



(12) 发明专利申请

(10) 申请公布号 CN 103699636 A

(43) 申请公布日 2014. 04. 02

(21) 申请号 201310718598. 3

(22) 申请日 2013. 12. 20

(71) 申请人 大连大学

地址 116622 辽宁省大连市经济技术开发区
学府大街 10 号大连大学

(72) 发明人 裴炳南 李顺宏

(74) 专利代理机构 大连智高专利事务所(特殊
普通合伙) 21235

代理人 胡景波

(51) Int. Cl.

G06F 17/30(2006. 01)

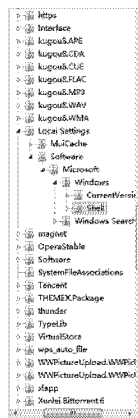
权利要求书1页 说明书5页 附图8页

(54) 发明名称

一种在 WIN7-32/64bit 操作系统下使文件夹 / 文件位置随意移动的方法

(57) 摘要

本发明公开了一种在 WIN7-32/64bit 操作系统下使文件夹 / 文件位置随意移动的方法, 包括: 打开注册表编辑器, 分别打开注册表编辑器的主键与子键, 直接删除上述目录下的子键 Shell, 导入已经编辑好的注册表文件。实现了在 WIN7-32/64bit 下随意移动处于非桌面目录下的文件 / 文件夹的位置。



1. 一种在 WIN7-32/64bit 操作系统下使文件夹 / 文件位置随意移动的方法,其特征在于,包括:打开注册表编辑器,分别打开注册表编辑器的主键与子键,直接删除上述目录下的子键 Shell,导入已经编辑好的注册表文件。

2. 如权利要求 1 所述的一种在 WIN7-32/64bit 操作系统下使文件夹 / 文件位置随意移动的方法,其特征在于,所述打开注册表编辑器,分别打开注册表编辑器的主键与子键,直接删除上述目录下的子键 Shell,导入已经编辑好的注册表文件,包括:

打开注册表编辑器,定位到如下目录:HKEY_CURRENT_USER\Software\Classes\LocalSettings\Software\Microsoft\Windows\Shell\Bags\AllFolders\Shell\,打开上述目录下的子键 Shell,在 HKEY_CURRENT_USER\Software\Classes\LocalSettings\Software\Microsoft\Windows\Shell\Bags\AllFolders\Shell 目录下新建一个子键,并进行命名;

打开新建的子键,在新建两个二进制值 FFlags 和 Mode,其值分别指定为 0x43000001 和 0x4;

在 HKEY_CURRENT_USER\Software\Classes\LocalSettings\Software\Microsoft\Windows\Shell\Bags\AllFolders\Shell 下再分别新建四个子键;

然后再分别在这四个新建的子键下新建两个二进制 FFlags、Mode,其值分别指定为 0x43000001 和 0x4;

定位到目录 HKEY_CURRENT_USER\Software\Classes\LocalSettings\Software\Microsoft\Windows\Shell\Bags,删除以数字命名的项。

3. 如权利要求 1 所述的一种在 WIN7-32/64bit 操作系统下使文件夹 / 文件位置随意移动的方法,其特征在于,还包括:将 AllFolders 项导出保存,删掉整个 HKEY_CURRENT_USER\Software\Classes\LocalSettings\Software\Microsoft\Windows\Shell 目录下的 Bags 项;

然后再在 HKEY_CURRENT_USER\Software\Classes\LocalSettings\Software\Microsoft\Windows\Shell 目录下新建一个叫 Bags 的项,把刚刚导出的 AllFolders 项直接导入到 Bags 项中;

把 HKEY_CURRENT_USER\Software\Classes\LocalSettings\Software\Microsoft\Windows 目录下的 Shell 子键导出保存;

保存修改,关闭注册表编辑器。

一种在 WIN7-32/64bit 操作系统下使文件夹 / 文件位置随意移动的方法

技术领域

[0001] 本发明涉及电学领域,尤其涉及一种在 WIN7-32/64bit 操作系统下使文件夹 / 文件位置随意移动的方法。

背景技术

[0002] 目前,Windows 操作系统注册表概述 :注册表是 Microsoft Windows 中的一个重要的数据库,用于存储系统和应用程序的设置信息。早在 Windows3.0 推出 OLE 技术的时候,注册表就已经出现。随后推出的 Windows NT 是第一个从系统级别广泛使用注册表的操作系统。但是,从 Microsoft Windows95 开始,注册表才真正成为 Windows 用户经常接触的内容,并在其后的操作系统中继续沿用至今。

