

# 天气预报WEB服务 接口说明

www.webxml.com.cn

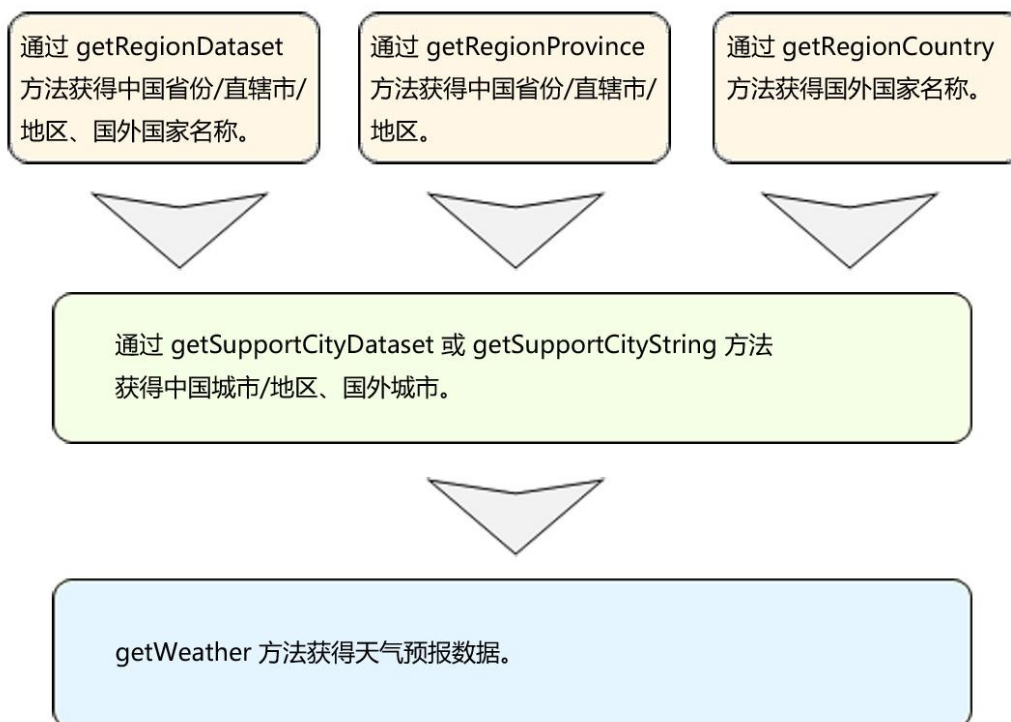
**Endpoint:** <http://ws.webxml.com.cn/WebServices/WeatherWS.asmx>

**Disco:** <http://ws.webxml.com.cn/WebServices/WeatherWS.asmx?disco>

**WSDL:** <http://ws.webxml.com.cn/WebServices/WeatherWS.asmx?wsdl>

## 简要说明:

- 为让更多的开发人员学习WEB服务开发，此服务支持免费用户使用（说明：免费用户24小时内查询不超过50次并且获取二次数据大于间隔 600ms），[注册会员](#)可获得试用。
- 为支持多种平台开发，此WEB服务接口提供了多种返回类型可选择，包括.NET的 DataSet 和 一维数组。
- 开发流程：



## getRegionDataset 方法

获得中国省份、直辖市、地区；国家名称（国外）和与之对应的ID

输入参数：无，返回数据：DataSet

此 DataSet("getRegion") 包含二个表：1. Tables("Province") 中国省份、直辖市、地区表； 2. Tables("Country") 外国国家表。

Item("RegionName") 类型 String，为省份、直辖市、地区或国外国家名称；

Item("RegionID") 类型 String 为对应的 ID。

### 提示和推荐：

- 因为此数据不常变化，请缓存此数据以提高访问速度。
- .NET开发者可以直接把DataSet保存为本地XML数据。请看下面示例：

'VB

```
Public Shared Sub WriteXmlToFile(ByVal filePath As String, ByVal thisDataSet As DataSet)
    If thisDataSet Is Nothing Then
        Return
    End If
    Ds.WriteXml(filePath, XmlWriteMode.WriteSchema)
End Sub
```

