记得录屏

RPC 包含的要素

webservice

协议层：tcp/ip

应用层：http协议

SOAP： http+xml

# 分布式通信框架-webservice分析

# 什么是webservice

webservice也可以叫xml web service webservice, 轻量级的独立的通讯技术

1. 基于web的服务：服务端提供的服务接口让客户端访问
2. 跨平台、跨语言的整合方案

# 为什么要使用webservice

跨语言调用的解决方案

# 什么时候要去使用webservice

电商平台，订单的物流状态。

.net实现的webservice服务接口

# webservice中的一些概念

## WSDL(web service definition language webservice web服务定义语言)

webservice服务需要通过wsdl文件来说明自己有什么服务可以对外调用。并且有哪些方法、方法里面有哪些参数

wsdl基于XML（可扩展标记语言）去定义的

1. 对应一个.wsdl的文件类型
2. 定义了webservice的服务器端和客户端应用进行交互的传递数据和响应数据格式和方式
3. 一个webservice对应唯一一个wsdl文档

## SOAP（simple object access protocal简单对象访问协议）

http+xml

webservice通过http协议发送和接收请求时， 发送的内容（请求报文）和接收的内容（响应报文）都是采用xml格式进行封装

这些特定的HTTP消息头和XML内容格式就是SOAP协议

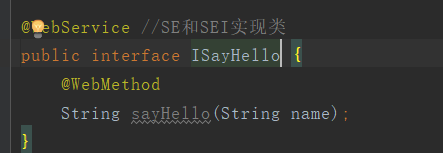
1. 一种简单、基于HTTP和XML的协议
2. soap消息：请求和响应消息
3. http+xml报文

