



# **IP2Location™ ActiveX Component User Manual**



Developed by Hexasoft Development Sdn. Bhd.

Copyright © 2001-2008 IP2Location.com. All Rights Reserved.

Visit our Web site at  
<http://www.ip2location.com>

*~ Bringing Geography to the Internet ~*

# INTRODUCTION

**IP2Location™** is the solution for identifying the *country origin, region, city, latitude, longitude, ZIP code, ISP, domain name, time zone and connection* of your site visitors in real-time by using IP address. This documentation provides detailed information to help you quickly get up to speed with our products and master it in no time at all.

Developers use this ActiveX/COM component for projects related to:

- **Display native language and currency**
- **Redirect based on country**
- **Digital rights management**
- **Prevent password sharing and abuse of service**
- **Reduce credit card fraud**
- **Web log stats and analysis**
- **Auto-selection of country on forms**
- **Filter access from countries you do not do business with**
- **Geo-targeting for increased sales and click-through**

IP2Location™ ActiveX/COM component is written and optimized for Windows® operating system performance.

## SYSTEM REQUIREMENT

IP2Location™ ActiveX/COM Component has been developed to run exclusively under Windows based x86/x64 machines with support for both standalone desktop and web server application. This component has built-in IP-COUNTRY database but can read a local IP2Location™ BIN database for lookup purpose. No Internet connection required to execute this component.

### Minimum Configuration Requirements

This section describes the minimum configuration requirements for a computer where the IP2Location™ ActiveX/COM control is to be installed. If the minimum requirements are not met, IP2Location™ setup file might block the installation of the redistributable package. Specifically, note that you cannot install the setup package on a computer running the DOS or Microsoft Windows 95 operating system.

### Minimum Platform Requirements

- Microsoft® Windows 98
- Microsoft® Windows 98 Second Edition
- Microsoft® Windows Millennium Edition (Windows Me)
- Microsoft® Windows NT 4 (Workstation or Server) with Service Pack 6a
- Microsoft® Windows 2000 (Professional, Server, or Advanced Server) with the latest Windows service pack and critical updates available from the Microsoft Security Web site.
- Microsoft® Windows XP (Home or Professional)
- Microsoft® Windows Server 2003 family
- Microsoft® Windows Vista family

## Minimum Configuration Requirements

Scenario	CPU Required	RAM Required
Client	Pentium 90 MHz*	32 MB**
Server	Pentium 133 MHz*	128 MB**

\* Or the minimum CPU required running the operating system, whichever is higher.

\*\* Or the minimum RAM required running the operating system, whichever is higher.

## Quick Start Guide

In this document, you will go through these basic steps to use IP2Location<sup>TM</sup> ActiveX/COM component:

1. Download and installing IP2Location ActiveX/COM component from Web site
2. Test your ActiveX/COM component installation using demo source codes
3. Start using IP2Location<sup>TM</sup> ActiveX/COM in your projects
4. Purchase license to remove random 5 second delay in demo version
5. Update IP2Location<sup>TM</sup> ActiveX/COM component database in monthly basis

Note: For demo version, you will have a random 5-second delay. Registered version has unlimited real-time query without delay.

## 1. DOWNLOAD AND INSTALLING IP2LOCATION ACTIVEX/COM COMPONENT

### 1.1. Download Setup File

Please use the following URL to download the latest IP2Location™ .NET Component 2.0 together with sample database.

<http://www.ip2location.com/download/ip2location-activex.zip>

### 1.2. ActiveX Component Installation

Installing the IP2Location ActiveX/COM component system is very easy. You just need to unzip the setup file.

General steps in installing IP2Location™ ActiveX/COM component.

