

基于MFC的文件快速查找软件开发

李传可

(上海政法学院图书馆, 上海 201701)

摘要: 在VS2010开发平台上以MFC基于对话框为开发框架, 实现对指定目录路径的自动保存, 并在该路径下根据选择的文件类型, 进行快速检索的功能。实现逻辑为读取EDIT控件上的路径, 然后调用MFC基类CFileFind的成员函数, 遍历该路径下的所有文件, 并将文件名等相关数据同步到Report格式的List控件中, 最后通过鼠标双击响应函数来调用系统API函数ShellExecute()打开指定文件。

关键词: MFC; List控件; 遍历; 软件开发; 文件查找

中图分类号: TP311.5 **文献标识码:** A

Software Development of Fast File Searching Based on MFC

LI Chuanke

(Shanghai University of Political Science and Law, Shanghai 201701, China)

Abstract: In Visual Studio 2010 platform, use dialog-based MFC as developing frame, to realize the auto-saving of the specified path to configurate file and quick search in accordance with the selected file types through such path. After reading the path of Edit Control, call member function of CFileFind Class to traverse all files under that path, then copy file names and other related data to list control in LVS_REPORT Mode. Lastly call the ShellExecute API function on the MouseEvent function with Double-Clicking to open selected file by ShellExecute().

Keywords: MFC; List control; traversal; software development; file search

1 引言(Introduction)

MFC(Microsoft Foundation Classes)是微软公司开发的具有图形界面的C/C++基础类库, 封装了大部分的Windows API函数。通过VS2010(Microsoft Visual Studio 2010)一个集成的开发环境, 在该平台上新建一个MFC的工程, 会自动生成相应的程序文件及外部库文件等。由于该应用程序框架提供了大量的Windows句柄封装类, 以及控件、组件的封装类, 就大大提高了软件的开发效率, 使软件工程师可以把更多的精力放在算法及逻辑的实现上。同时因具备图形化操作界面, 使用户可以快速适应并掌握软件的使用技巧^[1]。

2 开发该软件的必要性(The necessity of the software development)

随着办公文件的不断增加, 如我校图书馆采编部由于需要整理查找大量的图书相关数据, 导致整个桌面几乎都被excel文件占据, 最终影响了用户的正常操作。对于在工作中需处理大量文件的工作人员而言, 如果能快速有效地找到并打开目标文件将极大地提高工作效率^[2]。虽然Windows系统自带了文件搜索功能, 但因其功能有限, 实际使用中并不能很有效的解决用户所遇到的问题, 因此需要开发一款符合用户

实际操作需求, 用于文件查找的专业软件。

3 Windows系统搜索功能的局限性(Limitations of the search function of Windows OS)

3.1 无法自动保存查找的路径

用户一般会事先将相关文件放置在同一目录下, 而系统自带的搜索功能由于无法对上次搜索的目录进行记忆, 只要系统重启或者文件查找框被关闭后就需要重新输入目录, 才能够在指定目录下进行文件的查找, 即存在重复的操作, 降低了用户的使用体验。

3.2 不能针对某种文件类型查找

由于使用习惯或者用途的不同, 目前使用最多的是Excel、Word、txt等常用类型文件, 用户往往希望能指定某种文件类型进行查找, 而系统自带的搜索功能却又无法针对某种或几种类型进行专门查找, 只能盲目地将所有包含该关键字的文件都进行查找导致查找效率降低。

3.3 缺少关键字模糊查找功能

目前, Windows系统仅支持关键字100%匹配度的查找法, 即前后字符顺序必须完全匹配才行, 但实际操作过程中, 用户并非完全能保证前后顺序的正确性, 甚至不确定是

否有这些关键字，这种情况下就对用户的查找增加了难度。

4 开发该软件的整体框架和布局(The overall framework and layout of the software development)

在微软推出的VS2010开发平台上进行MFC基于对话框的软件开发，利用自动生成的MFC类库可快速完成整个MFC程序的软件框架搭建，通过自带且功能强大的控件功能，不仅快速完成桌面的布局，且由于MFC封装的很浅，通过继承基类，及调用外部库文件函数，可根据自身功能需求完成子类控件及功能类的二次开发。开发语言使用C++进行编写^[3]，利用VS2010集成开发环境提供的调试和编译环境，不断进行测试和错误追踪，逐步地完成所有功能的开发。在界面的布局上，利用MFC编程的可视化操作^[4]，对各控件的布局进行科学合理的优化，最大限度地实现文件的高效查找目标。

