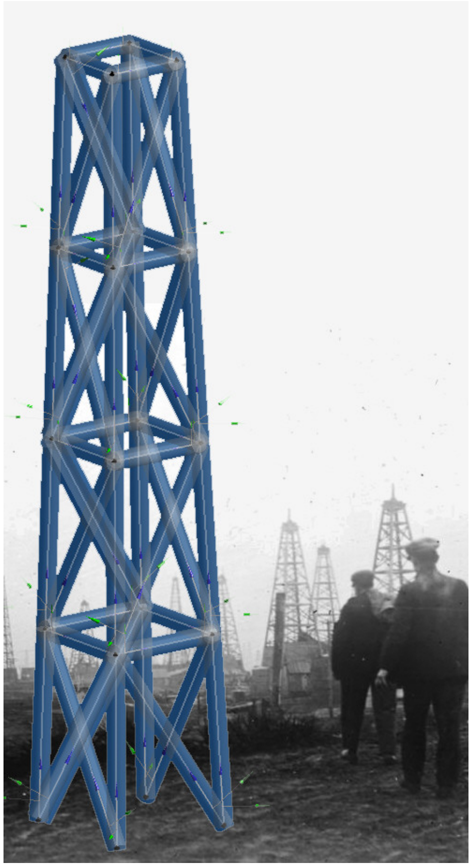
The Focus

October 6, 2006 | A Publication for ANSYS Users | Issue 52



Beam me Up, Point File

By Doug Oatis

One capability that seems to have fallen through the cracks in Workbench is beam modeling. Using Design Modeler, you can easily define the cross-section for common beam sections, or define a user-integrated real constant set. One way to create beams is to define construction points on one or more sketching planes, and then define line bodies between points. This method, however, is rather impractical for defining large truss bodies.

Luckily, Design Modeler has a feature that allows it to read point files. This allows the user to define a listing of vertices in terms of x/y/z position. The only restriction is the point file must follow the following format:

Group ID	Point ID	X	Y	Z
INT	*INT*	*REAL*	*REAL*	*REAL*
1	1	1.2	0	0
1	2	1.2	1.35	0
2	1	1.2	0	1.3
...	...	...	...	...

The Group and Point Numbers must be integers, and you can use the same Point Number for different Group Numbers (i.e. You can have Group 1 - Point 1 and Group 2 - Point 2). The x/y/z coordinates are interpreted relative to the global coordinate system and active unit system. You only need a space between each of the data columns.

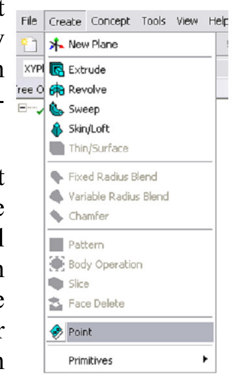


Figure 1: Insert Point

To read in the point file, simply go Create > Point. This will create a 'Point' item in the model tree. In the Details window for this 'Point', click on the (Cont. on Pg. 2.)

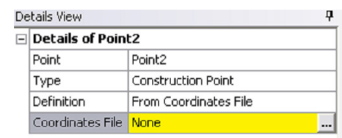


Figure 2: Point Detail

Must Have ANSYS Skills

By Rod Scholl

Although the title sounds like a want ad – it is meant to refer to what skills a new user often misses, but should probably sit down and learn. Mostly, these 20 items are a list of things that I didn't want to learn for one reason or another (usually laziness) but later proved to have large bang for the buck. These things took many hours and tricks to avoid learning, when just learning them would've saved time and heartache.

So if you scan down this list, and say "yep, yep,… got it… got it… don't need it… sorta got it… I'll learn that next time it comes up… have a workaround… I just avoid that… never heard of that, probably a typo…." And so on – well then you're probably a lot like me and this article will fade quickly from mind and be mostly useless – but you could always forward to a new user who might appreciate it. But, on the off chance that you're more studious and visionary than me – well then shazaam! Read on!

