

Google Earth Animation



動態 KML

台灣地理資訊學會
臺北市地理資訊推動顧問案計畫主持人
孫樹國 博士
E-mail: sksun@cs.nctu.edu.tw

Google Earth Animation 範例



- Fly wingman with a 3d model plane
- Polygon morph France into Spain!
- Dot-to-dot tour of European cities
- How Dams Could Worsen Climate Change
- Rotterdam bicycle

Google Earth Animation



動態元素介紹



控制動態元素



時間軸之應用

動態元素介紹

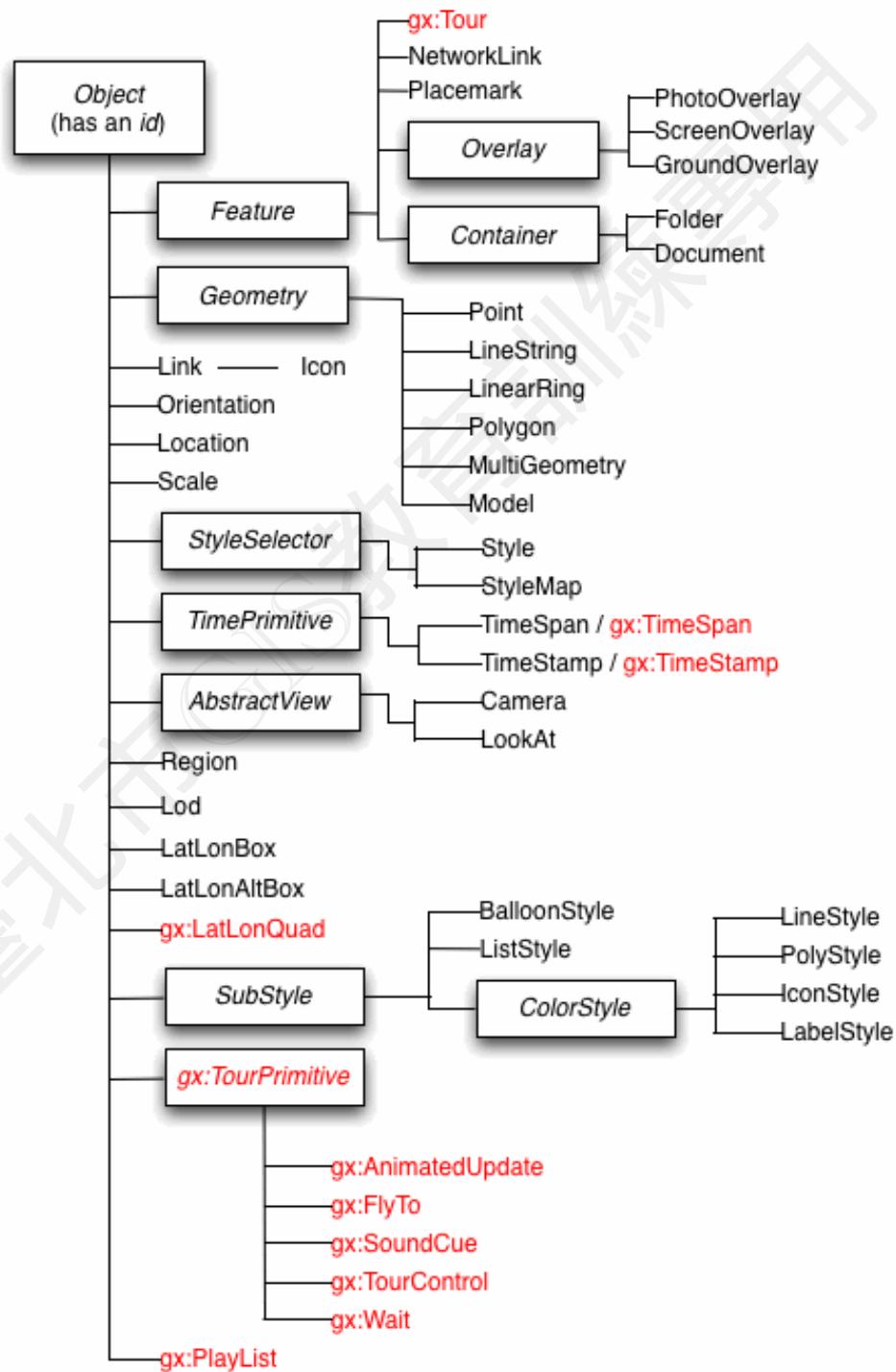


KML中的遊覽
遊覽時間表

:::建立遊覽:::



KML元素類樹



KML中的遊覽



- Google 地球5.0 中的新功能，可以透過地理空間數據控制飛行進行遊覽。
- 具有**gx**前綴(如遊覽相關gx:Tour, gx:FlyTo等)的元素，為OGC KML標準之擴展集。
- 要使用它們，必須在KML文件裡的開放<kml>元素中包含正確的名稱空間URI：

```
<kml xmlns="http://www.opengis.net/kml/2.2"  
      xmlns:gx="http://www.google.com/kml/ext/2.2" >
```

