

实验四 设计 KML 几何类

实验目的

1. 了解面向对象的基本概念和使用面向对象技术进行程序设计的基本思想；
2. 理解了解 C++ 程序设计的基本概念和使用 C++ 进行程序设计的基本思想，进行一般面向对象的程序设计；
3. 熟练掌握 C++ 数据类型、类和对象、控制结构、函数等基础知识；
4. 利用 VC.net 读写文本文件，并能够初步将 GIS 数据文件读取；
5. 了解 Google KML 几何类的继承结构，为后续的地理信息系统专业课程打下一个良好的基础。

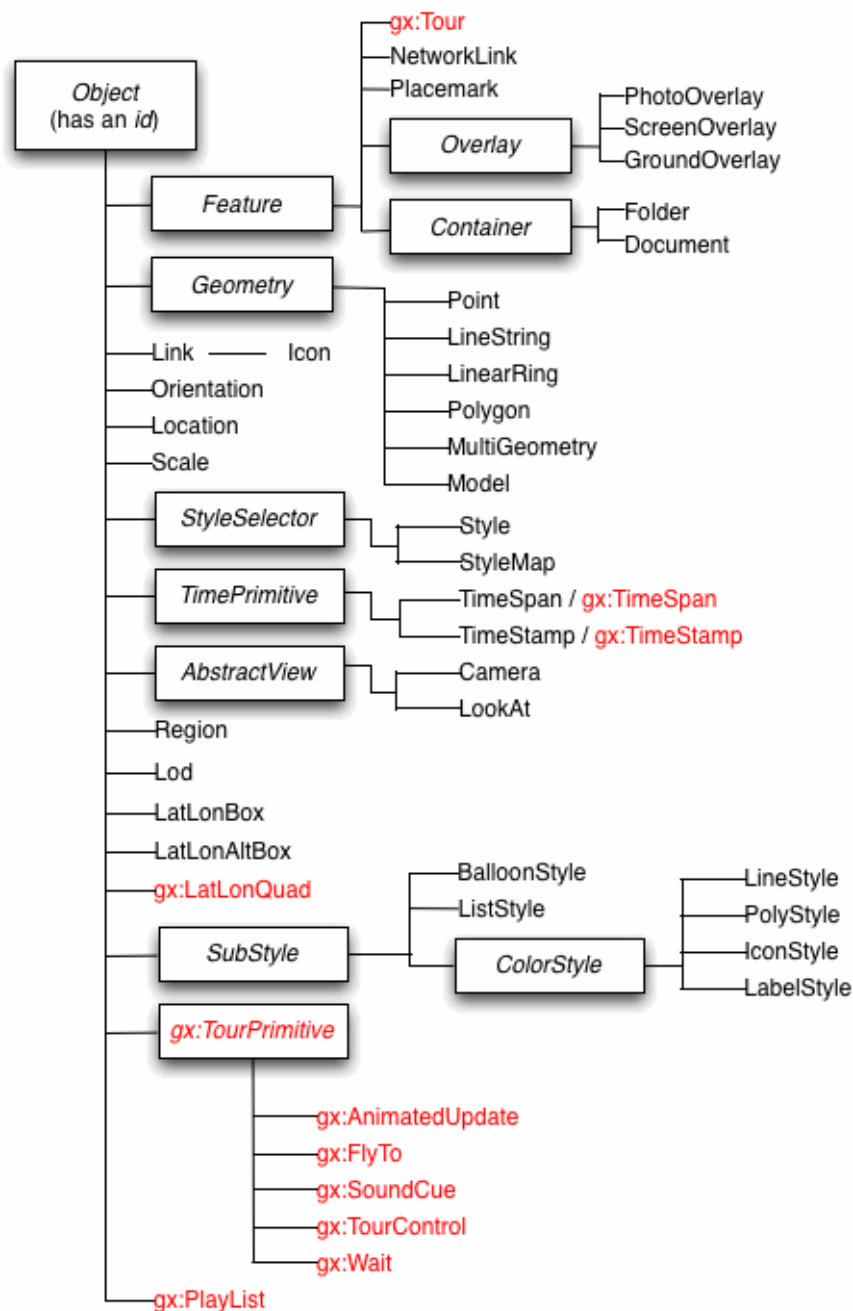
实验内容

KML 是一种文件格式，用于在地球浏览器（例如 Google 地球、Google 地图和谷歌手机地图）中显示地理数据。KML 使用含有嵌套的元素和属性的基于标记的结构，并符合 XML 标准，Google 已经将 KML2.2 提交到 OGC，成为 OGC 的规范。所有标记都区分大小写，而且完全如 KML 参考中列出的那样显示。该参考指出了哪些标记是可选的。在指定元素内，标记必须按照参考中显示的顺序显示。如果您尚不熟悉 KML，请查看 <http://code.google.com/intl/zh-CN/apis/kml/documentation/>。

