

SYBASE  
技术服务园地  
连载 (25)

SYBASE TECHNOLOGY SERVICE FIELD

解决方案

## 使用 PowerDesigner 12 创建 Web 服务(3)

### 1 简介

前两篇文章介绍了PowerDesigner 12对面向服务的业务过程建模的全面支持,并阐述了业务过程开发模型与面向服务的对象模型之间的双向关系。本文将重点描述PowerDesigner 12对Web服务的对象模型的支持,这些服务将最终导入业务过程模型,并与相应地业务流程集成。Web服务描述了在网络上可通过SOAP消息进行访问的操作集合,其接口和部署位置(可选)由WSDL说明。Web服务以一种标准格式提供集成现有遗留系统的或新增的功能,并且允许用户和应用程序从任何位置访问它们。PowerDesigner 12同时针对Java和Microsoft .NET引入了完善的Web服务设计和生成机制。限于篇幅,本文只介绍Java的服务设计与生成。

### 2 PowerDesigner 的 Web 服务功能

PowerDesigner 12 允许您新建 Web 服务组件、通过对 WSDL 进行反向工程来创建 Web 服务组件、浏览 UDDI、搜索 WSDL、从 Web 服务组件的定义生成 WSDL、生成 Java (JAX-RPC 和 JAXM) 的服务器端代码以及生成 Java 的客户端代理。为了设计 Web 服务组件,PowerDesigner 使用了 UML 类图、组件图和部署图。Web 服务由组件图的组件表示,而已部署的 Web 服务则由部署图的组件实例来表示。

#### 2.1 创建 Web 服务组件

创建 Web 服务组件有5个步骤。

第1步:创建面向对象模型(OOM)。选用Java语言,在此模型上会自动附加用于WSDL的扩展模型定义(XEM)。XEM定义了WSDL的生成模板和扩展属性。

第2步:创建类。首先创建一个包(Package),然后在此包中创建新的类。图1显示了一个创建完成的类(Order)。

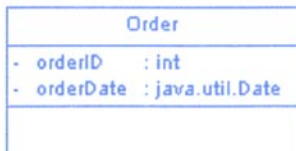


图1 创建新类(Order)

第3步:使用Web服务向导将创建的类转换成Web服务组件。右键单击该类,然后选择“创建Web服务组件”命令,或在选中该类后选择“工具->创建Web服务组件”命令。图2显示了一个Web服务向导的示例。Web服务类型可以是接口或实现。如果选择“接口”,则表明您只希望定义此Web组件并生成相应的WSDL。对于Java,此组件类

型可能是“标准”、“服务程序”或“无状态会话Bean”。这几种Java组件类型的实现方法分别是:“标准”类型使用JAX-RPC、“服务程序”类型使用JAXM,而“无状态会话Bean”类型将使用J2EE(JSR109)Web服务规范。当向导结束时创建一个组件和类。图4显示了组件图中的组件符号示例。

点击“Next”进入下一个界面,如图3所示Web服务类型选择“实现(Implementation)”,组件类型选择“Web Services for J2EE (Stateless Session Bean)”选择完之后点击“Next”进入下一个页面,如图4所示在PowerDesigner 12中可以选择是否为新建的服务和组件生成图形符号。

第4步:添加Web方法。您需要新建一个操作(Operation)、打开该操作的属性页、单击“Web方法”复选框、定义参数和返回类型以及编写该操作的实现代码。图3显示了Web方法GetStockQuote的实现示例。

最后一步是对WSDL生成进行自定义。可以自定义参

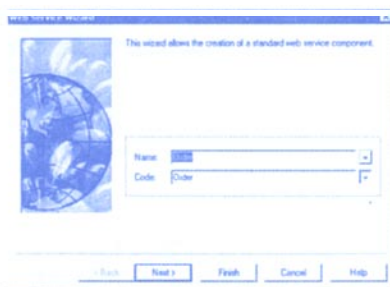


图2 Web 服务向导



图3 Web 服务类型定义界面

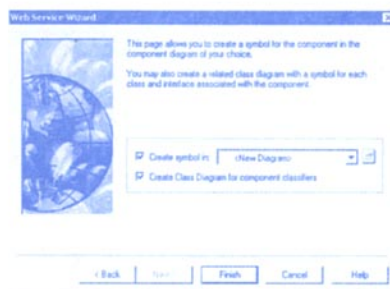


图4 图形组件符号选择界面



图5 组件符号

数和返回值的 WSDL 数据类型、操作的扩展属性以及组件的属性以及扩展属性。若要检验生成的 WSDL, 您可以打开组件属性页的 WSDL 选项卡。生成 WSDL 的任务可以交给 PowerDesigner 完成, 也可以由您自行定义 (用户定义)。如果 Web 服务组件来源于对 WSDL 的反向工程, 原始的 WSDL 将得到保留。此外, 可以在类属性页的“预览”选项卡中预览生成的实现代码。