KML中的遊覽



在gx擴展集內與遊覽相關的元素

<gx:Tour>

<gx:playMode>

<gx:Playlist>

<gx:Wait>

<gx:Flyto>

<gx:AnimatedUpdate>

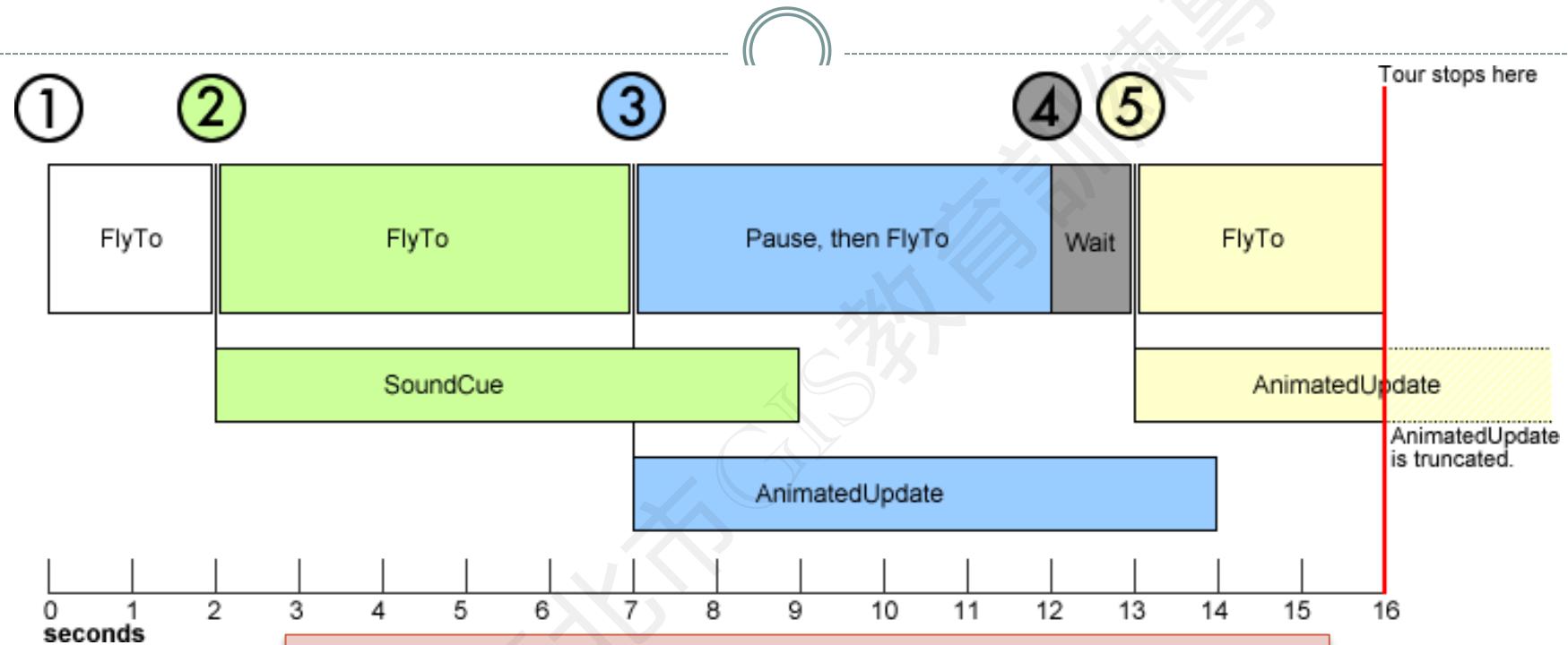
<gx:flyToMode>

<gx:SoundCue>

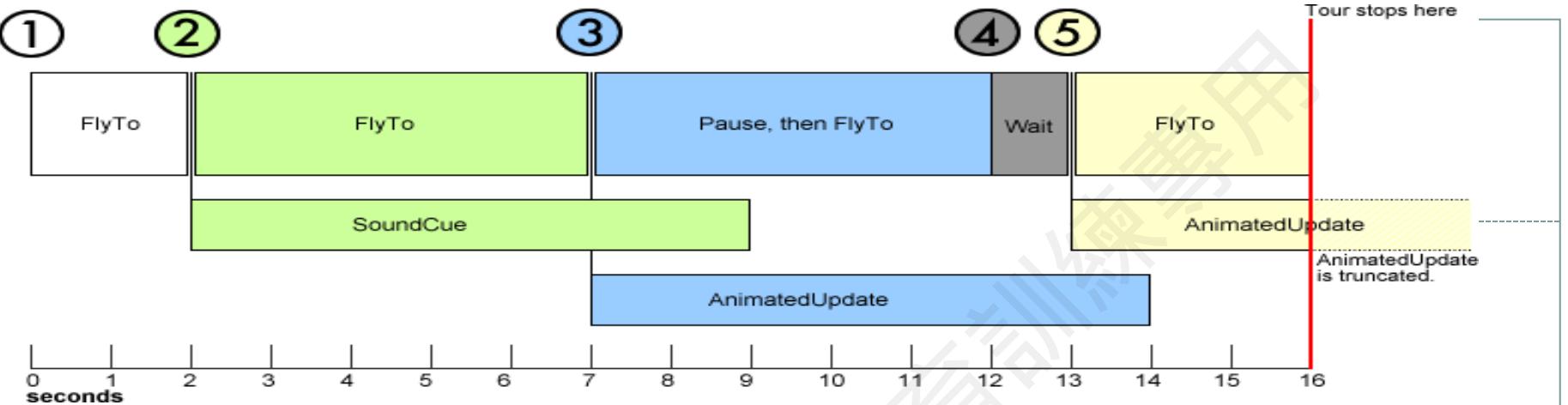
<gx:TourControl>

<gx:duration>

KML遊覽時間表 (gx:Tour)



```
<kml xmlns="http://www.opengis.net/kml/2.2"
      xmlns:gx="http://www.google.com/kml/ext/2.2">
  ...
  <gx:Tour>
    <name>Tour timeline example</name>
    <gx:Playlist>
      (1)
      ...
      (5)
    </gx:Playlist>
    <gx:Tour>
  </kml>
```