//C#

```
public static void WriteXmlToFile(string filePath, DataSet thisDataSet)
{
    if (thisDataSet == null) { return; }
    Ds.WriteXml(filePath, XmlWriteMode.WriteSchema);
}
```

## getRegionProvince 方法

获得中国省份、直辖市、地区和与之对应的ID

输入参数：无，返回数据：一维字符串数组 String()

结构：用英文逗号分割的省份（直辖市、地区）和对应ID。示例：

```
' VB
```

```
Array(0) = "上海, 31112"
```

```
.....
```

```
.....
```

```
Array(n) = "城市/地区, ID"
```

```
//C#
```

```
Array[0] = "上海, 31112";
```

```
.....
```

```
.....
```

```
Array(n) = "城市/地区, ID";
```

### 提示和推荐：

- 因为此数据不常变化，请缓存此数据以提高访问速度。

## getRegionCountry 方法

获得国外的国家名称和与之对应的ID

输入参数：无，返回数据：一维字符串数组 String()

结构：用英文逗号分割的国外国家名称和对应ID。

```
' VB
```

```
Array(0) = "美国, 3460"
```

```
.....
```

```
.....
```

```
Array(n) = "国家名称, ID"
```

```
//C#
```

```
Array[0] = "美国, 3460";
```

```
.....
```

```
.....
```

```
Array(n) = "国家名称, ID";
```

### 提示和推荐:

- 因为此数据不常变化，请缓存此数据以提高访问速度。

## getSupportCityDataset 方法

获得本天气预报WEB服务支持的城市/地区名称和与之对应的ID

输入参数： 参数名： theRegionCode ， 类型： 字符串， 输入内容可以为以下数据之一（可通过 getRegionDataset、getRegionProvince、getRegionCountry 方法获得）：

1. 国外的国家名称（中文名称）
2. 国外的国家 ID
3. 中国的省份、直辖市、地区名称（名称请不要包含“省”、“市”、“自治区”等，如：四川）
4. 中国的省份、直辖市、地区 ID
5. 参数为空字符串时默认“上海”

返回数据： DataSet

此 DataSet("Region") 包含一个表， Tables("City")

Item("CityName") 类型 String， 城市或地区名称； Item("CityID") 类型 Int32 为对应的 ID。

### 提示和推荐：

- 因为此数据不常变化，请缓存此数据。
- .NET开发使用，可以直接把DataSet保存为本地XML数据，也可以在WEB开发的时候使用 HttpContext.Current.Cache 来缓存此数据，以提高访问速度。.NET开发使用CACHE缓存数据请看下面示例：

'VB

```
Public Shared Function GetDataset(ByVal RegionID As String) As DataSet
    Dim CacheName As String = "Weather_City_" & RegionID.Trim()
    Dim Ds As DataSet = CType(HttpContext.Current.Cache(CacheName), DataSet)
    If Ds Is Nothing Then
        Dim getWebserviceDataset As DataSet = getSupportCityDataset(RegionID) '通过 WEB SERVICE
        getSupportCityDataset 方法获得支持的城市或地区名称和ID
        Ds = getWebserviceDataset
        SyncLock CacheName
            HttpContext.Current.Cache.Insert(CacheName, Ds, Nothing, DateTime.MaxValue,
            TimeSpan.FromMinutes(30), CacheItemPriority.NotRemovable, Nothing) '这里设置缓存：从被访问后
            缓存30分钟数据
        End SyncLock
    End If
    Return Ds
End Function
```

//C#

```
public static DataSet GetDataset(string RegionID)
```

```
{
    string CacheName = "Weather_City_" + RegionID.Trim();
    DataSet Ds = (DataSet)HttpContext.Current.Cache(CacheName);
    if (Ds == null) {
        DataSet getWebserviceDataset = getSupportCityDataset(RegionID); //通过 WEB SERVICE
        getSupportCityDataset 方法获得支持的城市或地区名称和ID
        Ds = getWebserviceDataset;
        lock (CacheName) {
            HttpContext.Current.Cache.Insert(CacheName, Ds, null, DateTime.MaxValue,
            TimeSpan.FromMinutes(30), CacheItemPriority.NotRemovable, null); //这里设置缓存: 从被访问后缓
            存30分钟数据
        }
    }
    return Ds;
}
```