1) Know what the .log file is. Read it often. If you are on Linux/Unix – use a command to read it on the fly:
```
tail -f file.log
```

2) Always check the .err file. If I had a dime for every time I struggled with non-convergence or some other difficulty, only to find hours later that the .err file gave a telltale warning, or even an exact description of my erroneous ways. Note that not every error/warning is written to the output window. I still get caught by this over and over…

3) APDL – this is really the thing to get excited about (link to matrix article). (Cont. on Pg. 5.)


Figure 1: Misc. Vacation Shot from Durango CO to add color to Rod's Article

Contents

(Points, Cont.)

yellow box labeled "Coordinates File". Click on '…' and this will allow you to browse to the coordinates file location. Hit OK and then generate. You'll then see all of your points generated in the graphics window.

You can now use the "Concept > Lines from…" by selecting point pairs to generate your line bodies. After you're done creating your line bodies, you can create a cross-section using Concept > Cross Section. Fill out the dimensions for your cross section, and then use the parts list (bottom of DM model tree) to assign a cross section to each line body.

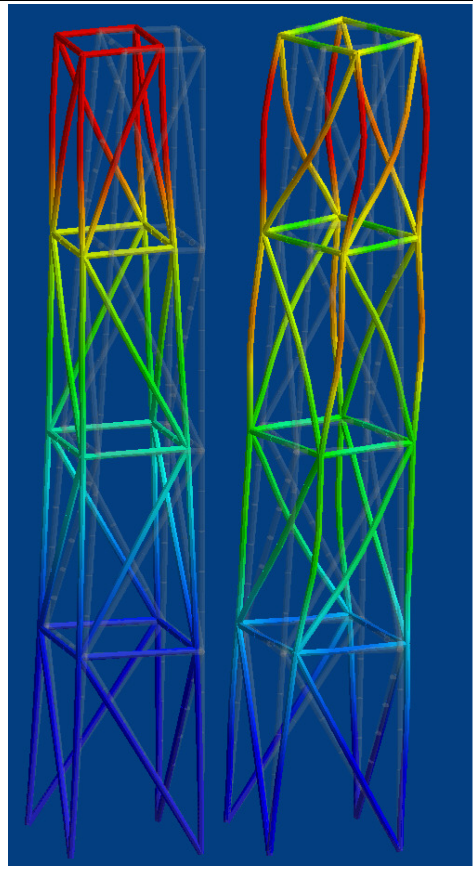
If you've gone through and generated your entire model, and then realize you need to move some points around, fret not! Simply modify your point file (I always rename it so I have a trail to follow back) and save. Then, click on the 'Point' object in the model tree and browse to the new file. Hit "generate" and all of your model will regenerate. Points will shift to new positions, line bodies will be automatically generated and cross-sections will be defined and assigned (if you defined/assigned them before-hand).

If you need to create some type of truss structure at an angle or relative to a different coordinate system, there is a way to work around the global origin interpretation. Simply create your model as usual, then use the Create > Body Operation to move the line bodies to another coordinate system.

To create these point files, I use Excel's "concatenate" feature. This is a very clean way to define Excel columns for user input, and have a DM input format automatically generated. In my excel file, I have 6 columns of information. The first 5 are reserved for DM info (Group, Point, X, Y, Z), and the 6th is a single line containing all of the info with a single space in-between each column.

Group #	Point #	X	Y	Z	Put this into a text file
1	1	1	1	0	1 1 1 1 0
1	2	1	-1	0	1 2 1 -1 0
1	3	-1	1	0	1 3 -1 1 0
1	4	-1	-1	0	1 4 -1 -1 0
2	1	0.9	0.9	1	2 1 0.9 0.9 1
2	2	0.9	-0.9	1	2 2 0.9 -0.9 1
2	3	-0.9	0.9	1	2 3 -0.9 0.9 1
2	4	-0.9	-0.9	1	2 4 -0.9 -0.9 1
3	1	0.8	0.8	2	3 1 0.8 0.8 2
3	2	0.8	-0.8	2	3 2 0.8 -0.8 2
3	3	-0.8	0.8	2	3 3 -0.8 0.8 2
3	4	-0.8	-0.8	2	3 4 -0.8 -0.8 2
4	1	0.7	0.7	3	4 1 0.7 0.7 3
4	2	0.7	-0.7	3	4 2 0.7 -0.7 3
4	3	-0.7	0.7	3	4 3 -0.7 0.7 3
4	4	-0.7	-0.7	3	4 4 -0.7 -0.7 3
5	1	0.6	0.6	4	5 1 0.6 0.6 4
5	2	0.6	-0.6	4	5 2 0.6 -0.6 4
5	3	-0.6	0.6	4	5 3 -0.6 0.6 4
5	4	-0.6	-0.6	4	5 4 -0.6 -0.6 4

After some practice, you'll be on your way to creating complex (and flexible) truss structures!