區分「連續出現的基元」及「並行出現的基元」。

●連續出現的基元。

<gx:FlyTo>

<gx:Wait>

<gx:TourControl>

□這些基元逐個出現，在播放清單移至任何類型的其他元素之前必須完全完成

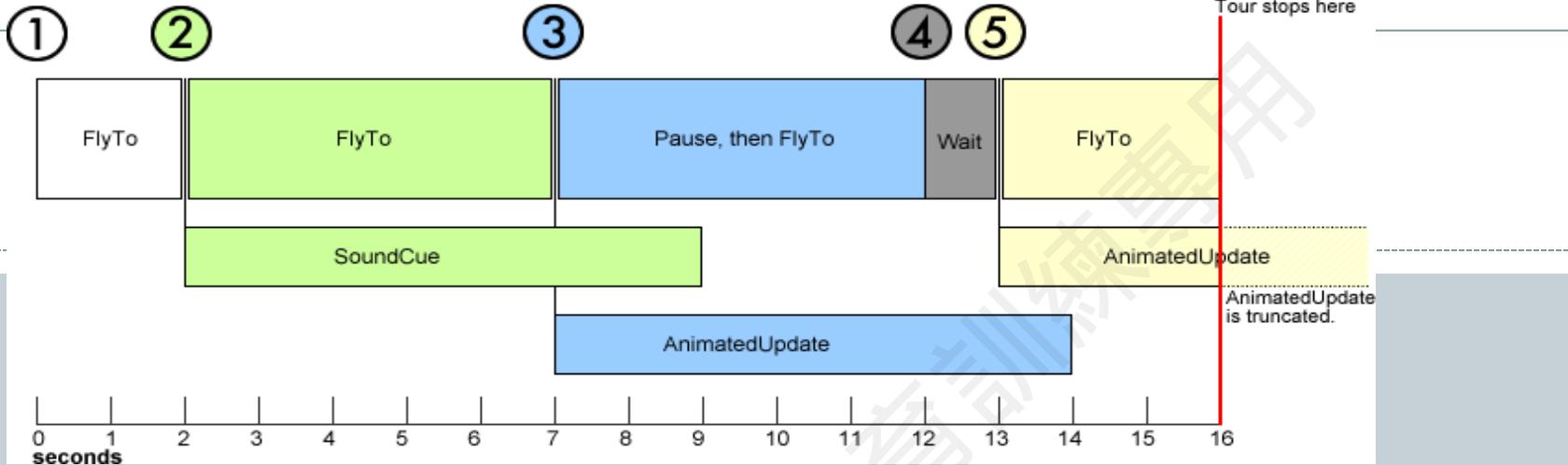
□一次只可以出現一個連續型的基元

●並行出現的基元

<gx:AnimatedUpdate>

<gx:SoundCue>

任何數量的 SoundCue 和/或 AnimatedUpdated 可以同時出現



```
<gx:FlyTo>
<gx:duration>2</gx:duration>
<gx:flyToMode>bounce</gx:flyToMode>
...
</gx:FlyTo>
```

```
<gx:FlyTo>
<gx:duration>5</gx:duration>
<gx:flyToMode>bounce</gx:flyToMode>
...
</gx:FlyTo>
<...>
<gx:SoundCue>
```

```
<gx:TourControl>
<gx:playMode>pause</gx:playMode>
</gx:TourControl>
```

```
<gx:AnimatedUpdate>
<gx:duration>7</gx:duration>
```

```
<gx:Wait>
<gx:duration>1</gx:duration>
</gx:Wait>
```

```
<gx:FlyTo>
<gx:duration>3</gx:duration>
<gx:flyToMode>bounce</gx:flyToMode>
...
</gx:FlyTo>
```

遊覽介紹



FlyTo

- <gx:FlyTo>

- 遊覽從一個位置轉到另一個位置。

- <gx:duration>

- 點之間的速度控制

<gx:FlyTo>

<gx:duration>4.0</gx:duration>

<LookAt>

<longitude>-121.303179</longitude>

<latitude>45.993041</latitude>

<altitude>0</altitude>

<heading>-27.923387</heading>

<tilt>71.600075</tilt>

<range>22570.546801</range>

<altitudeMode>relativeToGround</altitudeMode>

</LookAt>

</gx:FlyTo>

遊覽介紹



FlyToMode

- <gx:flyToMode>
 - 可指定smooth(平穩)或bounce(跳躍)
 - 初始FlyTo為跳躍模式(bounce)

```
<gx:FlyTo>
<gx:duration>4.0</gx:duration>
<gx:flyToMode>smooth</gx:flyToMode>
<LookAt>
<longitude>-121.303179</longitude>
<latitude>45.993041</latitude>
<altitude>0</altitude>
<heading>-27.923387</heading>
<tilt>71.600075</tilt>
<range>22570.546801</range>
<altitudeMode>relativeToGround</altitudeMode>
</LookAt>
</gx:FlyTo>
```

建立遊覽 (*gx:FlyTo*)



示意範例：從 A 視點至 B 視點

```
<gx:Tour>
<gx:Playlist>

<gx:FlyTo>
<gx:duration>2</gx:duration>
<gx:flyToMode>bounce</gx:flyToMode>
<LookAt>
<gx:horizFov>60</gx:horizFov>
<longitude>121.561286224182</longitude>
<latitude>25.0375558211563</latitude>
<altitude>0</altitude>
<heading>91.1309470039876</heading>
<tilt>60</tilt>
<range>100</range>
<gx:altitudeMode>relativeToGround</gx:altitudeMode>
</LookAt>
</gx:FlyTo>

<gx:FlyTo>
<gx:duration>9.34326733800614</gx:duration>
<gx:flyToMode>smooth</gx:flyToMode>
<LookAt>
```



建立遊覽 (*gx:flyToMode*)



bounce

- https://developers.google.com/kml/documentation/kmlfiles/bounce_example.kml

smooth

- https://developers.google.com/kml/documentation/kmlfiles/smooth_example.kml

控制動態元素



動態 KML 製作流程

:::PLACEMARK逐漸放大效果:::

:::加入模型產生動態效果:::

:::新增動態路徑:::



動態控制元素



AnimatedUpdate

- <**gx:AnimatedUpdate**>
 - 瀏覽KML地圖項過程中使用 **Update**控制更改。
- <**targetHref**>
 - 可指定.kml或.kmz網址中之元素進行修改。
- <**Update**>
 - 必須具有已定義之id，作為 targetId屬性中引用，進行變更、新增或刪除。

```
<gx:AnimatedUpdate>
<gx:duration>0.0</gx:duration>
<Update>
  <targetHref>...</targetHref>
  <Change>...</Change>
  <Create>...</Create>
  <Delete>...</Delete>
</Update>
</gx:AnimatedUpdate>
```

動態KML製作流程 (AnimatedUpdate)



Step 1：設定元素

- Placemark (Model、LineString)

Step 2：建立遊覽

- Tour
- FlyTo (duration)

Step 3：新增動態效果

- AnimatedUpdate (duration)

範例一

Placemark逐漸放大效果

(控制元素：
Placemark)

(example1.kml)



控制動態元素

範例一：Placemark逐漸放大效果



Step 1：設定元素

- 設定樣式

```
<Style id="pushpin">  
  <IconStyle id="mystyle">  
    <Icon>  
      <href>http://maps.google.com/mapfiles/kml/pu  
      shpin/ylw-pushpin.png</href>  
      <scale>1</scale>  
    </Icon>  
  </IconStyle>  
</Style>
```

- 新增**Placemark**

```
<Placemark id="mountainpin1">  
  <name>Pin on a mountaintop</name>  
  <styleUrl>#pushpin</styleUrl>  
  <Point>  
    <coordinates>170.1435558771009,  
    43.60505741890396,o</coordinates>  
  </Point>  
</Placemark>
```

範例一：Placemark逐漸放大效果



Step 2：建立遊覽

```
<gx:Tour> <name>Play me!</name> <gx:Playlist>  
<gx:FlyTo>  
<gx:flyToMode>bounce</gx:flyToMode>  
<gx:duration>3</gx:duration>  
<Camera>  
<longitude>170.157</longitude>  
<latitude>-43.671</latitude>  
<altitude>9700</altitude>  
<heading>-6.333</heading>  
<tilt>33.5</tilt>  
</Camera>  
</gx:FlyTo>
```

... (Step3: 新增動態效果)

```
<gx:Wait> <gx:duration>5</gx:duration> </gx:Wait>  
</gx:Playlist> </gx:Tour>
```

