

浅谈计算机软件数据接口的几种实现思路和应用

张晓鹏

(太原铁路局晋太集团顺新科研所,山西太原,030031)

摘要:阐述了计算机软件数据接口的设计原则,介绍了计算机软件数据接口的几种实现思路及其具体应用,以实现这些数据在不同软件之间的交互。

关键词:计算机软件;数据接口;文件交换模式;应用程序接口函数模式;中间数据库模式

中图分类号:TP311.5

文献标识码:A

随着计算机应用技术的发展和应用程序的日益普及,人们生产生活中要使用各种各样的计算机软件。对于这些软件来说,主要是操作使用自己定义的各种数据结构,而相对于普通用户来说,这些软件操作的数据都是他们用到的基本数据。可是因为生产计算机软件开发商的不同,这些软件使用的数据结构都各有不同,使得用户要操作两套甚至更多软件的时候,要按计算机软件提供不同格式的基础数据,要按计算机软件保留不同格式的数据结果。这样在实际使用中,不仅给用户带来各种重复劳动,而且整个计算机软件处理流程链也出现了中断,而要解决这一系列问题,最好的方法就是在各软件程序之间采用计算机软件数据接口。

1 计算机软件数据接口

计算机软件数据接口是计算机软件开发商提供给用户或者第三方软件开发商的一系列标准规范,用于特定数据的交流,既保持了开发商软件数据的保密性,又提供了一定的灵活性,使用户或者第三方开发商能更加有效地使用自己的软件。这些规范可以是固定格式的数据文件,可以是封装好的应用程序接口函数(通常所说的API函数),还可以是以数据库形式存在的中间数据库。

软件接口在系统A、B结构中所处的位置见图1。

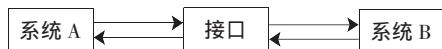


图1 接口在计算机系统结构所处位置

2 计算机软件数据接口设计原则

2.1 满足用户的需求

软件决定了软件数据接口能够提供的服务功能,软件数据接口又扩展了程序的使用范围。软件数据接口的设计,最终目的是为了实现在用户的需求,而怎么最大化地满足用户的需求,则要具体问题具体分析。

2.2 面向对象的原则

科学地设计软件数据接口能达到事半功倍的效果。采用面向对象的原则,尽可能全面准确地描述提供的功能。对于开发者的接口越复杂越详细越好。这样可以减少每个程序模块之间的耦合性,从而降低开发的难度,节约成本。

2.3 软件数据接口的高健壮性及高容错性

健壮性又称鲁棒性,是指软件对于规范要求以外的输入情况的处理能力。所谓健壮的系统是指对于规范要求以外的输入能够判断出这个输入不符合规范要求,并能有合理的处理方式。由于软件数据接口涉及多方的程序开发,所以一定要保证程序接口的高健壮性及高容错性,提供强大的错误处理机制。这样也使得用户在使用接口时能准确找到错误产生的原因,同时避免数据接口的错误影响到软件系统本身。

2.4 软件数据接口的可扩展性

软件接口是基于软件本身的。随着用户应用程度的提高,软件本身的升级会带来软件数据接口的升级。软件数据接口的升级必然影响到第三方开发商的软件接口使用。因此,尽可能使现有的软件数据接口具有一定的可扩展性,应减少对方厂商程序大规模变动,最终使用户受益。

2.5 尽量遵守本行业的标准数据接口规范

现在计算机应用已经十分普遍,一个行业有很多的计算机软件开发商,而开发商采用不同的编程语言,不同的数据库,而且又各自定义了自己的接口标准。这些没有共同标准的软件数据接口,不仅给软件升级以及后续软件开发商的开发带来很大的困难,而且也给用户软件维护带来很大的困难。因此,国家已经制定出一些软件数据接口行业标准规范。作为一个负责任的软件开发商,应尽量遵守这些规范。

3 计算机软件数据接口实现思路

3.1 文件交换模式

文件交换模式是指采取指定数据结构的数据文件来完成软件开发商、用户和第三方开发商的数据交互。用户或第三方开发商首先按开发商指定的数据结构,提交一个文件,当软件扫描到用户或第三方开发商提交的文件后,根据系统内部规则向用户返回一个文件。用户或第三方开发商按开发商制定的规则读取此文件即可,以此达到数据交互。

