

Iterate Files by Tcltk

eryar@163.com

Abstract. Tcl/Tk provide a programming system for developing and using graphical user interface(GUI) applications. Tcl stands for “tool command language” and is pronounced “tickle”, is a simple scripting language for controlling the extending applications. The blog use Tcl/Tk to iterate all the files for a given directory, this is useful to some automation work, such as change all the file names for a given directory; add copyright info for the source code files.

Key Words. Tcl/Tk, Iterate Files, 遍历文件夹中所有文件

1. Introduction

Tcl/Tk 是一种用于易于使用的脚本语言，可以用来对程序进行扩展及完成一些自动化的工作，加上内置的一些命令，其功能要比 Windows 中的 DOS 的批处理命令功能更强大，使用更方便。Tcl 脚本语言是开源免费的，可以方便获取且免费使用。

OpenCASCADE 中使用了 Tcl/Tk 来实现了一个自动化测试体系。使用在 OpenCASCADE 中使用自定义的 Tcl 命令，可以快速来检验算法的结果。通过编写脚本文件，实现了测试的自动化。所以学习一下 Tcl/Tk 脚本语言，并在实际的工作中加以应用，可以将一些机械的劳动交给计算机自动完成。

本文主要说明如何使用 Tcl/Tk 来遍历指定文件夹中所有文件。利用此功能，可以稍微加以扩展，就可以完成一些实际的重复劳动。如遍历指定目录中所有的源文件或指定类型的文件，添加上版权信息等。

2. Tcl/Tk Code

要遍历指定目录下所有的文件，包括子文件夹，需要用到命令 `glob` 及一个递归函数。脚本代码如下所示：

```
#
# Tcl/Tk script to iterate all the files for a given directory.
# eryar@163.com
# 2015-01-18
#

package require Tcl
package require Tk

wm title . "Iterate Files"

label .labelDirectory -text "Directory "
entry .entryDirectory -width 30 -relief sunken -textvariable aDirectory
button .buttonDirectory -text "..." -command {chooseDirectory .entryDirectory}

button .buttonApply -text "Apply" -command {perform $aDirectory}
button .buttonCancel -text "Cancel" -command {exit}

grid .labelDirectory .entryDirectory .buttonDirectory
```

```

grid .buttonApply .buttonCancel

# chooseDirectory--
# choose the directory to iterate.
#
proc chooseDirectory {theEntry} {
    set dir [tk_chooseDirectory -initialdir [pwd] -mustexist 1]

    if {[string compare $dir ""]} {
        $theEntry delete 0 end
        $theEntry insert 0 $dir
        $theEntry xview end
    }
}

# perform--
# perform the algorithm.
#
proc perform {theDirectory} {
    puts "Iterate all the files in $theDirectory"

    if {[string length $theDirectory] < 1} {
        tk_messageBox -type ok -icon warning -message "Please select the directory!"
        -parent .
        return
    }

    # process the iterate...
    process $theDirectory
}

# process--
# recursion every folder and file.
#
proc process {theFolder} {

    set aFiles [glob -nocomplain -directory $theFolder *]

    foreach aFile $aFiles {
        if {[file isfile $aFile]} {
            # just output the file name here.
            # you can do something such as rename for the file.
            puts "$aFile \n"
        } else {
            process $aFile
        }
    }
}

```

程序用法为打开 Tcl 解释器，使用命令 source 加载脚本文件，如下图所示：

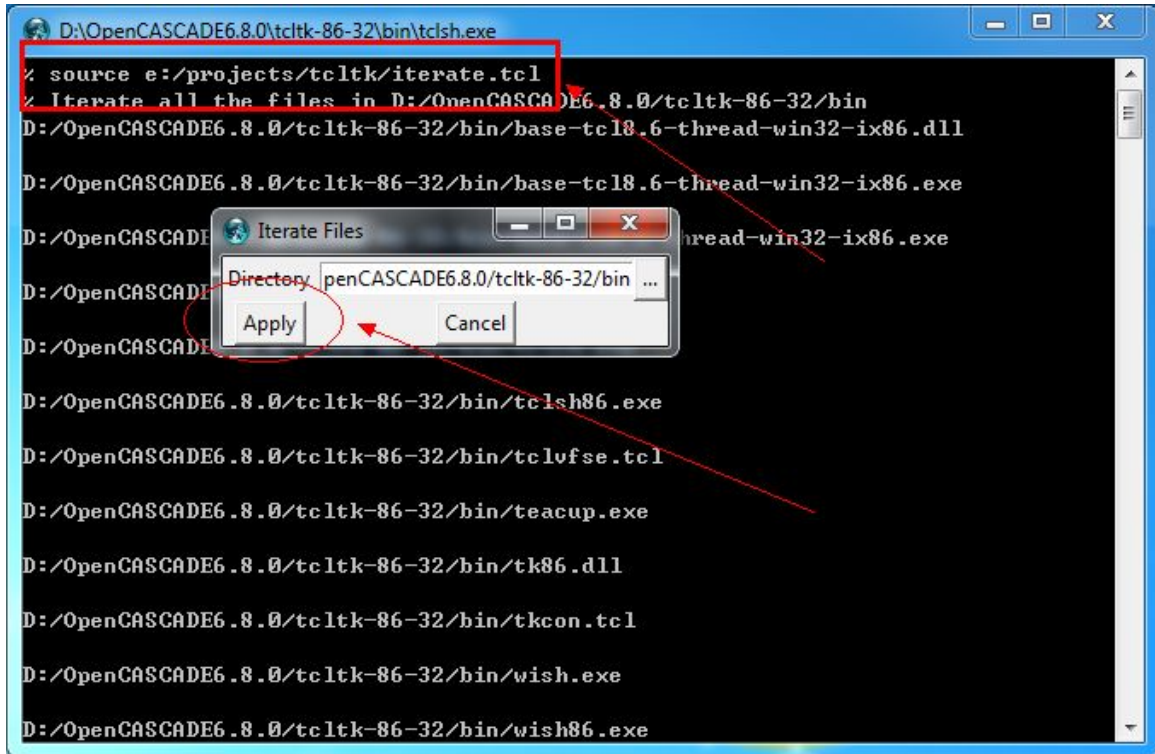


Figure 2.1 Tcl usage

3. Conclusion

通过应用 Tcl/Tk 来体验脚本编程的乐趣，并加深对 Tcl/Tk 的理解。从而对 OpenCASCADE 的模块 Draw Test Harness 更好地理解。

如果有编程基础，Tcl/Tk 会很快入门的。入门后，可以应用其直接编写一些有意思有脚本，来实现一些重复工作的自动化。也可将 Tcl 加入到自己的程序中，增加程序的二次开发功能。

可见，玩一玩脚本语言，还是非常有趣的！