1. Unzip the demo package into a temporary directory.
2. Copy ActiveX component "ip2location.dll" and sample database "IP-COUNTRY-SAMPLE.BIN" to a system directory in Web server. Directory *Windows\system32\* or *Winnt\system32\* is recommended.
3. Register the DLL component using one of the following options :

#### Option 1

From the command prompt, navigate to the directory where you installed ip2location.dll and then run the command :

```
C:\Windows\system32> regsvr32 ip2location.dll
```

**OR**

#### Option 2

Click on the Windows® "Start" button, then "Run" and typing in:

```
regsvr32 <fullpath>\ip2location.dll
```

Make sure and replace "<fullpath>" with the actual full path that the ip2location.dll resides in.

4. In Explorer, right click ip2location.dll, select "Properties" and go to "Security" tab. Add "Read & Execute" permission to the Internet user account i.e. "IUSR\_<machine>".
5. In Explorer, right click IP-COUNTRY-SAMPLE.BIN, select "Properties" and go to "Security" tab. Add "Read" permission to the Internet user account. i.e. "IUSR <machine>"

**\* Note: Please use x64ip2location.dll in 64 Bit operating system.**

## 2. TESTING IP2LOCATION™ .NET COMPONENT 2.0

### 2.1 Folders and Files Structure

Several sample files were enclosed in component zip file. The sample filenames by default are :

/asp/ip2location.asp	(ASP demo script)
/php/ip2location.php	(PHP demo script)
/vb6/ip2location.frm	(Visual Basic 6 demo application)
/asp.net/vb.net/ip2location.aspx.vb	(VB.NET demo script)
/asp.net/c#/ip2location.aspx.cs	(C# demo script)

Copy one of these samples to a Web directory configured to run Web project in specific language. You need to copy the IP2Location Sample BIN file /IP-COUNTRY-SAMPLE>BIN file to the directory as well. Then run these samples using Web Browser.

If the IP2Location™ has been installed correctly, the sample script will execute and demonstrate results from the IP2Location products. For demo version, there is a 5-minutes delay per query. Registered version enjoys unlimited real-time query without delay.

*\*Note: If you are getting this error "Attempted to load a 64-bit assembly on a 32-bit platform. Use ReflectionOnlyLoad() instead if trying to load for reflection purposes." Please disable "Enable 32-Bit Applications" in your IIS Application Pools.*

### 2.2 Download Other IP2Location Sample BIN Files

You can access to more IP2Location sample BIN files for evaluation purpose. The demo version of Binary (.BIN) data file consists of IP address range from 0.0.0.0 to 99.255.255.255. The complete Binary (.BIN) data file is available upon software registration.

IP2Location DB18 Sample bin File

<http://www.ip2location.com/download/samples-db18.zip>

Please visit <http://www.p2location.com/developers.aspx> to get all links for sample database.

### 3. IMPLEMENTING IP2LOCATION™ ACTIVEX/COM COMPONENT

#### 3.1 IP2Location™ ActiveX Method Reference

Method	Description
Initialize ( RegCode As String )	Initialize component and apply registration code to remove 5-seconds query delay. Please use "demo" for evaluation purpose.
CountryCodeToName (Code As String)	Return full country name by country code. Refer Appendix I.
LookUpFullName ( IPAddress As String)	Return full country name of a specified IP address using Internal IP database.
LookUpShortName ( IPAddress As String)	Return country code (two letters) of a specified IP address using Internal IP database. Refer Appendix I.
Version ()	Return component version.
IsRegistered()	Return true if component have been registered correctly and n delay in query.
LookUp ( BINPath As String, IPAddress As String)	Query external IP2Location BIN database as in BINPath and stores all values in Object Properties.

