

# 创信 WS 接口开发说明

文件状态： <input type="checkbox"/> 草稿 <input checked="" type="checkbox"/> 正式发布 <input type="checkbox"/> 正在修改		文件标识：	创信 WS 接口开发说明		
		当前版本：	1.2		
		作 者：	赵强		
		完成日期：	2015-11-12		
日期	完成人	版本	修改内容	备注	
2015-10-27	赵强	1.0	新建		
2015-11-12	赵强	1.2	回调接口里面加CodeMsg		

## 目录

1	文档介绍	3
1.1	文档范围	3
1.2	读者对象	3
1.3	术语与缩写解释	3
2	目的	3
3	规范说明	3
3.1	访问地址	3
3.2	主界面	3
3.3	通用说明	4
3.3.1	XML 调用方法	4
3.3.2	JSON 调用方式	4
3.3.3	InputXML 参数说明 ( XML )	6
3.3.3.1.1	参数举例：	6
3.3.4	具体说明：	6
3.3.5	返回结果 ( XML )	6
3.3.6	InputXML 参数说明 ( JSON )	7
3.3.6.1.1	参数举例	7
3.3.6.1.2	具体说明	7
3.3.6.1.3	返回结果 ( JSON )：	7
3.3.6.1.4	举例说明	7
3.3.6.1.4.1	成功举例 ( 多表数据 )：	7
3.3.6.1.4.2	成功举例 ( 没有数据 )：	8
3.3.6.1.4.3	成功举例 ( 没有数据 1 )：	8
3.3.6.1.4.4	错误举例：	8
3.3.6.1.5	详细说明：	9
3.4	流量派发接口	9
3.4.1	接口说明	9
3.4.2	传入案例	10
3.4.3	参数特别说明	11
3.4.4	结果返回	11
3.5	派发超时查询接口	13
3.5.1	接口说明	13
3.5.2	传入案例	13
3.5.3	结果返回	14

1 文档介绍

1.1 文档范围

成都创信信息技术有限公司 WSRR 接口说明。

1.2 读者对象

WSRR 开发相关人员，

1.3 术语与缩写解释

缩写、术语	解 释
WS	Web Service
XML	Extensible Markup Language 可扩展标记语言
JSON	JavaScript Object Notation 常用的轻量级的数据交换格式
WSRR	Web Service Registry and Repository(WS 服务注册与存储)
...	

