

# Engineering Sketch Pad (ESP) Training

## Session 8: Writing a UDP

John F. Dannenhoffer, III

Syracuse University

Bob Haimes

Massachusetts Institute of Technology

Revised for v1.08





# Overview

- Review of EGADS geometry and topology models
- EGADS documentation
- Steps to writing a UDP
- Sample UDP
  - structure of code
  - code walk-through
  - stand-alone execution
  - execution as a UDP
- Tire UDP

<div>Top Down</div> <div>↓</div> <div>↑</div> <div>Bottom Up</div>	Topological Entity	Geometric Entity	Function
	Model		
	Body	Solid, Sheet, Wire	
	Shell		
	Face	<b>surface</b>	$(x, y, z) = \mathbf{f}(u, v)$
	Loop		
	Edge	<b>curve</b>	$(x, y, z) = \mathbf{g}(t)$
	Node	<b>point</b>	

- *Solids* are open at machine precision – tolerances
  - Node points that bound Edges may not be on the curve
  - Edge curves that bound the Faces (through Loops) may not be on the underlying surface



# EGADS Documentation

Included in ESP distribution

- Overview
- Objects
  - Geometry
  - Topology
  - Tessellation — Others
- API
  - Utility & IO Functions
  - Attribution
  - Geometry
  - Topology
  - Tessellation
  - High-Level Functions



# Steps to Writing a UDP

- **Draw a picture**
- Define input and output parameters
  - name (case-insensitive)
  - type (ATTRSTRING, ATTRINT, -ATTRINT, ATTRREAL, -ATTRREAL, ATTRREALSEN)
  - size
  - default value(s)
- Build the Body (stand-alone)
  - bottom-up
  - top-down
  - combination
- Test stand-alone with vTess
- Write a .csm file
- Test the UDP

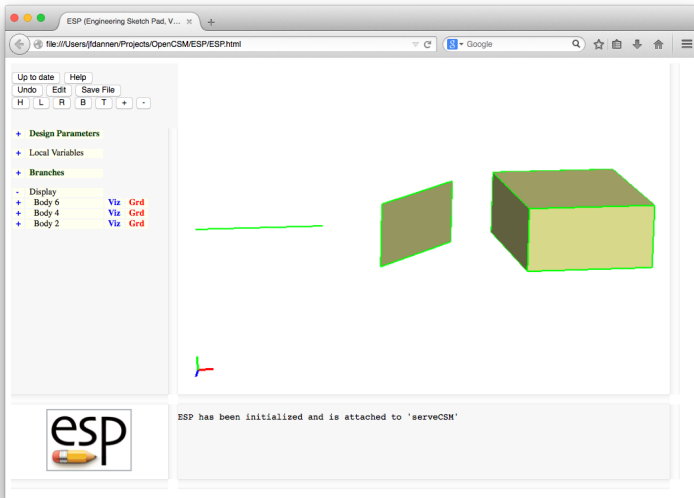


# Sample UDP

- Inputs:
  - `dx` — `ATTRREALSEN`, `default=0`
  - `dy` — `ATTRREALSEN`, `default=0`
  - `dz` — `ATTRREALSEN`, `default=0`
- Outputs:
  - `area` — `-ATTRREAL`, `default=0`
  - `volume` — `-ATTRREAL`, `default=0`
- Notes:
  - if `dx`, `dy`, and `dz` are all positive, create a box
  - if two of `dx`, `dy`, and `dz` are positive, create a plate
  - if one of `dx`, `dy`, and `dz` is positive, create a beam
  - otherwise, raise an error



# Sample UDP





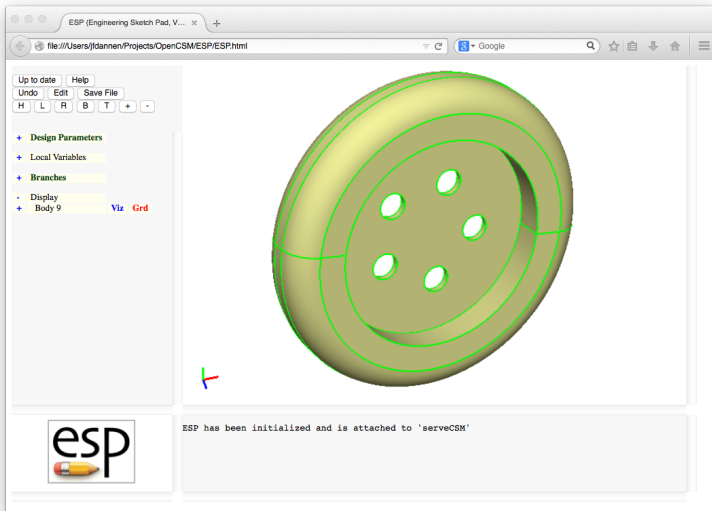
# Code Walk-through

- Source file is `udpSample.c`
- To build under LINUX/OSX
  - `make -f sample.make`
- To build under Windows
  - `nmake -f sample.mak`
- To run stand-alone
  - `sample`
  - `$ESP_ROOT/bin/vTess sample.egads`
    - open browser on `$ESP_ROOT/bin/wv.html`
- To run in ESP
  - `$ESP_ROOT/bin/serveCSM sample`