1. KML 几何类设计

通过 KML 向 Google 地球用户界面中添加的地图项：地标、说明、地面叠加层、路径和多边形等。KML 中使用要素 Feature 来描述现实世界中的实体对象，并将与空间坐标相关的属性抽取出来，称为几何体 (Geometry)。利用面向对象的程序设计和 C++ 类的组织方法，组织建立起一个基本的 KML 几何类。设计地标 Placemark 和文档 Document 等类型要素对应的 C++ 类；实现处理点 Point、折线 LineString、封闭环 LinearRing 及多边形区域 Polygon 等几何图形，针对每类几何图形设计对应的 C++ 类，把每个图形元素作为一个独立的对象来管理。

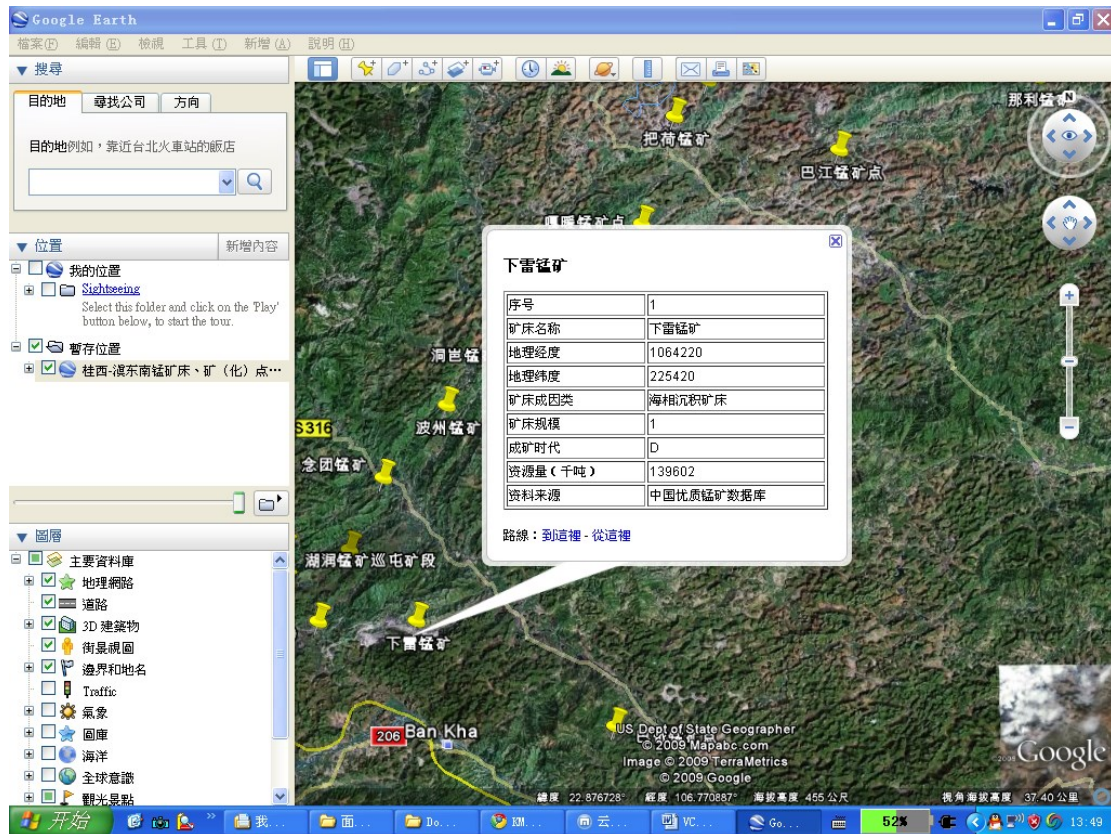
对各类要素 Feature 和几何图形 Geometry 进行分析，可以发现多个类都具有有一些相同的属性和操作功能，可以把这些共性的东西（属性和操作）组织存放在一个基类中，其他类由这个类派生。