2 目的

3 规范说明

3.1 访问地址

正式：<http://120.26.72.15:9001/WSRR.aspx>

3.2 主界面



### 3.3 通用说明

#### 3.3.1 XML 调用方法

常规 WSDL 调用方法。

传入参数：XmlInput

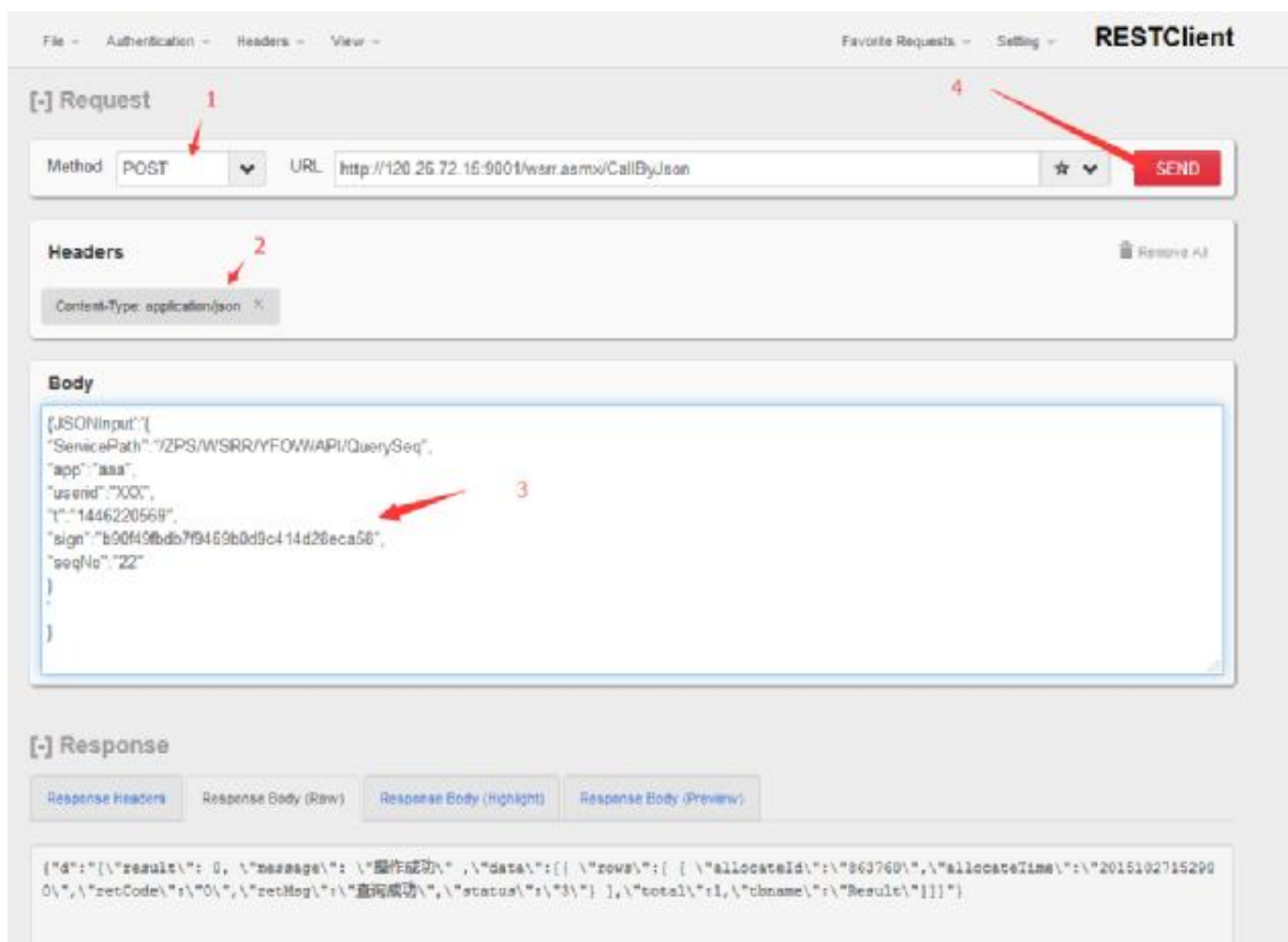
如 C# 中，添加 WEB 服务，并按普通类方法进行操作。

#### 3.3.2 JSON 调用方式

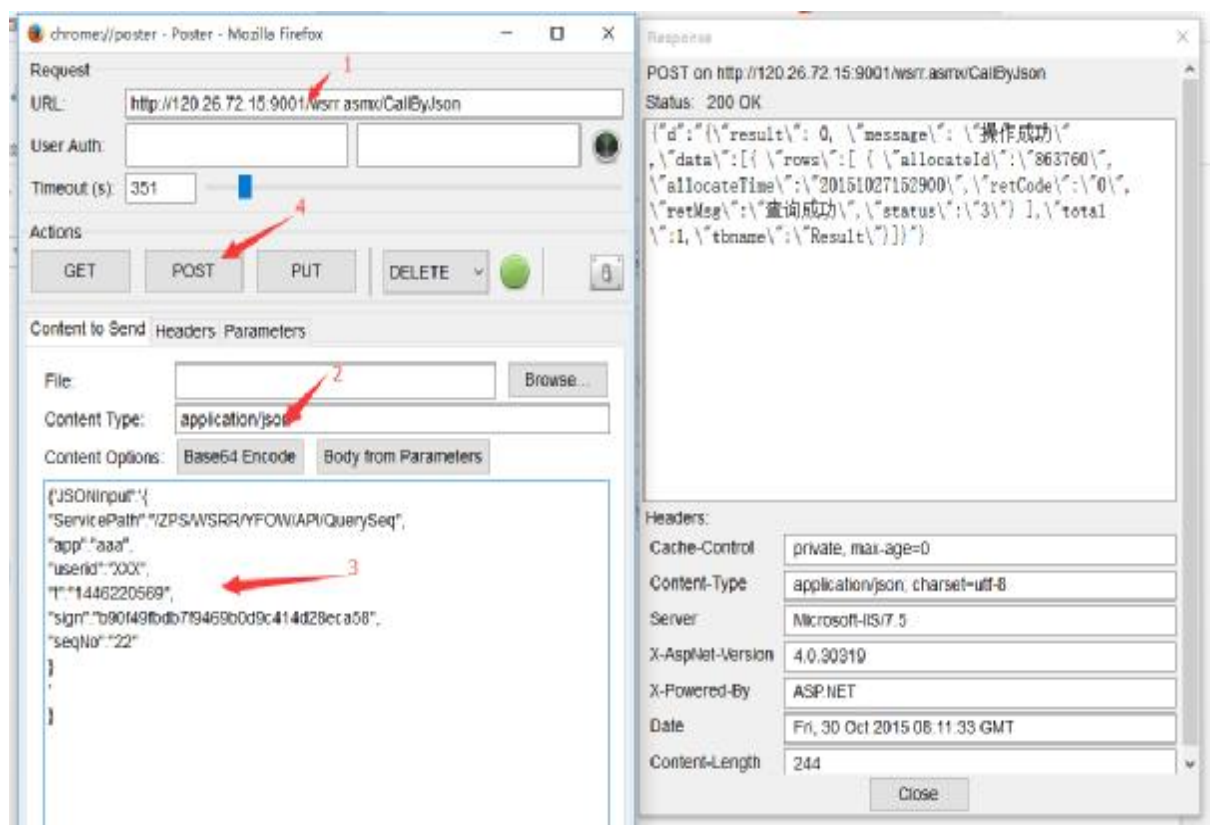
以 Http 的请求方式，Content-Type 必须设置为：application/json，以 Post 方式提交，并传入参数 JsonInput（Json 格式）。

调用方法简单举例：

使用火狐的 RESTClient 组件



使用火狐的 Poster 组件



使用 JQuery 的 Ajax 代码：

```
var objData = {
    "ServicePath": "/ZPS/WSRR/YFOW/API/QuerySeq",
    "app": "aaa",
    "userid": "XXX",
    "t": "1446220569",
    "sign": "b90f49fbd7f9469b0d9c414d28eca58",
    "seqNo": "22"
};

$.ajax({
    type: "POST",
    contentType: "application/json",
