

基于 TCL 语言的 HyperMesh 平台 二次开发范例

屈晓峰

北京北汽福田汽车有限公司

摘要: 本文通过对 TCL 脚本的简要分析,介绍了 TCL 语言的基本特点,结构形式。并结合工程实际,给出一个基于 TCL 语言的 HyperMesh 平台二次开发范例。通过对其进行简要的解释和分析,展示了在 HyperWorks 平台上应用 TCL 语言进行二次开发的灵活性与功能的强大性。并使初学者对其有一个初步的了解,以便进一步学习和应用。

关键词: TCL 语言, HyperWorks, 二次开发, CAE 仿真分析

1 TCL 语言简介

TCL(Tool Command Language)是一种解释执行的脚本语言(Scripting Language)。它拥有一个固有的核心命令集,同时还具有和 C/C++语言类似的控制结构:if 控制、循环控制和 switch 控制等,并支持过程的定义和调用,对数组和字符串等简单数据结构也提供了支持。

另外,因为 TCL 是一种比 C\C++语言有着更高抽象层次的语言,使用 TCL 可以在一种更高的层次上编写程序,它屏蔽掉了编写 C\C++程序时必须涉及到的一些较为烦琐的细节,可以大大地提高开发测试的速度。而且,我们使用 TCL 语言写的测试脚本,即使作了修改,也用不着重新编译就可以调用 TCL 解释器执行。可以省却不少时间。

TCL 是一种简明,高效,可移植性好的编程语言,它几乎在所有的平台上都可以解释运行,功能强大。特别是在信息产业、自动控制、机械电子等领域应用广泛。TCL 目前已成为自动测试中事实上的标准。

2 TCL 脚本示例

对于我们机械行业 CAE 分析工程师来说, TCL 也是一种重要的工具语言。很多 CAE 分析软件的深入应用都离不开 TCL 语言。今天笔者通过实际工程范例,对基于 TCL 语言的 HyperWorks 软件平台二次开发流程进行一个简要的阐述。先以一个 TCL 过程为例了解一下 TCL 语言:

```
#####  
#令三个输入的实数按从小到大排列。
```

```

#定义实现排列功能的过程，过程名为 arrange，x、y、z 为其参数。
proc arrange{x y z}{
#设置结果输出文件
set output [open "arrange_output.log" w];
set t 1;
if(x>y){
    set t $x;
    set x $y;
    set y $t;
}
if(x>z){
    set t $x;
    set x $z;
    set z $t;
}
if(y>z){
    set t $y;
    set y $z;
    set z $t;
}
#输出结果
puts $output "$x  $y  $z";
close $output;
}
# 完成
#####

```

其中，#是注释符，TCL 编译器不会执行以#开头的语句。set 命令是定义变量并赋值。这相当于 C 语言中的 int（或 float 等其它变量定义命令）。Proc 命令是程序员自己定义的过程，相当于 C 语言里的自定义函数，x、y、z 是过程的参数。在这里，名为 arrange 的过程完成了 x、y、z 三个值从小到大的排列。在使用此过程时，直接输入过程名 arrange 及三个参数即可完成排列功能，并将排列结果输入到文本文件中进行保存。符号\$和"分别为变量置换符和双引号置换符。变量置换符\$作用是得到变量的值，而非变量自身的符号。双引号置换符"作用则是解释器把分隔符和各种置换符（除换行符、\$置换符和[]置换符）等特殊字符当作普通字符，而不作特殊处理。Open、puts、close 等命令是 TCL 语言文件访问命令，在这里完成了排列后的三个数值的输出。如果读者想深入学习请参考 TCL 语言学习资料。

3 TCL 脚本在 HyperMesh 中的应用

由以上的小例子可知，TCL 语言简单、灵活。我们不光可以直接应用 TCL 自身的固有命令，还可以定义自己的过程，提高语言应用性。最重要的是，TCL 语言与 HyperWorks

