CFDAB

国 家 食 品 药 品 监 管 信 息 化 标 准

CFDAB/T 0402—2013

食品药品监管应用支撑平台通用技术规范

General technical specifications for application supporting platform of food and drug administration

（征求意见稿）

2013-xx-xx 发布　　20xx-xx-xx 实施

国家食品药品监督管理总局　发 布

目 次

前 言

本标准依据 GB/T 1.1—2009 给出的规则起草。

本标准由国家食品药品监督管理局信息中心提出。

本标准由国家食品药品监督管理总局科技和标准司归口。

本标准起草单位：国家食品药品监督管理局信息中心、中科软科技股份有限公司、广东省食品药品监督管理局。

本标准主要起草人：陈锋、张原、陆颖、刘靓、赵坤、李宗波、张翔、刘吕昕、史先东、李建魁。

食品药品监管应用支撑平台通用技术规范

1 范围

本标准提出了食品药品监管信息化工程应用支撑平台应遵循的技术规范，对应用支撑平台总体架构、统一接入认证、统一用户管理、权限管理、行为审计、服务建设等提出了要求。

本标准适用于食品药品监管信息化工程的设计、开发、建设实施和管理维护。

2 规范性引用文件

下列文件对于本文件的应用是必不可少的。凡是注日期的引用文件，仅注日期的版本适用于本文件。凡是不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

GB/T 11457—2006 信息技术 软件工程术语

GB/T 29262—2012 信息技术 面向服务的体系结构（SOA）术语

CFDAB/T 0102.1—2013 食品药品监管信息化基础术语 第1部分：技术基础术语

3 术语、定义和缩略语

3.1 术语和定义

CFDAB/T 0102.1—2013 界定的以及下列术语和定义适用于本文件。

3.1.1

面向服务的体系结构 service oriented architecture

遵循面向服务原则、具有松耦合特性的体系结构风格。

[GB/T 29262—2012，定义 2.37]

3.1.2

服务总线 service bus

服务接入与消费的中介基础设施，为基本服务提供了基于标准的事件驱动消息路由。其基本功能包括：服务路由、消息转换、事件处理，提供服务调用，及相关中介服务，支持 Web Service 或 JMS 连接，连接各种应用，服务，信息，平台资源。

[GB/T 29262—2012，定义 2.10]

3.1.3

Web 服务 web service

一种应用编程接口或 Web 应用编程接口，通过标准的规约进行定义、并通过标准进行访问和使用。

[GB/T 29262—2012，定义 2.57]

3.1.4

接口 interface

a）一个共享的边界。信息跨越边界传送。

b）连接两个或多个其他部件，为了相互间传送信息的硬件或软件部件。

c）连接两个或多个部件，为了在相互间传送信息。

d）作为如 b）中连接的或被连接的部件。

[GB/T 11457—2006，定义 2.795]

3.1.5

服务接口 service interface

用于实现使用者和提供者之间的通信合约，它提供了服务的功能、位置等执行通信时所需要的所有细节的描述。

[GB/T 29262—2012，定义 2.27]