- 推荐使用国家或省份的ID为参数查询，如文件编码和WEB服务编码不一致（WEB服务文件编码为 UTF-8）。在GET或POST等方法时，使用中文文字作为参数的话可能不能正确传递参数查询，这需要对中文参数进行URL编码，JAVA、JavaScript、.NET示例如下：

```
//JavaScript
```

```
//如果您的网页文件编码是UTF-8可以使用: escape() 方法, 如果网页文件编码不是UTF-8请使用:
```

```
encodeURIComponent() 方法或 encodeURIComponent() 方法
```

```
encodeURIComponent("参数")
```

```
'VB
```

```
System.Web.HttpUtility.UrlEncode("参数", System.Text.Encoding.UTF8)
```

```
//C#
```

```
System.Web.HttpUtility.UrlEncode("参数", System.Text.Encoding.UTF8);
```

```
//JAVA
```

```
java.net.URLEncoder.encode("参数", "UTF-8");
```

## getSupportCityString 方法

获得支持的城市/地区名称和与之对应的ID

输入参数：同 getSupportCityDataset 方法

返回数据：一维字符串数组 String()。结构：用英文逗号分割的城市或地区名称和对应ID。

如查询结果为空，输出以下结果：

```
'VB
```

```
Array(0) = "无城市,000000"
```

```
//C#
```

```
Array[0] = "无城市,000000";
```

### 提示和推荐：

- 因为此数据不常变化，请缓存此数据。

## getWeather 方法

获得天气预报数据，（输入有效 theUserID 用户ID参数可获得七天的天气预报数据）

输入参数： 参数名： theUserID 用户ID，免费用户填写空字符串 theUserID = ""。参数名： theCityCode ， 类型： 字符串，输入内容可以为以下数据之一（可通过 getSupportCityDataset 或 getSupportCityString 方法获得）：

1. 城市或地区中文名称
2. 城市或地区 ID
3. 参数为空字符串默认“上海”

返回数据： 一维字符串数组 String()，结构如下：

```
Array(0) = "省份 地区/洲 国家名（国外）"  
Array(1) = "查询的天气预报地区名称"  
Array(2) = "查询的天气预报地区ID"  
Array(3) = "最后更新时间 格式： yyyy-MM-dd HH:mm:ss"  
Array(4) = "当前天气实况： 气温、风向/风力、湿度"  
Array(5) = "第一天 空气质量、紫外线强度"  
Array(6) = "第一天 天气和生活指数"  
Array(7) = "第一天 概况 格式： M月d日 天气概况"  
Array(8) = "第一天 气温"  
Array(9) = "第一天 风力/风向"  
Array(10) = "第一天 天气图标 1"  
Array(11) = "第一天 天气图标 2"  
Array(12) = "第二天 概况 格式： M月d日 天气概况"  
Array(13) = "第二天 气温"  
Array(14) = "第二天 风力/风向"  
Array(15) = "第二天 天气图标 1"  
Array(16) = "第二天 天气图标 2"  
.....  
..... 每一天的格式同： Array(12) -- Array(16)  
.....  
Array(n-4) = "最后一天 概况 格式： M月d日 天气概况"  
Array(n-3) = "最后一天 气温"  
Array(n-2) = "最后一天 风力/风向"  
Array(n-1) = "最后一天 天气图标 1"  
Array(n) = "最后一天 天气图标 2"
```

如查询结果为空，输出以下结果：

```
'VB
```



```
Array(0) = "查询结果为空"
```

```
//C#
```

```
Array[0] = "查询结果为空";
```

如出现错误将会有以下提示:

```
Array(0) = "发现错误: 用户验证失败"
```

```
Array(0) = "发现错误: 免费用户不能使用高速访问"
```

```
Array(0) = "发现错误: 免费用户24小时内访问超过规定数量"
```