[0003] 注册表的由来 :在 Windows3.x 操作系统中,注册表是一个极小文件,其文件名为 Reg. bat,里面只存放了某些文件类型的应用程序关联,大部分的设置放在 Win. ini、System. ini 等多个初始化 INI 文件中。由于这些初始化文件不便于管理和维护,时常出现一些因 INI 文件遭到破坏而导致系统无法启动的问题。为了使系统运行得更为稳定、健壮,Windows95/98 设计师们借用了 Windows NT 中的注册表的思想,将注册表引入到 Windows95/98 操作系统中,而且将 INI 文件中的大部分设置也移植到注册表中,因此,注册表在 Windows95/98 操作系统的启动、运行过程中起着重要的作用。

[0004] 注册表的作用 :注册表是 windows 操作系统中的一个核心数据库,其中存放着各种参数,直接控制着 windows 的启动、硬件驱动程序的装载以及一些 windows 应用程序的运行,从而在整个系统中起着核心作用。这些作用包括了软、硬件的相关配置和状态信息,比如注册表中保存有应用程序和资源管理器外壳的初始条件、首选项和卸载数据等,联网计算机的整个系统的设置和各种许可,文件扩展名与应用程序的关联,硬件部件的描述、状态和属性,性能记录和其他底层的系统状态信息,以及其他数据等。

[0005] 注册表是为 Windows NT 和 Windows95 中所有 32 位硬件 / 驱动和 32 位应用程序设计的数据文件。16 位驱动在 WinNT 下无法工作,所以所有设备都通过注册表来控制,一般这些是通过 BIOS 来控制的。在 Win95 下,16 位驱动会继续以实模式方式设备工作,它们使用 system. ini 来控制。16 位应用程序会工作在 NT 或者 Win95 下,它们的程序仍然会参考 win. ini 和 system. ini 文件获得信息和控制。

[0006] 在没有注册表的情况下,操作系统不会获得必须的信息来运行和控制附属的设备和应用程序及正确响应用户的输入。

[0007] 注册表的存放位置 :2000/XP 注册表文件按功能来分,也是由系统注册表文件和用户注册表文件两类组成的注册表文件存放系统的所有设置信息 :系统设置和缺省用户配置数据存放在系统 \ 系统文件夹 \SYSTEM32\CONFIG 文件夹下的 6 个文件,DEFAULT、SAM、SECURITY、SOFTWARE、USERDIFF 和 SYSTEM 中,而用户的配置信息存放在系统所在磁盘的 \Documents and Setting\ 文件夹,包括 ntuser. dat ntuser. ini ntuser. dat. log

[0008] 注册表的数据结构:注册表由键(或称“项”)、子键(子项)和值项构成。一个键就是分支中的一个文件夹,而子键就是这个文件夹中的子文件夹,子键同样是一个键。一个值项则是一个键的当前定义,由名称、数据类型以及分配的值组成。一个键可以有一个或多个值,每个值的名称各不相同,如果一个值的名称为空,则该值为该键的默认值。如图 1,注册表共有 5 个主键:

[0009] HKEY_CLASSES_ROOT

[0010] HKEY_CURRENT_USER

[0011] HKEY_LOCAL_MACHINE

[0012] HKEY_USERS

[0013] HKEY_CURRENT_CONFIG

[0014] 文件/文件夹的属性:在 Windows XP 或者 Windows7 系统下随便打开一个文件夹,查看属性下的“自定义”选项,下拉菜单中有“常规项”、“文档”、“图片”、“音乐”、“视频”这几项,说明 Windows 系统下,所有文件或者文件夹的类型都是这五种中的一种,这五种类型在注册表中的名称分别为:

[0015] 常规项 {5C4F28B5-F869-4E84-8E60-F11DB97C5CC7}

[0016] 文档 {7D49D726-3C21-4F05-99AA-FDC2C9474656}

[0017] 图片 {B3690E58-E961-423B-B687-386EBFD83239}

[0018] 音乐 {94D6DDCC-4A68-4175-A374-BD584A510B78}

[0019] 视频 {5FA96407-7E77-483C-AC93-691D05850DE8}

[0020] Windows XP 和 Windows7 在移动文件/文件夹位置方面的不同:在 Windows XP 操作系统下,可以任意移动文件/文件夹的位置,无论这些文件/文件夹是在什么目录下,不管是在桌面还是在一般的盘符下,文件/文件夹的位置都是可以由用户任意移动的,如图 2-1 和图 2-2;但在 WIN7-32/64bit 下,文件/文件夹的位置无法随意移动,文件夹是否可以随意移动是和注册表的控制信息有关的。

发明内容

[0021] 本发明所要解决的问题就是如何在 WIN7-32/64bit 下随意移动处于非桌面目录下的文件/文件夹的位置,因此本发明提出了一种通过修改注册表的方式,在 WIN7-32/64bit 操作系统下使文件夹/文件位置随意移动的方法。