上图给出了 google KML 类之间的层级结构图，其中带方框的为抽象基类，无方框的为可实例化的类。

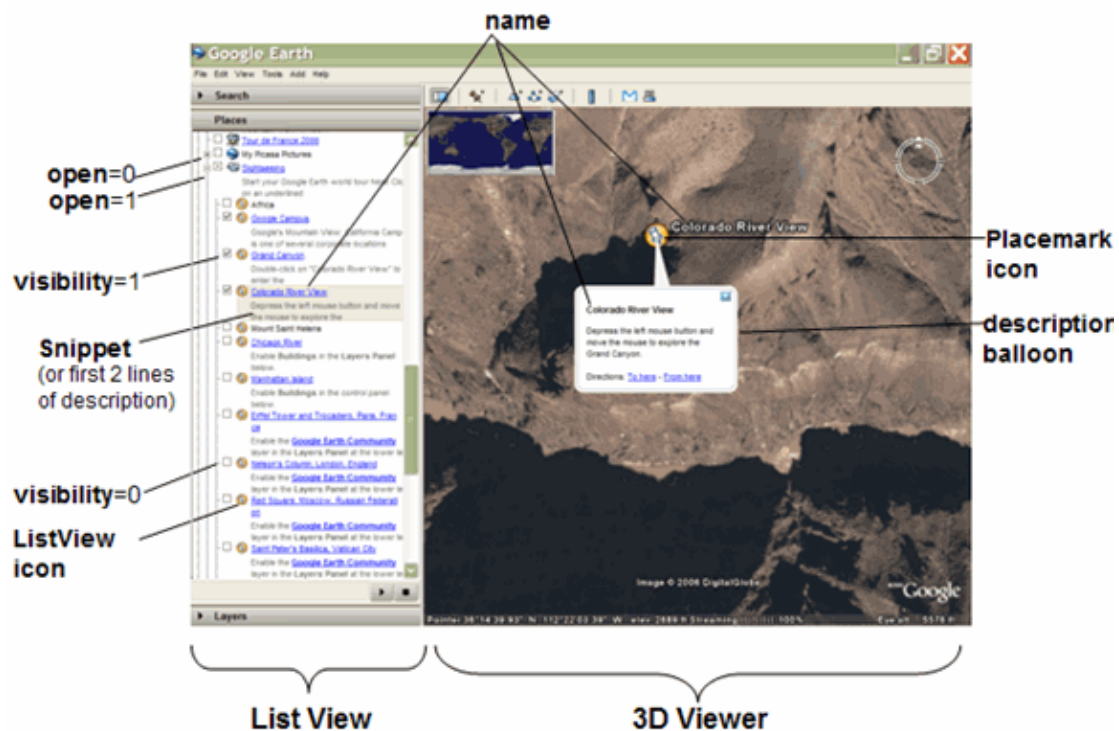
要素类 Feature 和几何类 Geometry 由同一个抽象基类——对象类 Object 派生。

要素抽象基类 Feature 包含有一个名称 name 和描述 description 属性，其中 description 属性支持 HTML 编码。



要素抽象基类 Feature 的派生类：

1. 地标 Placemark：1 个几何 Geometry 派生类对象；
2. 容器 Container：抽象基类，派生文档类 Document 和文件夹类 Folder；
3. 文档 Document：1-n 个 Feature 派生类对象；
4. 文件夹 Folder：1-n 个 Feature 派生类对象。



几何图形抽象基类 Geometry 的派生类:

1. 点 Point: 1 个坐标 Coord(Longitude, Latitude, Altitude);
2. 折线 LineString: 由 2-n 个坐标组成, 数目是不确定的;
3. 封闭环 LinearRing: 由 4-n 个坐标组成, 数目是不确定的, 首尾坐标重合形成封闭环;
4. 多边形: 由一个外封闭环 outerBoundary 和 0-n 个内封闭环 innerBoundary 组成;
5. 几何组合 MultiGeometry: 1-n 个几何 Geometry 派生类对象。

2. 示例: KML 文本文件输出

从 Microsoft Excel 的 CSV 文件(逗号分隔符文本文件)中读取数据, 生成多个地标 Placemark, 并将地标放在一个文档 Document 中; 将文档 Document 及其内的地标 Placemarks 输出为用 “.kml” 或 “.kmz” 扩展名保存文本文件 (出现乱码时, 将文本文件另存为 UTF-8 编码方式即可), 地球浏览器会知道如何显示它。

也可以设计文件转换类 Csv2Kml 来实现下类功能:

1. 读取 CSV 文件;
2. 创建 Document 及其 Placemarks 对象;
3. 输出 KML 文件。

KML 文件示例

```
<?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">
```

```
<Document>
  <name>桂西-滇东南锰矿床、矿（化）点一览表</name>
  <Placemark>
    <name>下雷锰矿</name>
    <description>
      <![CDATA[
<table border="1" padding="3" width="300">
<tr><td>序号</td><td>1</td></tr><tr><td>矿床名称</td><td>下雷锰矿
</td></tr><tr><td>地理经度</td><td>1064220</td></tr><tr><td>地理纬
度</td><td>225420</td></tr><tr><td>矿床成因类</td><td>海相沉积矿床
</td></tr><tr><td>矿床规模</td><td>1</td></tr><tr><td>成矿时代
</td><td>D</td></tr><tr><td>资源量（千吨）
</td><td>139602</td></tr><tr><td>资料来源</td><td>中国优质锰矿数据
库</td></tr></table>
]]>

      </description>
    <Point>
      <coordinates>
        106.702, 22.9061, 0
      </coordinates>
    </Point>
  </Placemark>
  <Placemark>
    <name>富宁县洞波乡至周锰矿</name>
    <description>
      <![CDATA[
<table border="1" padding="3" width="300">
<tr><td>序号</td><td>36</td></tr><tr><td>矿床名称</td><td>富宁县洞
波乡至周锰矿</td></tr><tr><td>地理经度</td><td>1054239-
1054628</td></tr><tr><td>地理纬度</td><td>234254-
234504</td></tr><tr><td>矿床成因类</td><td>沉积锰矿床
</td></tr><tr><td>矿床规模</td><td>4</td></tr><tr><td>成矿时代
</td><td>D1</td></tr><tr><td>资源量（千吨）
</td><td>866</td></tr><tr><td>资料来源</td><td>中国优质锰矿数据库
</td></tr></table>
]]>

      </description>
    <MultiGeometry>
      <Point>
        <coordinates>
          105.735, 23.7314, 0
        </coordinates>
      </Point>
    </MultiGeometry>
  </Placemark>
</Document>
```

```

        </Point>
        <LineString>
            <coordinates>
                105.701,23.7064,0
                105.768,23.7564,0
            </coordinates>
        </LineString>
        <LineString>
            <coordinates>
                105.701,23.7564,0
                105.768,23.7064,0
            </coordinates>
        </LineString>
        <Polygon>
            <outerBoundaryIs>
                <LinearRing>
                    <coordinates>
                        105.701,23.7064,0
                        105.768,23.7064,0
                        105.768,23.7564,0
                        105.701,23.7564,0
                        105.701,23.7064,0
                    </coordinates>
                </LinearRing>
            </outerBoundaryIs>
        </Polygon>
    </MultiGeometry>
</Placemark>
</Document>
</kml>

```

实验要求

1. 写出程序，并调试程序，要给出实验代码和实验结果 KML。

要求实现添加点、线、多边形和几何组合的地标及对应描述的函数，参考代码如下：

```

int main()
{
    cout << "Hello KML World!\n";
    ofstream outfile;
    outfile.open(_T("data//Travel.kml"));
    //

```



```

Document doc;
string docName = "MyTravel";
doc.setName(docName);
tArray<Placemark> pa;
//点, 封装成函数
Point pt;
pt.setLongitude(123.85);
pt.setLatitude(42.28);
pa[0].setGeometry(&pt);
string des = "<![CDATA[ \n";
des += "<table border=\"1\" padding=\"3\" width=\"300\">\n";
//第一行
des += "<tr>";
des += "<td> Name </td>";
des += "<td> Travel </td>";
des += "</tr>";
//第二行
des += "<tr>";
des += "<td> Year </td>";
des += "<td> From 2022 </td>";
des += "</tr>";

des += "</table>\n";
des += "]]>\n";

pa[0].setDescription(des);
//线, 封装成函数
LineString ls;

//多边形, 封装成函数
Poly poly;

//几何组合, 封装成函数
MultiGeometry mg;

doc.setPlacemarks(pa);

outfile << "<?xml version=\"1.0\" encoding=\"UTF-8\"?>" << endl;
outfile << "<kml xmlns=\"http://" << "/www.opengis.net/kml/2.2\">" << endl;
outfile << "xmlns:gx=\"http://" << "/www.google.com/kml/ext/2.2\">" << endl;
//<!-- required when using gx-prefixed elements -->
outfile << doc;
outfile << "</kml>" << endl;

```

```
    outfile.close();  
}
```

2. 整理上机步骤，总结经验和体会。
3. 完成实习报告和上交程序。