You can download the files used for this article at ftp.padtinc.com /public/downloads/Point_Beam_Files.zip

Avoiding Extinction: Using Shell Sections

By Eric Miller

Realizing you are a dinosaur can be an embarrassing thing. This process is becoming all too familiar: A customer asks if some new feature can do this or that and you confidently tell them no, not possible, then find out from some one that there is a whole rich set of options to do what they want. This happened to me the other day, and the person that pointed out that I didn't know the new functionality was a senior manager in development. Ouch! So, as penance and to erase my shame I've decided to learn about the feature in question, defining shell properties with sections instead of real constants, and then write an article.

You use this capability with the newer generation of shell elements: SHELL181 and SHELL281 are the 3D structural versions we will focus on. Many of the commands also apply to layered solids (186,190), Axisymmetric shells (208,209) and some thermal shells (131, 132) Although they support some real constants for the most common options, to really use these elements for composites, offsets, and greater accuracy, you need to use Shell Section commands documented in Table 1.

Take a look at the documentation for each command. They are fairly typical ANSYS commands and are easy to implement. SCT1.MAC is a simple macro that makes a cantilever beam with thickness of 0.125" and three integration points across the thickness. The two bolded section commands replace the older style real constants. One thing that is nice is that you can assign a name to the section. This comes in handy when plotting or if you use the GUI to manage your sections. (Cont. on Pg. 3.)

SECTYPE	Define a section and its type. For shells type = SHELL
SECDATA	Define the geometry. For shells you give the thickness, material, angle, number of integration points and a name for each layer
SECCONTROL	Overrides calculated properties. For shells you can define your own transverse shear stiffnesses, added mass, hourglass control and drill stiffness
SECFUNCTION	Links a function to shell
SECNUM	Defines section number to use for new elements (the TYPE/REAL/MAT for sections)
SECOFFSET	Defines offset of shell. Can be Top, Mid, Bot or by some number
SECPLOT	Plots geometry of a shell section definition including layers and fiber directions
SLIST	Summarizes section properties
SDELETE	Deletes a section definition

Table 1: Section Commands Used on Shells

(Sections, Cont.)

We can get fancier and create a composite shell that is offset by adding SECDATA commands for each new layer and the SECOFFSET command as shown in Code Snip 1. Go ahead and replace the two bolded sections with the snip and run the model. Issue an EPLOT and zoom in. Notice how ANSYS shows the different layers, thicknesses and materials for you as shown in Figure 1. Turn on the working plane and zoom in on the origin to check the offset.

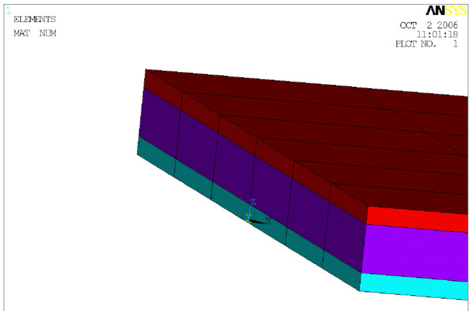


Figure 1: Plot Showing Layers

If you go back to PREP7 and issue a SECPLOT,1 you will see the section information shown in Figure 2. Not only does this informative plot let you catch mistakes in material assignment and orientation, but it is also great for documentation, as is the SLIST command.

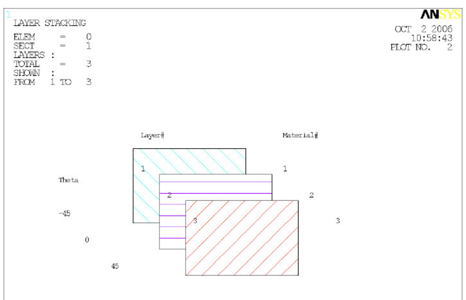


Figure 2: Section Layer Stacking Plot

One little thing that it took me a while to realize is that there is no add/delete/move command for the section properties. This is because every time you issue the SECTYPE command, the previous definition for the ID number given is erased and a new one is created using any commands that follow the SECTYPE.