## SEI（webservice endpoint interface webservice的终端接口）

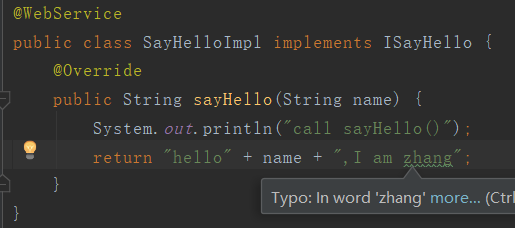
webservice服务端用来处理请求的接口，也就是发布出去的接口。

# 开发一个webservice的实例

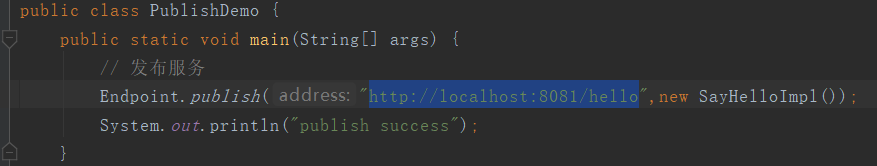
接口



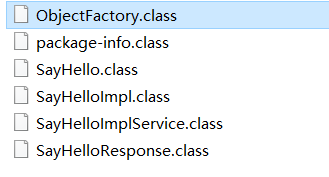
实现类



发布：



生成文件: wsimport <http://localhost:8081/hello> jdk自带工具



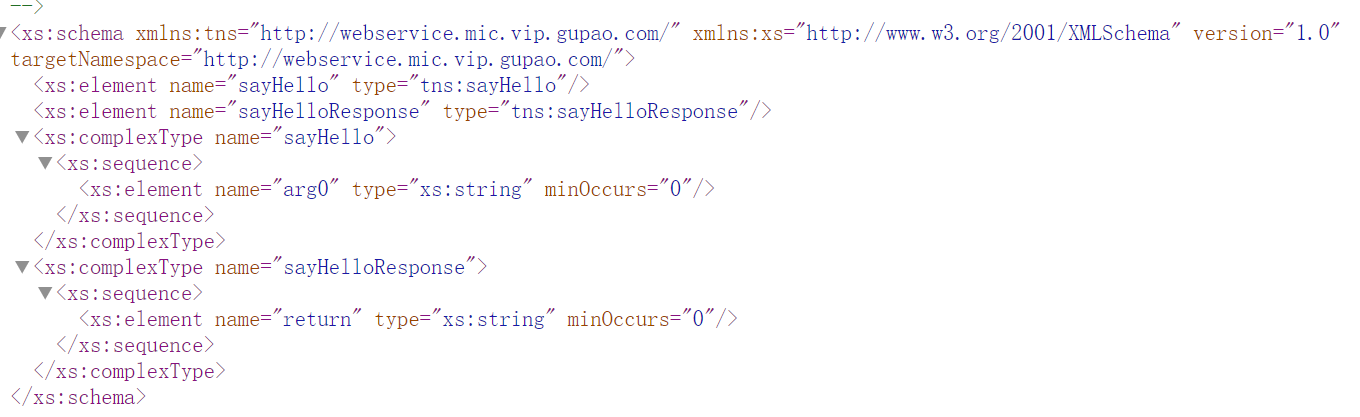
# 分析WSDL文档



## Types标签

定义整服务端的数据报文

### Schema标签



<sayHello>

<arg0>String</arg0>

</sayHello>

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://webservice.mic.vip.gupao.com/">