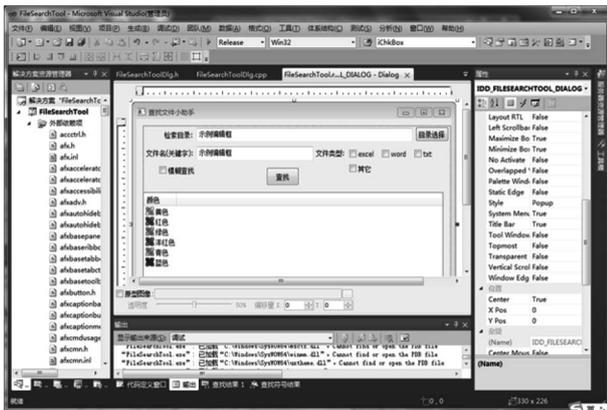


图1 MFC基于对话框生成的开发环境

Fig.1 The development environment of dialog-based MFC

4.1 源文件及资源视图设计

源文件无疑是整个程序最重要的可执行文件，也是软件开发人员编写代码的地方。其中外部依赖库是已经封装好功能强大的代码库，对于开发人员而言，只需包含调用即可。因此开发与设计的重点在于头文件类成员变量与函数的声明及源文件的逻辑设计。该软件的头文件主要有：FileSearchTool.h与FileSearchToolDlg.h，源文件主要有：FileSearchTool.cpp与FileSearchToolDlg.cpp，资源视图文件主要是FileSearchTool.rc。头文件主要用于声明各控件类对象、成员变量及成员函数等。而源文件负责所有程序的执行流程，主要包括文件路径的读写、指定路径下的文件查找及各个控件的响应函数等。资源视图界面则负责软件整个窗口及各控件的布局，其中Edit控件主要有检索目录输入框、文件名关键字输入框及选定文件的完整路径显示框；Button控件主要是目录选择按钮、查找按钮；Check Box控件包括文件类型、模糊查找复选框；List控件则用于文件信息显示列表框。根据用户使用习惯调整各控件尺寸大小并进行合理布局^[5]。

4.2 静态库文件及Unicode字符集

为保证软件的可移植性，以及对所有语言的支持，选择使用静态库，即在程序编译时就会被连接到目标代码中，程序运行时就不再需要该静态库，保证在其他用户电脑中也能正常使用。而动态库则是在程序运行时才被载入，程序运行时就一直需要动态库文件的存在，因此不利于程序的移植。同时为保证程序的跨语言、跨平台，实现对多国语言支持，选择Unicode字符集，使字符在不同的语言系统环境下仍能正常显示不出现乱码，解决了传统字符编码方案的局限性^[6]。



图2 配置属性设置

Fig.2 Setting configuration properties

4.3 文件查找功能实现逻辑

遍历指定目录下的所有文件，根据选定的文件类型，找到含有关键字的文件，并将该文件的名称、路径等信息全部插入到List控件中，然后根据先后顺序将符合查找条件的文件名从上往下逐行排列显示。最后用户通过鼠标双击目标文件名的方式来快速打开文件。除此之外，为保证软件的实用性特意增加了模糊查找等功能。



图3 实现逻辑图

Fig.3 A image of implementation logic diagram

5 各个功能模块的编程实现(Programming implementation of each functional module)

5.1 目录选择按钮

用于用户手动选择所需要的检索目录，通过鼠标单击弹出浏览文件夹对话框，根据目录选择指定的文件夹。获取的路径信息将赋值给成员变量m_strPath,用于保存指定路径的信息。实现方法为：首先定义BROWSEINFO结构体变量，然后给该变量的各个成员变量赋值，主要是标题、窗口样式等，再调用::SHBrowseForFolder()函数打开文件夹选择对话框，最后使用::SHGetPathFromIDList()函数获取选择的目录路径。

5.2 检索目录输入框

该Edit控件负责显示要查找的目录路径，执行查找功能后会将该路径写入到配置文件中，实现每次打开软件后自动读取配置文件中的数据，通过对配置文件进行读写操作，节约了输入路径或者目录选择的时间。具体实现步骤为：首先定义CFile类对象file，接着调用该对象的文件类操作函数file.Open(_T("filepath.txt"), CFile::modeCreate|CFile::modeReadWrite)函数，配置文件打开成功后再执行写操作：file.Write()和读操作：file.ReadString()^[7]。



图4 配置文件

Fig.4 Configuration file

5.3 文件类型判断功能

文件类型主要分为Excel、Wrod、txt及其他类型，通过调用Check-box控件的消息响应函数来判断选择的文件类型，用户根据目标文件类型可进行单选或多选。如下图所示通过只输出选定类型的文件名信息，检索出来的文件数量就大大减少，提高了查找效率。实现方法为：首先定义CFileFind类对象find，然后获取文件扩展名的前三个字符CString strFileType=find.GetFileName().Mid(find.

GetFileName().Find(_T("."))+1,3),最后与选定的文件扩展名前三个字符进行比较:if((0==iChkBox[0]&&_T("xls")==strFileType)).

