

Services Web – publication et découverte

Fabrice Rossi

<http://apiacoa.org/contact.html>.

Université Paris-IX Dauphine

Services Web – publication et découverte – p.1/15

Solutions services web

La solution principale est UDDI :

- *Universal Description, Discovery, and Integration*
- créé par IBM, Ariba (?) et Microsoft
- évolution :
 - V1 septembre 2000
 - V2 juin 2001
 - V3 juillet 2002
- permet de :
 - publier la description d'un service web
 - découvrir les services web disponibles
- construit à partir de l'infrastructure des services web :
 - données décrites en XML
 - accès par messages SOAP
- accès en Java : *Java Api for XML Registries*

Services Web – publication et découverte – p.3/15

Publication et découverte

- Problèmes classiques des systèmes répartis :
 - comment indiquer qu'un service est disponible
 - comment découvrir un service
- Solutions classiques, les annuaires :
 - le *portmapper* des RPC :
 - associe un programme (identifié par un numéro) à un port UDP ou TCP
 - permet d'obtenir la liste des programmes enregistrés
 - le service de noms de CORBA (*COSNaming*) :
 - organisé de façon hiérarchique
 - permet d'associer un nom à un objet (identifié sous forme d'un numéro unique)
 - associé à des services très évolués : découverte et publication dynamique de l'interface d'un service
 - le *rmiregistry* de Java (RMI), même principe que *COSNaming* (en moins évolué)

Services Web – publication et découverte – p.2/15

Concepts UDDI

- UDDI est un annuaire orienté *Business*
- il décrit 4 types d'entités
 - les *businesses* (en gros les entreprises)
 - les *business services* (les services fournis par les entreprises, au sens non technique du terme)
 - les *technical models* (la description abstraite des services proposés)
 - les *service bindings* (l'implémentation des services)
- organisées selon trois méthodes
 - les pages blanches : annuaire classique par nom (sur les entreprises)
 - les pages jaunes : annuaire thématique (organisé selon une taxonomie)
 - les pages vertes : annuaire technique

Services Web – publication et découverte – p.4/15

Evolutions

- version 2
 - notion de liens entre entreprises (filiales, co-entreprises, etc.)
 - taxonomies plug-ins
 - internationalisation
 - recherches évoluées
 - réplication souple entre noeuds UDDI
- version 3
 - structure hiérarchique entre les annuaires (à la DNS)
 - améliorations sur la sécurité (signatures, etc.)
 - recherches approximatives
 - mécanisme d'abonnement et de notification
 - etc.

Détails techniques

- chaque élément est identifié par un numéro unique (au sein d'un annuaire), le UUID (Universally Unique ID)
- les UUID sont utilisés pour lier les éléments entre eux (par exemple un `tModel` avec un `businessService`)
- `businessEntity` :
 - la description d'une "entreprise"
 - contient des informations de haut niveau, destinées essentiellement à des humains :
 - nom de l'entreprise
 - liste de contacts
 - etc.
 - peut contenir des informations de classification (taxonomie) pour les pages jaunes

Enregistrement d'un service

Etapes :

1. Couper en deux le fichier WSDL, en séparant la partie abstraite de la partie concrète
2. Enregistrer la partie abstraite comme un *technical model* (attention, le `tModel` ne contient pas le WSDL)
3. S'enregistrer comme une *Business Entity*
4. Enregistrer un *Business service* associant la *Business Entity* avec le *technical model*

Remarques :

- tout est décrit en XML
- l'idée de base est que le `tModel` fait référence à la partie abstraite du WSDL (mais ce n'est pas obligatoire)
- de même, un *service binding* peut faire référence à la partie concrète

Exemple

```
1 <businessEntity businessKey="">
2   <discoveryURLs>
3     <discoveryURL useType="wsil">
4       http://localhost/inspection.wsil
5     </discoveryURL>
6   </discoveryURLs>
7
8   <name>DougCo Software</name>
9   <description xml:lang="en">
10     Sample business created for
11     the UDDI discovery demo.
12   </description>
13
14   <contacts>
15     <contact useType="Technical contact">
16       <personName>Doug Tidwell</personName>
17       <phone useType=
18         "voice">1-919-555-5583</phone>
19       <phone useType=
20         "fax">1-919-555-2389</phone>
21       <phone useType=
22         "mobile">1-919-555-9385</phone>
23       <email>dtidwell@us.ibm.com</email>
```

