<http://code.google.com/apis/kml/documentation/kml_tut.html>

**1. Giới thiệu KML**

KML là một định dạng file để hiển thị dữ liệu địa lý trong một trình duyệt Earth như Google Earth, Google Maps, Google Maps cho di động. KML sử dụng một cấu trúc dựa trên tag với các thành phần lồng nhau và các thuộc tính và được dựa trên chuẩn XML. Tất cả các tag phân biệt dạng chữ và phải xuất hiện chính xác [KML Reference](http://code.google.com/apis/kml/documentation/kmlreference.html). Danh mục phụ lục chỉ ra tag nào tag tùy chọn. Bên trong một thành phần được trao, các tag phải xuất hiện theo thức tự hiển thị trong danh mục ([KML Reference](http://code.google.com/apis/kml/documentation/kmlreference.html)).

Placemark

Một placemark là một trong những tính năng được sử dụng phổ biến trong Google Earth. Nó đánh dấu một vị trí trên bề mặt của Earth, sử dụng một icon hình cái đinh màu vàng. Một Placemark đơn giản nhất chỉ gồm một thành phần <Point>, chỉ rõ vị trí của Placemark. Chúng ta có thể chỉ rõ tên và một icon túy ý cho Placemark, và cũng có thể thêm các thành phần hình học vào nó. Dưới đây là một file code placemark đơn giản :

<?xml version="1.0" encoding="UTF-8"?>

<kml xmlns="http://www.opengis.net/kml/2.2">

<Placemark>

<name>Simple placemark</name>

<description>Attached to the ground. Intelligently places itself

at the height of the underlying terrain.</description>

<Point>

<coordinates>-122.0822035425683,37.42228990140251,0</coordinates>

</Point>

</Placemark>

</kml>

Cấu trúc file được chia như sau:

* Phần đầu là thông tin của file XML. Đây là phần nằm ở dòng thứ nhất của mỗi file kml. Không có khoảng trắng hay ký tự nào xuất hiện trước dòng này.
* Một phần khai báo về namespace KML. Đây là dòng thứ 2 trong mỗi file KML 2.2.
* Một đối tượng Placemark chứa các thành phần sau:
* Một tag *name* được sử dụng như một tên cho Placemark.
* Một tag *description* xuất hiện trong “balloon” đính kèm vào Placemark
* Một tag *Point* chỉ ra vị trí của một Placemark trên bề mặt của Earth (*longitude*, *latitude*, và một thuộc tính tùy chọn *altitude*).

**HTML mô tả trong Placemark**

Sử dụng thành phần CDATA

Nếu bạn muốn viết HTML chuẩn bên trong một tag <description> , bạn có thể đạt nó bên trong một tag CDATA. Nếu không, dấu ngoặc nhọn phải được viết như một tham chiếu để thực thể để chặn Google Earth từ việc phân tích cú pháp HTML một cách không đúng (ví dụ, biểu tượng > phải được viết như &gt; và biểu tượng < được viết như &lt). Đây là một tính năng tiêu chuẩn của XML và không duy nhất đối với Google Earth.

Xem xét sự khác nhau giữa HTML với các tag CDATA và không có CDATA. Đầu tiên, đây là <description> với các tag CDATA:

<?xml version="1.0" encoding="UTF-8"?>

<kml xmlns="http://www.opengis.net/kml/2.2">

<Document>

<Placemark>

<name>CDATA example</name>

<description>

<![CDATA[

<h1>CDATA Tags are useful!</h1>

<p><font color="red">Text is <i>more readable</i> and

<b>easier to write</b> when you can avoid using entity

references.</font></p>

]]>

</description>

<Point>

<coordinates>102.595626,14.996729</coordinates>

</Point>

</Placemark>

</Document>

</kml>

Và dưới đây là <description > với các tag CDATA, vì vậy các ký tự đặc biệt phải sử dụng các tham chiếu thực thể:

<?xml version="1.0" encoding="UTF-8"?>

<kml xmlns="http://www.opengis.net/kml/2.2">

<Document>

<Placemark>

<name>Entity references example</name>

<description>

&lt;h1&gt;Entity references are hard to type!&lt;/h1&gt;

