

Arc en Ciel Ltd.

## Address Fulfilment Webservice FactSheet

### Overview

Our address fulfilment webservice provides a facility whereby an application or a webpage can send a postcode to our server and get back the address details for the postcode.

To see how it works, simply paste the following link into the address field of your browser (you'll have to do it in two parts). The clearest results are given in Internet Explorer, as Firefox and Google Chrome don't display XML tags:

[www.addressgeo.co.uk/perclick/lookup.aspx?  
function=getroadaddress&user=test&password=password&postcode=cb11du](http://www.addressgeo.co.uk/perclick/lookup.aspx?function=getroadaddress&user=test&password=password&postcode=cb11du)

You will see the result:-

```
<?xml version="1.0" encoding="utf-8" ?>
  <Result>
    <Label>Prospect Row;CAMBRIDGE;CB1 1DU</Label>
    <STR>Prospect Row</STR>
    <TWN>Cambridge</TWN>
    <CTY>(Cambridgeshire)</CTY>
    <PCD>CB1 1DU</PCD>
  </Result>
```

It's easy to automate this in your website code so that the user types their postcode into a text box and their address appears, needing only the addition of a house number to complete.

The above example is for a road-level lookup. If you need full details of the premises at a postcode, simply substitute "function=getpremiseaddress" for "function=getroadaddress" above. You will then see:-

```
<?xml version="1.0" encoding="utf-8" ?>
  <Result>
    <Label>Prospect Row;CAMBRIDGE;CB1 1DU</Label>
    <NUM_R>1A;2;3;4;5;6</NUM_R>
    <ORG>Company Hair Ltd;The Free Press</ORG>
    <NUM_N>1;7</NUM_N>
    <STR>Prospect Row</STR>
    <TWN>Cambridge</TWN>
    <CTY>(Cambridgeshire)</CTY>
    <PCD>CB1 1DU</PCD>
  </Result>
```

Now you also have a list of all premise numbers (separated into residential and non-residential) and all organisation names at the postcode. With a little extra coding a drop-down list can be filled so the user can select their house number or organisation. This usage is especially valuable if your website serves businesses, as it saves a lot of time in typing a business name. It's also useful when data cleanliness is important as it ensures that a house number is valid for that postcode.

On very rare occasions (about 1 case in 2500) a postcode will cover more than one road. The URLs used above will return multiple <Result> tags as below:-

```
<?xml version="1.0" encoding="UTF-8"?>
<Results>
  <Result>
    <Label>New Hey;Sandfield Park;LIVERPOOL;L12 1LH</Label>
    <STR>New Hey</STR>
    <DLO>Sandfield Park</DLO>
    <TWN>Liverpool</TWN>
    <CTY>(Merseyside)</CTY>
    <PCD>L12 1LH</PCD>
  </Result>
  <Result>
    <Label>South Drive;Sandfield Park;LIVERPOOL;L12 1LH</Label>
    <STR>South Drive</STR>
    <DLO>Sandfield Park</DLO>
    <TWN>Liverpool</TWN>
    <CTY>(Merseyside)</CTY>
    <PCD>L12 1LH</PCD>
  </Result>
</Results>
```

A common requirement on websites is to find the map reference for a postcode. This can be used for example to organise deliveries or to advise on local items for sale. To get this information use "function=getpostzon" which for postcode CB1 1DU gives the result:-

```
<?xml version="1.0" encoding="utf-8" ?>
<Result>
  <LNG> 0.130118</LNG>
  <LAT> 52.204877</LAT>
  <GRE>545656</GRE>
  <GRN>258437</GRN>
  <CCD>E10000003</CCD>
  <CNM>Cambridgeshire</CNM>
  <DCD>E07000008</DCD>
  <DNM>Cambridge</DNM>
  <WCD>E05002709</WCD>
  <WNM>Market</WNM>
  <NHS>E18000006</NHS>
  <SCD>55131</SCD>
  <PCD>CB1 1DU</PCD>
</Result>
```

In this result, the first four fields give the map reference of the centre of the postcode,

firstly as a longitude/latitude pair, then as a grid reference pair. Effectively, the latter are co-ordinates in metres from a notional origin off the Scilly Isles. So in this case, postcode CB1 1DU is 545.6 km. east of the Scillies and 258.4 km. north. Now it's simple to find the straight-line distances to your branches and display the nearest one.

The next six fields give Local Authority data – the postcode is in the county of Cambridgeshire; the District of Cambridge and the Electoral Ward of 'Market'. The last three fields are the NHS Authority; the Mailsort code used for bulk mailing purposes and the correctly-formatted postcode.

## JavaScript Object Notation (JSON)

The results shown so far have been in XML format, which is widely used. However, for users of JavaScript, it is also possible to produce output in JSON notation. To get this add the parameter `“output=JSON”` to the calling URL. You may also add `“callback=<fname>”` to supply a callback function name. If you omit this the name `“LoadJSON”` will be used.

