </xenc:CipherData>
    </xenc:EncryptedData>
  </S11:Body>
</S11:Envelope>
```

Segurança - Exemplo (Timestamp)

```
<S11:Envelope ...>  
  <S11:Header>  
    ...  
    <wsse:Security>  
      <wsu:Timestamp wsu:Id="T0">  
        <wsu:Created>2001-09-13T08:42:00Z</wsu:Created>  
      </wsu:Timestamp>  
    </wsse:Security>  
    ...  
  </S11:Header>  
  ...  
</S11:Envelope>
```

- ▶ Diferentes plataformas que disponibilizam serviços
- ▶ Diversas capacidades dependentes de configuração de servidor:
 - ▶ Sistema Operativo
 - ▶ Bibliotecas de sistema
 - ▶ Algoritmos/sistemas/propriedades
 - ▶ *Hashing* (MD5, SHA-1, etc.)
 - ▶ Autenticação (Kerberos, Username/Password, etc.)
 - ▶ Qualidade de Serviço
 - ▶ etc.

- ▶ Web Service Policy Framework
- ▶ Conjunto de mecanismos para expressar políticas específicas de um serviço Web
- ▶ Constructos permitem especificar alternativas e combinação de asserções
- ▶ NÃO específica como as políticas são descobertas ou ligadas a um serviço Web
- ▶ Usado em várias especificações (WSDL, UDDI, etc.)

- ▶ *Política*: Conjunto de alternativas de políticas (/wsp:Policy)
- ▶ *Alternativa de política*: Conjunto de asserções de política (/wsp:Policy/wsp:ExactlyOne, /wsp:Policy/wsp:ExactlyOne/wsp:All)
- ▶ *Asserção de política*: Requisito, capacidade ou outra propriedade de um comportamento
- ▶ Providencia operadores sobre conjuntos para semânticas mais ricas (alternativas)

```
<wsp:Policy>
  <wsp:ExactlyOne>

    <wsse:SecurityToken>
      <wsse:TokenType> wsse:Kerberosv5TGT </wsse:TokenType>
    </wsse:SecurityToken>

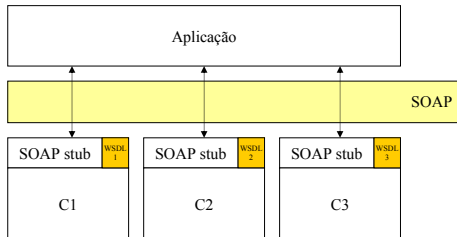
    <wsse:SecurityToken>
      <wsse:TokenType> wsse:X509v3 </wsse:TokenType>
    </wsse:SecurityToken>

