

WSDL 文档

WSDL 文档仅仅是一个简单的 XML 文档。
它包含一系列描述某个 web service 的定义。

WSDL 文档结构

WSDL 文档是利用这些主要的元素来描述某个 web service 的：

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

一个 WSDL 文档的主要结构是类似这样的：

```
<definitions>

<types>
  data type definitions.....
</types>

<message>
  definition of the data being communicated....
</message>

<portType>
  set of operations.....
</portType>

<binding>
  protocol and data format specification....
</binding>

</definitions>
```

WSDL 文档可包含其它的元素，比如 extension 元素，以及一个 service 元素，此元素可把若干个 web services 的定义组合在一个单一的 WSDL 文档中。

WSDL 端口

<portType> 元素是最重要的 WSDL 元素。
它可描述一个 web service、可被执行的操作，以及相关的消息。
可以把 <portType> 元素比作传统编程语言中的一个函数库（或一个模块、或一个类）。

WSDL 消息

`<message>` 元素定义一个操作的数据元素。

每个消息均由一个或多个部件组成。可以把这些部件比作传统编程语言中一个函数调用的参数。

WSDL types

`<types>` 元素定义 web service 使用的数据类型。

为了最大程度的平台中立性，WSDL 使用 XML Schema 语法来定义数据类型。

WSDL Bindings

`<binding>` 元素为每个端口定义消息格式和协议细节。

WSDL 实例

这是某个 WSDL 文档的简化的片段：

```
<message name="getTermRequest">
  <part name="term" type="xs:string"/>
</message>

<message name="getTermResponse">
  <part name="value" type="xs:string"/>
</message>

<portType name="glossaryTerms">
  <operation name="getTerm">
    <input message="getTermRequest"/>
    <output message="getTermResponse"/>
  </operation>
</portType>
```

在这个例子中，`<portType>` 元素把 "glossaryTerms" 定义为某个端口的名称，把 "getTerm" 定义为某个操作的名称。

操作 "getTerm" 拥有一个名为 "getTermRequest" 的输入消息，以及一个名为 "getTermResponse" 的输出消息。

`<message>` 元素可定义每个消息的部件，以及相关的数据类型。

对比传统的编程，glossaryTerms 是一个函数库，而 "getTerm" 是带有输入参数 "getTermRequest" 和返回参数 getTermResponse 的一个函数。

[← WSDL 简介](#)[WSDL 端口 →](#)[✍ 点我分享笔记](#)