[0022] 一种在 WIN7-32/64bit 操作系统下使文件夹/文件位置随意移动的方法,包括:打开注册表编辑器,分别打开注册表编辑器的主键与子键,直接删除上述目录下的子键 Shell,导入已经编辑好的注册表文件。

[0023] 具体的:打开注册表编辑器,定位到如下目录:HKEY_CURRENT_USER\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\AllFolders\Shell\,打开上述目录下的子键 Shell,在 HKEY_CURRENT_USER\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\AllFolders\Shell 目录下新建一个子键,并进行命名;

[0024] 打开新建的子键,在新建两个二进制值 FFlags 和 Mode,其值分别指定为 0x43000001 和 0x4;

- [0025] 在 HKEY_CURRENT_USER\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\AllFolders\Shell 下再分别新建四个子键；
- [0026] 然后再分别在这四个新建的子键下新建两个二进制 FFlags、Mode, 其值分别指定为 0x43000001 和 0x4；
- [0027] 定位到目录 HKEY_CURRENT_USER\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags, 删除以数字命名的项。
- [0028] 上述方法, 还包括: 将 AllFolders 项导出保存, 删掉整个 HKEY_CURRENT_USER\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell 目录下的 Bags 项；
- [0029] 然后再在 HKEY_CURRENT_USER\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell 目录下新建一个叫 Bags 的项, 把刚刚导出的 AllFolders 项直接导入到 Bags 项中；
- [0030] 把 HKEY_CURRENT_USER\Software\Classes\Local Settings\Software\Microsoft\Windows 目录下的 Shell 子键导出保存；
- [0031] 保存修改, 关闭注册表编辑器。
- [0032] 本发明的效果: 实现了在 WIN7-32/64bit 下随意移动处于非桌面目录下的文件 / 文件夹的位置。

附图说明

- [0033] 为了更清楚的说明本发明的实施例或现有技术的技术方案, 下面将对实施例或现有技术描述中所需要使用的附图做一简单地介绍, 显而易见地, 下面描述中的附图仅仅是本发明的一些实施例, 对于本领域普通技术人员来讲, 在不付出创造性劳动的前提下, 还可以根据这些附图获得其他的附图。
- [0034] 图 1 为本发明实施例提供的注册表的示意图；
- [0035] 图 2-1 为本发明实施例提供的文件 / 文件夹的位置图；
- [0036] 图 2-2 为本发明实施例提供的文件 / 文件夹的位置图；
- [0037] 图 3-1 为本发明实施例提供的注册表编辑器的位置示意图；
- [0038] 图 3-2 为本发明实施例提供的分别打开注册表的主键和子键的示意图；
- [0039] 图 4 为本发明实施例提供的 Shell 目录下有很多子键示意图；
- [0040] 图 5 为本发明实施例提供的新建的子键的示意图；
- [0041] 图 6 为本发明实施例提供的各个子键的示意图；
- [0042] 图 7 为本发明实施例提供的批处理程序运行的框图。

具体实施方式

- [0043] 为使本发明的实施例的目的、技术方案和优点更加清楚, 下面结合本发明实施例中的附图, 对本发明实施例中的技术方案进行清楚完整的描述：
- [0044] 本发明实施例提供一种在 WIN7-32/64bit 操作系统下使文件夹 / 文件位置随意移动的方法, 包括：
- [0045] 1, 打开注册表编辑器, 例如: 打开方式可以为直接在开始的位置输入 regedit

即可,具体的,如图 3-1。

[0046] 2,分别打开如下主键与子键。如图 3-2,如:

[0047] HKEY_CURRENT_USER\Software\Classes\Local Settings\Software\Microsoft\Windows。

[0048] 3,直接删除上述目录下的子键 Shell。

[0049] 4,导入已经编辑好的注册表文件(.reg)。

[0050] 首先,修改注册表文件的目录下子键 Shell 的一些信息,然后再把 Shell 导出来保存。

[0051] 具体的修改过程如下:打开注册表编辑器,定位到如下目录:HKEY_CURRENT_USER\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\AllFolders\Shell\。

[0052] 打开上述目录下的子键 Shell,可以看到 Shell 目录下有很多子键,如图 4 所示。

[0053] 在 HKEY_CURRENT_USER\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\AllFolders\Shell 目录下新建一个子键,命名为 {5C4F28B5-F869-4E84-8E60-F11DB97C5CC7}。

[0054] 打开新建的子键,会看到一个默认的字符串值(REG_SZ),它的值是未指定的,对于这个字符串值,不需要进行任何的修改。然后在新建两个二进制值(REG_DWORD)FFlags、Mode,将他们的值分别指定为 0x43000001 和 0x4,如图 5。

[0055] 在 HKEY_CURRENT_USER\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\AllFolders\Shell 下再分别新建四个子键