To get one step fancier, replace the section commands with Code Snip 2. This defines a composite layer dropoff. Notice that you drop off a layer by setting its thickness to zero. Also notice how the material number of the top most layer is used for each dropped layer. This is done because ANSYS plots a zero thickness layer so to get the colors right, change the material number.

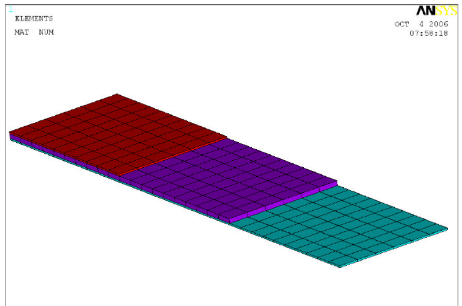


Figure 3: Plot Showing Layer Dropoff

The last example can be run by swapping out Code Snip 3. This is the same model but the thickness is defined with the table in the snippet. If you exaggerate the thickness with /ESHAPE,10,1 you can see how the thickness varies. One important thing to note is that the thickness of a given layer is scaled based upon its share of the total thickness as specified in the SECDATA command.

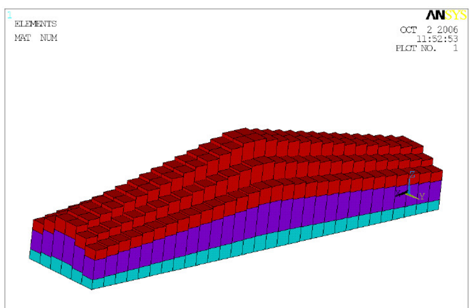


Figure 4: Thickness Defined by Table

These examples have all used APDL. You can achieve the same effect with a nice GUI tool under: Preprocessor->Sections->Shell->Lay-Up->Add/Edit. One of the advantages of the GUI is that you can use the Tools menu to copy layers as well as impose a symmetric layup scheme.

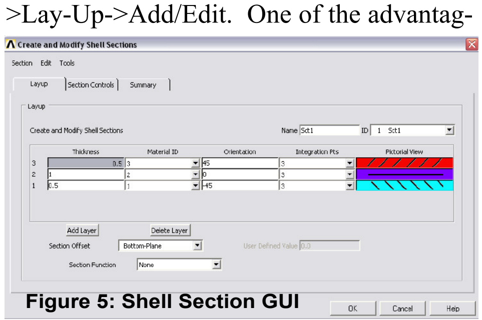


Figure 5: Shell Section GUI

```
! Start up a clean model
finish $/clear $/prep7
! Build a 2 x 10 rectangle
rectng,0,3.333,-1,1
rectng,3.333,6.666,-1,1
rectng,6.666,10.0,-1,1
aglue,all
numcomp,area
!Set up mat/esize/etype
et,1,181
keyopt,1,8,1
esize,.35
mp,ex,1,10e6 $mp,nuxy,1,.23
mp,ex,2,10e6 $mp,nuxy,2,.23
mp,ex,3,10e6 $mp,nuxy,3,.23
!------Section Definition
sect,1,shell,,Sct1
secdata,0.125,1,0,3
!--------End Section Definition
!Mesh, load and constrain
amesh,all
sfa,all,1,pres,-1
nsel,s,loc,x,0
d,all,all
nsel,all
!Plot Elements
/eshape,1 $/num,1 $/pnum,mat,1
/view,1,.25,-1,.25 $/vup,1,z
eplot
!Solve and Plot Results
finish $/solu $solve
finish $/post1 $/dscale,all,1
plnsol,u,sum,1,1
```

CODE SNIP 1

```
sect,1,shell,,Sct1
secdata,0.03,1,-45,3
secdata,0.08,2,0,3
secdata,0.03,3,45,3
secoffset,bottom
```