Step 3：新增動態效果

```
<gx:AnimatedUpdate>  
<gx:duration>5</gx:duration>  
<Update>  
<targetHref></targetHref>  
<Change>  
<IconStyle targetId="mystyle">  
<scale>10.0</scale>  
</IconStyle>  
</Change>  
</Update>  
</gx:AnimatedUpdate>
```

範例二

加入模型產生動態效果
(控制元素 : Model)



控制動態元素

範例二：加入模型產生動態效果

Step1: 設定模型

- <Model>
 - 新增一個模型於Placemark中
- <altitudeMode>
 - clampToGround (內定值)、relativeToGround 、absolute
- <Location>
 - 用經度、緯度和海拔高度指定模型原點的精確坐標。
- <Orientation>
 - 模型坐標之朝向、傾度及旋轉
- <Scale>
 - 模型坐標空間沿X、Y、Z軸方向之縮放

```
<Placemark>
  <name>yaris</name>
  <Model id="model_1">
    <altitudeMode> relativeToGround </altitudeMode>
    <Location id="yarisLocation">
      <longitude>121.5601003926589</longitude>
      <latitude>25.03899395043037</latitude>
      <altitude>0</altitude>
    </Location>
    <Orientation id="yarisOrientation">
      <heading>0</heading>
      <tilt>0</tilt> <roll>0</roll>
    </Orientation>
    <Scale> <x>0.1</x> <y>0.1</y> <z>0.1</z> </Scale>
    <Link>
      <href>C:\Users\ntugis\Desktop\dae\yaris.dae</href>
    </Link>
  </Model>
</Placemark>
```

範例二：加入模型產生動態效果



Step 2：建立遊覽

```
</Update>
</gx:AnimatedUpdate>

<gx:FlyTo>
<gx:duration>2</gx:duration>
<LookAt>
  <gx:horizFov>70</gx:horizFov>
  <longitude>121.562093987949</longitude>
  <latitude>25.037561257978</latitude>
  <altitude>0</altitude>
  <heading>91.4494376853489</heading>
  <tilt>70</tilt>
  <range>100</range>
  <altitudeMode>
    relativeToGround</altitudeMode>
  </LookAt>
</gx:FlyTo>

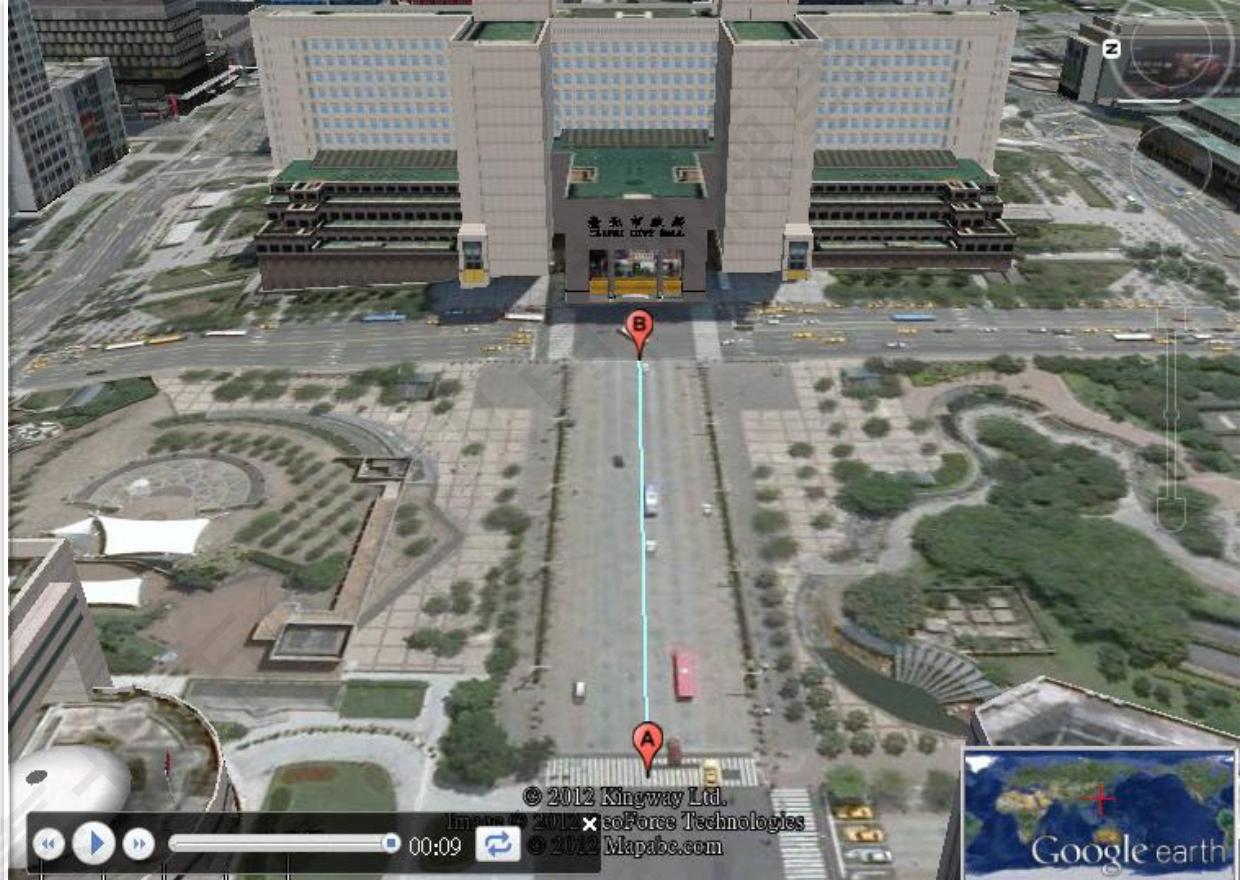
<gx:AnimatedUpdate>
<gx:duration>7.99582998597238</gx:duration>
<Update>
<targetHref></targetHref>
```

Step3：模型動態控制

```
<gx:AnimatedUpdate>
<gx:duration>7.99582998597238</gx:duration>
<Update>
<targetHref></targetHref>
<Change>
<Location targetId="yarisLocation">
<longitude>121.563416242834</longitude>
<latitude>25.0375278011507</latitude>
<altitude>0</altitude>
</Location>
<Orientation targetId="yarisOrientation">
<heading>91.4494376853489</heading>
</Orientation>
</Change>
</Update>
</gx:AnimatedUpdate>
```

範例三

新增動態路徑
(控制元素：LineString)



控制動態元素

範例三：新增動態路徑



Step1：設定元素

- 可調整線段顏色、寬度

```
<Style id="m_icon">  
    ...  
    <LineStyle>  
        <color>fffffaa</color>  
        <width>4</width>  
    </LineStyle>  
</Style>
```

- 新增**LineString**

```
<Placemark>  
    <styleUrl>#m_icon</styleUrl>  
    <LineString id="line1">  
        <coordinates>121.562093,25.0375612,0</coordinates>  
    </LineString>  
</Placemark>
```

只有一
個起點!!