### 3.2 IP2Location™ ActiveX Property Reference

Property	Type	Description
IPAddress	String	IP Address.
CountryShort	String	Two-character country code based on ISO 3166.
CountryLong	String	Country name based on ISO 3166.
Region	String	Region or state name.
City	String	City name.
Latitude	String	City latitude.
Longitude	String	City longitude.
ZIPCode	String	United States ZIP code.
ISP	String	Internet Service Provider (ISP) name.
Domain	String	Internet domain name associated to IP address range.
Timezone	String	Time zone of city
NetSpeed	String	Connection speed
IDDCode	String	The IDD prefix to call the city from another country
AreaCode	String	A varying length number assigned to geographic areas for call between cities
WeatherStationCode	String	Special code to identify the nearest weather observation station
WeatherStationName	String	Name of the nearest weather observation station

**\*Note :** The result returns a not supported message if the field of the result is not supported by the current database.

### 3.3 Sample Codes in PHP

*Please refer to the sample PHP codes in the setup package for more information.*

### 3.4 Sample Codes in Visual Basic 6

*Please refer to the VB6 sample codes in the setup package for more information.*

### 3.5 Sample Codes in C#

*Please refer to the C# sample codes in setup package for more information.*

### 3.6 Sample Codes in VB.NET

*Please refer to the VB.NET sample codes in the setup package for more information.*

### 3.7 Sample Codes in ASP

*Please refer to the ASP sample codes in the setup package for more information.*

### 3.8 Sample Codes in Cold Fusion

#### Query Country Information using Internal Database

```
<!-- connect to COM object -->
<CFOBJECT ACTION="Create" NAME="ipObj" CLASS="IP2Location.Country">
<!-- initialize IP2LocationT Component -->
<CFSET result_code = ipObj.Initialize("demo")>
<!-- it will return "US" -->
<CFSET result_code = ipObj.LookUpShortName("205.188.145.215")>
<!-- it will return "United States" -->
<CFSET result_code = ipObj.LookUpFullName("205.188.145.215")>
<!-- it will return "Japan" -->
<CFSET result_code = ipObj.CountryCodeToName("JP")>
<!-- it will return the version number -->
<CFSET result_code = ipObj.Version()>
```

#### Query Other Information using External Database

```
<!-- connect to COM object -->
<CFOBJECT ACTION="Create" NAME="ipObj" CLASS="IP2Location.Country">
<!-- initialize IP2LocationT Component -->
<CFSET result_code = ipObj.Initialize("demo")>
<!-- query external IP2Location database -->
<CFSET result_code = ipObj.LookUp("IP-COUNTRYSAMPLE.
BIN", "15.188.145.215")>
<CFSET result_code = ipObj.CountryShort>
<CFSET result_code = ipObj.CountryLong>
<CFSET result_code = ipObj.Region>
<CFSET result_code = ipObj.City>
<CFSET result_code = ipObj.Latitude>
<CFSET result_code = ipObj.Longitude>
<CFSET result_code = ipObj.ZIPCode>
<CFSET result_code = ipObj.ISP>
<CFSET result_code = ipObj.Domain>
<CFSET result_code = ipObj.TimeZone>
<CFSET result_code = ipObj.NetSpeed>
<CFSET result_code = ipObj.IDDCode>
<CFSET result_code = ipObj.AreaCode>
<CFSET result_code = ipObj.WeatherStationCode>
<CFSET result_code = ipObj.WeatherStationName>
```



### 3.9 Sample Codes in Visual C++

#### Query Country Information using Internal Database

```
#import "c:\winnt\system32\ip2location.dll" no_namespace
HRESULT hr = OleInitialize(NULL);
ICountry ipObj("IP2Location.Country");
if ((hr == S_OK) || (hr == S_FALSE))
{
    // initialize IP2LocationT Component
    if (ipObj->Initialize("demo") != "OK")
    {
        // initialization failed
    }
    // it will return "US"
    AfxMessageBox(ipObj->LookupShortName("205.188.145.215"));
    // it will return "United States"
    AfxMessageBox(ipObj->LookupFullName("205.188.145.215"));
    // it will return "Japan"
    AfxMessageBox(ipObj->CountryCodeToName("JP"));
    // it will return the version number
    AfxMessageBox(ipObj->Version());
}
OleUninitialize();
```