The equivalents of the getroadaddress example will then be:-

```
LoadJSON({
  "Results":
  {
    "Result":
    [
      {
        "Label": "Prospect Row;CAMBRIDGE;CB1 1DU",
        "STR": "Prospect Row",
        "TWN": "Cambridge",
        "CTY": "(Cambridgeshire)",
        "PCD": "CB1 1DU"
      }
    ]
  }
})
```

If this is sent to a function as `LoadJSON(data)` then each element can be loaded with lines like:-

```
var label = data.Results.Result[0].Label;
```

## Implementation

### Javascript

This is the easiest way to add rapid addressing to a webpage. A complete HTML file is given below. It sends a request for a result in JSON (JavaScript Object Notation) form. It works by encapsulating the URL string into a script element and appending it to the document. This issues the URL. The reply is a function call which is received by LoadJSON. This parses the data into a selection list.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8"/>
<title>Address Fulfilment Demonstration</title>

<script type="text/javascript">
function getAddress()
{
    // Load postcode
    var postcode = document.getElementById('txtPostcode').value;

    // Create a script element that will issue the URL
    var loader = document.getElementById('jsonp_loader');
    if (!(loader == null)) loader.parentNode.removeChild(loader);
    var script = document.createElement("script");
    script.setAttribute("id", "jsonp_loader");
    var url = "https://www.addressgeo.co.uk/perclick/lookup.aspx?" +
        "user=test&password=password&" +
        "function=getroadaddress&output=JSON&postcode=" + postcode;
    script.setAttribute("src", url);

    // Append the element to the document, which will issue the URL.
    // The reply will call the LoadJSON function
    document.getElementsByTagName("head")[0].appendChild(script);
}

function LoadJSON(data)
// This is the callback function that will receive the reply from the
webservice.
// Its name is the default. If you use the callback=<function name>
parameter be sure to change it.
{
    // Clear any existing results
    document.getElementById('lstAddress').innerHTML = '';

    // Check for error message
    var err = data.Results.Result[0].ErrorMessage;
    if (err != null) alert(err);
    else
    {
        // Take each address in turn (normally just one)
        for (var i=0; i<data.Results.Result.length; i++)
        {
```

```
// Create an Option and add it to the Selection List
var opt = document.createElement("option");
var item = data.Results.Result[i];

opt.setAttribute('value', i);
opt.text = item.Label;
document.getElementById('lstAddress').appendChild(opt);

    }
}

</script>
</head>

<body>
Postcode <input id="txtPostcode" type="text"/>
<input type="button" onclick="getAddress()" value = "Find Address"/>
<br />
<br />
Address <select id="lstAddress" multiple='multiple' size='10'
style="width:514px; height: 26px;"></select>
</body>
</html>
```

## ASP.NET with Visual Basic

When using the .NET Framework, results are best obtained in the native XML format. Put the following declarations at the head of the source file.

```
Imports System.NET
Imports System.IO
```

Then to perform a lookup use:-

```
' Set the lookup string with a test postcode
Dim sPostcode As String = "cb11du"
Dim sURL As String =
"https://www.addressgeo.co.uk/perclick/lookup.aspx?user=test&" + _
    "password=password&function=getroadaddress&" + _
    "postcode=" + sPostcode

' Read the response from the website into a string
Dim wc As WebClient = New WebClient()
Dim arr As Byte() = wc.DownloadData(sURL)
Dim sXML As String = ASCIIEncoding.ASCII.GetString(arr)

// sXML now holds the address in XML format
```

## ASP.NET with C#

Put the following declarations at the head of the source file:-

```
using System.Net;
using System.Text;
```

To perform a lookup use:-

```
// Set the lookup string with a test postcode
string sPostcode = "cb11du";
string sURL = "https://www.addressgeo.co.uk/perclick/lookup.aspx?" +
    "user=test&password=password&function=getroadaddress&" +
    "postcode=" + sPostcode;

// Read the response from the website into a string
WebClient wc = new WebClient();
byte[] arr = wc.DownloadData(sURL);
string sXML = ASCIIEncoding.ASCII.GetString(arr);

// sXML now holds the address in XML format
```

## PHP

In PHP the code is simply:-

```
// Set the lookup string with a test postcode
$postcode = "cb11du";
$url =
    "https://www.addressgeo.co.uk/perclick/lookup.aspx?user=test&" +
    "password=password&function=getroadaddress&postcode=" +
    $postcode;

// Perform the lookup
$xmlResult = file_get_contents( $url );