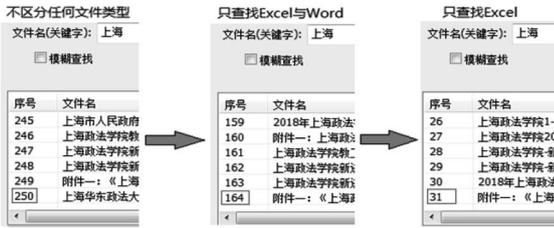


图5 不同文件类型的查找结果比较

Fig.5 Comparing the search results of different file types

5.4 关键字与模糊查找功能

关键字查找顾名思义,是将关键的字符与所有的文件名进行逐一匹配,一旦含有该关键字就将该文件名进行输出。关键字查找是所有数据库检索系统都必备的功能,本软件将在该功能基础上再研发出模糊查找的功能,是一种不需要关键字前后顺序必须一致,且所有字符都匹配,仍能具有一定匹配度的文件名进行输出的算法。实现逻辑为根据关键字的总字符数,从左至右依次取一个字符循环判断文件名中是否存在该字符,若存在就使成员变量iMatchNum自增,最后若自增值大于等于总字符数的50%以上就认为匹配,输出该文件名。通过使用该功能极大地提高了关键字查找成功率。核心代码与效果图,如图6所示。

```
int iMatchNum = 0;
for (int i=0; i<iKeyNameLen; i++)
{
    if (find.GetFileName().Find(strKeyName.Mid(i,1))>= 0)
    {
        iMatchNum++;
    }
}
if (2*iMatchNum < iKeyNameLen)
    continue;
```

图6 模糊查找功能核心代码

Fig.6 The core code of fuzzy search function

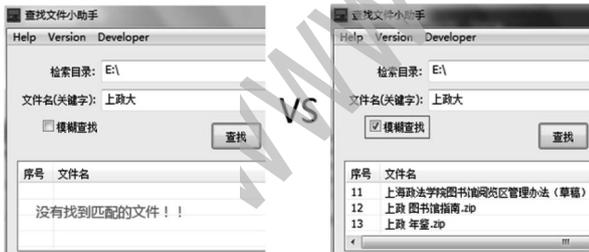


图7 模糊查找功能的效果

Fig.7 The results of fuzzy search function

5.5 列表控件信息显示及双击事件

单击界面查找按钮来执行文件查找时,通过调用函数:m_list.InsertItem(iCount,strCount)与m_list.SetItemText(iCount,1,find.GetFileName()),将每个符合条件的文件信息(序号、文件名、文件路径)逐行插入到List控件中,并以Report风格显示,实现类似于Excel表的功能,方便用户快速浏览文件信息,然后通过响应鼠标双击事件,获取目标文件完整路径然后调用ShellExecute(NULL,_T("open"),strSelectFile,NULL,NULL,SW_SHOWNORMAL)函数来打

开。

5.6 整个窗口及控件大小可变

为进一步提高用户对软件的使用体验,实现通过鼠标自由拉伸改变软件及控件的大小,以满足用户特殊情况下的使用需求。与窗口一同拉伸或移动的控件有检索目录输入框、完整路径显示框、列表控件及目录选择按钮。通过响应消息处理函数OnSize(),实时调整窗体及相关控件的大小与位置。定义CRect结构体变量rec,以及Cwnd类的指针变量*pWnd使指向目标控件,然后获取目标控件的大小pWnd->GetWindowRect(&rec),再转换为在对话框中的区域坐标ScreenToClient(rec),再根据变化后的窗体大小,实时修改rec结构体变量的值,最后执行pWnd->MoveWindow(rec)修改目标控件的大小及位置。



图8 使用鼠标调整窗口尺寸及控件位置

Fig.8 Use mouse to adjust window size and controls position

6 结论(Conclusion)

利用MFC基于对话框的文件查找软件开发,通过强大的类库功能、合理的代码逻辑设计及界面控件的布局,为广大用户提供了一种新的专业文件查找工具,有效地弥补了Windows系统自带文件搜索功能的不足,进一步提高了查找的效率及成功率。

为进一步满足不同用户使用需求,提高软件的实用性,需要不断完善该软件的功能,如增加文件内容的关键字查找,以及显示查找进度等功能。

参考文献(References)

- [1] 谭振江,朱冰,逯洋.基于MFC的随机点名系统设计与实现[J].吉林师范大学学报(自然科学版),2018,39(01):110-115.
- [2] 水疙瘩.善用Windows 7搜索让查找效率更高找[J].个人电脑,2012,18(03):94-96.
- [3] 谭浩强.C++程序设计(第2版)[M].北京:清华大学出版社,2011.
- [4] 董荣伟,姚月琴,陶冶,等.基于VC++智能粮库环境参数监控软件设计[J].河北农机,2018(01):19.
- [5] 郑阿奇,丁有和,郑进,等.Visual C++实用教程(第3版)[M].北京:电子工业出版社,2007.
- [6] 张驰.基于C++语言的跨平台软件开发的设计与实现[D].北京:北京交通大学,2010.
- [7] 陈秀川,杨云,郭琦.对VC~(++MFC类库中CFile类的派生类的研究和示例[J].南昌航空工业学院学报(自然科学版),2004(04):76-80;84.

作者简介:

李传可(1985-),男,硕士,助理馆员.研究领域:软件开发,人工智能。