CODE SNIP 2

```
sect,1,shell,,Sct1
secdata,0.03,1,-45,3
secdata,0.08,2,0,3
secdata,0.03,3,45,3
secoffset,bottom
sect,2,shell,,Sct2
secdata,0.03,1,-45,3
secdata,0.08,2,0,3
secdata,0,2
secoffset,bottom
sect,3,shell,,Sct3
secdata,0.03,1,-45,3
secdata,0,1
secdata,0,1
secoffset,bottom
! Define sections for areas
asel,s,,,2
aatt,1,1,1,0,2
asel,s,,,3
aatt,1,1,1,0,3
asel,all
```

CODE SNIP 3

```
*dim,thkvl,table,3,3,,X,Y
thkvl(1,0) = 0,5,10
thkvl(0,1) = -1,.08,.125,.08
thkvl(0,2) =  0,.125,.25,.125
thkvl(0,3) =  1,.08,.125,.08
sect,1,shell,,Sct1
secdata,.5,1,-45,3
secdata,1,2,0,3
secdata,.5,3,45,3
secfunction,%thkvl%
secoffset,bottom
```

ANSYS CFX: Importing Meshes into CFX

By J. Luis Rosales

The purpose of this article is to illustrate the process of preparing and transferring a mesh model from ANSYS and ANSYS Workbench to CFX. You can create meshes for CFX in a variety of places, with ICEM CFD or CFX being the prefered location. But if you don't have these tools, you can use the meshers in ANSYS or WB Simulation. Although a lot of thought and practice are needed to create a suitable mesh for a CFD simulation, this article will only cover the steps required to transfer the mesh. The first example will show the steps for creating boundaries and moving the mesh structure from ANSYS to CFX. The second example will show how to setup and transfer the mesh models from WB to CFX.

Example 1: ANSYS to CFX

An image of the mesh structure in ANSYS is shown below in Fig. 1. The model is a simple straight tube with flow coming in at one end and leaving from the other end. A quadrilateral grid was generated on one of the end-faces of the model and swept through the volume.

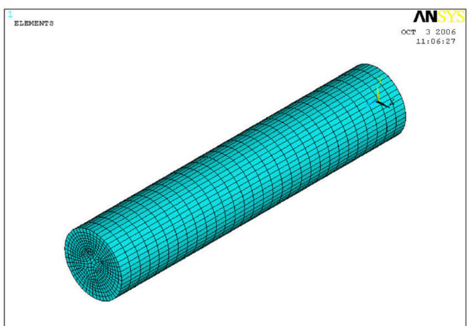


Figure 1: Mesh in ANSYS

Note: MESH200 elements were used to make the quadrilateral faces and thermal SOLID70 elements were used in the volume. Once the volume is meshed, the Mesh200 elements must be deleted since that will cause a problem when transferring the model to CFX. PADT has successfully used both SOLID70 and FLUID142 elements, so any single type of volume element should be fine.

The next step is to define the boundaries for a CFD analysis. In this case, only an inlet, an outlet and tube wall boundaries are needed. First, isolate the nodes attached to the inlet area by using the Select Entities window. Select the area by picking and then isolate the nodes attached to that area. Plot the nodes to ensure that you have the right set. Figure 2 shows the isolated inlet nodes after following the procedure above.

Now that the nodes are isolated, create a nodal component called INLET. This can be done using the Component Manager or with APDL. Do the same thing for the nodes at the outlet and call the component OUTLET. The nodes on the tube wall can also be isolated and placed in a component called WALL. If the nodes are plotted by components, ANSYS should display a picture similar to that shown in Fig. 3 below.