#### Query Other Information using External Database

```
#import "c:\winnt\system32\ip2location.dll" no_namespace
HRESULT hr = OleInitialize(NULL);
ICountry ipObj("IP2Location.Country");
if ((hr == S_OK) || (hr == S_FALSE))
{
    // initialize IP2Location Component
    if (ipObj->Initialize("demo") != "OK")
    {
        // initialization failed
    }
    AfxMessageBox(ipObj->Lookup("IP-COUNTRYSAMPLE.BIN", "15.188.145.215"));
    AfxMessageBox(ipObj->CountryShort);
    AfxMessageBox(ipObj->CountryLong);
    AfxMessageBox(ipObj->Region);
    AfxMessageBox(ipObj->City);
    AfxMessageBox(ipObj->Latitude);
    AfxMessageBox(ipObj->Longitude);
    AfxMessageBox(ipObj->ZIPCode);
    AfxMessageBox(ipObj->ISP);
    AfxMessageBox(ipObj->Domain);
    AfxMessageBox(ipObj->TimeZone);
    AfxMessageBox(ipObj->NetSpeed);
    AfxMessageBox(ipObj->IDDCCode);
    AfxMessageBox(ipObj->AreaCode);
    AfxMessageBox(ipObj->WeatherStationCode);
    AfxMessageBox(ipObj->WeatherStationName);
}
OleUninitialize();
```

### 3.10 Sample Codes in Delphi

#### Query Country Information using Internal Database

```

uses ComObj;
var
ipObj: Variant;
nResult: Variant;
begin
    // create IP2LocationT object
    ipObj:=CreateOleObject('IP2Location.Country');
    // initialize IP2LocationT Component
    nResult := ipObj.Initialize('demo');
    if nResult = 'OK' then begin;
        // initialization passed
    else
        // initialization failed
    end;
    // it will return "US"
    nResult := ipObj.LookUpShortName('205.188.145.215');
    // it will return "United States"
    nResult := ipObj.LookUpFullName('205.188.145.215');
    // it will return "Japan"
    nResult := ipObj.CountryCodeToName('JP');
    // it will return the version number
    nResult := ipObj.Version();
end;

```

#### Query Other Information using External Database

```

uses ComObj;
var
ipObj: Variant;
nResult: Variant;
begin
    // create IP2LocationT object
    ipObj:=CreateOleObject('IP2Location.Country');
    // initialize IP2LocationT Component
    nResult := ipObj.Initialize('demo');
    nResult := ipObj.LookUp('IP-COUNTRY-
SAMPLE.BIN', '15.188.145.215');
    nResult := ipObj.CountryShort;
    nResult := ipObj.CountryLong;
    nResult := ipObj.Region;
    nResult := ipObj.City;
    nResult := ipObj.Latitude;
    nResult := ipObj.Longitude;
    nResult := ipObj.ZIPCode;
    nResult := ipObj.ISP;
    nResult := ipObj.Domain;
    nResult := ipObj.TimeZone;
    nResult := ipObj.NetSpeed;
    nResult := ipObj.IDDCCode;
    nResult := ipObj.AreaCode;
    nResult := ipObj.WeatherStationCode;
    nResult := ipObj.WeatherStationName;
end;

```

### 3.11 Sample Codes in Perl

#### Query Country Information using Internal Database

```
use OLE;
$code = Win32::OLECreateObject('IP2Location.Country',$ipObj) or
die "Couldn't create new ipObj Object!";
# initialize IP2LocationT Component
if ($ipObj->Initialize("demo") <> "OK")
{
    # initialization failed
}
# it will return "US"
$result = $ipObj->LookUpShortName("205.188.145.215");
# it will return "United States"
$result = $ipObj->LookUpFullName("205.188.145.215");
# it will return "Japan"
$result = $ipObj->CountryCodeToName("JP");
# it will return the version number
$result = $ipObj->Version();
# free object
$ipObj->Quit();
```

