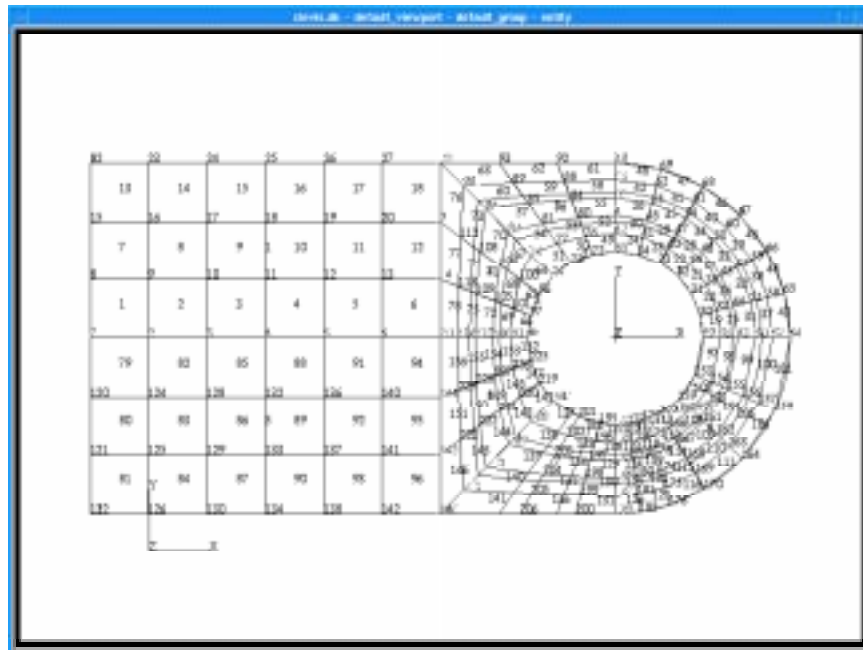


## Exercise 5

# *Mesh Lug Model*



### Objective:

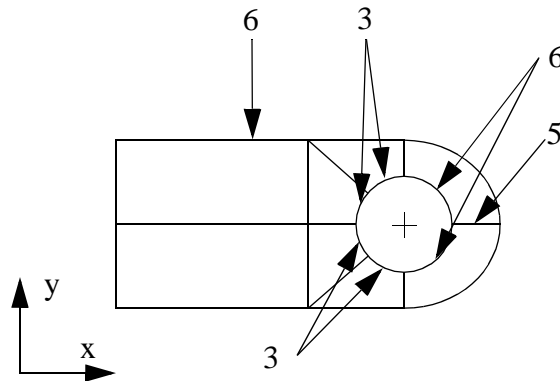
- Write a function to mesh the geometry of the lug.



**Exercise Description:**

This exercise, `lug_mesh ( )`, meshes a 2-dimensional model of a lug with parametric cubic patches. When complete, the model will have mesh seeds and a QUAD4 mesh.

Apply mesh seeds to the geometry resulting in the following number of element edges.

**Files:**

All the files that are used in this exercise are listed below. Each list includes the file, where it originated, and a summary of information of how it relates to the exercise.

File	Supplied/Created	Description
<code>lug_mesh.ses</code>	Created	You will create this file while you are building the mesh for the model.
<code>lug_mesh.pcl</code>	Created	This will be created from the session file after it is made into a PCL function.

**Exercise Procedure:**

1. Open up the `lug.db` database that you created in Exercise 4.
2. Create a new session file called **`lug_mesh.ses`**. As before we are going to use the session file as the body of our function.

**File/Session/Record ...**

---

lug\_mesh.ses

Apply

3. First place mesh seeds on the model of your lug. The picture on the previous page shows the number of seeds that should be placed on each curve.

◆ **Finite Elements**

Action:

Create

Object:

Mesh Seed

Type:

Uniform

◆ **Number of Elements**

Number =

6

Curve List

See figure on previous page  
for the placement of the seeds

**Note:** Use the same technique for placing all the seeds on the model. The number implies the number of elements that should be created on each curve.

4. Mesh the surfaces of the lug that you just created. Use the IsoMesh with quad4 elements. Select all the surfaces in the lug (this is most easily done by clicking and dragging over the lug).

Action:

Create

Object:

Mesh

Type:

Surface

Element Topology

Quad 4

◆ **IsoMesh**

Surface List

Surface 1:8

Apply

## Mesh Lug Model

Don't forget to equivalence your model as well

<i>Action:</i>	<b>Equivalence</b>
<i>Object:</i>	<b>All</b>
<i>Type:</i>	<b>Tolerance Cube</b>
<b>Apply</b>	

- When you are done creating the mesh stop recording the session file.

<b>File/Session/Record ...</b>	<b>lug_mesh.ses</b>
<b>Stop</b>	
<b>Cancel</b>	

- Go into the file that you just created called `lug_mesh.ses`, using a text editor such as `vi` or `jot`. Any line that begins with a dollar sign is a comment and can be deleted or ignored. Call the function `lug_mesh()`, remember to end the function.