&lt;p&gt;&lt;font color="green"&gt;Text is

&lt;i&gt;more readable&lt;/i&gt;

and &lt;b&gt;easier to write&lt;/b&gt;

when you can avoid using entity references.&lt;/font&gt;&lt;/p&gt;

</description>

<Point>

<coordinates>102.594411,14.998518</coordinates>

</Point>

</Placemark>

</Document>

</kml>

**Đường đi (path)**

Nhiều loại đưởng đi khác nhau có thể được tạo ra trong Google Earth. Trong KML, một đường đi được tạo ra bởi thành phần <LineString>. Hãy xem một đọa code trong trong file KML đưới đây bạn sẽ thấy đường đi được tại ra như thế nào:

<?xml version="1.0" encoding="UTF-8"?>

<kml xmlns="http://www.opengis.net/kml/2.2">

<Document>

<name>Paths</name>

<description>Examples of paths. Note that the tessellate tag is by default

set to 0. If you want to create tessellated lines, they must be authored

(or edited) directly in KML.</description>

<Style id="yellowLineGreenPoly">

<LineStyle>

<color>7f00ffff</color>

<width>4</width>

</LineStyle>

<PolyStyle>

<color>7f00ff00</color>

</PolyStyle>

</Style>

<Placemark>

<name>Absolute Extruded</name>

<description>Transparent green wall with yellow outlines</description>

<styleUrl>#yellowLineGreenPoly</styleUrl>

<LineString>

<extrude>1</extrude>

<tessellate>1</tessellate>

<altitudeMode>absolute</altitudeMode>

<coordinates> -112.2550785337791,36.07954952145647,2357

-112.2549277039738,36.08117083492122,2357

-112.2552505069063,36.08260761307279,2357

-112.2564540158376,36.08395660588506,2357

-112.2580238976449,36.08511401044813,2357

-112.2595218489022,36.08584355239394,2357

-112.2608216347552,36.08612634548589,2357

-112.262073428656,36.08626019085147,2357

-112.2633204928495,36.08621519860091,2357

-112.2644963846444,36.08627897945274,2357

-112.2656969554589,36.08649599090644,2357

</coordinates>

</LineString>

</Placemark>

</Document>

</kml>

**2. Các truy xuất thông dữ liệu từ file KML**

Để truy xuất thông tin từ file KML mà google map trả về ta sử dụng XML DOM. Đây là mô hình đối tượng và giao diện lập trình chuẩn cho văn bản XML. XML DOM định nghĩa các đối tượng và thuộc tính của tất cả thành phần XML, và các phương thức (interface) để truy cập chúng

**Truy xuất thông tin địa điểm**

Ta hãy xem xét đoạn code dưới để lấy về thông tin đia điểm

ArrayList<PlaceModel> lstPlaceMode = **new** ArrayList<PlaceModel>();

//Build url

StringBuilder urlString = **new** StringBuilder();

urlString.append("http://maps.google.com/maps?q=");

urlString.append(keyword); //nội dung tìm kiếm

urlString.append(Double.*toString*(lat));

urlString.append(",");

urlString.append(Double.*toString*(lng));

urlString.append("&num=");

urlString.append(Integer.*toString*(num));

urlString.append("&ie=UTF8&output=kml");

// get the kml (XML) doc. And parse it. Lất về file kml (xml). Và đôc thông tin

Document doc = **null**;

HttpURLConnection urlConnection= **null**;

URL url = **null**;

**try**

{

//Declare connection

url = **new** URL(urlString.toString());

urlConnection=(HttpURLConnection)url.openConnection();

urlConnection.setRequestMethod("GET");

urlConnection.setDoOutput(**true**);

urlConnection.setDoInput(**true**);

urlConnection.connect();

DocumentBuilderFactory dbf = DocumentBuilderFactory.*newInstance*();

DocumentBuilder db = dbf.newDocumentBuilder();

doc = db.parse(urlConnection.getInputStream());

**if**(doc.getElementsByTagName("Placemark").getLength()>0)

{

**for**(**int** i=0;i<doc.getElementsByTagName("Placemark").getLength();i++)

{

PlaceModel placeModel = **new** PlaceModel();

//Get ATM name

String name = doc.getElementsByTagName("Placemark")

