

文章编号: 1005-0523(2006)04-0071-04

基于 Mapinfo 开发的南昌市地图位置与公交查询系统

占自才

(华东交通大学 电气与电子工程学院, 江西 南昌 330013)

摘要:介绍了一种基于 Mapinfo 二次开发的地图位置及公交查询系统. 本系统可以很方便查询出用户所要查找的地图上有的位置, 并把它按一定比例放大定位在视图中央. 由于用户有时只记得要查地名的部分名称, 所以该系统还设计了模糊查询. 在公交查询的设计里, 用户很容易找到从起始点到目的点所有公交车, 指明要乘几站, 便于用户作出具体选择. 当没有直达车时并提供了最佳转车位置. 本系统独特与创新之处就是查询效率高, 设计简单.

关键词:二次开发; 定位查询; 公交查询

中图分类号: TH163

文献标识码: A

1 引言

基于 Mapinfo 开发的各种 GIS 信息系统很多, 有城市的规划图、水土保持以及一些查询系统等. 但很多查询系统很少进行模糊查找和精确定位, 并结合开发公交查询系统. 城市公共交通运输以其覆盖面广、经济快捷的特点, 目前仍然是绝大多数出行者的首选方式, 也是各地城市政府大力发展的一种交通方式^[1]. 如果能够提供一种服务, 为市民特别是外来旅游、出差、就医等急需了解本地道路情况的人提供方便、快捷、经济、高效地利用公交线路的方案, 将方便他们的出行和生活, 同时减少不必要的交通流量, 提高交通运输的效率和城市的地位. 这已是一个越来越迫切的现实问题. 另外在城市交通查询系统中, 有一个重要的问题就是地面查询, 因为有些市民在出差、旅游、探亲、访友或就医时往往只知道目的地的地名, 甚至目的地的地名也记不全, 而他们的目的地在城市的具体位置往往是不知道的, 所以在基于 Mapinfo 的基础上开发了一种很实用的模糊地名精确定位查询系统, 方便了市民出

行. 在西方国家(如美国、日本等), 在城市公交网络系统上投入了极大的财力, 构建了利用计算机网络和先进的通信系统的智能运输系统 ITS (Intelligence Transport System), 从而实现了大范围内全方位发挥的实时、准确、高效的运输综合管理, 使人、车、路密切配合, 和谐统一, 大大改善了交通环境等^[2]. 国内在这方面的起步较晚, 但是各地政府都给予了极大的重视, 相继建立了网上公交查询系统^[3]. 但效果不是很好, 通常图形浏览方式查询, 查询速度慢, 方法单一. 本文结合基于 Mapinfo 模糊地面精确地理位置查询和模糊目的地公交线路查询设计的地名与公交查询系统较为方面地满足了用户的需求.

2 用 VC++ 及其 MFC 基础类调用 DMapinfo 的函数实现地名查询

用 VC++ 建立单文档程序框架, 在建立程序向导中第四步选择 Automation 多选项按完成即可^[4]. 把 MapBasic 里的 mapbasic.h 文件拷贝到工程文件夹中并加入到工程的头文件中^[5]. 在 VC++ View 菜单的 Class Wizard... 中打开 MFC 类向导对话框, 在 Add Class... 菜单选项中选择 From a type li-

收稿日期: 2006-05-21

作者简介: 占自才(1969-), 男, 江西乐平人, 讲师. 主要从事图像处理、地理信息及智能控制方面的工作.

brary...加入 Mapinfo 类,从安装了 Mapinfo 的子目录 Mapinfo/Professional 文件夹中选择 Mapinfo.tlb 文件加入,选择包含 DMapinfo 类相关类加入到项目中.这样就把 DMapinfo 类加入到了工程中.在工程中打开的 Mapinfo.h 文件,在该文件的最后加入如下语句:

```
extern DMapInfo mapinfo;
#include "mapbasic.h"
```

在程序框架的应用程序类、主帧框类、文档类及视图类的 CPP 文件中加入语句:

```
#include "Mapinfo.h"
```

在应用程序类的 CPP 文件中构成函数后面加入全局变量:

```
DMapInfo mapinfo;
```

在 CMainFrame 类的 OnCreate()函数的最后加入下列语句:

```
CString strCommand;
```

```
strCommand.Format("Set Application Window %lu", (long)m_hWnd);
```

```
mapinfo.Do(strCommand);
```

用 Mapinfo 软件制作地图,把最新版的南昌地图制作成“南昌.tab”文件,并加入地面位置图层,其文件名为“位置.tab”,其中有“地名”字段且能索引,再加入公交站点图层为“公交.tab”,这三个图层都放在“南昌地图”文件夹中,再把整个文件夹放入与查询系统应用程序相同的文件夹中.

在 CView 类的 OnCreate()函数中使用打开图层语句加入各图层:

```
mapinfo.Do("Open Table \". . . \ 南昌地图 \ 南昌.tab \ " ReadOnly");
```

其中省略号表示南昌地图所在的文件夹,其它图层也类似加入.