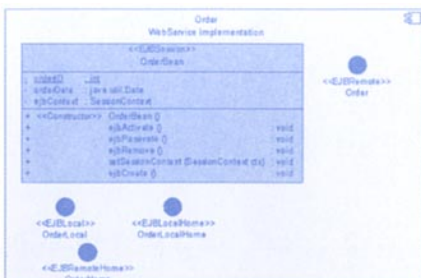


图6 组件图

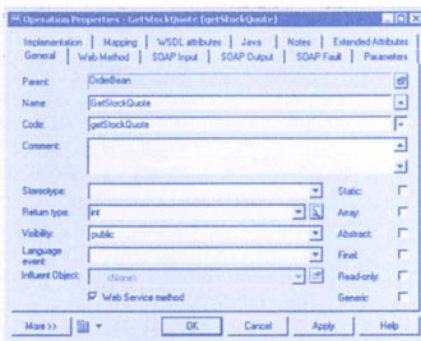


图7 Web 方法 GetStockQuote 的实现示例

## 2.2 对 WSDL 进行反向工程

WSDL 可以仅定义 Web 服务接口, 也可以同时定义接口和访问点的 URL。Web 服务接口可能被多个服务提供者用来提供相同的服务, 这些服务会具有不同的实现和访问点。如果拥有 WSDL 文档的文件路径或 URL, 您就可以通过对它进行反向工程来创建 Web 服务组件。

为了对 WSDL 进行反向工程, 选择“Language->Import WSDL...”菜单并输入 WSDL 文件的路径或 URL。在查找感兴趣的 WSDL 时, 甚至可以使用 UDDI 浏览器在 UDDI 注册服务中进行搜索。进行反向工程之后, 会在组件图中创建一个组件, 同时在类图中创建一个类。所创建的类将仅包含该 Web 服务的操作, 若想实现该 Web 服务, 还需要定义这些操作的实现代码。

## 2.3 生成 Java 类型的 Web 服务

如果 Web 服务的实现类型为“标准”, PowerDesigner 将使用 JAX-RPC 来实现。如果实现类型为“服务程序”, PowerDesigner 将使用 JAXM 来实现。如果实现类型为“无状态会话 Bean, 则 PowerDesigner 将使用 J2EE (JSR109) Web 服务规范来实现。

JAX-RPC 定义了 RPC 类型的 Web 服务调用。虽然它易于开发和使用的, 但为了避免复杂的对象/XML 映射, 它只能局限于简单的消息格式。而基于 JAXM 的 Web 服务组件显然更具优越性, 它允许您灵活地处理复杂的消息结构。

若要使用 JAX-RPC 或 JAXM, 您需要安装 Java Web Services Developer Pack (JWS DP)。此程序包可从 Sun 公司的网站下载。在部署 JAX-RPC 或 JAXM 类型的 Web 服务组件时, 需要有支持 JAX-RPC 或 JAXM 的服务器。支持 JAX-RPC 或 JAXM 的 Apache Tomcat 附带提供了 JWS DP。

至于基于 JAX-RPC 的 Web 服务组件, 您只需实现其 Web 方法的代码。若要生成 WSDL、服务器端代码和 / 或客户端代理, 可选择“语言>生成 Java 代码”, 然后选择“使用 XRPC 工具生成 WSDL (服务器端)”和 / 或“使用 XRPC 工具生成 Web 服务的代理代码”命令 (见图 11)。XRPC 命令是 JWS DP 的一个工具。它可以生成 JAX-RPC 类型的服务器端代码或客户端代理。在部署生成的 Web 服务时, 需要创建一个包含所有生成文件的 .WAR 文件, 并将此 .WAR 文件部署在支持 JAX-RPC 的服务器上, 比如说 Apache Tomcat。

## 3 结束语

PowerDesigner 12 简化了 Web 服务的设计和生成过程。可以使用 PowerDesigner 设计、创建 Web 服务, 以及对已有的 Web 服务进行反向工程。回顾 3 篇文章, 我们介绍了 PowerDesigner 12 的集成化企业级过程建模能力, 并阐述了其对 SOA 的支持。业务过程开发模型与面向服务的对象模型之间的双向关系是 PowerDesigner 12 的独有的特性, 使业务分析人员与 IT 技术人员可以通过解析和描述业务功能下面的流程, 并与面向服务的应用集成环境协调行动。大大提高了企业业务流程的分析, 设计与开发能力。

文 / Sybase 公司 PowerDesigner 上海研发中心 张晓光

## SYBASE 常见问题解答 (上接第 3 期)

### 常见 Error Msg 的处理

#### Error 8206

解决方案:  
调参数 'number of open objects'

#### Error 8402

解决方案:  
对有关表执行 dbcc 检查, 如果是索引损坏, 则将有关

non-clustered index 删除重建 如果是表损坏, 则将表中数据导出后删除重建。

#### Error 9502

Dataexception - string data right truncated  
解决方案:

检查有关 sql 语句中变量的数据长度, 类型与要操作的表, 其中列的数据类型和长度是否匹配。

SYBASE 软件 (中国) 有限公司 电话: 86-10-59215888