#### Query Other Information using External Database

```
use OLE;
$code = Win32::OLECreateObject('IP2Location.Country',$ipObj) or
die "Couldn't create new ipObj Object!";
# initialize IP2Location Component
if ($ipObj->Initialize("demo") <> "OK")
{
    # initialization failed
}
$result = $ipObj->LookUp('IP-COUNTRY-SAMPLE.BIN','15.188.145.215');
$result = $ipObj->CountryShort();
$result = $ipObj->CountryLong();
$result = $ipObj->Region();
$result = $ipObj->City();
$result = $ipObj->Latitude();
$result = $ipObj->Longitude();
$result = $ipObj->ZIPCode();
$result = $ipObj->ISP();
$result = $ipObj->Domain();
$result = $ipObj->TimeZone();
$result = $ipObj->IDDCode();
$result = $ipObj->AreaCode();
$result = $ipObj->WeatherStationCode();
$result = $ipObj->WeatherStationName();
# free object
$ipObj->Quit();
```

## 4. Purchase IP2LOCATION™ ActiveX/COM COMPONENT

### 4.1 License Agreement

A license is required for each machine the product is installed on, including development or staging machines. Please refer to the end of this document for the complete license agreement.

### 4.3 Purchase Instructions

Proceed to our order page. Fill out the online form and choose the correct number of licenses. Once your order has been approved, you will receive your license file (License.Key) immediately through email.

Initialize the ActiveX component with the registration code in your script will remove the 5-seconds query delay.

Secure Order Page:

<https://www.ip2location.com/buyonline.aspx>

For example, if the registration code is "demo@yahoo.com-JGSVXgzjdDNS"

### ASP

```
Set ipObj = Server.CreateObject("IP2Location.Country")
result = ipObj.Initialize("demo@yahoo.com-JGSVXgzjdDNS")
...
Set ipObj = Nothing
```

### PHP

```
$ipObj = new COM("IP2Location.Country");
$result = $ipObj->Initialize(demo@yahoo.com-JGSVXgzjdDNS);
...
$ipObj = null;
```

## 5. UPDATE COMPONENT DATABASE

### 5.1 Update Component Internal Database

The IP2Location™ ActiveX/COM Component depends to an external database for non-country information reverse lookup. All users with valid license will be allowed to download the monthly updates from the IP2Location™ download area during subscription period. To update, users need to replace the BIN database with the latest one from the download area. The component will query the latest database once the local copy has been replaced.

## 6. UPDATE OR UNINSTALL COMPONENT

### 6.1 Upgrade Component

To upgrade a new version of IP-Country database, you can overwrite the IP2Location™ ActiveX DLL Component after uninstalling the old component.

**Note:** You need to stop the IIS web server before overwrite your old component.

First, stop the Web server

```
C:\>net stop iisadmin /y
```

The, un-register the component and update it with the latest component.

```
C:\>regsvr32 c:\windows\system32\ip2location.dll /u  
C:\>copy c:\new\ip2location.dll c:\windows\system32\ip2location.dll
```

NOTE: In this case, "c:\new\ip2location.dll" is the latest version of component. Meanwhile "c:\path\ip2location.dll" is the destination. We assume the installation path is c:\windows\system32.

Finally, register component and restart the web server.

```
C:\>regsvr32 c:\windows\system32\ip2location.dll  
C:\>net start w3svc
```

### 6.2 Uninstall Component

To uninstall IP2Location™ ActiveX/COM DLL Component, you need to stop and restart the IIS services to remove the old version of the DLL from the memory, as follows:

```
C:\>net stop iisadmin /
```

Then

```
C:\>regsvr32 c:\windows\system32\ip2location.dll /u  
C:\>delete c:\windows\system32\ip2location.dll
```

```
C:\>net start w3svc
```

## APPENDIX I: ISO3166 COUNTRY CODE

This table lists all valid ISO3166 two characters country codes that returns from IP2Location™ .NET component and explain the full country names for these country codes.