    url: "http://120.26.72.15:9001/WSRR.asmx/CallByJson",
    data: "{ 'JsonInput': '" + JSON.stringify(objData) + "' }",
    dataType: 'json',
    success: function (result) {
        var result = eval("[ "+result.d+" ]");
    }
});
```

### 3.3.3 InputXML 参数说明 ( XML )

#### 3.3.3.1.1 参数举例：

```
<Interface ServicePath="/"ZPS/WSRR/WebService/Platform " ">
  <userid>XXX</userid>
  <mobile> XXXXXXXXXXXXX</mobile>
</Interface>
```

### 3.3.4 具体说明：

#### 基本信息：

Interface：为必须节点，固定格式  
WSID：为必须节点，固定格式。  
APP：为必须节点，固定格式  
DeviceID：必须节点可以为空，固定格式  
SessionID：必须节点可以为空  
User：必须节点可以为空  
Pwd：必须节点可以为空，以 DES 加密得出

#### XML 参数信息：

Root 为必须节点，固定格式  
List：为必须节点，固定格式。  
Rows 及以下：Rows 以下的内容为列名。Rows 可以出现多次，也就是多行记录，可以有多个 Rows。  
如果是多行记录，每一个 Rows 里面的列必须全部相同。列的内容请参考接口表结构。