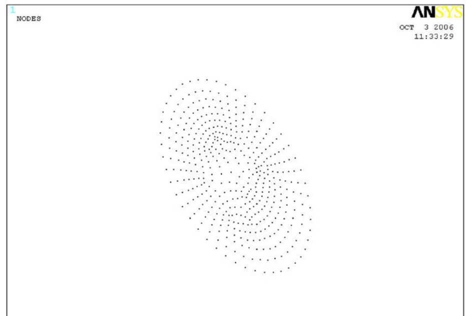


Figure 2: Nodes on Inlet

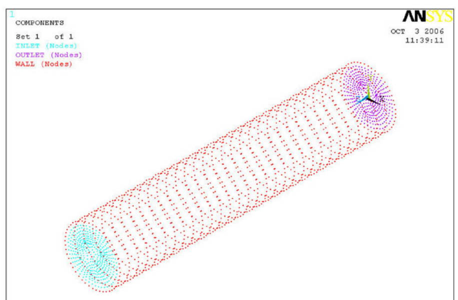


Figure 3: Components in ANSYS

Note that any surface nodes that are not placed into a component may be placed in a default boundary or boundaries by CFX. If the grouped nodes are all in the same type of boundary with the same boundary condition, then they don't really have to be defined as components in ANSYS.

The next step is to create a file that can be read by CFX. This is done by creating an archived file of the type All Associated FE and IGES (2 files). The two files created have an extension *cdb and *iges. The *iges file will not be used. The *cdb file can now be loaded into CFX after starting a new project and importing the mesh using the file type ANSYS (*cdb, *inp). The boundaries that were created in ANSYS Classic are now available for use in CFX.

Example Problem 2: ANSYS Workbench to CFX

The IGES file created by archiving the mesh model in ANSYS is used to show the steps required to produce a CFX file in ANSYS Workbench. The mesh is created in the Simulation module in WB and not in CFX Mesh. After importing the IGES file into the Simulation module, the different boundary areas of interest are named and sized for meshing. The boundary names for the model are created by right clicking on Environment > Insert > Pressure. The value of the pressure is arbitrarily set to a value of 1 MPa. The pressure boundary is used in this example to create the inlet, outlet and wall by selecting each area individually. The boundaries are easily renamed by right clicking on the pressure labels and typing INLET, OUTLET and WALL, respectively for each pressure label created.

The next step is to create the mesh. The areas have a default mesh sizing that can be changed by creating a sizing label under Mesh > Insert > Sizing. The wall setting is left with a default value while the inlet and outlet areas are resized to a smaller value. To view the mesh, select Mesh > Preview mesh. The resulting mesh visible in the WB viewer is shown below in Fig. 4

The next step is to define a result variable. Although no solution will be solved, this is required before exporting the mesh to CFX. Right click on Solution > Insert > Stress Tool > Max Equivalent Stress. Note that the solution variable is arbitrary. Before we export the mesh, make sure that no contact surfaces or springs are activated, as this will cause a problem.

Highlight Solution in the tree structure and then from the utility menu scroll down the Tools menu and click on "Write ANSYS Input File …" The output file created will end with *inp. After reading this file into CFX using the same procedure as outlined for the ANSYS Classic transfer, the resulting view will be just like that shown in Fig. 4.

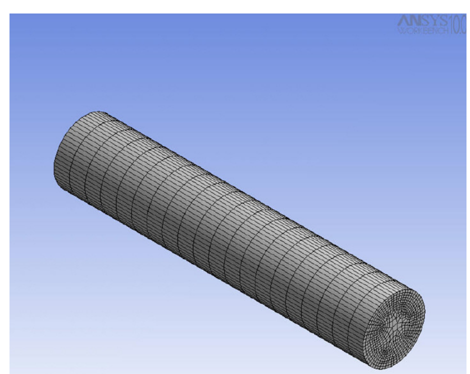


Figure 4: Mesh in WB Simulation

(Skills, cont.)

Taking a class (Focus 25) is about the best ROI on training one can expect. If you want to explore a little on your own, and don't know where to get started here's a suggested approach: a) Make and save your .db as normal, b) Write this script file:

```
resume,myjobname,db
/solu
Solve
/post1
plns,u,sum
!or whatever your DOF solved is
```

And c) Presto! You're a scripting master… eventually you might find you put more in the script, and less in the saved .db. A perfect example of this which will help archiving file size is to mesh in the script, or solve for multiple scenarios:

```
resume,myjobname,db
/prep7
vmesh,all
/solu
Solve
D,47,ux,.01       !add   another
displacement condition
Solve
…
```

This is how I got started… and over time scripting just sort of takes over where its easier, and I still use the GUI for some things for which I prefer the interface (although I usually just cut and paste this out of the log file into my script).