[0056] {7D49D726-3C21-4F05-99AA-FDC2C9474656}、

[0057] {B3690E58-E961-423B-B687-386EBFD83239}、

[0058] {94D6DDCC-4A68-4175-A374-BD584A510B78}、

[0059] {5FA96407-7E77-483C-AC93-691D05850DE8}

[0060] 然后再分别在这四个新建的子键下新建两个二进制(REG_DWORD)FFlags、Mode,值分别指定为 0x43000001 和 0x4。

[0061] 当以上的工作都完成时,进行下面的操作:

[0062] 定位到目录 HKEY_CURRENT_USER\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags。可以看到,里面有很多以数字命名的项和一个叫 AllFolders 的项,现在要做的就是删除这些以数字命名的项,因为正是这些项的存在,将随意移动文件/文件夹变得不可能,如图 6。这些以数字命名的项实在是太多了,一一删除变得很麻烦。

[0063] 下面介绍一个好的方法:先将 AllFolders 项导出保存,然后直接删掉整个 HKEY_CURRENT_USER\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell 目录下的 Bags 项,然后再在 HKEY_CURRENT_USER\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell 目录下新建一个叫 Bags 的项,把刚刚导出的 AllFolders 项直接导入到 Bags 项即可。

[0064] 把 HKEY_CURRENT_USER\Software\Classes\Local Settings\Software\Microsoft\Windows 目录下的 Shell 子键导出保存。

[0065] 保存修改,关闭注册表编辑器。

[0066] 当用户有这种习惯或者需求时,需要导入的注册表文件 (.reg)。如果当用户不喜欢这种修改,想要回到 WIN7-32/64bit 的默认状态时,也是可能的,因为在导入该注册表文件之前,用户备份了未修改之前的 HKEY_CURRENT_USER\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell 子键,为避免麻烦,可以写一个小的批处理程序 (.bat),使用户免去寻找 HKEY_CURRENT_USER\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell 子键,并手动导出保存的麻烦。使用该批处理程序,将使得导出的注册表文件保存在 C 盘根目录下,名为“不随意移动.reg”,此批处理程序运行的框图如图 7。

[0067] 此批处理程序的源代码如下:

```
[0068] @echo off
```

```
[0069] reg export "HKEY_CURRENT_USER\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell" c:\不随意移动.reg
```

```
[0070] @pause
```

[0071] 虽然仅仅是三行代码,但是使用户省去了很多麻烦,节省了大量时间。

[0072] 通过上述注册表修改后,如上所述,本发明存在两个文件,一个是用于导出并保存注册表的批处理程序 (.bat),另一个是已经修改好所导出的能够使文件 / 文件夹被随意移动的注册表文件 (.reg),用户只需要两步即可完成操作:运行批处理程序,和双击编辑好的注册表文件,导入即可。在进行备份注册表以及导入注册表操作时,任何分区以及任何盘符下的任何文件 / 文件夹均不能打开,否则会造成某一分区下文件 / 文件夹位置移动属性修改失败。批处理程序必须放在桌面或者 U 盘里运行。必须要等到批处理程序运行结束后,关闭程序对话框,才能进行注册表的导入工作。

[0073] 需要说明的是:在实际操作证明,在文件 / 文件夹的所有 8 种状态模式中,除了在“列表”和“状态”模式下仍然不能自由移动文件 / 文件夹外,在其余的 6 种模式“超大图标”、“大图标”、“中等图标”、“小图标”、“详细信息”、“扩展的平铺”下,修改是成功的,这已经能够满足绝大多数人的需求了。如果用户想恢复到系统默认情况,只需要将名叫“不随意移动”的注册表文件 (.reg) 重新导入到注册表即可(导入方法是直接双击该注册表文件)。

[0074] 本发明的实施例中,应用平台优势明显,不仅能够在 WIN7-32bit 上成功操作,还能在 WIN7-64bit 上成功修改,无论是哪一台计算机,只要是 WIN7 系统,就能成功修改。文件小巧。在不压缩的情况下,注册表文件和批处理程序总大小还不超过 3M,在使用 RAR 程序压缩的情况下,总大小不到 150KB,使得拷贝和存放变得极其简单和轻松。能够使得众多习惯了 Windows XP 的用户将操作系统更换成 Windows7 后,还能继续使用 XP 系统中的文件 / 文件夹位置随意移动特性,多了一份亲切感和归属感。能够提高文件 / 文件夹摆放位置的美观度,提升工作效率。转换轻松。用户能够很容易的从系统默认状态转换到本创作的目标修改状态,也可以轻松的恢复到系统默认状态,即轻轻双击鼠标即可实现。