输入有效 `theUserID` 参数可以使用 `getWeather` 接口获得七天的天气预报数据（暂时只支持中国城市，国外城市五天数据）。

#### 提示和推荐:

- 此数据 2.5 小时左右更新一次，如访问量比较大可按访问情况缓存或保存数据（可使用数据库）
- 请尽可能使用城市ID作为参数调用，以避免相同名称城市和参数传递编码问题（可参考上面 `getSupportCityDataset` 方法的提示）
- 使用名称查询如遇城市/地区名称相同只显示默认：（已发现相同地名，左边第一个为默认，）

```
东兴（广西省 防城港）、东兴（四川省 内江）
```

```
河口（山东省 东营）、河口（云南省 昆明）
```

```
松江（上海市）、松江（吉林省 延吉）
```

```
清水河（青海省 果洛）、清水河（内蒙古 呼和浩特）
```

```
东乡（江西省 抚州）、东乡（甘肃省 临夏）
```

**相关链接:**

部分JAVA开发工具在调用由.NET开发的WEB服务时WSDL出错请访问以下链接:

<http://www.webxml.com.cn/h/6C533855614658756258513D.aspx>

Web Services GET 方法调用:

<http://www.webxml.com.cn/h/7A636E424F763363446B303D.aspx>

Java调用.Net WebService 发生异常: 服务器未能识别 HTTP 头 SOAPAction 的值:

<http://www.webxml.com.cn/h/5A416E4C4F6A31526D4D453D.aspx>

JAVA读 .NET开发的WEB服务 DataSet 数据可参考以下链接:

<http://download.csdn.net/source/698391> ,

<http://www.webxml.com.cn/d/44587637472B4E426377453D.aspx>,

<http://www.webxml.com.cn/h/4F65627450716E5A6832493D.aspx>

C# 转 VB.net: <http://labs.developerfusion.co.uk/convert/csharp-to-vb.aspx>

VB.net 转 C#: <http://labs.developerfusion.co.uk/convert/vb-to-csharp.aspx>

服务支持 [http://www.webxml.com.cn/zh\\_cn/support.aspx](http://www.webxml.com.cn/zh_cn/support.aspx)

下载中心 [http://www.webxml.com.cn/zh\\_cn/download\\_center.aspx](http://www.webxml.com.cn/zh_cn/download_center.aspx)

天气图标下载: <http://www.webxml.com.cn/images/weather.zip>

天气现象和图例: [http://www.webxml.com.cn/zh\\_cn/weather\\_icon.aspx](http://www.webxml.com.cn/zh_cn/weather_icon.aspx)

部分城市/地区代表图片: [http://www.webxml.com.cn/files/city\\_photo.zip](http://www.webxml.com.cn/files/city_photo.zip)

部分城市/地区介绍和气候背景资料: [http://www.webxml.com.cn/files/about\\_city.zip](http://www.webxml.com.cn/files/about_city.zip)

以上相关链接内的资料内容均来自互联网,我们不对此数据、资料的完整性、真实性和可靠性负责,请您先核对后再使用。如果数据、资料侵犯了您的权益或版权,请立即联系我们:

[http://www.webxml.com.cn/zh\\_cn/contact\\_us.aspx](http://www.webxml.com.cn/zh_cn/contact_us.aspx) ,我们将立即删除改正。

除访问 <http://www.webxml.com.cn/> 网站外,链接到第三方的Web网站仅作为一种方便服务提供给您。如果您决定访问任何第三方链接的网站,由您自己承担风险,且负责采取所有保护措施防止病毒或其它破坏性因素。对任何链接的网站或其上出现的信息及所述产品和服务,或者通过使用它们可能获得的任何结果我们不作担保或说明,也不表示认同。

声明和使用条款: [http://www.webxml.com.cn/zh\\_cn/notice.aspx](http://www.webxml.com.cn/zh_cn/notice.aspx)

版权所有 © 上海思集信息科技有限公司保留所有权利