Exemple (2)

```
24     <address>
25       <addressLine>1234 Main Street
26     </addressLine>
27     <addressLine>Anytown, TX 73958
28     </addressLine>
29   </address>
30 </contact>
31 </contacts>
32
33 <categoryBag>
34 <keyedReference keyName=
35   "uddi-org:iso-ch:3166-1999"
36   keyValue="US-NC" tModelKey=
37   "UUID: 4E49A8D6-D5A2-4FC2-93A0-
38   0411D8D19E88"/>
39 <keyedReferencekeyName= "ntis-gov:naics:1997"
40   keyValue="541511" tModelKey=
41   "UUID: COB9FE13-179F-413D-8A5B-5004DB8E5BB2"/>
42 <keyedReference keyName="unspsc-org:unspsc"
43   keyValue="43.16.26.05.00" tModelKey=
44   "UUID: CD153257-086A-4237-B336-6BDCBDC6634"/>
45 </categoryBag>
46 </businessEntity>
```

Services Web – publication et découverte – p.9/15

Détails techniques (2)

- **businessService :**
 - description d'un service
 - réalise le lien entre une interface de service (un tModel) et son implémentation, par l'intermédiaire de sous-éléments `bindingTemplates`
 - un `bindingTemplate` se contente de pointer vers une description externe
- **tModel :**
 - décrit l'interface d'un service
 - se contente de pointer vers une description externe
- **Remarques importantes :**
 - l'annuaire UDDI ne stocke pas les descriptions WSDL
 - l'api ne propose pas le téléchargement de ces descriptions
 - UDDI n'oblige pas à utiliser WSDL

Services Web – publication et découverte – p.10/15

Exemple

```
1 <tModel tModelKey="">
2   <name>EightBall Interface</name>
3   <description xml:lang="en">
4     The interface-only definition
5     of the EightBall service.
6   </description>
7   <overviewDoc>
8     <description xml:lang="en">
9       The mystical powers of the EightBall, channelled into Java.
10    </description>
11    <overviewURL>
12      http://localhost/eightball.org/eightballinterface.wsdl
13    </overviewURL>
14  </overviewDoc>
15  <categoryBag>
16    <keyedReference
17      tModelKey="UUID:C1ACF26D-9..."
18      keyName="uddi-org:types"
19      keyValue="wsdlSpec"/>
20  </categoryBag>
21 </tModel>
```

Services Web – publication et découverte – p.11/15

Exemple

```
1 <businessService serviceKey=""
2   businessKey="00EB06FC-3C45-42B8-B394-1
3   F7F35602B2E">
4   <name>EightBallService</name>
5   <descriptionxml:lang="en">
6     The power of the MagicEightBall, channelled through code.
7   </description>
8
9   <bindingTemplates>
10    <bindingTemplatebindingKey=""
11      serviceKey="">
12      <descriptionxml:lang="en">
13        Ties this implementation to the EightBallInterface.
14      </description>
15      <accessPointURLType="http">
16        http://localhost:8080/axis/services/EightBall
17      </accessPoint>
18      <tModelInstanceDetails>
19        <tModelInstanceInfo tModelKey=
20          "UUID:0B6E1227-329A-4657-89B5-35DA36CA2554">
21          <descriptionxml:lang="en">
22            Implementation of the EightBallInterface.
23          </description>
```

Services Web – publication et découverte – p.12/15

Exemple (2)

```
24      service-concret.xml
25      <instanceDetails>
26        <overviewDoc>
27          <overviewURL>
28            http://localhost/EightBallImplementation.wsdl
29          </overviewURL>
30        </overviewDoc>
31      </instanceDetails>
32    </tModelInstanceInfo>
33  </tModelInstanceDetails>
34 </bindingTemplate>
35 </bindingTemplates>
</businessService>
```

Solutions concurrentes/complémentaires

- UDDI est intéressant comme annuaire global
- solution concurrente : ebXML, un standard OASIS (même esprit)
- solution complémentaire, WS-Inspection :
 - idée de base : donner la liste des services web disponibles sur un serveur
 - dialecte XML
 - compatible UDDI : la description d'un service peut faire référence à l'enregistrement UDDI de celui-ci
 - compatible WSDL : même chose avec WSDL

Exemple

```
1 <?xml version="1.0"?>
2 <inspection
3   xmlns="http://schemas.xmlsoap.org/ws /2001/10/inspection/"
4   xmlns:wsilwsdl="http://schemas.xmlsoap.org/ws/2001/10/inspection/wsdl/">
5 <service>
6   <abstract xml:lang="en-US">
7     WSDL description for the Eight Ball service
8   </abstract>
9   <name xml:lang="en-US">EightBall</name>
10  <descriptionreferencedNamespace= "http://schemas.xmlsoap.org/wsdl/"
11    location= "http://localhost/EightBall.wsdl"/>
12 </service>
13 </inspection>
```