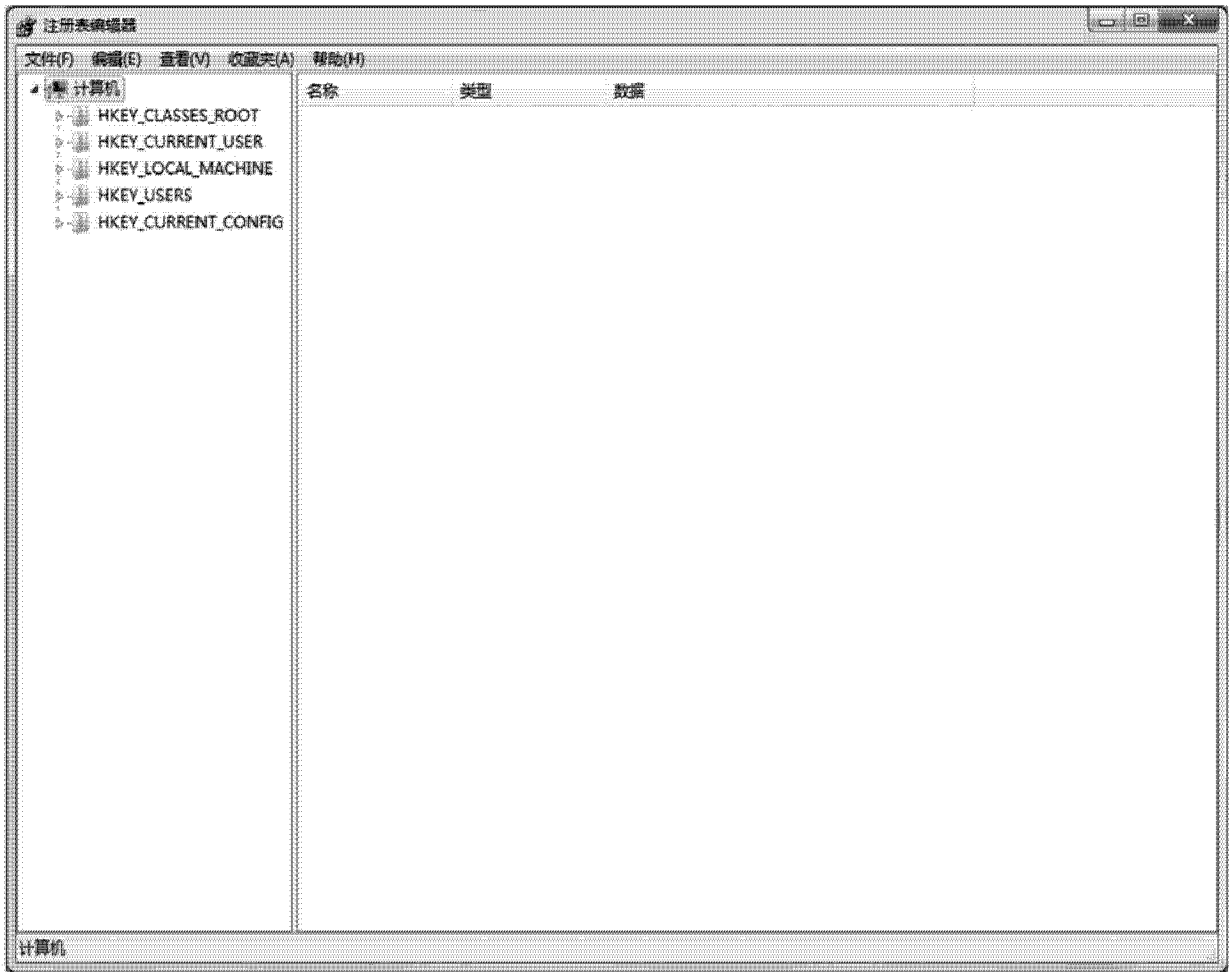


图 1



图 2-1



图 2-2

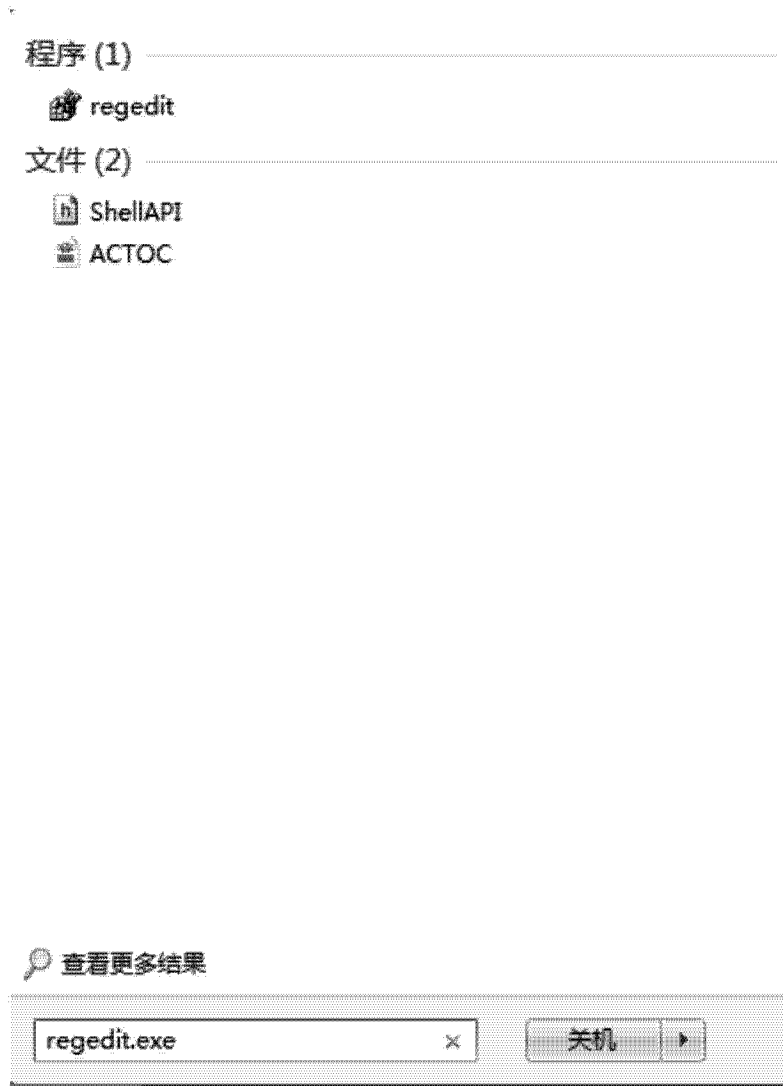