4. Learn the /filename command. Do all your files still look like this?

   ```
   File.log, file.rst, file.db
   ```

   If so you can get away with this for a while, but the time will come where you need to specify and change the names… and with the added organization benefit of controlling your file names, I think its best just to take the 5 minutes to learn how this works early.

5. Speaking of /filename – learn to do restarts (RESCON) (Focus 12). This is necessary whenever you are doing a nonlinear analysis. That is, if you want to get it finished early and make happy hour.

6. Detach the input window and resize it. This input line detaches and is resizeable so you can scroll up to recent commands. (link to article)

7. Learn to specify the –m and –db settings (Focus 21) on startup, or you might spend a lot more time staring at an hourglass.

8. Best way to get an upgrade for your 833mhz PC is to show the actual difference in solve times and the relatively insignificant cost of hardware upgrade. Run the PADT benchmarks and show them to your boss compared to hardware. (padtinc.com Benchmark Page)

9. Never miss geometric symmetry. Easy to do – and look silly later. If it's any solace, we see experts miss symmetry all the time. Maybe even get fancy and recognize possibilities for anti-symmetry and beyond (Focus 32).

10. Learn when to use the PCG solver vs. the sparse. Sometimes the sparse is faster, but in most cases you should start with the PCG – if it was the noticeably slower solver, ANSYS will sometimes give you a warning to change to sparse. I use 500-800 iterations as a warning that sparse would have been faster – less than that and you will appreciate the sometimes notably faster solution times.

11. Don't mess with unmeshable geometries. Give it a couple ESIZE or SMRT settings, then just mesh it in Workbench. This can save days of pointless struggle within PREP7 by using methods in Workbench which snap a mesh onto a surface in a way that it skips over sliver areas, unsewn surfaces etc. (Search Focus for Meshing)

12. Use /DSCALE when post-processing. I often exaggerate deflections to get a feel for different parts' behavior.

13. When using contact, consider bonded contact and other options available per KEYOPT 12. Often this can eliminate a nonlinearity early in the solve cycle and save oodles of equilibrium iterations, and even substeps. Consider using a FKN of 0.01, if you can live with the error in penetration. Also, always use KEYOPT 10=1 (or 2) to speed convergence. (link to article)

14. Avoid the friction trap. Adding friction has a rule of thumb of doubling solve times. It is also path dependent, and for many components friction changes over the life of the product with corrosion, wear, lubrication. Maybe a hand-calc of contact pressure after a linear solution can convince you is static friction (not sliding) – or that the resistance/stress due to friction is not a factor in your analysis. At least COF's are easy to vary… so you might have to enter a high and low value to bound the problem.

15. Learn *TREAD format. At some point you will have to take columnar data into ANSYS. There are many ways to do this, but the way I used for ages was to format the way I wanted in Excel (including columns of commas) – and then using the forgiving *TREAD to bring it into ANSYS. Then if you need it as an array you can convert it over. (Focus 48)

16. Learn birth and death. This sounds hard, but is really two commands. EALIVE and EKILL. Powerful and necessary in some situations, and a cinch to learn to implement. (Focus 22).

17. Use a backup tool. How often is this skipped? A good tool will let you choose what size file to backup giving its extension. I backup .rst's up to 50MB, but I backup anything with a .inp or .mac, for example. And I never backup a .osav. I find AISBackup to be the best tool out there. (Focus 36).

18. Have a short macro that makes a .png or .jpg file, and make them willy-nilly. You should be able to do this with a few keystrokes to clutter your directory up with interesting images. Later making .ppt's or trying to explain a phenomena you can browse them and put them to use.

19. Understand how to make, plot and list an ETABLE – including the NMISC and SMISC items. (Focus 38)

20. Learn *VWRITE and use APDL to write scripts in some circumstances. The need occasionally arises, and for all the years I avoided this, a couple example formats would have saved all sorts of workarounds.

Awesome APDL: Counting Components

Cruising through the macro pile at PADT we came across a simple little macro that does a sort of string search on component names. This can be useful if you group a bunch of items by using components with common roots like "holes_" or "fet". You can use it to find out how many components you have and use that for some other macro.

It is also a good example of string functions, something that can really add a lot of power to your macros.