数据文件可以是任何形式的文件。常用的有ini文件和txt文件等。开发商可以采取自定义格式文件,也可以利用现有的通用格式文件。下面介绍常用的文件交换模式实际使用的一些例子。

3.1.1 ini文件

ini文件是Windows的系统配置文件,Windows的各项配置,但由于Windows提供操作ini文件的API函数,因此很多软件使

用 ini 文件来记录自己的配置信息。通过操作 ini 文件也可以达到交互数据的目的。

ini 文件样例：

```
[his]
```

```
code=000001
```

```
licence= 141201000000540a0a1287701c286937
```

下面是常用的读写 ini 文件的 API 函数：

读取 ini 文件函数

```
Declare Function GetPrivateProfileString Lib "kernel32" Alias
"GetPrivateProfileStringA" (ByVal lpApplicationName As String,
ByVal lpKeyName As Any, ByVal lpDefault As String, ByVal
lpReturnedString As String, ByVal nSize As Long, ByVal lpFileName
As String) As Long
```

写 ini 文件函数

```
Declare Function WritePrivateProfileString Lib "kernel32"
Alias "WritePrivateProfileStringA" (ByVal lpApplicationName As
String, ByVal lpKeyName As Any, ByVal lpString As Any, ByVal
lpFileName As String) As Long
```

3.1.2 txt 文件

txt 文件是 Windows 中的文本文件。程序员在开发软件的使用也经常使用 txt 来记录一些特殊的格式。比如字符串 ##AAA##BB##C##,以“##”分割字符串,读取##之间的数据即可。而操作 txt 的方式在各种开发语言中都有提供。比如 vb 中：

· 读取文件

```
Dim b As String
```

```
Open "f:\Read.txt" For Input As #1
```

```
b = StrConv(InputB(LOF(1), 1), vbUnicode)
```

```
Close #1
```

· 写入文件

```
Dim b As String
```

```
Open "f:\Write.txt" For Output As #2
```

```
Print #2, b
```

```
Close #2
```

文件交换模式对于程序员来说开发起来比较自由,不受编程语言的限制,可以灵活地完成对方的需求。但是采用这种方式,需要由第一开发商定期检测第三方开发商是否提交了接口文件。文件交换模式见图 2。



图 2 文件交换模式图

3.2 应用程序接口函数模式

应用程序接口函数模式是指开发商提前定义好一系列的函数,完成需要用户或者第三方开发商介入的数据操作,具体实现过程对用户或第三方开发商不透明,起到数据保密的作用。用户或者第三方开发商只要调用提供的函数即可完成开发商允许的数据操作。

比如定义以下函数：

```
int getMembers(char*nd, char* bm, char* info)
```

返回值：

0 成功

小于 0 失败

使用方法：使用时需要提前声明函数：

```
Public Declare Function getMembers Lib "NhHisImp.dll" ( _
```

```
ByVal nd As String, _
```

```
ByVal bm As String, _
```

```
ByVal info As String) As Long
```

然后调用函数 n=getMembers("2010","000001",info)

应用程序接口函数模式是目前比较常用的接口模式。很多软件厂商公布自己的接口函数,方便用户更深入地使用产品。应用程序接口函数模式见图 3。



图 3 应用程序接口函数模式

3.3 中间数据库模式

中间数据库模式是指软件开发商建立一个公用的数据库,授权用户或者第三方开发商访问,双方按一定的规则来操作数据库中的数据来完成数据的交互。这种模式需要用户或者第三方开发商访问指定的数据库格式,所采用的数据库可以为当前的各种主流数据库。比如 Access, SqlServer, Oracle, sybase 等。

这种方式对第三方开发商来说比较容易开发,操作数据的灵活性比较强。但是由于使用到额外的数据库,有的数据库本身的配置比较复杂,相对而言使用起来具有一定难度,因此这种接口模式使用范围较小。中间数据库模式见图 4。

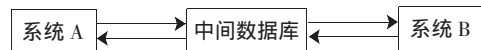


图 4 中间数据库模式

4 结语

计算机软件数据接口在整个软件操作流程中起到一个承上启下的作用,使一个个孤立的计算机软件形成了一条计算机软件流水线,用户工作的信息化进程形成了一个整体,在行业内部大规模使用计算机软件系统的可行性大幅提高。随着计算机软件应用的日益普及,使用到计算机软件数据接口的软件也越来越多。对于同一行业内部使用统一标准数据接口规范的呼声也越来越高。作为一名程序开发人员,也期待着每个行业都能有统一的标准软件接口规范,以便接口程序能更好地服务于每个用户。

参考文献

[1] 阎宏.Java 与模式[M].北京:电子工业出版社,2002:59-63.