图 3-1

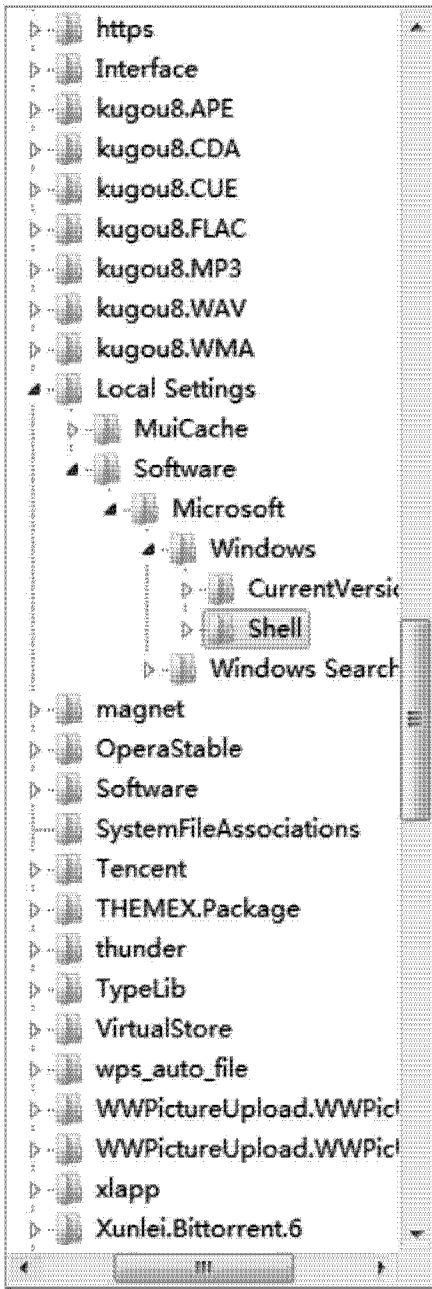


图 3-2