# Tire UDP: Finished Product





## Tire UDP: Inputs and Outputs

Name	Description	Default
width	width of tire	0
minrad	minimum radius of tire	0
maxrad	maximum radius of tire	0
fillet	fillet radius at outside of tire	0 (for none)
platethick	wheel thickness	0 (for none)
bolts	number of bolt holes	0 (for none)
patternrad	radius of bolt hole circle	0
boltrad	radius of each bolt hole	0 (for none)
volume	volume	output

# Tire UDP: Strategy (1)

- Draw a sketch, with Nodes, Edges, and Faces numbered
- Define the inputs and outputs
  - check size (scalar vs. multi-valued)
  - check validity
- Build basic tire from bottom up
  - 8 Nodes
  - 8 Edges (linear) at the equator
    - generate a linear curve
    - inverse evaluate at Nodes to get  $t_{\text{beg}}$  and  $t_{\text{end}}$
    - make the Edge
  - 8 Edges (circular)
    - generate the circular curve
    - inverse evaluate at Nodes to get  $t_{\text{beg}}$  and  $t_{\text{end}}$
    - ensure  $t_{\text{end}} > t_{\text{beg}}$  by increasing  $t_{\text{beg}}$  by  $2\pi$  if needed
    - make the Edge
  - ...



# Tire UDP: Node Numbers

The screenshot displays the ESP (Engineering Sketch Pad) software interface. The main workspace shows a tire model with 8 numbered nodes (1-8) and green construction lines. The left sidebar contains a 'Design Parameters' table and a 'Local Variables' section.

Design Parameters	
width	5
minrad	8
maxrad	12
fillet	0
platethick	0
patternrad	4
bolts	5
boltrrad	1

Below the table are sections for 'Local Variables', 'Branches', and 'Display'.

The console log at the bottom shows the following messages:

```
Parameter 'platethick[1,1]' has been changed to 0 =====> Re-build is needed <=====
Unknown command (keyPress=0, modifier=0, keyCode=13). Use '?' for help
Re-building...
Entire build complete, which generated 1 Body(s)
```



# Edge Numbers for Tire UDP

The screenshot shows the ESP (Engineering Sketch Pad) software interface. The main window displays a tire model with 16 numbered edges. The edges are labeled 1 through 16. The tire is composed of several concentric rings and a central hub. The edges are numbered as follows: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16. The edges are highlighted in green. The software interface includes a menu bar (File, Edit, Save File, Help), a toolbar (H, L, R, B, T, +, -), and a Design Parameters panel on the left. The Design Parameters panel lists parameters such as width, minrad, maxrad, fillet, platethick, patternrad, bolts, and boltrrad. The bottom status bar shows the message: "Parameter 'platethick[1,1]' has been changed to 0 =====> Re-build is needed <===== Unknown command (keyPress=0, modifier=0, keyCode=13). Use '?' for help Re-building... Entire build complete, which generated 1 Body(s)".

ESP (Engineering Sketch Pad, V...)

file:///Users/jfdannen/Projects/OpenCSM/ESP/ESP.html

Up to date Help

Undo Edit Save File

H L R B T + -

Design Parameters

- width 5
- minrad 8
- maxrad 12
- fillet 0
- platethick 0
- patternrad 4
- bolts 5
- boltrrad 1

Local Variables

Branches

Display

Parameter 'platethick[1,1]' has been changed to 0 =====> Re-build is needed <=====

Unknown command (keyPress=0, modifier=0, keyCode=13). Use '?' for help

Re-building...

Entire build complete, which generated 1 Body(s)

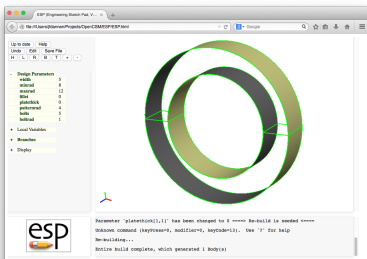
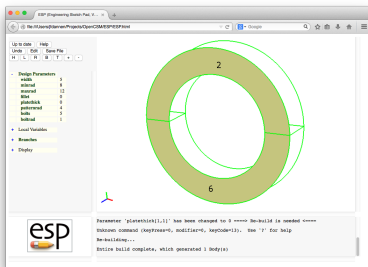
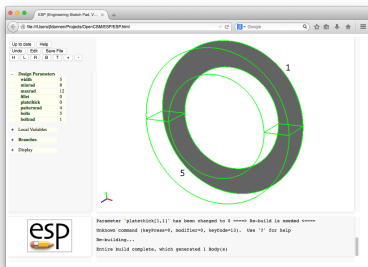