範例三：新增動態路徑



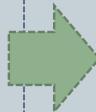
Step2: 建立遊覽

```
<gx:Tour>
<name>Click to Play!</name>
<gx:Playlist>

<gx:FlyTo>
  <gx:duration>7</gx:duration>
  <LookAt>
    <gx:horizFov>60</gx:horizFov>
    <longitude>121.562093987949</longitude>
      <latitude>25.037561257978</latitude>
    <altitude>0</altitude>
    <heading>91.4494376853489</heading>
    <tilt>60</tilt>
    <range>100</range>
    <altitudeMode>relativeToGround</altitudeMode>
  </LookAt>
</gx:FlyTo>
</gx:Playlist>
</gx:Tour>
```

Step3：新增動態路徑

```
<gx:AnimatedUpdate>
  <gx:duration>7</gx:duration>
  <Update>
    <targetHref></targetHref>
    <Change>
      <LineString targetId="line1">
        <coordinates>
          121.5620939879489,25.03756125797801,0
          121.5634162428335,25.03752780115074,0
        </coordinates>
      </LineString>
    </Change>
  </Update>
</gx:AnimatedUpdate>
```



時間軸之應用



認識時間軸
動態KML製作流程

===== EXAMPLE =====

:::臺北市區界圖:::

:::利用GOOGLE EARTH產生動態效果:::

:::TIMESTAMP和TIMESPAN之差異性:::



認識時間軸



TimeSpan

- 代表由開始和結束 “**dateTimes**” 限定的時間範圍。
- <**begin**>
 - 描述某時間段的開始瞬間。
- <**end**>
 - 描述某時間段的結束瞬間。
- 如果缺少<**begin**> 或<**end**>，則不會限定時段的起點或終點。

- **示例：**

- 以下示例顯示代表科羅拉多州存在時間的時間段。它只包含<**begin**> 標籤，因為科羅拉多州在**1876** 年**8** 月**1** 日成為一個州，之後一直是一個州

```
<Placemark>
  <name>Colorado</name>
  ...
  <TimeSpan>
    <begin>1876-08-01</begin>
  </TimeSpan>
</Placemark>
```

認識時間軸



TimeStamp

- 表示時間軸中的某個瞬間。
- <when>
 - 其值是“dateTime”，可以是以下其中一個：
 - “dateTime”指明秒精度
 - “date”指明日精度
 - “gYearMonth”指明月精度
 - “gYear”指明年精度

```
<TimeStamp>
<when>1997</when>
</TimeStamp>
```

```
<TimeStamp>
<when>1997-07</when>
</TimeStamp>
```

```
<TimeStamp>
<when>1997-07-16</when>
</TimeStamp>
```

```
<TimeStamp>
<when>1997-07-16T07:30:15Z</when>
</TimeStamp>
```

認識時間軸



<gx:TimeSpan>和<gx:TimeStamp>

- <**TimeSpan**>和<**TimeStamp**>元素的副本。
- 允許在**AbstractViews**中加入時間值（<**Camera**>和<**LookAt**>）。
- 用於控制歷史影像、日照效果以及具有時間性之地圖呈現項目。

- **示例：**

```
<Placemark>
  <name>Sutro Baths in 1946</name>
  <Camera>
    <gx:TimeStamp>
      <when>1946-07-29T05:00:00-08:00</when>
    </gx:TimeStamp>
    <longitude>-122.518172</longitude>
    <latitude>37.778036</latitude>
    <altitude>221.0</altitude>
    <heading>70.0</heading>
    <tilt>75.0</tilt>
  </Camera>
</Placemark>
```

動態KML製作流程 (TimeLine)



Step 1：元素（加入時間軸）

- Placemark (TimeStamp)
- Polygon (TimeSpan)