平台是无缝连接的，HyperMesh 和 HyperView 中均提供了 TCL 语言的命令接口。通过对 HyperWorks 的二次开发，我们可以实现各种丰富的功能，并使工作高效、轻松。下面笔者将在工程项目中应用到的一个 TCL 脚本文件进行分析，与大家一同学习。这个 TCL 脚本作用是将所选部件抽取中面，并将这个中面所在的组件名字改为“原部件组件名字_midsurface”形式。有下划线的文字是对脚本的解释。

```
#####
# BatchMidSurface Version 1.0
# This script was wrote for getting midsurface of solid automatically.
#####
# The codes developed and maintained by fotonQ.
#suggestion or bugs to jat_radon@126.com
#2010.02.02
#####
以上为本TCL脚本编写目的、版本、建议等信息记载。
*createmarkpanel comps 1 "Select comps to get midsurface";
set CompIds [hm_getmark comps 1];
set NumComps [llength $CompIds];
*clearmark comps 1;
选取要抽取中面的组件（components），将这些组件的Id列表及其长度分别赋予变量CompIds和NumComps。
if { $NumComps < 1 } {
  return;
}
如果没有选取到组件，则返回。
#
#
set NumFail 0;
set fid [open "mid_surface_error.log" w];
puts $fid "The file is generated by MidSurface.";
puts $fid "comps";
打开出错信息输出文件通道，并对其进行写入操作。
#
for {set i 0} {$i < $NumComps} {incr i} {
  set Compid [lindex $CompIds $i];
  指针指向当前组件的Id，并赋给变量Compid。
  set Compname [hm_getcollectorname comps $Compid];
  获得当前组件的名字。
  hm_createmark solids 1 "by component" $Compid;
  建立当前组件中solid的组。
  set SolidIds [hm_getmark solids 1];
  set NumSolids [llength $SolidIds];
  if { $NumSolids < 1 } {
    incr NumFail;
  }
}
```

```

    puts -nonewline $fid $NumFail;
    puts -nonewline $fid ". ";
    puts -nonewline $fid "[";
    puts -nonewline $fid $Compname;
    puts -nonewline $fid "\] ";
    puts $fid ", Failure=None Solid in comps.";
    continue;
}

```

通过solids的Id的长度来判断是否出错，并把出错信息写入Log文件中。

```

#
*setsurfacenormalsdisplaytype 1 ;
*normalsoff;
*midsurface_extract_10 solids 1 3 0 1 1 0 0 2 0 0 10 0 10 -2 undefined 0 0 1;
*midsurface_remove_edit_bodies;
*release_temp_fixed_vertices;
*normalsoff;
*retainmarkselections 1;

```

抽取中面。

```

*renamecollector components "Middle Surface" "$Compname _midsurface " ;

```

更改中面组件名字。

```

*retainmarkselections 0;
}
close $fid;
hm_usermessage "Done."
#
# The end.

```

完成。

#####

带*的语句是 HyperMesh 格式命令。HyperMesh 在读入脚本后会把这些命令转化为 HyperMesh 的动作过程。HyperMesh 的这些动作命令有成百上千条，都记住是不可能的，同时也是不必要的。实际上我们在用 HyperMesh 时，每一步操作都会在 command.cmf 文件中留下相关的记录语句。诸如上面带*的命令。我们可以先在 HyperMesh 中抽取一个 solid 的中面，然后在 command.cmf 中将这几行语句提出来，适当修改(根据在程序中起的作用)。

4 总结

TCL 二次开发能够很好地完成 HyperWorks 中的许多批处理工作。在我公司近期的分析项目中，在 HyperWorks 中由 TCL 脚本实现的批处理控制发挥了巨大的作用，大大提高了工作效率，减轻了工程师的工作强度，保证了项目节点。对于 CAE 工程师来说，用好这一工具，实现 CAE 分析过程自动化，有着重要的意义和作用。

5 参考文献

- [1] 陈旭盛 TCL 培训教程 深圳市华为技术有限公司研究管理部文档中心 2000
- [2] HyperWorks/help/Reference Guide/Scripts