### 3.3.5 返回结果 ( XML )

#### 案例：

TODO:

```
<ROOT>
  <Result>0</Result>
  <Message>成功！</Message>
  <LIST TBName="Result">
    <Rows>

    </Rows>
  </LIST>
</ROOT>
```

说明：Result 指示结果是否成功。0 表示成功，1 表示失败。如果失败，会在 Message 告知失败原因，本次的所有数据将不会更新，请重新调用。

### 3.3.6 InputXML 参数说明 ( JSON )

#### 3.3.6.1.1 参数举例

```
{
  "ServicePath":"/ZPS/WSRR/WebService/Platform ",
  "app":"evmandroid",
  "optype":"GetData",
  "tbname":"Value",
}
```

#### 3.3.6.1.2 具体说明

整体：所有关键字全部小写

ServicePath：为必须节点，固定格式。

app：为必须节点，固定值（evmandroid）

sessionid：可以为空，固定格式

data:可以为空，固定格式。如果有值，里面为 JSON 数组

data 对象说明：

- 1、整体为 JSON 数组
- 2、包含的每一个对象可以不一样，但是必须包含 rows 节点，不然会被认为无效
- 3、单个对象说明：
  - a) optype：是指操作说明，后台会根据这个字段来进行判断所需要的操作
  - b) tbname:指示这个对象的名称，**所有对象名称不可以重复**
  - c) rows:数组对象，里面的内容对象可以为多个，可以理解为一个表的多行数据，但是**每一行数据结构必须相同**：如

```
"rows":[ { "VIP_ID":"112AB3EE-C1C8-41F3-B056-1C743608F251"},{ "VIP_ID":"112AB3E
E-C1C8-41F3-B056-1C743608F251"}]
```

不允许：

```
"rows":[ { "VIP_ID":"112AB3EE-C1C8-41F3-B056-1C743608F251"},{ "CODE":"212AB3EE
-C1C8-41F3-B056-1C743608F263"}]
```

- 4、当 data 里面没有任何列或者没有数据时可以允许以下格式：

```
"data":[{"total": 0, "rows": []}]
```

#### 3.3.6.1.3 返回结果 ( JSON ):

#### 3.3.6.1.4 举例说明

##### 3.3.6.1.4.1 成功举例 ( 多表数据 ):

```
{
  "result": 0,
```

```
"message": "操作成功",
"data": [
  {
    "rows": [
      {
        "ID": "S0001",

        "Message": "测试 1"
      }, {
        "ID": "S0002",

        "Message": "测试 1"
      }
    ],
    "total": 2,
    "tbname": "Table"
  },
]
}
```

#### 3.3.6.1.4.2 成功举例（没有数据）：

```
{
  "result": 0,
  "message": "操作成功",
  "data": [
    {
      "total": 0,
      "rows": []
    }
  ]
}
```

#### 3.3.6.1.4.3 成功举例（没有数据 1）：

```
{"result": 0, "message": "操作成功" }
```

#### 3.3.6.1.4.4 错误举例：

```
{ "result": 1, "message": "用户 ID 不能为空，请核对！" }
```



**3.3.6.1.5 详细说明：**

result：固定格式。结果值：int 类型：0:表示成功，1：失败

message：固定格式。操作结果文字描述。如果失败，会提示错误信息

data：固定格式。JSON 数组，如果没有数据返回，可能为空。

data 详细说明:

rows:固定格式，与 InputXML 的 rows 一样，JSON 数组，并且每一行数据格式相同

total:根据后果实际情况返回，一般是返回本次数据行数，如果有分页的话，会返回表里的数据总行。

tbname：指示这个对象的名称，所有对象名称不会重复

**3.4 流量派发接口****3.4.1 接口说明**

接口参数：

参数	名称	说明
userid	用户 ID	用户 id
mobile	手机号码	必填，string
flow	派发流量	( 选填，int，单位 M ) 在平台中指定为随机派发流量或 单个号码派发流量后不需要该参数
round	流量匹配原则	( 必填，string，取值范围：up 向上接近，down 向下接近，no 没有匹配到的流量大小不进行充值，直接返回错误 ) 由于运营商只提供了几个固定流量大小的套餐包和价格，所以并不能派发任意金额的流量，只能取一个最接近的
exclusive	排除某些运营商的号码	( 选填，string，取值 1 电信、2 移动、3 联通，多个值用逗号分隔 ) 如果该号码属于被排除的运营商则不对其进行流量派发
callback	回调地址	( 选填，string ) 客户提供的回调接口 URL 地址，需要进行 utf-8 格式的 url encode 编码
t	请求时间戳	( 必填，int ) 从 Unix 纪元( 格林威治时间 1970 年 1 月 1 日 00:00:00 ) 到当前时间的秒数
sign	签名串	( 必填，string ) md5(apikey+t+mobile)，"+" 表示字符串连接，apikey 为双方约定的私有串
msg	是否指定短信	( 选填，string ) 0 不指定，使用默认短信模板；1 指定短信，需要填写参数 msgId 和 msgSign

msgId	短信 id	( 选填, string ) 例如: 747413。
msgSign	短信模板签名	( 选填, string ) 例如: 创信科技, 需要进行 utf-8 格式的 url encode 编码
seqNo	流水号	客户提供的充值流水号

回调请求说明:

URL 地址: http://xxxxx/xxxxx ( 由客户提供 )

URL 参数:

code          状态编码          ( 必填, string )

allocateId    流量派发流水 id ( 必填, string )

allocateTime    派发时间          ( 必填, string, 格式 yyyyMMddHHmmss )

seqNo          客户提供的充值流水号      ( 选填, string )

mobile        手机号码              ( 选填, string )

codeMsg        充值状态说明              ( 选填, string )

返回值:

固定返回 1, 判断客户是否收到结果

注: 将流量派发结果通过客户提供的回调地址传给客户, 如客户没有提供该回调地址, 则由客户自行登陆我们平台查询充值结果。

code 参数取值:

code 值	说明
0	已派发成功
10	派发失败, 请登陆平台网站查询

### 3.4.2 传入案例

JSON格式:

```
{
  "ServicePath":"/ZPS/WSRR/YFOW/API/Allocate",
  "app":"aaa",
  "userid":"XXX",
  "mobile":"XXXXXXXXXXXX",
  "flow":"1",
  "round":"up",
  "exclusive":"1",
  "callback":"",
  "t":"1446139447",
```

```
"sign":"07c568e980eb51b4344a8f03bcc713df",
"msg":"0",
"msgId": "",
"msgSign": "",
"seqNo": "25"
}
```

XML 格式：

```
<Interface ServicePath="/ZPS/WSRR/YFOW/API/Allocate" App="aaa" >
  <userid>XXX</userid>
  <mobile>XXXXXXXXXXXX</mobile>
  <flow>2</flow>
  <round>up</round>
  <exclusive>1</exclusive>
  <callback/>
  <t>1446139447</t>
  <sign>07c568e980eb51b4344a8f03bcc713df </sign>
  <msg>0</msg>
  <msgId/>
  <msgSign/>
  <seqNo>21</seqNo>
</Interface>
```

### 3.4.3 参数特别说明

详见参数说明

### 3.4.4 结果返回

以 JSON 格式调用返回格式：

#### 失败

```
{ "result": 1, "message": "13 流水号重复" }
```

#### 成功

```
{
  "result": 0,
  "message": "操作成功",
  "data": [{
    "rows": [ {
      "allocateId": "906064",
      "retCode": "0",
      "retMsg": "受理成功"
    }
  ]
}
```

```
        }],
        "total":1,
        "tbname":"Result"
    ]}
}
```