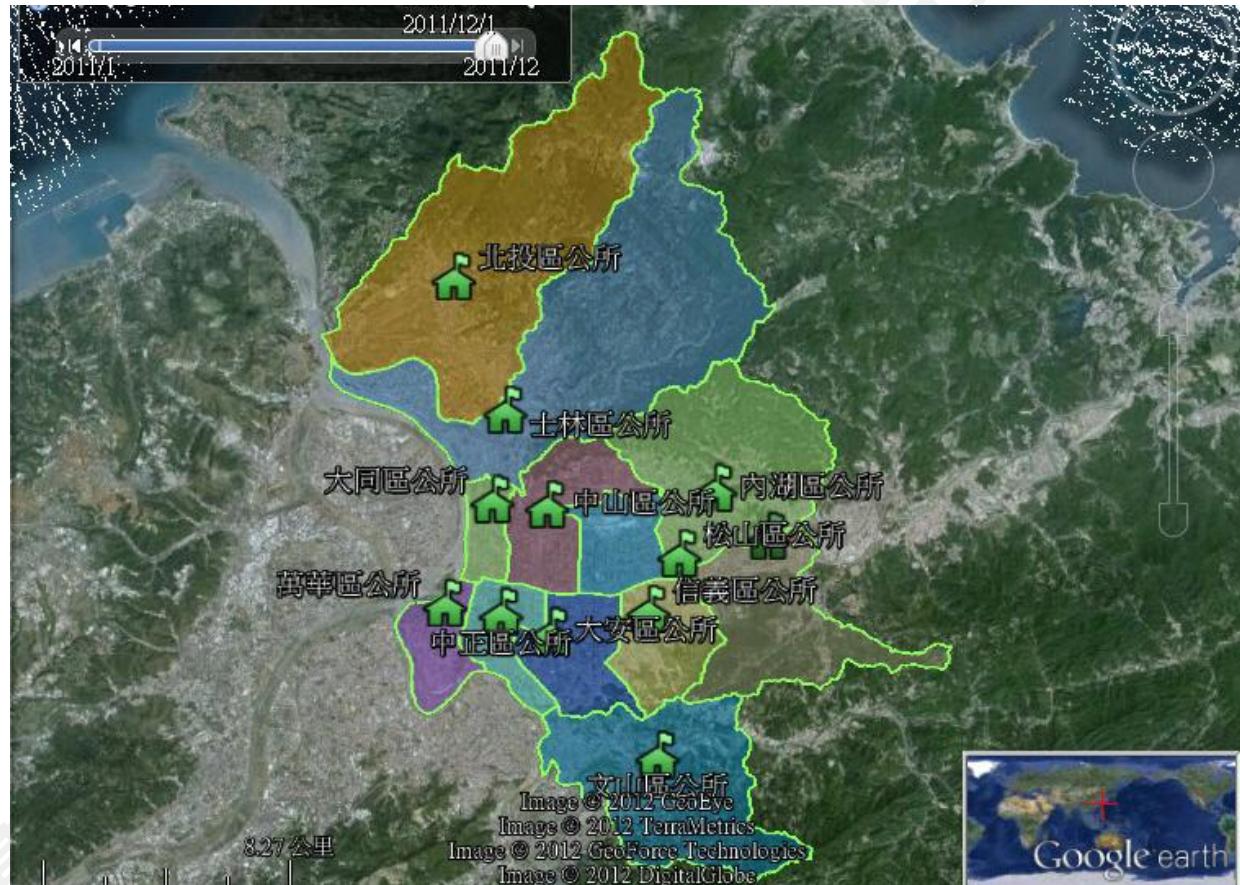
Step 2：建立遊覽（加入時間軸）

- Tour
- FlyTo (duration)

Step 3：產生連動效果

範例四

臺北市區界圖 (TimeSpan)



時間軸之應用

臺北市區界圖

Step 4-1 :

- 新增Polygon元素

Step 4-2 :

- 在每個多邊形地標中
插入TimeSpan之時
間差

```
...
<Placemark>
  <name>臺北市文山區</name>
  <TimeSpan>
    <begin>2011-01</begin>
  </TimeSpan>
  <styleUrl>#m_ylw-pushpin</styleUrl>
  <Polygon>
    <outerBoundaryIs>
      <LinearRing>
        <coordinates>
          121.542195656376,24.9879387022263,0
          121.542303947707,24.9879507491782,0
          ...
        </coordinates>
      </LinearRing>
    </outerBoundaryIs>
  </Polygon>
</Placemark>

<Placemark>
  <name>臺北市萬華區</name>
  <TimeSpan>
    <begin>2011-02</begin>
  </TimeSpan>
  <styleUrl>#m_ylw-pushpin</styleUrl>
  <Polygon>
    <outerBoundaryIs>
      ...
      ...
    </outerBoundaryIs>
  </Polygon>
</Placemark>
```

臺北市區界圖

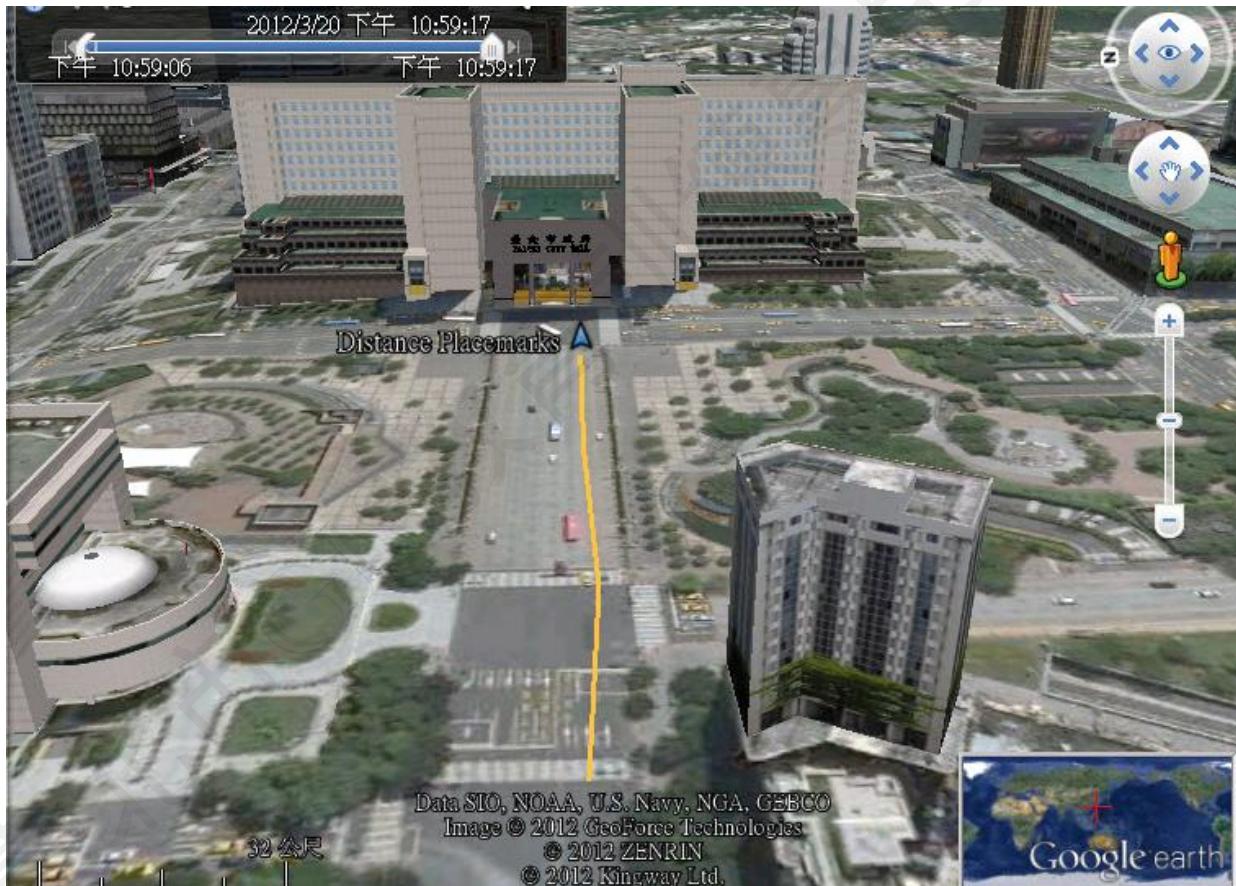
Step 4-3：

- 完成後可於左上時間軸播放動態效果
- 設定時間播放速度及重播控制如右圖



範例五

利用Google Earth
產生動態效果
(TimeStamp)