  </wsp:ExactlyOne>
</wsp:Policy>
```


- ▶ *Software system* empresarial = manta de retalhos de serviços:
 - ▶ Aplicações de legado
 - ▶ Módulos que comunicam por Middleware orientado a mensagens
 - ▶ Serviços Web baseados em SOAP
 - ▶ *JavaBeans*
 - ▶ ...
- ▶ Quebra de código quando se muda de bibliotecas
- ▶ Módulo altera forma de comunicação
- ▶ Nova localização do serviço

- ▶ *Software system* empresarial = manta de retalhos de serviços:
 - ▶ Aplicações de legado
 - ▶ Módulos que comunicam por Middleware orientado a mensagens
 - ▶ Serviços Web baseados em SOAP
 - ▶ *JavaBeans*
 - ▶ ...
- ▶ Quebra de código quando se muda de bibliotecas
- ▶ Módulo altera forma de comunicação
- ▶ Nova localização do serviço
- ▶ **Problemas de manutenção e de migração de código**

► Web Services Invocation Framework



Revisão

UDDI

- Motivação
- O que é?
- Registo
- Estruturas de dados
- tModel
- API
- Pesquisa
- Armazenamento
- Ligação dinâmica

Coordenação

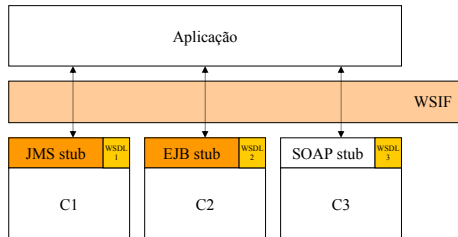
Outras normas

- Endereçamento
- Segurança
- Políticas
- Invocação**

Conclusões

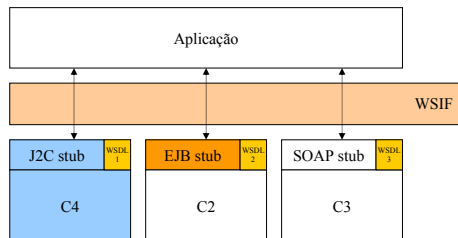
Referências

▶ Web Services Invocation Framework



- ▶ NÃO usa apenas SOAP para invocação de serviços Web
- ▶ WSDL é a base de descrição dos serviços, WSIF complementa com invocação
- ▶ API Java usada para encapsular e invocar serviços (stubs ou invocação dinâmica)

► Web Services Invocation Framework



- NÃO usa apenas SOAP para invocação de serviços Web
- WSDL é a base de descrição dos serviços, WSIF complementa com invocação
- API Java usada para encapsular e invocar serviços (stubs ou invocação dinâmica)

Revisão

UDDI

Motivação
O que é?
Registo
Estruturas de dados
tModel
API
Pesquisa
Armazenamento
Ligação dinâmica

Coordenação

Outras normas

Endereçamento
Segurança
Políticas
Invocação

Conclusões

Referências

Invocação - Exemplo (WSDL)

```
<definitions ...>
  <!-- type, message & port type declarations -->

  <binding ...>
    <java:binding />
    <operation name="getAddressFromName">
      <java:operation methodName="getAddressFromName" ... />
      ...
    </operation>
  </binding>

  <service name="AddressBookService">
    <port ...>
      <java:address
        className="localjava.service.AddressBookImpl"/>
    </port>
  </service>
</definitions>
```

Revisão

UDDI

Motivação

O que é?

Registo

Estruturas de dados

tModel

API

Pesquisa

Armazenamento

Ligação dinâmica

Coordenação

Outras normas

Endereçamento

Segurança

Políticas

Invocação

Conclusões

Referências

Invocação - Exemplo (Serviço)

```
package localjava.service;

import localjava.client.stub.addressbook.wsifypes.Address;
import localjava.client.stub.addressbook.wsifservice.AddressBook;

public class AddressBookImpl implements AddressBook
{
    private HashMap name2AddressTable = new HashMap();

    public Address getAddressFromName(String name)
    {
        return (Address)name2AddressTable.get(name);
    }
}
```

Invocação - Exemplo (Cliente)

```
import localjava.client.stub.addressbook.wsifservice.AddressBook;
import localjava.client.stub.addressbook.wsiftypes.Address;
import org.apache.wsif.*;

public class Run
{
    public static void main(String[] args)
    {
        // create a service factory
        WSIFServiceFactory factory = WSIFServiceFactory.newInstance();

        // parse WSDL
        WSIFService service = factory.getService(args[0],
            null, null,
            "http://wsifservice.addressbook/",
            "AddressBook");

        // create the stub
        AddressBook stub = (AddressBook)service.getStub(AddressBook.class);

        // do the invocation
        Address address = addressBook.getAddressFromName("Zé Ninguém");
    }
}
```


Conclusões

- ▶ Serviços Web
- ▶ UDDI
- ▶ Coordenação entre UDDI, WSDL, SOAP, etc.
- ▶ Outras normas

- ▶ Alonso, G. et al - Web Services: Concepts, Architecture and Applications
- ▶ UDDI: <http://www.uddi.org>
- ▶ WS-Addressing: <http://www.w3.org/2002/ws/addr>
- ▶ WS-Security: <http://www.oasis-open.org/specs/index.php#wssv1.0>
- ▶ WS-Policy: <http://www-106.ibm.com/developerworks/library/specification/ws-polfram>
- ▶ WSIF: <http://ws.apache.org/wsif>