在 Cview 类的 OnInitialUpdate()函数中加入以下语句:

```
CString strCommand;
```

```
strCommand.Format("Set Window Info Parent %u", m_hWnd);
```

```
mapinfo.Do(strCommand);
```

```
mapinfo.Do("Set Window Info ReadOnly");
```

```
CFindzipDoc * pDoc = GetDocument();
```

```
if (pDoc! = NULL) {
```

```
pDoc->SetCallback();
```

及对 Mapinfo 进行回调.到这里,查询系统各图层已经加入到系统中了.

调用缩放、手抓等功能,通过调用 mapinfo.RunMenuCommand(UINT ref)函数,ref 为功能键在 Mapinfo 中的 ID 号,这个可以到 Resource.h 文件中查询.如手抓 ID 号为 MM_TOOLBAR_GRABBER 或 1702.

在视图菜单或工具条加入一个消息函数,在此函数中处理地面查询.把南昌地图中所有标明的地址都写入一个文本文件中,设计一个对话框,在对话框中可以输入用户地名的关键字,在文本文件中搜索出具有关键字的所有地名放入一个列表控键中让用户挑选,用户选中的地名返回到消息函数中.调用下列语句实现地名查询:

```
mapinfo.Do("Find Using 位置(地名)");
```

```
CString strCommand;
```

```
strCommand.Format("Find \">%s \", (LPCTSTR)用户选出的地名字符串);
```

```
mapinfo.Do(strCommand);
```

```
if (atoi(mapinfo.Eval("CommandInfo(3)")) >= 1) {
```

```
strCommand.Format (
```

```
"Set Map Window %lu Center (CommandInfo(1), CommandInfo(2)) Zoom 0.075", m_hWndID);
```

```
mapinfo.Do(strCommand);
```

这样,就已经实现了模糊地名的精确位置查询了,查询的结果是把用户查询的地名显示在地图的中央.

3 公交线路查询设计

在用 Mapinfo 制作表的时候,公交站点信息就已经标注在公交图层里了,用户可以浏览各路车的运行路程.但是这只相当于一张普通的地图,查询起来很慢且不方便.本文设计了一种公交查询方法,查询速度快且很方便.首先把所有的公交线路及其各站点都依次存放在一个文本文件“公交.Txt”中,设计一个对话框,在对话框中实现关键字始终点车线路查询.

用两个编辑框用来输入用户的始终点站关键字,从车线路文本文件中搜索出具有始点关键字所有站点名放入一个列表框中,搜索出具有终点关键字的站点名放入另一列表框中,用户必须从两个列表框中选择确切的始终点站名,按搜索键就可以找

到符合用户要求的公交线路乘坐. 因为搜索的结果包含用户从始点到终点的站数, 所以用户可以选择最佳线路乘车.

搜索的算法比较烦杂^[6], 现介绍如下:

步 1: 分别从始终点编辑框中输入准确的始终点车站名, 定义记录被搜索到公交线路的数目 $n=0$;

步 2: 判断是否到了“公交.Txt”文件末尾, 若是则转步 5, 否则从“公交.Txt”依次读入一条公交线路名及该线路的各个站点名(每个公交线路名及该线路的各个站点名都是一行文本, 公交线路名由数字命名, 中文冒号结束, 每个站点名之间也由一个中文冒号隔开.) 附给一个“CString”类对象变量“m_strGongjiao”;

步 3: 对“m_strGongjiao”每一个冒号之间的站点名都与待找的始点站名和终点站名比较是否相等, 若“m_strGongjiao”中既有与始点站名也有与终点站名相等的两个站名则转步 4, 否则转步 2;

步 4: $n=n+1$, 把被搜索到线路显示在静态文本框中, 并计算该公交车线路从被搜索到的起始站到终点站的站次和方向, 再把这些信息一并显示在对话框的同一静态文本框中让用查阅, 同时转步 2;

步 5: 判断 $n>0$ 否, 若 $n>0$, 则结束本次搜索, 否则继续下一步;

步 6: 因为 $n=0$, 说明没有被搜索到从目的始点站到目的终点站的直达公交线路, 故而进行转车线路和转车站点的搜索, 使“公交.Txt”指到文件头准备重新搜索, 并定义 $l=0$ 记录可以换乘的站点的个数;

步 7: 判断是否搜索文件末尾, 若是则转步 9, 否则搜索“公交.Txt”文件, 找到具有目的始点站的公交线路或目的终点站的公交线路分别附给字符串类成员变量 $m_strGongjiao1$, $m_strGongjiao2$, 判断 $m_strGongjiao1$, $m_strGongjiao2$ 两个成员变量中是否有相同的站点名, 若有则转步 8, 否则转步 7;

步 8: $l=l+1$, 并把搜索到的相同站点名显示在对话框的静态文本中, 此站点即时换车站点, 并转步 7;

步 9: 判断 $l>0$ 否, 若是, 则结束整个搜索, 否则显示信息“至少要换乘两次公交车”在静态文本框中.