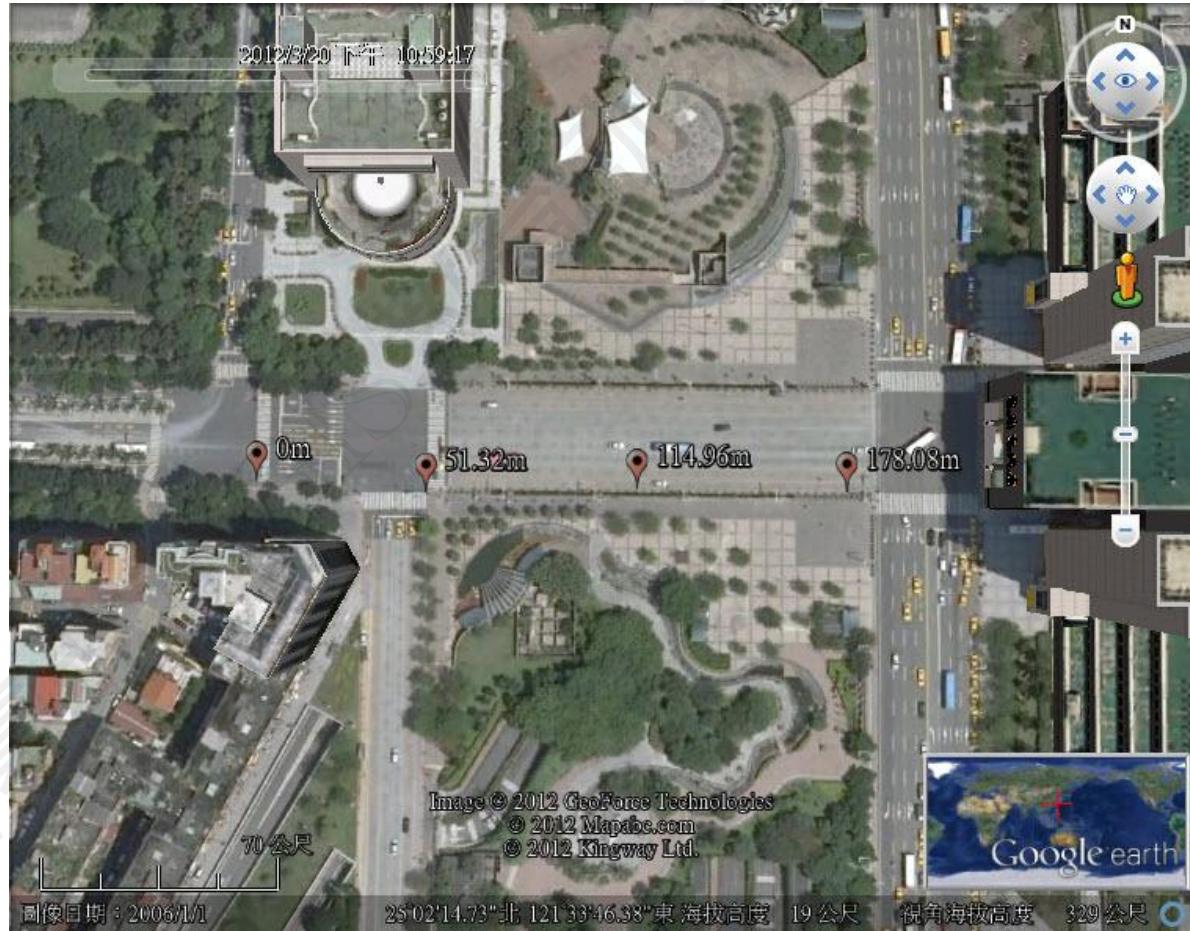


時間軸之應用

利用Google Earth 產生動態效果

Step 5-1 :

- 於Google Earth中新增數個地標(Placemark)
- 將位置另存為KML



利用Google Earth 產生動態效果

Step 5-2 :

- 開啟KML編輯器
- 在每個Placemark間依循插入時段性之
<TimeStamp>
- 完成編輯後，重新開啟KML檔案

```
...
<Placemark>
<name>om</name>
<TimeStamp>
  <when>2012-03-20T14:59:06Z</when>
</TimeStamp>
<styleUrl>#m_icon</styleUrl>
<Point>
  <coordinates>
    121.5615659865427,25.03749721513404,0
  </coordinates>
</Point>
</Placemark>

<Placemark>
<name>51.32m</name>
<TimeStamp>
  <when>2012-03-20T14:59:09Z</when>
</TimeStamp>
<styleUrl>#m_icon</styleUrl>
<Point>
  <coordinates>
    121.5620739133503,25.03746231185273,0
  </coordinates>
</Point>
</Placemark>
...
...
```

利用Google Earth 產生動態效果

Step 5-3：

- 開啟KML後，Google Earth左上角即呈現時間軸
- 在KML的folder按右鍵「以歷程格式複製」
- 同KML的folder按右鍵「貼上」



利用Google Earth 產生動態效果

Step 5-4：

- 貼上後呈現畫面如右
- 利用左上時間軸可控制速度



利用Google Earth 產生動態效果

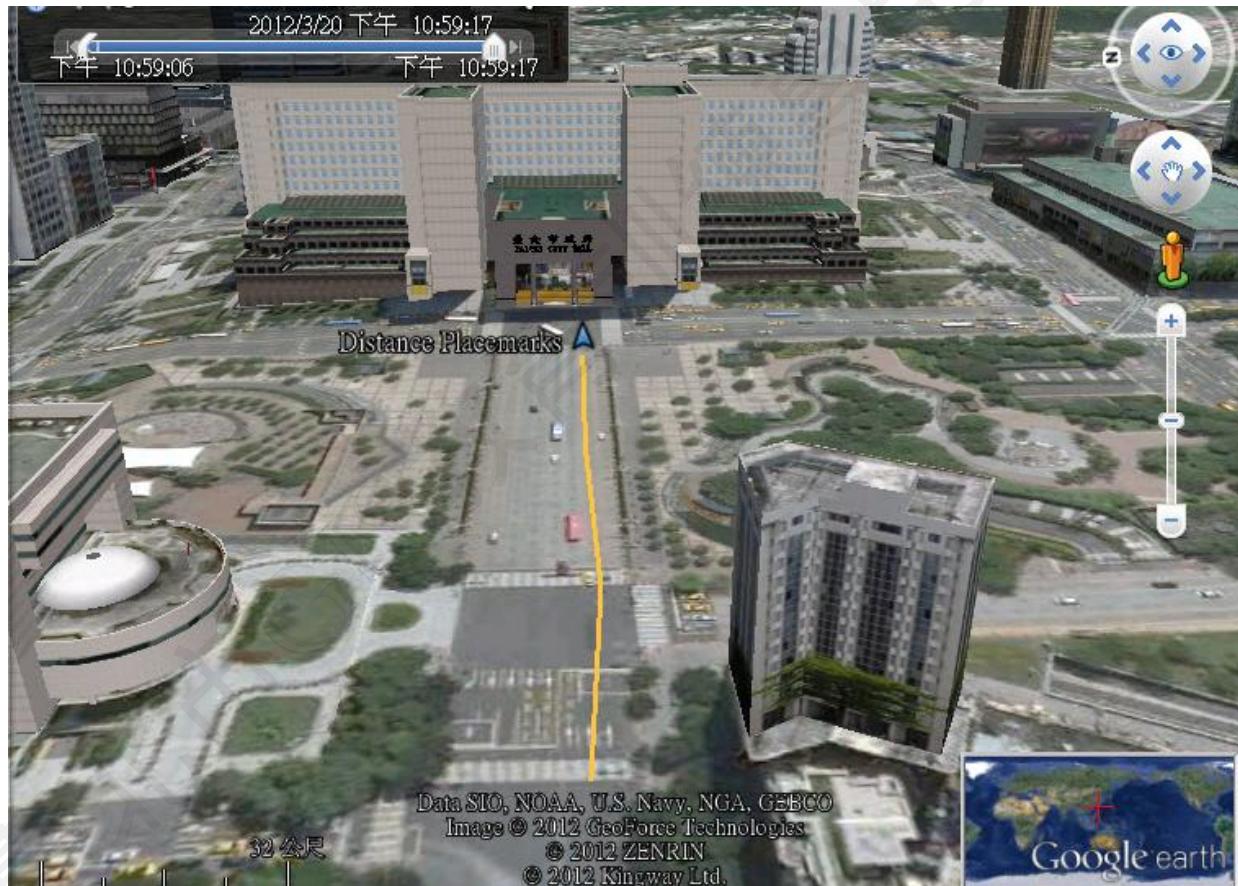
Step 5-4 :