3.2 缩略语

ACL ：访问控制列表（Access Control List）

CA：数字证书认证中心（Certificate Authority）

HTTP：超文本传输协议（HyperText Transfer Protocol）

JMS：Java 消息服务（Java Message Service）

LDAP：轻量目录访问协议（Lightweight Directory Access Protocol）

RBAC：基于角色的访问控制（Role-Based Access Control）

SOA ：面向服务的体系结构（Service-Oriented Architecture）

WSDL：Web 服务的描述语言（Web Services Description Language）

4 应用支撑平台总体架构

图 1 为食品药品监管信息化工程建设的应用支撑平台总体框架。

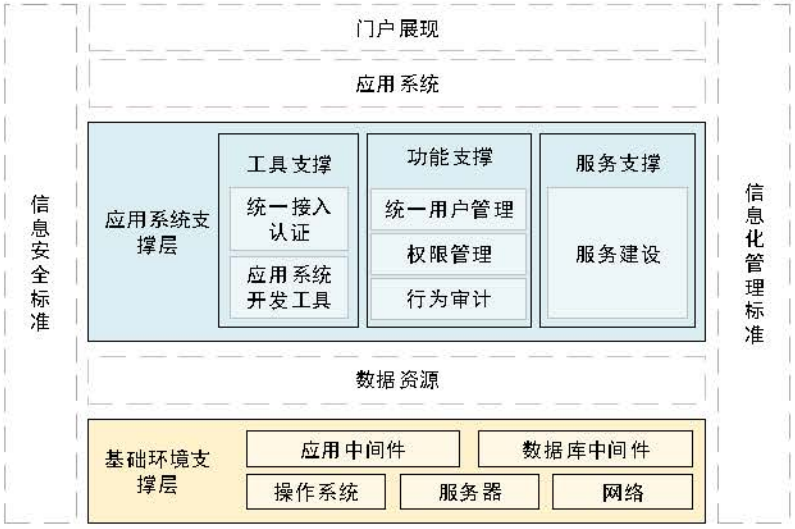

图 1 应用支撑平台总体架构图

其中，应用支撑平台的基础环境支撑层和应用系统支撑层包括：

a）基础环境支撑层：主要包括应用系统运行所需要的基础软硬件环境，包括服务器、网络和系统软件等，为应用系统提供基础的环境支撑；

b）应用系统支撑层：为应用系统和业务门户提供工具支撑、功能支撑、服务支撑。工具支撑由一系列的开发工具和统一接入认证功能组成；功能支撑提供应用系统的基础功能，包括：统一用户管理、权限管理、行为审计等；服务支撑提供数据服务，并对提供的服务进行管理维护，对服务建设提出要求。

以下重点对应用支撑层的统一接入认证规范、统一用户管理、权限管理规范、行为审计规范、服务建设规范提出要求。

5 统一接入认证规范

5.1 概述

应用支撑平台的统一认证功能，实现对各应用系统的单点登录。各应用系统应遵循统一接入认证规范的要求，使用单点登录实现应用系统的登录认证，并通过接口获取登录用户信息、使用统一用户管理提供的用户和机构信息、使用权限管理维护应用系统登录权限。

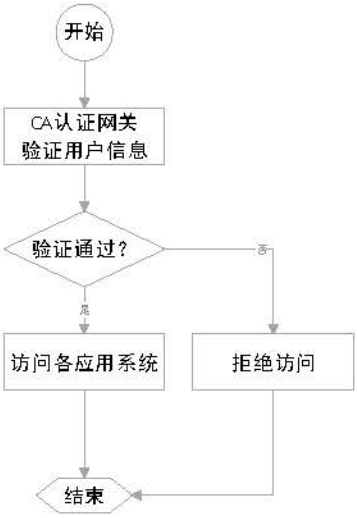

图 2 统一接入认证流程图

如图 2 所示，用户发起服务请求后，首先由 CA 安全认证网关进行用户信息验证，如果用户信息验证通过，则可访问具有权限的应用系统，如果验证不通过则拒绝用户的访问。

5.2 单点登录规范

单点登录基于 CA 安全认证体系实现，应符合以下要求：

a）食品药品监管信息化工程建设基于 CA 安全认证体系，需要用户身份认证的应用系统应基于食品药品监管信息化工程 CA 安全认证体系进行用户身份认证。

b）对于已建成且使用自身 CA 安全认证体系的应用系统，可继续使用其 CA 安全认证体系，在需要时，与食品药品监管信息化工程 CA 证书实现相互认证，CA 证书的注销列表通过数据交换方式定期同步。

c）应用支撑平台应提供安全认证网关实现单点登录。用户单点登录成功后，可使用统一的用户证书访问不同应用系统，无需多次登录。

d）单点登录的实现对应用系统建设有以下要求：

——各应用系统的用户信息应与安全认证网关使用的 LDAP 用户信息同步，且用户关键标识一致。

——各应用系统根据安全认证网关的认证接口获取用户信息。

6 统一用户管理规范

应用支撑平台应对应用系统提供统一用户管理功能，各应用系统用户的管理和维护由应用支撑平台统一实现，包括组织机构管理、用户管理、系统账号管理等功能。各应用系统在开发时，可以通过接口调用和同步数据表的方式获取用户信息，各应用系统的用户信息和组织机构信息需要与应用支撑平台保持一致。设计思路如图 3 所示：