通过搜索得到的有可能的三种结果: 一种以上的直达车到达目的地并知道到达目的的共有几站; 提供了一种以上的换车站点名; “至少要换乘两

次公交车”信息提供给用户. 根据不同的搜索结果, 用户可以采取相应的处理方式. 若第一种结果, 用户可以直接选择自己要乘最有效的公交线路; 若是第二种结果, 以目的始点站为始点站, 换车站点为终点站搜索出换乘前的公交线路, 再以换车站点为始点站, 原目的终点站为终点站搜索出换乘后的公交线路; 若是第三种结果, 用户只有换目的始点站名或目的终点站名再次搜索直到得到满意的结果.

4 系统界面及操作的简单说明

系统操作的主界面如图 1 所示, 一打开主界面, 放大、缩小与手抓浏览功能都放在视图菜单与工具条中. 点击工具条中左数第五个工具就弹出一个查找目的对话框(如图 2 所示). 用户只要输入目的地的关键字, 按“搜索”键就可把含有关键字的所有地名都列在下面的编辑框中, 选中编辑框中的一个地名, 如华东交通大学, 并双击它或按“确定”键, 华东交通大学就按一定的缩放比例显示在界面的最中央.

点击主界面工具条的最后一个图标就弹出一个查询公交车对话框(如图 3 所示). 如用户要查华东交通大学到青山路口的公交线路可采用以下方法: 在图 3 的对话框中分别输入如图的始终点站名的关键字, 按“搜索”键就得到所有具有相应关键字的始终点站名, 选定确切的始终点站名, 按“搜索搭乘线路”键, 则被搜索到的所有符合要求的公交线路就显示在对话框的文本显示框中. 用户根据这些信息就可作出相应判断, 为自己的乘车出行提供方便.

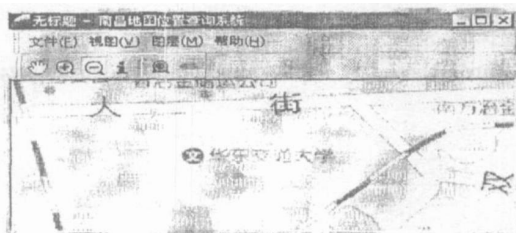


图 1 系统主界面

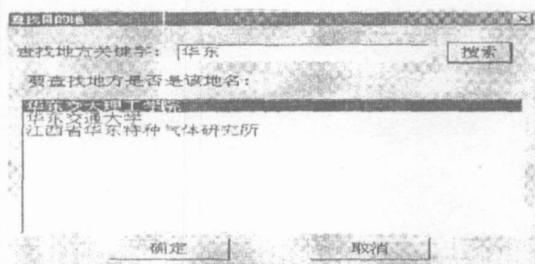


图 2 查找目的地对话框

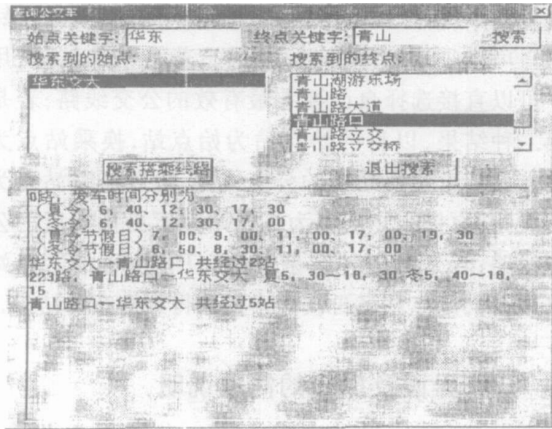


图3 查询公交车对话框

5 结语

本文利用 Mapinfo 二次开发功能,借助于 VC++ 高效的编程与运行效率设计的一种实用而又简

单的地理信息查询系统,为今后的城市交通信息与智能控制系统提供了一种有效的思路.这样既可得到满意的结果,又能减轻开发时间.

参考文献:

- [1] 陈立潮.城市交通智能咨询系统的设计与实现[J].计算机工程,2003,(1):32~34.
- [2] 邵祖峰.城市道路交通堵塞治理研究[J].城市交通,2005,(1):71~73.
- [3] 金凌,吴存钱.公交网络地理信息系统平台的研究开发[J].城市公共交通,2005,(1):40~43.
- [4] David J. Kruglinski, Scot Wingo. Visual C++ 6.0 技术内幕[M].北京:北京希望电子出版社,1999.
- [5] 祝晓鹰,余峰,惠玉洁.用 VB、VC 实现 Mapinfo 的回调[J].计算机系统应用,2004,(7):71~75.
- [6] 徐兵,谢仕义.基于站点优先级的公交换乘算法实现[J].计算机时代,2005,(7):16~17.

A System about Nanchang Map Position and Demanding Based on Mapinfo Impoldering

ZHAN Zi-cai

(School of Electronical and Electrical Eng., East China Jiaotong Univ., Nanchang 330013, China)

Abstract: This paper introduced a system about map position and demanding based on Mapinfo secondary impoldering. User can expediently look up his demanding position in the system and orient it in center of view. Because user can only know few words about his position, the system projected also fuzzy demanding. User can easily find his bus and counts of station to his position. If no direct bus to his position, he will be told place where he change. It is efficient and easy project that the system's unique innovation.

Key words: secondary impoldering; position demanding; bus demanding