After you are done editing the file, save it and then rename it:

**lug\_mesh.pcl**

- In the MSC.Patran command window enter the following command:

**!!input lug\_mesh.pcl**

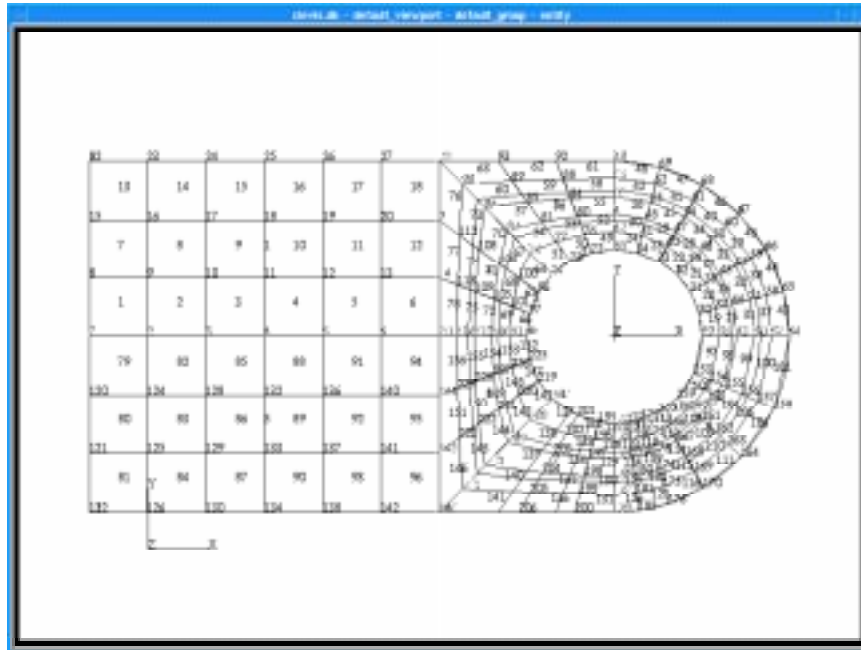
- Before you execute the function delete the Finite Elements and the Mesh Seeds on the model.

<i>Action:</i>	<b>Delete</b>
<i>Object:</i>	<b>Any</b>
<i>Curve List</i>	<b>Select all on Screen</b>
<b>Apply</b>	

- In the command window type:

**lug\_mesh()**

Your model should reappear as shown as shown below.



10. You will be using this model for the next exercise as well, so leave the database open and go on to the next exercise.

**Sample Solution:**

```

FUNCTION lug_mesh( )

/*
 * Purpose:
 * Mesh a previously created 2-dimensional model of a lug with
 * parametric cubic patches.
 *
 * Side effects:
 * A 2D lug is meshed from the specified dimensions
 */

STRING lug_geometry[], surface_2[], surface_5[], surface_6[]
STRING surface_7[], surface_8[], curve_1[], curve_2[], curve_3[]

INTEGER fem_create_mesh_surfa_num_nodes
INTEGER fem_create_mesh_surfa_num_elems
STRING fem_create_mesh_s_nodes_created[VIRTUAL]
STRING fem_create_mesh_s_elems_created[VIRTUAL]
REAL fem_equiv_all_x_equivtol
INTEGER fem_equiv_all_x_segment

/*
 * Begin by seeding the mesh
 */

mesh_seed_display_mgr.init()
mesh_seed_create( surface_2/"1 " , 1, 5, 0., 0., 0. )
mesh_seed_create( surface_6/"1 " , 3, 6, -3., 0., 0. )
mesh_seed_create( surface_7/"1 " , 3, 3, 2., 0., 0. )
mesh_seed_create( surface_8/"1 " , 1, 3, 0., 0., 0. )
mesh_seed_create( curve_3 " , 1, 3, 0., 0., 0. )
mesh_seed_create( curve_2 " , 1, 3, 0., 0., 0. )
mesh_seed_create( curve_1 " , 1, 6, 0., 0., 0. )
mesh_seed_create( surface_5/"1 " , 1, 6, 0., 0., 0. )
mesh_seed_display_mgr.erase()

/*
 * Mesh the surfaces
 */

fem_create_mesh_surf( "IsoMesh", lug_geometry, 0.1, @
                    "Quad4", "#", "#", @
                    "Coord 0", "Coord 0", @
                    fem_create_mesh_surfa_num_nodes, @
                    fem_create_mesh_surfa_num_elems, @
                    fem_create_mesh_s_nodes_created, @
                    fem_create_mesh_s_elems_created )
fem_equiv_all_group2( [0], 0, 1, 0.00499999999, FALSE, @
                    fem_equiv_all_x_equivtol, @
                    fem_equiv_all_x_segment )
gm_segment_delete( fem_equiv_all_x_segment )

END FUNCTION

```