// $xmlResult now holds the address in XML format
```

## Data structure

### Road-level

The codes in the first column relate to road-level data. The TWN and PCD fields will always contain data. Any or all remaining fields may be blank. If a field contains no data it will not be included in the XML result.

| Code  | Description               | Comment  |
|-------|---------------------------|--|
| Label | Address label             | Convenient assembly of road-level fields suitable for display.   |
| DST   | Dependant Street          | May be a terrace or yard in the principal street   |
| STR   | Street                    | Street name  |
| DDL   | Double-Dependant Locality | Often an Industrial Estate or hamlet outside a village   |
| DLO   | Dependant Locality        | Often a village or small town  |
| TWN   | Post Town                 | Main town from where mail is distributed   |
| CTY   | County                    | Ceremonial county as defined by the Lieutenancies Act. If the county is not normally included in the address (e.g. Bedford, Bedfordshire) it is bracketed. |
| PCD   | Postcode                  |  |

### Premise-level

A postcode may contain one or more premises. These may be residential, non-residential or a mixture.

The NUM\_R and BNA\_R lists are matched; empty entries being present where there is no building name for a premise number or vice-versa.

Similarly, the ORG, NUM\_N and BNA\_N lists are matched.

| Code  | Description                       | Comment   |
|-------|-----------------------------------|---|
| NUM_R | Premise numbers (residential)     | List of house or flat numbers, semi-colon separated |
| BNA_R | Building names (residential)      | List of building names, semi-colon separated        |
| ORG   | Organisations                     | List of organisations, semi-colon separated         |
| NUM_N | Premise numbers (non-residential) | List of premise numbers, semi-colon separated.      |
| BNA_N | Building names (non-residential)  | List of building names, semi-colon separated        |
| POB   | Post Office Box number            | Only one can be present                             |



## Postzon

The following data is supplied for Postzon information.

| <b>Code</b> | <b>Description</b>            | <b>Comment</b>                       |
|-------------|-------------------------------|--------------------------------------|
| LNG         | Longitude                     | WGS84                                |
| LAT         | Latitude                      | WGS84                                |
| GRE         | Easting Grid Reference        | A x-coordinate in metres             |
| GRN         | Northing Grid Reference       | A y-coordinate in metres             |
| CCD         | Local Authority County code   | 9-character code assigned by the ONS |
| CNM         | County name                   |                                      |
| DCD         | Local Authority District code | 9-character code assigned by the ONS |
| DNM         | District name                 |                                      |
| WCD         | Electoral Ward code           | 9-character code assigned by the ONS |
| WNM         | Electoral Ward name           |                                      |
| NHS         | NHS Authority code            | 9-character code assigned by the ONS |
| SCD         | Mailsort code                 | Royal Mail code                      |
| PCD         | Postcode                      |                                      |

## Error Messages

If an error occurs, the XML response will look something like:-

```
<?xml version="1.0" encoding="utf-8" ?>
<Result>
  <ErrCode>4</ErrCode>
  <ErrorMessage>Invalid user name or password</ErrorMessage>
</Result>
```

The JSON equivalent is:-

```
LoadJSON(
{
  "Result":
  {
    "ERRCODE": "4",
    "ERRMESSAGE": "Invalid user name or password"
  }
})
```

A complete list of all errors that may be returned is:-

|     |  |
|-----|--|
| 1   | Missing user name  |
| 2   | Missing password   |
| 3   | Missing function   |
| 4   | Invalid user name or password                              |
| 5   | Invalid function   |
| 6   | No credit left for this function                           |
| 7   | Missing postcode   |
| 8   | Invalid postcode   |
| 9   | Only codes starting with CB1-CB3 are allowed while testing |
| 10  | Postcode outside licensed Areas                            |
| 100 | System error: could not initialise addressing system       |
| 101 | System error: duplicate user                               |
| 102 | System error: invalid credit check                         |
| 103 | System error: error in addressing system                   |

## Pricing

When you first purchase credits, you will be assigned a user name and password. These will be used in the web access. You can supply an email address and a top-up point. When your credits reduce to this level, you'll automatically be sent a message inviting you to top up.

To find your credit balance at any time, use: “[function=getcredit](#)”.

Unused credits expire a year after the date of purchase.

Pricing depends on whether your website is used by the public without payment, in which case it is classed as “external” use, or is used by your own staff or by associated users that pay a fee to access the website. In the latter case it is classed as “internal” use. The Royal Mail has made internal use far more expensive than external in order not to compete with other deployment types. If you have a number of computers each with a relatively low requirement for address lookup, Pay As You Go may well be the best option. Otherwise, you should consider our installed solutions.

Most users will find the Pay As You Go solution the most economical. However, if your website is very heavily used – over 250,000 hits per year – you can get unlimited use for a fixed cost.

Full pricing can be seen on our website at [www.arcenciel.com/web\\_use](http://www.arcenciel.com/web_use) or contact us directly to discuss your requirements.