<soapenv:Header/>

<soapenv:Body>

<web:sayHello>

<!--Optional:-->

<arg0>?</arg0>

</web:sayHello>

</soapenv:Body>

</soapenv:Envelope>

<sayHelloResponse>

<return>string</return>

</sayHelloResponse>

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">

<S:Body>

<ns2:sayHelloResponse xmlns:ns2="http://webservice.mic.vip.gupao.com/">

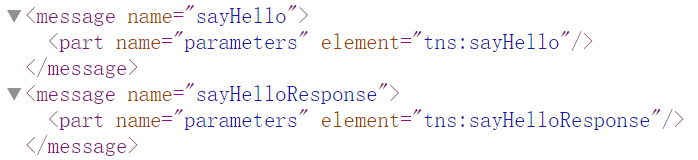
<return>Hello ,Mic,I'am 菲菲</return>

</ns2:sayHelloResponse>

</S:Body>

</S:Envelope>

## Message



定义了在通信中使用的消息的数据结构

## PortType



定义服务器端的SEI

input/output表示输入/输出数据

## Binding标签

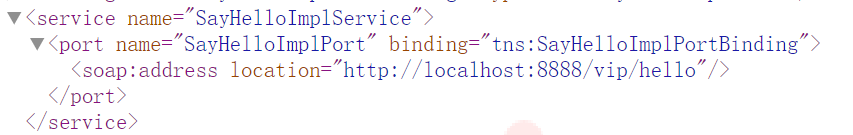


1. type属性： 引用porttype

<soap:binding style=”document”>

1. operation : 指定实现方法
2. input/output 表示输入和输出的数据类型

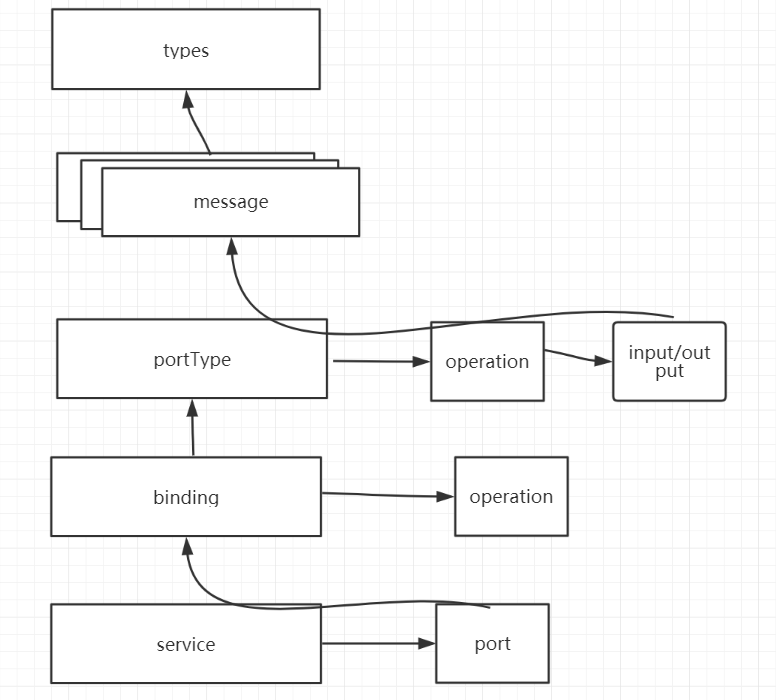
## Service标签

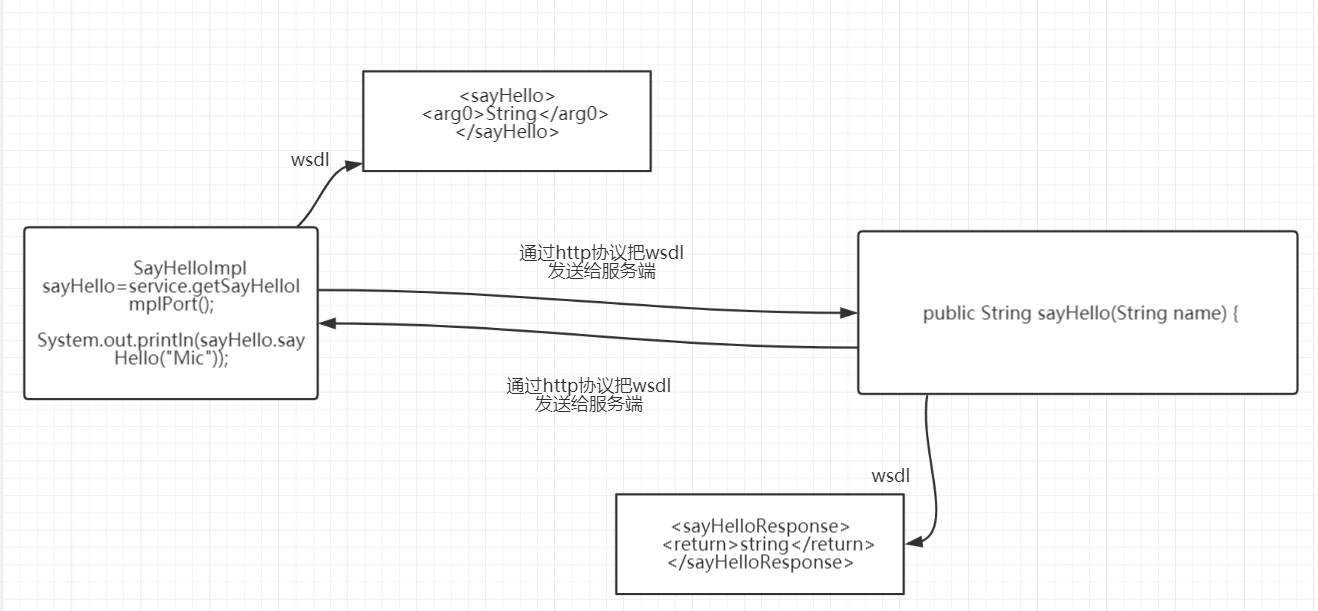


service： 服务器端的一个webservice的容器

name属性： 指定客户端的容器类

address： 当前webservice的请求地址





# Axis/Axis2

apache开源的webservice工具

## CXF

Celtix+Xfire 。 用的很广泛，因为集成到了spring

## Xfire

高性能的Webservice

HTTP+JSON (新的webservice)

HTTP+XML

# spring cxf+REST实现一个webservice服务

springmvc+REST实现的新webservice

linux： centos7; vm可以copy 设置一个备份点

jdk、tomcat

RMI、 http协议/https、webservice、 TCP协议、UDP协议、

socket编程、bio /nio模型、分布式架构、集群、架构演进过程