Country Code	Country Name
AD	ANDORRA
AE	UNITED ARAB EMIRATES
AF	AFGHANISTAN
AG	ANTIGUA AND BARBUDA
AI	ANGUILLA
AL	ALBANIA
AM	ARMENIA
AN	NETHERLANDS ANTILLES
AO	ANGOLA
AP	ASIA PACIFIC
AQ	ANTARCTICA
AR	ARGENTINA
AS	AMERICAN SAMOA
AT	AUSTRIA
AU	AUSTRALIA
AW	ARUBA
AZ	AZERBAIJAN
BA	BOSNIA AND HERZEGOWINA
BB	BARBADOS
BD	BANGLADESH
BE	BELGIUM
BF	BURKINA FASO
BG	BULGARIA
BH	BAHRAIN
BI	BURUNDI
BJ	BENIN
BM	BERMUDA
BN	BRUNEI DARUSSALAM
BO	BOLIVIA
BR	BRAZIL
BS	BAHAMAS
BT	BHUTAN
BV	BOUVET ISLAND
BW	BOTSWANA
BY	BELARUS
BZ	BELIZE
CA	CANADA
CC	COCOS (KEELING) ISLANDS
CD	CONGO, THE DEMOCRATIC REPUBLIC OF THE
CF	CENTRAL AFRICAN REPUBLIC
CG	CONGO

Country Code	Country Name
CH	SWITZERLAND
CI	COTE D'IVOIRE
CK	COOK ISLANDS
CL	CHILE
CM	CAMEROON
CN	CHINA
CO	COLOMBIA
CR	COSTA RICA
CS	CZECHOSLOVAKIA (FORMER)
CU	CUBA
CV	CAPE VERDE
CX	CHRISTMAS ISLAND
CY	CYPRUS
CZ	CZECH REPUBLIC
DE	GERMANY
DJ	DJIBOUTI
DK	DENMARK
DM	DOMINICA
DO	DOMINICAN REPUBLIC
DZ	ALGERIA
EC	ECUADOR
EE	ESTONIA
EG	EGYPT
EH	WESTERN SAHARA
ER	ERITREA
ES	SPAIN
ET	ETHIOPIA
EU	EUROPEAN UNION
FI	FINLAND
FJ	FIJI
FK	FALKLAND ISLANDS (MALVINAS)
FM	MICRONESIA, FEDERATED STATES OF
FO	FAROE ISLANDS
FR	FRANCE
FX	FRANCE, METROPOLITAN
GA	GABON
GB	GREAT BRITAIN
GD	GRENADA
GE	GEORGIA
GF	FRENCH GUIANA
GH	GHANA
GI	GIBRALTAR
GL	GREENLAND
GM	GAMBIA
GN	GUINEA
GP	GUADELOUPE
GQ	EQUATORIAL GUINEA
GR	GREECE

Country Code	Country Name
GS	SOUTH GEORGIA & SOUTH SANDWICH ISLANDS
GT	GUATEMALA
GU	GUAM
GW	GUINEA-BISSAU
GY	GUYANA
HK	HONG KONG
HM	HEARD ISLAND AND MCDONALD ISLANDS
HN	HONDURAS
HR	CROATIA
HT	HAITI
HU	HUNGARY
ID	INDONESIA
IE	IRELAND
IL	ISRAEL
IN	INDIA
IO	BRITISH INDIAN OCEAN TERRITORY
IQ	IRAQ
IR	IRAN, ISLAMIC REPUBLIC OF
IS	ICELAND
IT	ITALY
JM	JAMAICA
JO	JORDAN
JP	JAPAN
KE	KENYA
KG	KYRGYZSTAN
KH	CAMBODIA
KI	KIRIBATI
KM	COMOROS
KN	SAINT KITTS AND NEVIS
KP	KOREA, DEMOCRATIC PEOPLE'S REPUBLIC OF
KR	KOREA, REPUBLIC OF
KW	KUWAIT
KY	CAYMAN ISLANDS
KZ	KAZAKSTAN
LA	LAO PEOPLE'S DEMOCRATIC REPUBLIC
LB	LEBANON
LC	SAINT LUCIA
LI	LIECHTENSTEIN
LK	SRI LANKA
LR	LIBERIA
LS	LESOTHO
LT	LITHUANIA
LU	LUXEMBOURG
LV	LATVIA
LY	LIBYAN ARAB JAMAHIRIYA
MA	MOROCCO
MC	MONACO
MD	MOLDOVA, REPUBLIC OF