以 XML 格式调用返回格式

失败

```
<Root>
  <Result>1</Result>
  <Message>13 流水号重复</Message>
</Root>
```

成功

```
<Root>
  <Result>0</Result>
  <Message>操作成功</Message>
  <List TBName="Result" RowCount="1">
    <Rows>
      <allocateId>905945</allocateId>
      <retCode>0</retCode>
      <retMsg>受理成功</retMsg>
    </Rows>
  </List>
</Root>
```

接口调用返回说明：

Result 属性值取值：

Result 值	Message 值	说明
0	操作成功	操作成功
1	异常信息	(retCode+ retMsg)参数，网络，授权等异常信息

retCode 属性值取值：

retCode 值	说明
0	已成功加入派发队列
10	派发失败，余额不足，请前往平台充值
11	派发失败，已达派发限额，请前往平台重新创建派发规则
12	该号码属于被排除运营商
13	流水号重复

14	没有匹配到的流量大小
99	系统异常，异常说明从 retMsg 中获取

备注：操作成功会多返回如下数据

属性名称	说明
retCode	状态码
retMsg	状态说明
allocateId	为本次派发流水 id

3.5 派发超时查询接口

3.5.1 接口说明

取数条件：

认证方法：

接口参数：

参数	名称	说明
userid	用户 ID	用户 id
t	请求时间戳	( 必填 , int ) 从 Unix 纪元( 格林威治时间 1970 年 1 月 1 日 00:00:00 ) 到当前时间的秒数
sign	签名串	( 必填 , string ) md5(apikey+t+seqNo) , “+” 表示字符串连接 , apikey 为双方约定的私有串
seqNo	流水号	客户提供的充值流水号

3.5.2 传入案例

JSON格式：

```
{
  "ServicePath":"/ZPS/WSRR/YFOW/API/QuerySeq",
  "app":"aaa",
  "userid":"XXXX",
  "t":"1445960287",
  "sign":"27b7ede7fa59ae323a1830549a92bece",
  "seqNo":"22"
}
```

XML 格式：

```
<Interface ServicePath="/ZPS/WSRR/YFOW/API/QuerySeq" App="aaa" >
  <userid>XXXX</userid>
  <t>1446212761</t>
  <sign>797f37d0cbbc5dac035580f5d99498b8</sign>
  <seqNo>22</seqNo>
</Interface>
```

参数特别说明

详见参数说明

### 3.5.3 结果返回

以 JSON 格式调用接口：

失败：

```
{ "result": 1, "message": "99 IP 地址非法" }
```

成功：

```
{
  "result": 0,
  "message": "操作成功",
  "data": [{
    "rows": [ {
      "allocateId": "863760",
      "allocateTime": "20151027152900",
      "retCode": "0",
      "retMsg": "查询成功",
      "status": "3"
    } ],
    "total": 1,
    "tbname": "Result"
  }]
}
```

以 XML 格式调用接口：

失败：

```
<Root>
  <Resul t>1</Resul t>
  <Message>签名串错误，请核对！</Message>
</Root>
```

成功：

```
<Root>
```

```

<Result>0</Result>
<Message>操作成功</Message>
<List TBName="Result" RowCount="1">
  <Rows>
    <allocateId>863760</allocateId>
    <allocateTime>20151027152900</allocateTime>
    <retCode>0</retCode>
    <retMsg>查询成功</retMsg>
    <status>3</status>
  </Rows>
</List>
</Root>

```

Result 属性值取值：

Result 值	Message 值	说明
0	操作成功	操作成功
1	异常信息	(retCode+ retMsg)参数，网络，授权等异常信息

retCode 属性值取值：

retCode 值	说明
0	查询成功
其他	查询异常，异常说明从 retMsg 中获取

status 流量派发状态取值：

status 值	说明
0	未找到派发记录
1	任务已接受，准备派发中
2	流量派发中
3	派发成功
4	派发失败

备注：操作成功会多返回如下数据

属性名称	说明
retCode	查询状态码
retMsg	查询状态说明
status	流量派发状态说明
allocateId	为本次派发流水 id

allocateTime	派发时间
--------------	------