.item(i).getChildNodes().item(0)

.getFirstChild().getNodeValue();

//Get the address of ATM

String address = doc.getElementsByTagName("Placemark")

.item(i).getChildNodes().item(2)

.getFirstChild().getNodeValue();

placeModel.setName(name);

placeModel.setAddress(address);

lstPlaceMode.add(placeModel);

}

}

}

**catch** (MalformedURLException e)

{

e.printStackTrace();

}

**catch** (IOException e)

{

e.printStackTrace();

}

**catch** (ParserConfigurationException e)

{

e.printStackTrace();

}

**catch** (SAXException e)

{

e.printStackTrace();

}

Đầu tiên chúng ta để ý đến đối tượng PlaceModel đây là đối tượng để chứa các thông tin trả về gồm: tên địa điểm (name), địa chỉ của địa điểm (addresss) và tọa độ trên bản đồ (lat & lng).

Đoạn code:

StringBuilder urlString = **new** StringBuilder();

urlString.append("http://maps.google.com/maps?q=");

urlString.append(keyword); //nội dung tìm kiếm

urlString.append(Double.*toString*(lat));

urlString.append(",");

urlString.append(Double.*toString*(lng));

urlString.append("&num=");

urlString.append(Integer.*toString*(num));

urlString.append("&ie=UTF8&output=kml");

đây là nơi chúng ta sẽ xây dựng url cho việc lấy thông tin từ google map trong đó q=keyword là địa điểm chúng ta sẽ tìm kiếm, num= số lượng địa điểm tối đa được trả về, output=kml định dạng file trả về(ở đây chúng ta sẽ lấy về file KML để xử lý).

Tiếp theo:

String name = doc.getElementsByTagName("Placemark").item(i).getChildNodes().item(0) .getFirstChild().getNodeValue();

Đoạn code này sẽ đọc thông tin từ tag <name> trong file, và tương tự:

//Get the address of ATM

String address = doc.getElementsByTagName("Placemark")

.item(i).getChildNodes().item(2) .getFirstChild().getNodeValue();

Sẽ đọc thông tin từ <address> trong file KML.

Dưới đây là node thông tin trích ra từ file KML trả về:

<Placemark>

<name>HSBC</name>

<Snippet><![CDATA[No. 6 Nhà Th?, Hoan Kiem, Hanoi, Vietnam<br/>]]></Snippet>

<address>No. 6 Nhà Th?, Hoan Kiem&lt;br/&gt;Hanoi, Vietnam</address>

<StyleMap>

<Pair>

<key>normal</key>

<Style>

<IconStyle>

<Icon>

<href>http://maps.gstatic.com/intl/en\_ALL/mapfiles/kml/paddle/A.png</href>

</Icon>

<hotSpot x="0.500000" y="0.000000" xunits="fraction" yunits="fraction" />

</IconStyle>

<ListStyle>

<ItemIcon>

<href>http://maps.gstatic.com/intl/en\_ALL/mapfiles/kml/paddle/A-lv.png</href>

</ItemIcon>

</ListStyle>

</Style>

</Pair>

<Pair>

<key>highlight</key>

<Style>

<IconStyle>

<scale>1.300000</scale>

<Icon>

<href>http://maps.gstatic.com/intl/en\_ALL/mapfiles/kml/paddle/A.png</href>

</Icon>

<hotSpot x="0.500000" y="0.000000" xunits="fraction" yunits="fraction" />

</IconStyle>

<ListStyle>

<ItemIcon>

<href>http://maps.gstatic.com/intl/en\_ALL/mapfiles/kml/paddle/A-lv.png</href>

</ItemIcon>

</ListStyle>

</Style>

</Pair>

</StyleMap>

<Point>

<coordinates>105.850385,21.029302,0</coordinates>

</Point>

</Placemark>

Truy xuất thông tin đường đi

Tương tự đọa code dưới đây sẽ đọc thông tin tập hợp các điểm của một đường đi từ file KML mà google map trả về. Trong đó ta thấy ta đã sử dụng từ khóa saddr và daddr là địa điểm nguồn và địa điểm đích của đoạn đường