Country Code	Country Name
MG	MADAGASCAR
MH	MARSHALL ISLANDS
MK	MACEDONIA, THE FORMER YUGOSLAV
ML	MALI
MM	MYANMAR
MN	MONGOLIA
MO	MACAU
MP	NORTHERN MARIANA ISLANDS
MQ	MARTINIQUE
MR	MAURITANIA
MS	MONTSERRAT
MT	MALTA
MU	MAURITIUS
MV	MALDIVES
MW	MALAWI
MX	MEXICO
MY	MALAYSIA
MZ	MOZAMBIQUE
NA	NAMIBIA
NC	NEW CALEDONIA
NE	NIGER
NF	NORFOLK ISLAND
NG	NIGERIA
NI	NICARAGUA
NL	NETHERLANDS
NO	NORWAY
NP	NEPAL
NR	NAURU
NU	NIUE
NZ	NEW ZEALAND
OM	OMAN
PA	PANAMA
PE	PERU
PF	FRENCH POLYNESIA
PG	PAPUA NEW GUINEA
PH	PHILIPPINES
PK	PAKISTAN
PL	POLAND
PM	SAINT PIERRE AND MIQUELON
PN	PITCAIRN
PR	PUERTO RICO
PS	PALESTINIAN TERRITORY, OCCUPIED
PT	PORTUGAL
PW	PALAU
PY	PARAGUAY
QA	QATAR
RE	REUNION
RO	ROMANIA

Country Code	Country Name
RU	RUSSIAN FEDERATION
RW	RWANDA
SA	SAUDI ARABIA
SB	SOLOMON ISLANDS
SC	SEYCHELLES
SD	SUDAN
SE	SWEDEN
SG	SINGAPORE
SH	SAINT HELENA
SI	SLOVENIA
SJ	SVALBARD AND JAN MAYEN
SK	SLOVAKIA
SL	SIERRA LEONE
SM	SAN MARINO
SN	SENEGAL
SO	SOMALIA
SR	SURINAME
ST	SAO TOME AND PRINCIPE
SU	RUSSIAN FEDERATION
SV	EL SALVADOR
SY	SYRIAN ARAB REPUBLIC
SZ	SWAZILAND
TC	TURKS AND CAICOS ISLANDS
TD	CHAD
TF	FRENCH SOUTHERN TERRITORIES
TG	TOGO
TH	THAILAND
TJ	TAJIKISTAN
TK	TOKELAU
TM	TURKMENISTAN
TN	TUNISIA
TO	TONGA
TP	EAST TIMOR
TR	TURKEY
TT	TRINIDAD AND TOBAGO
TV	TUVALU
TW	TAIWAN, PROVINCE OF CHINA
TZ	TANZANIA, UNITED REPUBLIC OF
UA	UKRAINE
UG	UGANDA
UK	UNITED KINGDOM
UM	UNITED STATES MINOR OUTLYING ISLANDS
US	UNITED STATES
UY	URUGUAY
UZ	UZBEKISTAN
VA	HOLY SEE (VATICAN CITY STATE)
VC	SAINT VINCENT AND THE GRENADINES
VE	VENEZUELA