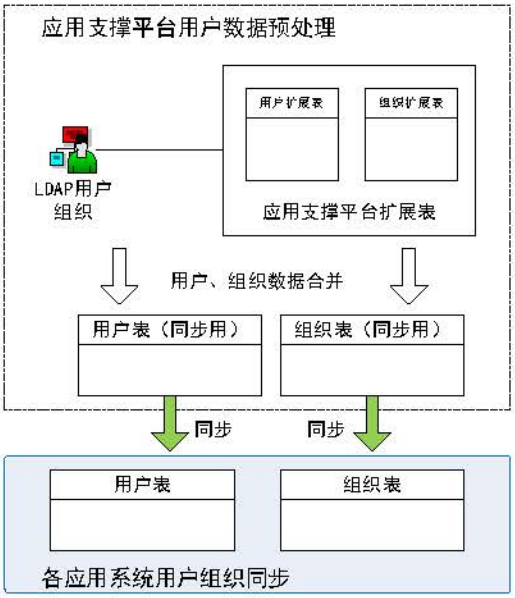

图 3 统一用户管理规范设计思路图

用户和组织机构信息发生变化时，由应用支撑平台进行预处理，各应用系统与应用支撑平台进行用户表和组织表的同步。

7 权限管理规范

7.1 权限基本要求

应用支撑平台应提供统一权限管理功能，负责系统级别权限过滤，即用户能够访问哪个系统。应用系统内部的功能级别权限过滤，由应用系统独立完成。如图 4 所示：

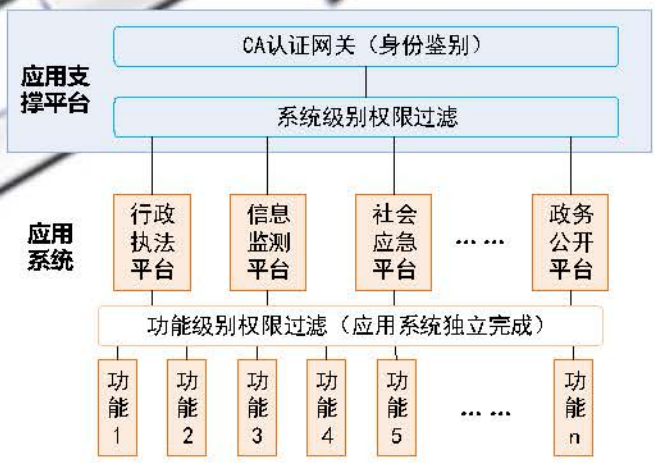

图 4 权限管理功能基本要求图

7.2 系统访问权限

应用支撑平台实现用户的系统访问权限，各应用系统开发时如需要判断用户是否具有指定系统的权限，可通过调用如下接口获取：

a）权限访问列表接口：根据用户登录名称获取该用户所具有访问权限的应用列表。

b）权限验证接口：根据用户登录名称和应用标识，验证该用户是否具有此应用的访问权限。

7.3 功能访问权限

功能访问权限由各应用系统根据本系统的需求进行控制，建议包括以下权限控制：

a）功能权限控制：基于 ACL、RBAC 等多种方式实现权限控制，权限控制粒度应到每一个功能点，基于角色可自由分配权限。

b）菜单权限控制：基于角色、菜单实现用户菜单级的授权管理，实现菜单权限可自由分配。

c）数据权限控制：根据业务需求，对数据进行分类，做到数据权限的控制。

8 行为审计规范

8.1 概述

应用支撑平台建设应包含行为审计功能。行为审计包括应用系统日志记录、系统管理日志记录、用户行为日志记录，应用系统应按要求分别调用应用退出系统日志记录接口、系统管理日志记录接口、用户行为日志记录接口，实现行为审计。