(责任编辑 张红)

第一作者简介:张晓鹏,男,1980 年 5 月生,2002 年毕业于太原理工大学计算机科学与技术专业,助理工程师,太原铁路局晋太集团顺新科研所,山西省太原市尖草坪区新店街 3 号,030003.

基于机载 LiDAR 数据制作大比例尺线划图方法初探

谢丽娟,侯国瑞

(山西亚太数字遥感新技术有限公司,山西太原,030006)

摘要:介绍了 TerraSolid 系列软件和 CASS 软件,研究了在 CASS 软件中利用机载 LiDAR 数据制作线划图的基本流程和方法,证明了该方法可加快测图速度、缩短成图周期以及用机载 LiDAR 数据生产大比例尺线划图的可行性。

关键词: TerraSolid 系列软件;机载 LiDAR 数据;大比例尺线划图;CASS 软件

中图分类号: P283

文献标识码: A

线划图是国家各项工程建设中必需的基础资料,从线划图上确定地物的位置和相互关系及地貌的起伏形态等情况,比实地更准确、全面、方便、迅速。在这个高速运转的社会体系中,只有加快工程用图的更新和生产,才能满足各项工程建设的需要。机载 LiDAR(Light Detection And Ranging)数据的获取速度快、精度高,外业工作量少,利用机载 LiDAR 数据生产大比例尺线划图,将会大大提高测图速度,缩短成图周期。本文将介绍一种利用 CASS 软件对机载 LiDAR 数据进行矢量化测图、快速生成线划图的方法。

1 软件介绍

本文所述测图法使用的软件主要是 TerraSolid 系列软件和南方 CASS 软件。TerraSolid 系列软件是一套基于 MicroStation 软件平台开发的商业化半自动 LiDAR 数据处理软件,包括多个模块,最常用的主要是 TerraScan、TerraModel、TerraPhoto 等模块。TerraScan 模块是处理机载或地面激光雷达数据的通用软件包,主要功能包括以多种格式读入、导出激光点;自定义点类别并根据激光点的坐标、颜色以及不同回波值等信息对激光数据进行分类、交互式判别三维目标等。TerraModel 模块是根据激光点数据、测量数据或 xyz 坐标数据建立地面或地表模型并对模型进行处理的软件,其主要功能包括建立三维剖面图、创建等高线、创

建彩色渲染图、在选定区域内移动和升降激光点等。TerraPhoto 模块是对原始数码影像进行处理和正射纠正的软件,可以进行自动空三平差、镶嵌等。CASS 软件是南方测绘仪器公司在 AutoCAD 上开发的新一代数字化地形地籍成图软件。

2 线划图制作

2.1 数据准备

TerraSolid 系列软件对机载 LiDAR 数据后处理生成的直接数字产品主要是 DEM、DOM 以及分类后的激光点 Laser,目前利用机载 LiDAR 数据制作线划图还需要借助其他软件平台进行。利用 CASS 软件对机载 LiDAR 数据进行测图所需要的数据有: TIF 格式的正射影像及对应的 TFW 格式的影像坐标文件,模型关键点坐标文件,地面点及非地面点坐标文件。

正射影像及其对应的坐标文件的获取过程:一是对机载 LiDAR 数据进行数据预处理,包括对激光数据的定位定向、数据检校、坐标转换以及确定影像的外方位元素,赋予这些数据三维坐标和姿态信息。二是对激光数据进行点分类,将地面点与非地面点分开。三是由预处理得到的数据,结合原始数码影像及分类后的地面点制作高精度 TIF 格式的正射影像及相应的 TFW 格式坐标文件。点坐标数据文件获取方法为:点坐标数据文件必须是 CASS 软件可以识别的 *.DAT 格式文件,对分类后的相应类别激

Talking about Several Implementation Ideas and Applications of Computer Software Data Interface

ZHANG Xiao-peng

ABSTRACT: This paper describes the design principles of computer software data interface, and introduces some implementation ideas and some specific applications of computer software data interface for realizing the interaction of these data among different software.

KEY WORDS: computer software; data interface; file exchange model; application program interface function model; intermediate database model