Altair HyperWorks

Altair HyperMesh 2019 Tutorials

HM-8050: Create Forces on Nodes and Add a Button on the User Page

HM-8050: Create Forces on Nodes and Add a Button on the User Page

In this tutorial you will:

- Determine the commands to create forces on nodes
- Create a **Utility** menu macro to execute the commands
- Create a new button on the **User** page of the **Utility Menu** to run the macro

Tools

In order to execute command file commands or Tcl scripts from a button on any of the HyperMesh **Utility Menu** pages, a **Utility Menu** macro must first be defined. A **Utility Menu** macro contains valid command file or templex commands that execute the appropriate operations, and is defined using the `*beginmacro` and `*endmacro` commands. Macros may accept data passed to them using the arguments `$1`, `$2`, etc. Each argument specifies where the values should be substituted. These macros are defined within the `.mac` files, including the `userpage.mac` file.

The following skeleton code shows the format of a **Utility** menu macro:

```
*beginmacro(macroname)
    command statements go here
*endmacro()
```

Utility menu macros consist of HyperMesh Tcl modify commands.

Load collectors can be created and edited using the **Model Browser**. Simply right click in the **Model Browser** and select *Create* > *Load Collector* to create one. To edit the name, color, or card image of a load collector, right click on the load collector name in the **Model Browser** and select *Edit*

The **Forces** panel can be accessed from the menu bar by selecting *BCs* > *Create* > *Forces*.

The **Forces** panel allows you to create and update forces.
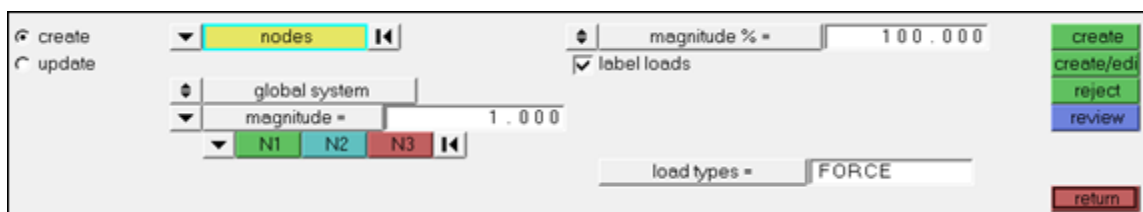
Exercise

In this exercise you will create a Tcl script from the command file commands, create a **Utility Menu** macro that runs the Tcl script and add a button on the *User* page that will launch the macro:

1. Define the task.
2. Delete the existing `command.cmf` file. This file is located in either the start-in directory or the current working directory.
3. Perform the operations in HyperMesh that the script should run.

4. Extract the commands from the `command.cmf` file.

5. Create a Tcl script by converting the commands to Tcl format and modifying as necessary.

6. Create a **Utility Menu** macro that runs a Tcl script.

7. Add macro button using `*createbutton` that calls the macro created in Step 6 with the appropriate Tcl script filename.

8. Reload the current `.mac` file into HyperMesh to load the modified `userpage.mac`.

9. Test the macro.

Step 1: Define the task.

The first step in creating a macro is to define the process you want to automate and recognize the individual tasks to reach the desired conclusion. Here, you want to create a one-button macro to automatically create forces on certain nodes. The actions necessary to complete this task are:

- Create a load collector for the forces

- Enter the **Forces** panel

- Apply forces to the nodes of interest

Step 2: Delete the existing `command.cmf` file.

The current `command.cmf` file is located in the current working directory. When first opening HyperMesh, the file is created in the directory HyperMesh is launched from. As soon as you begin working in HyperMesh all executed commands are written to the `command.cmf` file. If the file already exists, the commands are appended to the file. Deleting the file allows HyperMesh to create a new file and allows the user to easily find the relevant commands.

Step 3: Perform the operations in HyperMesh.

Execute the full process within HyperMesh. Every command issued in HyperMesh appears in the order executed and is reflected in the `command.cmf` file.

1. From the menu bar, select *File* > *Open* > *Model* and then load the file, `c_channel-tcl.hm`.

2. Right click in the **Model Browser** and select *Create* > *Load Collector*.

3. In the **Name** field enter the name `forces`.

4. Click *Create*.

5. Open the **Forces** panel.

6. Activate the *create* subpanel.

7. Click *nodes* and select one node in the model.

    -   For the direction of the force choose the *z-axis* option.

    -   For **magnitude=**, enter `23`.

    -   Toggle from **magnitude %** option to *uniform size* option for **load size** and set the **value** to `15`.

9. Click *create*.

Step 4: Extract the commands from the `command.cmf` file.

1. Open the `command.cmf` file using any text editor or use the **Open Command File** 🗐 option in the **Scripting Toolbar**.