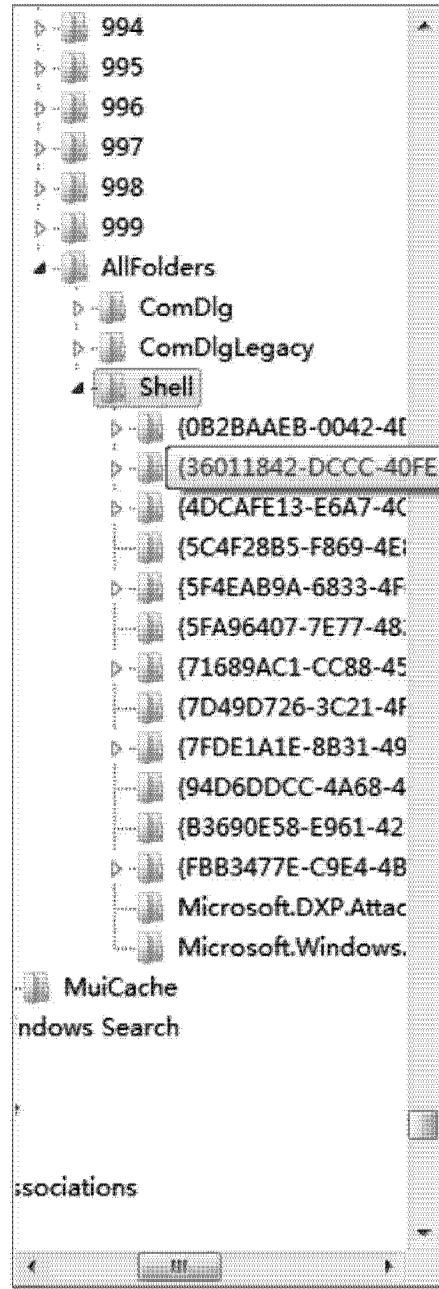


图 4

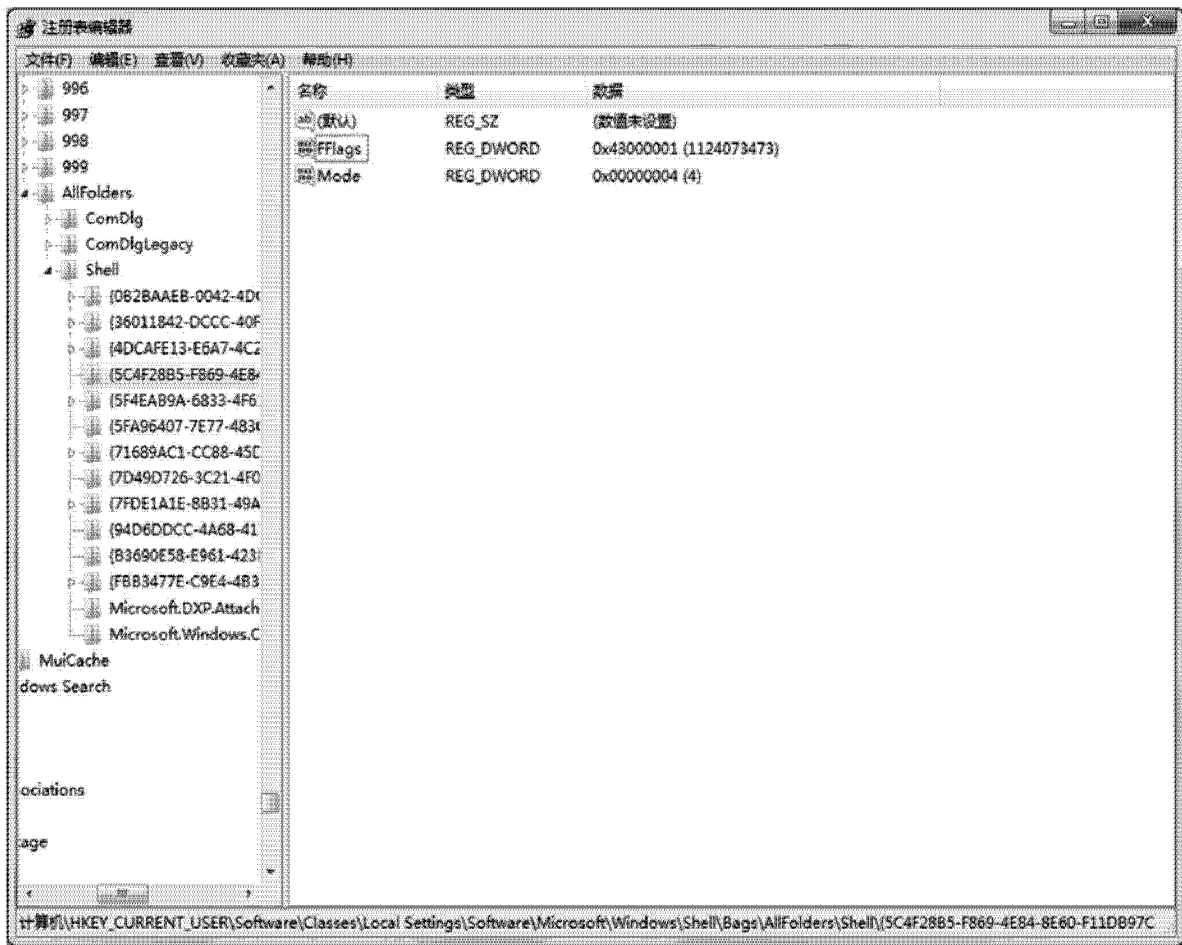


图 5