// Khai báo thông tin url đến google map

StringBuilder urlString = **new** StringBuilder();

urlString.append("http://maps.google.com/maps?f=d&hl=en");

urlString.append("&saddr=");//from

urlString.append( Double.*toString*((**double**)src.getLatitudeE6()/1.0E6 ));

urlString.append(",");

urlString.append( Double.*toString*((**double**)src.getLongitudeE6()/1.0E6 ));

urlString.append("&daddr=");//to

urlString.append( Double.*toString*((**double**)dest.getLatitudeE6()/1.0E6 ));

urlString.append(",");

urlString.append( Double.*toString*((**double**)dest.getLongitudeE6()/1.0E6 ));

urlString.append("&ie=UTF8&0&om=0&output=kml");

Log.*d*("xxx","URL="+urlString.toString());

//Lấy về file kml và đọc thông tin tọa độ

Document doc = **null**;

HttpURLConnection urlConnection= **null**;

URL url = **null**;

**try**

{

url = **new** URL(urlString.toString());

urlConnection=(HttpURLConnection)url.openConnection();

urlConnection.setRequestMethod("GET");

urlConnection.setDoOutput(**true**);

urlConnection.setDoInput(**true**);

urlConnection.connect();

DocumentBuilderFactory dbf = DocumentBuilderFactory.*newInstance*();

DocumentBuilder db = dbf.newDocumentBuilder();

doc = db.parse(urlConnection.getInputStream());

**if**(doc.getElementsByTagName("GeometryCollection").getLength()>0)

{

String path = doc.getElementsByTagName("GeometryCollection").item(0).getFirstChild().getFirstChild().getFirstChild().getNodeValue() ;

String [] pairs = path.split(" ");

String[] lngLat = pairs[0].split(","); // lngLat[0]=longitude lngLat[1]=latitude lngLat[2]=height

// src

GeoPoint startGP = **new** GeoPoint((**int**)(Double.*parseDouble*(lngLat[1])\*1E6),(**int**)(Double.*parseDouble*(lngLat[0])\*1E6));

mMapView01.getOverlays().add(**new** MyOverLay(startGP,startGP,1));

GeoPoint gp1;

GeoPoint gp2 = startGP;

**for**(**int** i=1;i<pairs.length;i++)

{

lngLat = pairs[i].split(",");

gp1 = gp2;

gp2 = **new** GeoPoint((**int**)(Double.*parseDouble*(lngLat[1])\*1E6),(**int**)(Double.*parseDouble*(lngLat[0])\*1E6));

mMapView01.getOverlays().add(**new** MyOverLay(gp1,gp2,2,color));

}

mMapView01.getOverlays().add(**new** MyOverLay(dest,dest, 3));

}

}

**catch** (MalformedURLException e)

{

e.printStackTrace();

}

**catch** (IOException e)

{

e.printStackTrace();

}

**catch** (ParserConfigurationException e)

{

e.printStackTrace();

}

**catch** (SAXException e)

{

e.printStackTrace();

}

Đưới đây là thông tin các tọa độ của một tuyến đường mà ta phải lọc ra để vẽ lại trên bản đồ:

<GeometryCollection><LineString><coordinates>106.702430,10.771050,0.000000 106.701460,10.773190,0.000000 106.700420,10.775660,0.000000 106.699900,10.776730,0.000000 106.695820,10.780500,0.000000 106.695820,10.780500,0.000000 106.698820,10.783700,0.000000 106.701610,10.786540,0.000000 106.705650,10.790870,0.000000 106.705650,10.790870,0.000000 106.707400,10.792750,0.000000 106.707400,10.792750,0.000000 106.709900,10.795390,0.000000 106.710040,10.795630,0.000000 106.710710,10.797360,0.000000 106.710930,10.798130,0.000000 106.711030,10.798640,0.000000 106.711270,10.801140,0.000000 106.711270,10.801140,0.000000 106.711470,10.801160,0.000000 106.711620,10.801270,0.000000 106.712340,10.801300,0.000000 106.713260,10.801220,0.000000 106.714010,10.801050,0.000000 106.716180,10.800350,0.000000 106.717940,10.799420,0.000000 106.717940,10.799420,0.000000 106.717860,10.799750,0.000000 106.716830,10.800260,0.000000 </coordinates></LineString></GeometryCollection>