2. Select and copy the following three lines in the file:

    ```
    *loadsize(1,15,0,1)

    *createmark(nodes,1) 3237

    *loadcreateonentity_curve(nodes,1,1,1,0,0,23,0,0,23,0,0,0,0,0)
    ```

    Observe the `*createmark` command and the list of entity id numbers. A mark is a storage buffer in HyperMesh. For some actions performed on entities, the entity ID is first entered into the designated mark.

    There are two marks available to the user (1 and 2) for each entity type (elements, nodes, lines, surfaces, points, etc.). At the execution of the command using the mark, the changes apply to all entities identified in the mark.

Step 5: Create a Tcl script named `create_force.tcl`, convert the commands to Tcl format and modify as necessary.

1. Create a new file named `create_force.tcl` using any text editor.

2. Paste the copied commands from the `command.cmf` file inside the `create_force.tcl` file.

3. Remove all `()` and `,` and replace them with spaces. The commands should look something like:

    ```
    *loadsize 1 15 0 1

    *createmark nodes 1 3237

    *loadcreateonentity_curve nodes 1 1 1 0 0 23 0 0 23 0 0 0 0 0
    ```

    Simply running the above commands will work without a problem, but note that the `*createmark` command is hard coded to the single node picked when generating the command file. Also notice that the magnitude is hard coded as well. This is not very useful for a generic utility.

4. Replace the `*createmark` command with the `*createmarkpanel` command.

    The command `*createmarkpanel` presents the user with a selection panel for the entity specified. The commands should now look like this:

    ```
    *loadsize 1 15 0 1

    *createmarkpanel nodes 1 "Select nodes for load creation"

    *loadcreateonentity_curve nodes 1 1 1 0 0 23 0 0 23 0 0 0 0 0
    ```

5. If you want to let the user specify the magnitude, prompt the user for a value using `hm_getfloat`. Then replace the hard coded magnitude in the `*loadcreateonentity_curve` command with the user defined value. The commands should now look like this:

    ```
    *loadsize 1 15 0 1

    *createmarkpanel nodes 1 "Select nodes for load creation"

    set mag_val [hm_getfloat "Magnitude=" "Enter force magnitude:"]

    *loadcreateonentity_curve nodes 1 1 1 0 0 $mag_val 0 0 $mag_val 0 0 0 0
    0
    ```

6. Save the `create_force.tcl` script.

Step 6: Create a Utility menu macro that runs a Tcl script.

1. Create a new **Utility Menu** macro that calls the `*evaltclscript` command to run a Tcl script, using the macro wrapper commands `*beginmacro` and `*endmacro`. In the `*beginmacro` command, name the macro `EvalTcl`.

    ```
    *beginmacro("EvalTcl")

        *evaltclscript($1,0)

    *endmacro()
    ```

    The macro name `EvalTcl` will be used to connect the button with the macro via the `macroName` field in the `*createbutton` command.

2. Save the `userpage.mac` file.

Step 7: Add the macro button.

Create a button on the **User** page to execute the macro.

1. Create a new button in the `userpage.mac` file.

    ```
    *createbutton(5,"Create Force",16,0,10,GREEN,"Create z-direction force on
    selected nodes","EvalTcl","create_force.tcl")
    ```

    This creates a button on page 5 (***User*** page), names it, places it in the 16th row, places its start at column 0, gives it a width of 10 columns, applies to it the color green, provides a help string and references the macro `create_force.tcl` defined in Step 6.

Notice that the full path is not used to reference the `create_force.tcl` script. A full path can be specified if the file is not located in one of the predefined paths that HyperMesh searches to find scripts. Users can add additional search paths using the `TCL_INCLUDE` environment variable. Relative paths can also be used from these search paths.

2. Save the `userpage.mac` file.

Step 8: Reload the current `.mac` file into HyperMesh to load the modified `userpage.mac`.

To reload the current macro menu `.mac` file while HyperMesh is open from the menu bar select ***Preferences*** > ***Menu Config*** and click on ***retrieve*** next to **macro file**. Make sure to load the proper .mac file from the `hm\scripts\<profile name>` directory based on the current user profile, or load the default `hm.mac` in `hm\bin\<platform>` if no user profile is loaded.

Step 9: Test the macro.

1. Click the **User** button on the **Utility Menu**.

   The new button labeled **Create Force** should be on the **User** page.

2. Click this button to run the Tcl script that automatically creates forces in the z-direction of the selected nodes.

   The new forces are created on the specified nodes with the given magnitude and placed in the current load collector If no load collector exists, the forces are placed in a load collector called **auto1**.

   It is often necessary to debug Tcl scripts using the **Command Window**. This allows you to run the Tcl script and easily review error messages, as well as print out debug information. Additional details can be found in the *Creating Tcl Scripts* and *Running Tcl Scripts* sections.

△ Altair