示例：以歷程格式複
製CODE

- <?xml version="1.0" encoding="UTF-8"?>
- <kml xmlns="http://www.opengis.net/kml/2.2"
 xmlns:gx="http://www.google.com/kml/ext/2.2"
 xmlns:kml="http://www.opengis.net/kml/2.2"
 xmlns:atom="http://www.w3.org/2005/Atom">
- <Placemark>
- <name>Distance Placemark</name>
- <open>1</open>
- **<gx:Track>**
- <when>2012-02-29T09:48:00Z</when>
- <when>2012-02-29T09:48:10Z</when>
- <when>2012-02-29T09:48:20Z</when>
- <when>2012-02-29T09:48:30Z</when>
- <when>2012-02-29T09:48:40Z</when>
- <when>2012-02-29T09:48:50Z</when>
- <gx:coord>121.5599426882465 25.038754521770</gx:coord>
- <gx:coord>121.5598461738954 25.037604726110</gx:coord>
- <gx:coord>121.5606048421189 25.0376172455710</gx:coord>
- <gx:coord>121.5616669789933 25.037467296690</gx:coord>
- <gx:coord>121.5627842886269 25.03740482450</gx:coord>
- <gx:coord>121.5633912234965 25.037404829740</gx:coord>
- <ExtendedData>
- <Data name="name">
- <value>Distance Placemark</value>
- </Data>
- </ExtendedData>
- **</gx:Track>**
- </Placemark>
- </kml>

範例六

TimeStamp和TimeSpan
之差異性



時間軸之應用

地標遊覽效果

Example 6-1 :

- 新增遊覽播放功能
- 加入
<gx:TimeStamp>
- 產生時間軸與地標之連動效果



```

<gx:Tour>
<name>Click to Play!</name>
<gx:Playlist>

<gx:FlyTo>
  <gx:duration>3 </gx:duration>
  <gx:flyToMode>smooth</gx:flyToMode>
  <LookAt>
    <gx:horizFov>60</gx:horizFov>
    <gx:TimeStamp>
      <when>2012-03-20T14:59:06Z</when>
    </gx:TimeStamp>
    <longitude>121.56207391335</longitude>
    <latitude>25.0374623118527</latitude>
    <altitude>0</altitude>
    <heading>93.9310228077966</heading>
    <tilt>60</tilt>
    <range>100</range>
    <gx:altitudeMode>relativeToGround</gx:altitudeMode>
  </LookAt>
</gx:FlyTo>
<gx:FlyTo>
  <gx:duration>4 </gx:duration>
  <gx:flyToMode>smooth</gx:flyToMode>
  <LookAt>
    <gx:horizFov>60</gx:horizFov>
    <gx:TimeStamp>
      <when>2012-03-20T14:59:09Z</when>
    </gx:TimeStamp>
    <longitude>121.562705619559</longitude>
    <latitude>25.0374685141091</latitude>
    <altitude>0</altitude>
    <heading>89.4374731675123</heading>
    <tilt>60</tilt>
    <range>100</range>
    <gx:altitudeMode>relativeToGround</gx:altitudeMode>
  </LookAt>
</gx:FlyTo>
...
  
```

經過時間14:59:06

經過時間14:59:09

地標及路徑 遊覽效果

Example 6-2 :

- 新增遊覽播放功能
- 加入`<gx:TimeSpan>`
- 產生時間軸和地標及路徑運動



```

<gx:Tour>
<name>Click to Play!</name>
<gx:Playlist>

    { <gx:FlyTo>
        <gx:duration>3 </gx:duration>
        <gx:flyToMode>smooth</gx:flyToMode>
        <LookAt>
        <gx:horizFov>60</gx:horizFov>
        <gx:TimeSpan>
            <begin>2012-03-20T14:59:06Z</begin>
            <end>2012-03-20T14:59:09Z</end>
        </gx:TimeSpan>
        <longitude>121.56207391335</longitude>
        <latitude>25.0374623118527</latitude>
        <altitude>0</altitude>
        <heading>93.9310228077966</heading>
        <tilt>60</tilt>
        <range>100</range>
        <gx:altitudeMode>relativeToGround</gx:altitudeMode>
        </LookAt>
    </gx:FlyTo>
    { <gx:FlyTo>
        <gx:duration>4 </gx:duration>
        <gx:flyToMode>smooth</gx:flyToMode>
        <LookAt>
        <gx:horizFov>60</gx:horizFov>
        <gx:TimeSpan>
            <begin>2012-03-20T14:59:06Z</begin>
            <end>2012-03-20T14:59:13Z</end>
        </gx:TimeSpan>
        <longitude>121.562705619559</longitude>
        <latitude>25.0374685141091</latitude>
        <altitude>0</altitude>
        <heading>89.4374731675123</heading>
        <tilt>60</tilt>
        <range>100</range>
        <gx:altitudeMode>relativeToGround</gx:altitudeMode>
        </LookAt>
    </gx:FlyTo>
    ...
    ...
}

```

經過3秒，結束時間累計3秒

經過4秒，結束時間累計7秒

完成動畫播放效果

- 左上為「時間軸」
- 左下為「播放遊覽」
- 按下所貼上歷程的播放遊覽，產生運動的動畫效果