8.2 应用系统日志记录接口

应用系统日志记录应用系统关键业务数据的操作信息，日志记录内容应包括：

a）客户端 IP 地址；

b）登录用户标识；

c）行为分类，如：用户行为、系统行为等；

d）动作名称，如：登录、退出等；

e）操作的应用系统名称；

f）操作结果，如：成功、失败；

g）操作安全级别，登录成功或退出安全级别为：1；登录失败安全级别为：10；

h）备注。

8.3 系统管理日志记录接口

系统管理日志记录管理员的操作，包括菜单注册、角色维护、角色授权、人员授权等操作，系统管理日志记录内容应包括：

a）客户端 IP 地址；

b）行为分类，如：用户行为、系统行为等；

c）动作名称，如：增加、删除、修改、启动、停止等操作；

d）操作的应用系统名称；

e）操作的对象；

f）操作结果，如：成功、失败；

g）操作安全级别，操作安全级别分为 1-10 级，操作行为对系统安全的影响越严重，安全级别越高，数字越大；

h）备注。

8.4 用户行为日志记录接口

用户行为日志记录应用系统对数据的增加、删除、修改以及重要数据的查询。用户行为日志记录内容应包括：

a）客户端 IP 地址；

b）行为分类，如：用户行为、系统行为等；

c）动作名称，如：增加、删除、修改、启动、停止等等操作；

d）操作的应用系统名称；

e）操作的对象；

f）操作结果，如：成功、失败；

g）操作安全级别，操作安全级别分为 1-10 级，操作行为对系统安全的影响越严重，安全级别越高，数字越大；

h）备注。

9 Web Service 建设规范

9.1 建设基本要求

Web service 建设宜基于服务总线进行，或按照以下 web service 设计规范、封装规范、开发规范进行建设。

9.2 设计规范

9.2.1 概述

Web service 作为应用系统之间的交互方式，应按照统一的标准进行开发，web service 的设计包括 web service 的识别与 web service 的定义。

9.2.2 Web service 的识别

Web service 的识别应符合以下要求：

a）Web service 的识别是指对业务进行分析和梳理，抽象出业务实现所需的 web service，并按 web service 的分类要求进行合理划分。

b）Web service 的识别应分析与业务功能或业务数据相关的接口以及约束该接口的契约，接口和契约采用中立、基于标准的方式进行定义，它独立于实现 web service 的硬件平台、操作系统和编程语言。

c）Web service 的识别应从业务的角度出发，包括但不限于下述几点：

——业务流程切入点：通过梳理、优化业务流程，将业务流程转化为可重用、更具灵活性的流程服务。

——用户体验切入点：关注用户体验需求，为终端用户提供增值、个性化、多渠道的服务，并据此来优化整合内部的应用和流程。

9.2.3 Web service 的定义

9.2.3.1 定义描述

Web service 的定义是在 web service 的识别的基础上定义 web service 的各项属性，描述 web service 的信息。web service 的属性包括：基本属性、技术属性、安全属性、配置属性。web service 的各项属性定义应分阶段进行，并逐步细化。web service 的识别阶段定义 web service 的基本属性，设计阶段定义 web service 的技术属性与安全属性，部署阶段定义 web service 的配置属性。

9.2.3.2 基本属性

Web service 的基本属性包括但不限于表 1 所述信息：

表 1 基本属性

序号	属性	说明	取值说明
1	编码	标识 web service 的唯一编码	可由英文字母、数字组成。
2	英文名称	web service 的英文概要名称，描述应简洁准确	可由英文单词组成。
3	中文名称	web service 的中文概要名称，描述应简洁准确	可由中文、字母、数字组成。
4	性质编码	描述 web service 特性的编码	可由英文字母、数字组成。如：entInfo
5	功能描述	对 web service 功能规格的详细描述。	
6	开发单位	实现 web service 的开发单位	如：XXX 公司

9.2.3.3 技术属性

Web service 的技术属性包括但不限于表 2 所述信息：

表 2 技术属性

序号	属性	说明	取值说明
1	版本号	web service 的版本号	如：V1.0
2	注册时间	web service 的正式注册时间	如：2013-01-01 10:00
3	依赖的服务	本 web service 需要调用的其它编码	可由英文字母、数字组成。如：entInfo
4	实现方式	具体技术实现方式	如：JAVA
5	类型	属于 web service 体系中的哪种类型	如：访问服务、数据服务、业务服务、流程服务、综合服务、展现服务
6	交互属性	是否需要人工交互	是/否
7	调用方式	客户端调用 web service 的具体方式	如：同步调用、异步无返回调用、异步有返回调用
8	接口方法	web service 提供的接口方法列表	如：创建信息的接口方法为 create(String name,String id)
9	接口协议	调用 web service 的通讯协议	如：HTTP
10	启用时间	web service 的正式启动时间	如：2013-01-10 8:00
11	停用时间	web service 的正式停用时间	如：2013-12-31 18:00