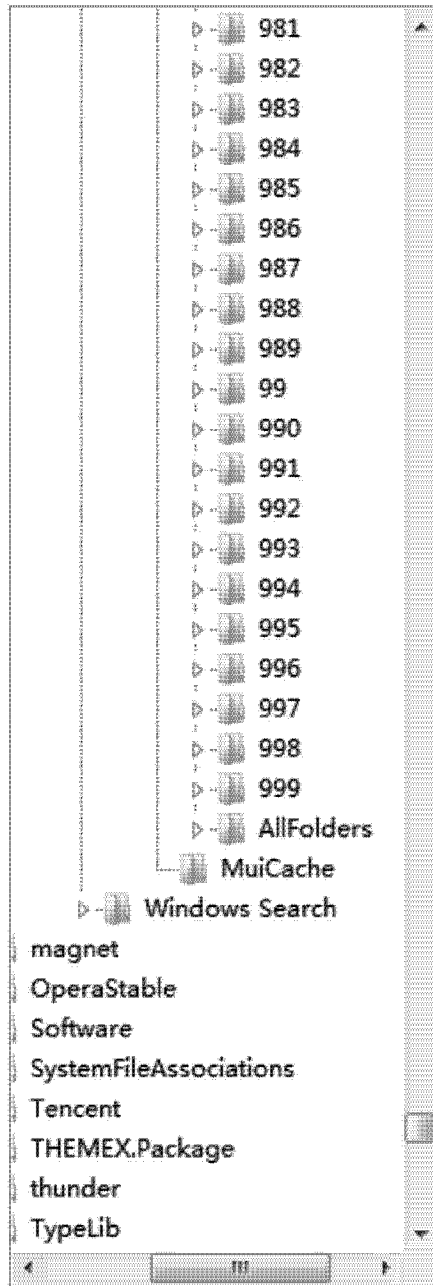


图 6

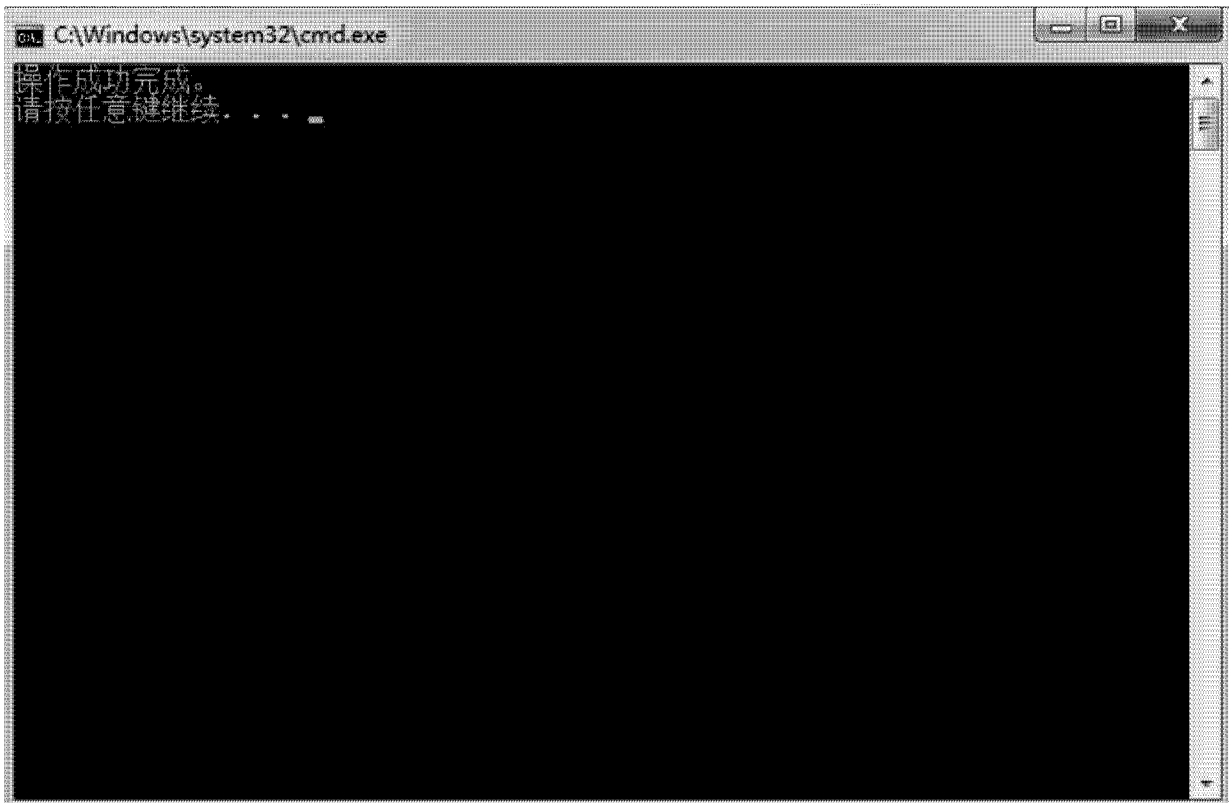


图 7