Country Code	Country Name
VG	VIRGIN ISLANDS, BRITISH
VI	VIRGIN ISLANDS, U.S.
VN	VIET NAM
VU	VANUATU
WF	WALLIS AND FUTUNA
WS	SAMOA
YE	YEMEN
YT	MAYOTTE
YU	YUGOSLAVIA
ZA	SOUTH AFRICA
ZM	ZAMBIA
ZW	ZIMBABWE

# IP2LOCATION™ ACTIVEX/COM COMPONENT LICENSE AGREEMENT

**IMPORTANT-READ CAREFULLY:**

This License Agreement is a legal agreement between you (either an individual or a single entity) and Hexasoft Development Sdn. Bhd., owner of IP2Location™ trademark, ("Hexasoft" or "we") for the Hexasoft developed IP2Location™ .NET Component (hereafter referred to as the SOFTWARE PRODUCT) accompanying this License Agreement, which includes web service routines and data result(s). By exercising your rights to make and use copies of the SOFTWARE PRODUCT, you agree to be bound by the terms of this License Agreement. If you do not agree to the terms of this License Agreement, you may not use the SOFTWARE PRODUCT.

**GRANT OF LICENSE.**

This License Agreement grants the following rights: You are granted the right to use the SOFTWARE PRODUCT files on one computer in the Internet or Local Area Network ("LAN"). You may not use the SOFTWARE PRODUCT files on multiple computers without matching number of licenses.

**DESCRIPTION OF LIMITATIONS.**

You may not reverse engineer except and only to the extent that such activity is expressly permitted by applicable law notwithstanding this limitation. Without prejudice to any other rights, Hexasoft may terminate this License Agreement if you fail to comply with the terms and conditions of this License Agreement. In such event, you must destroy all copies of the SOFTWARE PRODUCT and all of its parts.

**COPYRIGHT.**

All title and copyrights in and to the SOFTWARE PRODUCT and any copies of the SOFTWARE PRODUCT are owned by Hexasoft. The SOFTWARE PRODUCT is protected by copyright laws and international treaty provisions.

**NO WARRANTIES.**

Hexasoft expressly disclaims any warranty for the SOFTWARE PRODUCT. The SOFTWARE PRODUCT and any related documentation is provided "as is" without warranty of any kind, either express or implied, including, without limitation, the implied warranties of merchantability or fitness for a particular purpose. The entire risk arising out of use or performance of the SOFTWARE PRODUCT remains with you.

**LIMITATION OF LIABILITY.**

Hexasoft's entire liability and your exclusive remedy under this Agreement shall not exceed fifteen dollars (US \$15.00).

**NO LIABILITY FOR CONSEQUENTIAL DAMAGES.**

In no event shall Hexasoft nor anyone else who has been involved in the creation, production, or delivery of the SOFTWARE PRODUCT be liable for any damages whatsoever (including, without limitation, damages for loss of business profits, business interruption, loss of business information, or any other pecuniary loss) arising out of the use of or inability to use this Hexasoft product, even if Hexasoft has been advised of the possibility of such damages. Because some states and jurisdictions do not allow the exclusion or limitation of liability for consequential or incidental damages, the above limitation may not apply to you.

**MISCELLANEOUS**

"IP2Location" is a trademark of Hexasoft Development Sdn. Bhd..

"Microsoft" is a registered trademark of Microsoft Corporation.

"Windows" is a registered trademark of Microsoft Corporation.

All other trademarks are the property of their respective owners.

Copyright © 2008 Hexasoft Development Sdn. Bhd., All Rights Reserved.

**Hexasoft Development Sdn. Bhd.**

1-2-15 Mayang Mall Kompleks,  
Jalan Mayang Pasir 1,  
11950 Bayan Baru,  
Pulau Pinang,  
Malaysia. Tel: (6)-04-640-2380  
Fax: (6)-04-640-2381  
Email: [sales@ip2location.com](mailto:sales@ip2location.com)