9.2.3.4 安全属性

Web service 的安全属性包括但不限于表 3 所述信息：

表 3 安全属性

序号	属性	说明	取值说明
1	安全要求	调用 web service 时，是否需要进行安全认证	是/否
2	允许调用的角色	允许调用该 web service 的角色列表	如：Operator；Manager
3	自行安全认证	web service 被调用时，是否还进行自身的安全认证	是/否

9.2.3.5 配置属性

Web service 的配置属性包括但不限于表 4 所述信息：

表 4 配置属性

序号	属性	说明	取值说明
1	部署 IP 地址	提供 web service 功能的网络 IP 地址	如：127.0.0.1
2	接口定义文件	描述 web service 接口定义的文件路径	如：http://192.168.1.1/test.wsdl
3	可以使用的时间	可以使用该 web service 的时间段	如：0：00—24：00
4	是否支持重试	web service 调用失败后，是否支持重发调用	是/否

9.3 封装规范

9.3.1 封装原则

Web service 封装应遵循以下原则：

a)Web service 封装是 web service 实现的手段，web service 封装将应用系统可重用的功能或数据“剥离”出来，对外以相关的接口方式以及约束这个接口的契约提供给消费者调用。接口和契约的定义是中立的且基于标准的，并独立于实现 web service 的硬件平台、操作系统与编程语言。

b）Web service 封装应遵循包括但不限于下述原则：

——无状态原则：最大限度减少 web service 管理的状态信息以及状态期限。

——单一实例原则：避免功能冗余。

——接口定义原则：使用 WSDL 定义 web service 的接口，使用 WS-Policy 描述 web service 的契约，使用 XML 模式（Schema）定义 web service 交换的消息格式。web service 的使用者依赖 web service 的契约调用服务，web service 的定义应相对稳定，修改应通过审核批准。

——自包含和模块化原则：web service 封装的是那些在业务上稳定的、重复出现的活动和组件，组成的功能实体是完全独立自主的，可以独立进行部署、版本控制、自管理和恢复。

——粗粒度原则：确定 web service 的粒度时需要考虑性能需求，以及未来可能进行的更改对 web service 实现的影响，应尽可能使用粗粒度模式隐藏其中的细粒度 web service，这样有利于将 web service 与其实现的更改隔离开来。

——松耦合性原则： web service 的接口和实现相独立，web service 的使用者可以通过组合一

个或多个 web service 来构建应用，无须理解服务的底层实现。

——可重用原则：web service 是可重复使用的。

——策略声明原则：应利用策略声明描述对 web service 的期望，例如：安全性方面的要求、与业务有关的语义方面的要求以及 web service 级别方面的要求等。

9.3.2 封装方式

Web service 的封装方式应符合以下要求：

a）对于技术实现方式和接口不能满足或难以满足本标准 web service 的定义与封装要求的现有应用系统，建议通过适配器对现有应用系统进行集成，利用适配器对外提供的各类接口的方式实现 web service 的封装。

b）对于采用 J2EE 支持 web service 开发的现有应用系统，可以在不改变现有应用系统的技术实现方式与现有接口的前提下，通过增加对外接口以支持标准接口协议的方式，直接实现现有应用系统功能模块的封装。

c）对于新建的应用系统，应按本规范的 web service 的定义与封装的要求，直接采用支持 SOA 的开发工具进行封装。

9.4 开发规范

9.4.1 命名规范

9.4.1.1 命名要求

Web service 名称应遵循以下要求：

a）Web service 的方法名称为大小写混合的形式，以小写字母开头，名字中其它单词的首字母以大写字母开头，不宜使用下划线分隔单词。web service 的命名应该能描绘出方法的作用和功能，web service 的名字宜使用动词或动词短语。

b）Web service 的方法参数命名应用一个小写字母开头，后面的单词应用大写字母开头。

c）所有 web service 的方法应有完整的注释，注释内容包含方法功能介绍、参数说明、返回类型说明、例外类型，例如：

示例：

/** * 方法的功能描述。 * @param name 名称 * @return 返回值类型(INT)；返回-1 表示传入的参数存在类型错误或参数不足，* 返回 0 表示操作失败，返回 1 表示操作成功。 */
解释： **@param 格式如下：** @param 参数名 说明

