

《面向服务的体系结构》第五章作业

1. 解释UDDI的定义及其作用。

答：UDDI (Universal Description, Discovery, and Integration) 是一种用于描述、定位、集成和使用网络中业务列表的标准。它是一种XML-based的开放的、跨平台的框架，用于创建、描述和发现公开的Web服务，即它是一种目录服务，企业可以通过它对Web Service进行注册和检索。

UDDI技术是SOAP和WSDL之外的另一项Web Service的核心技术。它可以使提供Web Service的企业注册服务信息，从而使企业的合作伙伴或潜在的客户能够发现并访问这些Web Service，也可以使企业发现其他企业提供的服务，以便扩展潜在的业务伙伴关系。

UDDI相当于Web Service的一个公共注册表，可以理解成电子商务应用与服务的“网络黄页”

2. 简述UDDI的数据结构。

答：UDDI注册中心存储的信息以XML形式表示，称为UDDI数据结构，UDDI规范中定义了这些数据结构的含义以及彼此之间的关系。

UDDI的使用者可以使用这些数据结构以及相关的API来处理UDDI注册中心的信息，对UDDI注册中心的搜索结果也是使用这些数据结构来表达，所以UDDI数据结构实际上就是UDDI编程API的输入输出参数。

UDDI数据结构主要包括五种元素：

- **businessEntity**：这个结构代表一个业务实体，如公司或组织。它包含了关于这个实体的一些描述信息，比如公司名称、描述、联系信息等。每个businessEntity可以有多个businessService与之关联。
- **businessService**：这个数据结构表示业务实体提供的一个特定的服务。每个businessService都包含了服务的描述和分类信息，以及它的bindingTemplates，后者指明了如何与这个服务交互。
- **bindingTemplate**：用于描述与businessService相关的技术实现信息。这个结构通常包含了访问服务所需要的具体细节，如服务的访问点（access point，通常是一个URL）和使用的协议。
- **tModel**：这个结构提供了一种抽象的表示方法，用于描述服务的技术概念或者规范。它可以被用来指定businessService使用的一些标准接口或者协议。bindingTemplate可能会引用一个或多个tModel来明确它实现的具体协议或者格式。
- **PublisherAssertion**：它描述了两个businessEntity之间的关系。通过publisherAssertion，一个业务实体可以声明它与另一个业务实体有某种关系，比如合作伙伴或者附属机构。

3. 给出UDDI 2.0内置的标识系统和分类系统的名称。

答：在UDDI中，标识系统本身是通过tModel定义的，UDDI2.0有两个内置标识系统：

1. Dun&Bradstreet D-U-N-S 数字识别系统 (<http://www.dnb.com>)，tModel名为dnb-com:D-U-N-S；
2. Thomas Register供货商识别码系统 (<http://www.thomasregister.com>)，tModel名为Thomasregister-com:supplierID。

UDDI中的分类系统也是通过tModel定义的，UDDI注册中心一般都预置了四种常用的分类系统：

1. NAICS(North American Industry Classification System)北美工业分类系统(<http://www.census.gov/naics>), tModel名为ntis-gov:naics:1997;
2. UNSPSC(Universal Standard Products and Services Classification)统一标准产品和服务分类系统(<http://www.unspsc.org/>), tModel名为unspsc-org:unspsc:3-1;
3. ISO 3166地理分类系统(ISO 3166 Geographic Taxonomy, <http://www.iso.org/iso/en/prods-services/iso3166ma/index.html>), tModel名为iso-ch:3166:1999;
4. 其他分类法, 即通用的基于关键词分类法, tModel名为uddi-org:general_keywords。

4. 试述WSDL在UDDI中注册的方法。

答: 一个完整的WSDL文档包含3部分内容:

- 由type、message和portType元素组成的抽象接口
- 由binding元素组成的从端口类型到消息传输协议的绑定
- 由service元素组成的服务访问端口定义

type、message、portType和binding四种元素称为服务接口, service元素称为服务实现。服务接口和服务实现分别放在不同的文档, 服务接口就可以实现复用。

在UDDI中注册WSDL通常有以下几个步骤: 准备WSDL文件; 创建businessEntity; 创建businessService; 创建bindingTemplate; 引用tModel; 将WSDL的URL添加到bindingTemplate; 注册信息到UDDI注册中心:

具体如下:

- 每一个包含服务接口的文档都被映射为一个tModel(也可以把每一个binding元素映射为一个tModel),
- service元素被映射为一个商业服务, service元素包含的每个port元素都被映射为一个UDDI绑定模板。
- WSDL文档的地址映射到tModel的overviewURL元素。如果要为每一个binding元素建立一个tModel, 则要在后面附加上到binding的引用。
- WSDL文档的targetNamespace映射到tModel的name元素。
- WSDL文档的documentation元素映射到tModel的描述元素。
- tModel分类为“wsdlSpec”类型。
- 服务端口对应到UDDI商业服务的绑定模板, 其中端口地址映射到绑定模板的accessPoint属性;
- 服务实现对应的服务接口映射生成的tModel的主键在绑定模板tModel InstanceInfo元素的属性中声明;
- WSDL文档的实际存放地址映射为tModel InstanceInfo元素中的overviewURL元素。