Duncan Temple Lang, University of California at Davis

Table of Contents

Get the code for the C API from http://geolite.maxmind.com/download/geoip/api/c/ Get the binary form of the data http://www.maxmind.com/app/geolitecity

```
library(RCIndex)
col = genFunctionCollector()
parseTU("/usr/local/include/GeoIP.h", col$update)
```

```
funcs = col$funcs()
grep("^GeoIP", names(funcs), value = TRUE)
funcs = funcs[ grep("^GeoIP", names(funcs), value = TRUE) ]
```

The first thing we need to use any of the primary routines is a pointer to a GeoIP object. This is the first argument to most routines. We create this with the function GeoIP_new. This takes an integer value which is a combination of flags that control how this is created. These flags are found in the GeoIPOptions enumeration. Let's get those

```
col = genEnumCollector()
parseTU("/usr/local/include/GeoIP.h", col$update)
col$enums()
```

Web Service Queries

http://ipinfodb.com provides several REST-based Web services. The results can be returned as XML or JSON. We can get the country name for an IP address; get the city and longitude and latitude; process multiple IP addresses.

```
getIPCountry =
function(ip, ..., curl = getCurlHandle())
{
   txt = getForm("http://ipinfodb.com/ip_query_country.php", ip = ip, curl = curl,
   doc = xmlParse(txt, asText = TRUE)
   r = xmlRoot(doc)
   if(xmlValue(r[["Status"]]) == "OK") {
     sapply(c("CountryCode", "CountryName"),
             function(x)
                 xmlValue(r[[x]]))
   } else
      stop("Failed")
}
```

```
getIPLocation =
```

```
function(ip, ...., curl = getCurlHandle())
{
   multi = length(ip) > 1

   u = if(multi)
          "http://ipinfodb.com/ip_query2.php"
       else
          "http://ipinfodb.com/ip_query.php"
   tt = getForm(u, ip = if(multi) paste(ip, collapse = ",") else ip,
                timezone = 'false')

 #  xmlToDataFrame(xmlParse(tt, asText = TRUE))

  r = xmlRoot( xmlParse(tt, asText = TRUE) )
  structure(xmlSApply(r, xmlValue), names = names(r))
}
```