其中，“参数名”是指参数列表内的标识符，而“说明”代表可延续到后续行内的说明文字。一旦遇到新文档标记，就认为前一个说明结束。

@return 格式如下：

@return 说明

其中，“说明”是指返回值的含义。它可延续到后面的行内。

9.4.1.2 命名空间要求

targetNamespace 在食品药品监管信息化工程中的命名为：www.cfda.gov.cn。

9.4.2 WSDL 编写说明

9.4.2.1 编写要求

WSDL 文档利用如下主要元素描述 web service，如表 5 所示：

表 5 WSDL 文档描述元素

元素	定义
<portType>	web service 执行的操作
<message>	web service 使用的消息
<types>	web service 使用的数据类型
<binding>	web service 使用的通信协议

WSDL 文档可包含 service 元素，service 元素可以在一个 WSDL 文档中描述多个 web service 的定义，WSDL 文档也可包含其他元素，例如：extension 元素。WSDL 文档中各主要元素具体描述如下：

a）<portType>元素是最重要的 WSDL 元素，它可描述可被执行的操作，以及相关的消息，<portType> name 属性的值对应 Java 中的类名。

b）<message>元素定义了 web service 函数的参数，＜message＞元素中的每个子元素都对应 Java 中函数的参数。

c）<types>标签定义了 web service 用到的数据类型，这些数据类型用来定义 web service 方法的参数和返回值，数据类型可以是 Java 基本数据类型，也可以通过 XML Schema 语法自定义数据类型。

d）<binding>元素定义了 web service 消息格式和通讯协议的细节。

9.4.2.2 WSDL 实例

以下是 WSDL 文档的简化片段的举例：

示例：

```
    <?xml version="1.0" encoding="UTF-8" standalone="no"?>
    <wsdl:definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tns="http://www.cfda.gov.cn/SimpleService/" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" name="SimpleService"
targetNamespace="http://www.cfda.gov.cn/SimpleService/">
      <wsdl:types>
        <xsd:schema targetNamespace="http://www.cfda.gov.cn/SimpleService/">
```

```
        <xsd:element name="concatRequest">
          <xsd:complexType>
            <xsd:sequence>
                <xsd:element name="s1" type="xsd:string" />
                <xsd:element name="s2" type="xsd:string"></xsd:element>
            </xsd:sequence>
          </xsd:complexType>
        </xsd:element>
        <xsd:element name="concatResponse" type="xsd:string">
        </xsd:element>
      </xsd:schema>
    </wsdl:types>
    <wsdl:message name="concatRequest">
      <wsdl:part element="tns:concatRequest" name="parameters"/>
    </wsdl:message>
    <wsdl:message name="concatResponse">
      <wsdl:part element="tns:concatResponse" name="parameters"/>
    </wsdl:message>
    <wsdl:portType name="SimpleService">
      <wsdl:operation name="concat">
        <wsdl:input message="tns:concatRequest"/>
        <wsdl:output message="tns:concatResponse"/>
      </wsdl:operation>
    </wsdl:portType>
    <wsdl:binding name="SimpleServiceSOAP" type="tns:SimpleService">
      <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
      <wsdl:operation name="concat">
        <soap:operation soapAction="http://www.cfda.gov.cn/SimpleService/concat"/>
        <wsdl:input>
          <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
          <soap:body use="literal"/>
        </wsdl:output>
      </wsdl:operation>
    </wsdl:binding>
    <wsdl:service name="SimpleService">
      <wsdl:port binding="tns:SimpleServiceSOAP" name="SimpleServiceSOAP">
        <soap:address location="http://IP:PORT/axis2/services/SimpleService"/>
      </wsdl:port>
    </wsdl:service>
  </wsdl:definitions>
```

9.4.3 适配器开发规范

食品药品监管信息化工程适用的适配器类型包括：

a）SOCKET（通常也称作"套接字"）通讯方式适配器。

b）HTTP 通讯方式适配器。

c）JMS 通讯方式适配器。

适配器分为长连接和短连接，以及同步通讯方式和异步通讯方式。

通过适配器可以进行：

a）定义 web service：定义 web service 的方法、参数等信息。

b）转换 web service 的输入参数：把节点的输入参数转换为 web service 的方法参数。

c）调用 web service：调用 web service，并输入 web service 对应的参数。

d）打包 web service 的返回结果：把 web service 运行的结果打包返回。