## Tire UDP: Strategy (2)

- Continue bottom up build
  - 4 Faces (planar)
    - make a Loop of 4 Edges
    - make the (planar) Face
  - 4 Faces (cylindrical)
    - make cylindrical surface
    - make a PCurve for each Edge that bounds Face
    - make a Loop of 4 Edges and 4 PCurves
    - make the (cylindrical) Face
  - 1 Shell that combines the 8 Faces
  - 1 Solid Body from the Shell

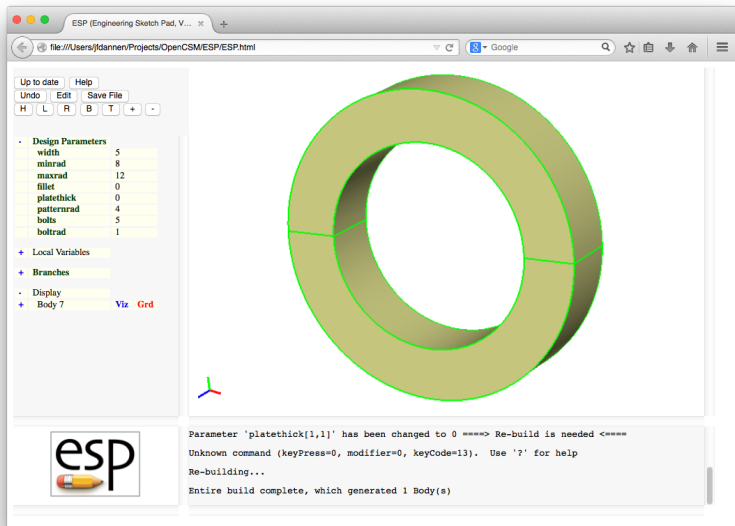


# Tire UDP: Face Numbers





# Tire UDP after Bottom-up Build





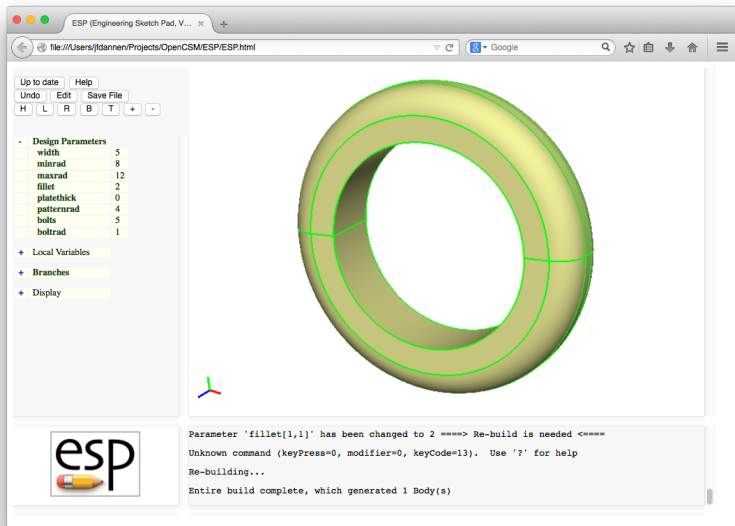


## Tire UDP: Strategy (3)

- Modify the Body top-down
  - fillet on outer Edges
    - identify the 4 Edges
  - add wheel
    - cylinder that is “unioned” with the tire
  - add pattern of holes
    - cylinders that are “subtracted” from the wheel
- Compute and return the “output” variables
- Note: this entire UDP could have been written top-down, but was broken up to show the steps needed in bottom-up construction

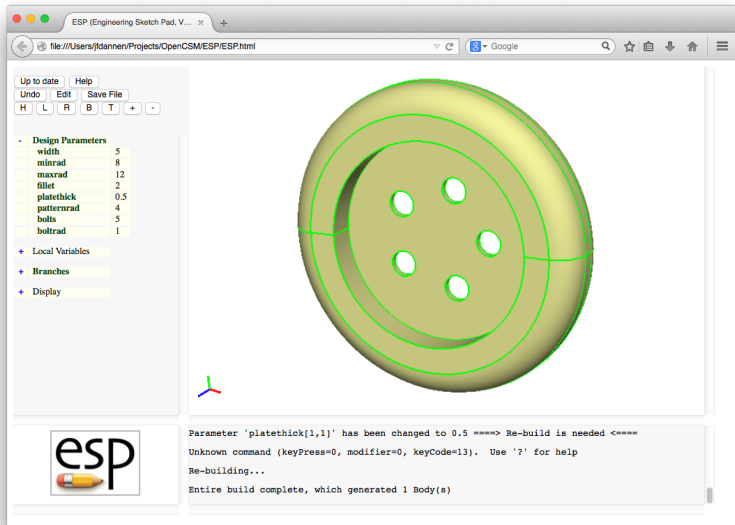


# Tire UDP: after Filleting





# Tire UDP: after Wheel with Holes





# Muddy Cards

- Questions / suggestions about writing UDPs
- Questions / suggestions about whole course
- Overall effectiveness of course