Take some time to look up the key routines used: STRLENG, UPCASE, and STRSUB

They are documented in the APDL Programmer's Guide under Appendix B: Get Function Summary. While you are there, check out all the other useful functions that are not clearly documented elsewhere.

```
!- cntcmp
! Counts Components with
! a given string in their name
!
!  CNTCMP,strng
!     Strng:  ANSYS char string
!             that is the root
!             you want to search
!             For.
!             Remember to place
!             in single quotes
/nopr
!Store Root in reg_name
req_name_= arg1

! Get Length of Name
d_test=STRLENG(req_name_)

! Convert to All Caps
req_name_=UPCASE(req_name_)

! Get number of Comps in Model
*get,num_comp,comp,,ncomp

! Loop on components
!  Store name in a numbered
!  Variable
*do,i,1,num_comp
  *get,comp_name%i%,comp,i,name
*enddo

! Zero out your counter
nam_count_=0

! Loop on components, look for
!   base name. On Match
!   Increment counter
*do,i,1,num_comp
  d=strsub(comp_name%i%,1,d_test)
  *if,d,EQ,req_name_,THEN
    nam_count_=nam_count_+1
  *endif
*enddo

! Write out results
*msg,,d_test, nam_count
Components with "%s": %g
/go
```

Resources

Struggling with messed up ECAD Data? We use LinkCAD to clean it up and get it into ANSYS. If you model electronics, download a trial version of LinkCAD for ANSYS: http://www.linkcad-ansys.com/

One set of FLUENT tools that PADT is getting very excited about and seeing a lot of interest in for electronics people, are the ICEPak packages. Visit www.icepak.com to learn more.

Upcoming Training Classes

Month	Start	End	#	Title	Location
Oct '06	2-Oct	4-Oct	101	Introduction to ANSYS, Part 1	Albq. NM
	5-Oct	6-Oct	203	Dynamics	Tempe, AZ
	9-Oct	10-Oct	100	Engineering with FE Analysis	Irvine, CA
	16-Oct	18-Oct	104	ANSYS Workbench, Intro	Albq, NM
	19-Oct	19-Oct	105	ANSYS Workbench, Struc NL	Albq, NM
	25-Oct	27-Oct	902	Multiphysics for MEMS	Tempe, AZ
Nov '06	1-Nov	3-Nov	101	Introduction to ANSYS, Part 1	Tempe, AZ
	8-Nov	9-Nov	107	ANSYS WB DesignModeler	Tempe, AZ
	13-Nov	14-Nov	301	Heat Transfer	Irvine, CA
	16-Nov	17-Nov	102	Introduction to ANSYS, Part 2	Tempe, AZ
	27-Nov	28-Nov	604	Introduction to CFX	Tempe, AZ
Dec '06	6-Dec	8-Dec	101	Introduction to ANSYS, Part 1	Irvine, CA
	11-Dec	13-Dec	104	ANSYS Worbench, Intro	Tempe, AZ
	14-Dec	14-Dec	105	ANSYS Workbench, Struc NL	Tempe AZ
	18-Dec	18-Dec	106	ANSYS WB DesignXplorer	Tempe, AZ

Links

Useful Spots on the ANSYS Web site:

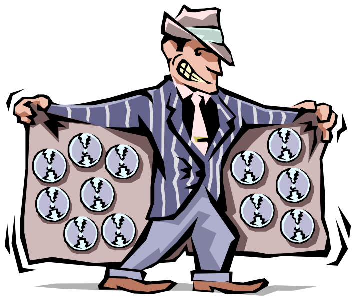
Learn more about new ANSYS, Inc. Products by visiting the demo room: http://www-harwell.ansys.com/demoroom/

Check for Service Packs on the Customer portal: http://www1.ansys.com/customer

News

- ANSYS used to simulate first European Powered Flight link
- ANSYS Supports new Microsoft Compute Cluster Server 2003 Operating System link
- ANSYS, Inc. Named to Sustained Success Honor Role link

The Focus is a periodic publication of Phoenix Analysis & Design Technologies (PADT). Its goal is to educate and entertain the worldwide ANSYS user community. More information on this publication can be found at: http://www.padtinc.com/epubs/focus/about



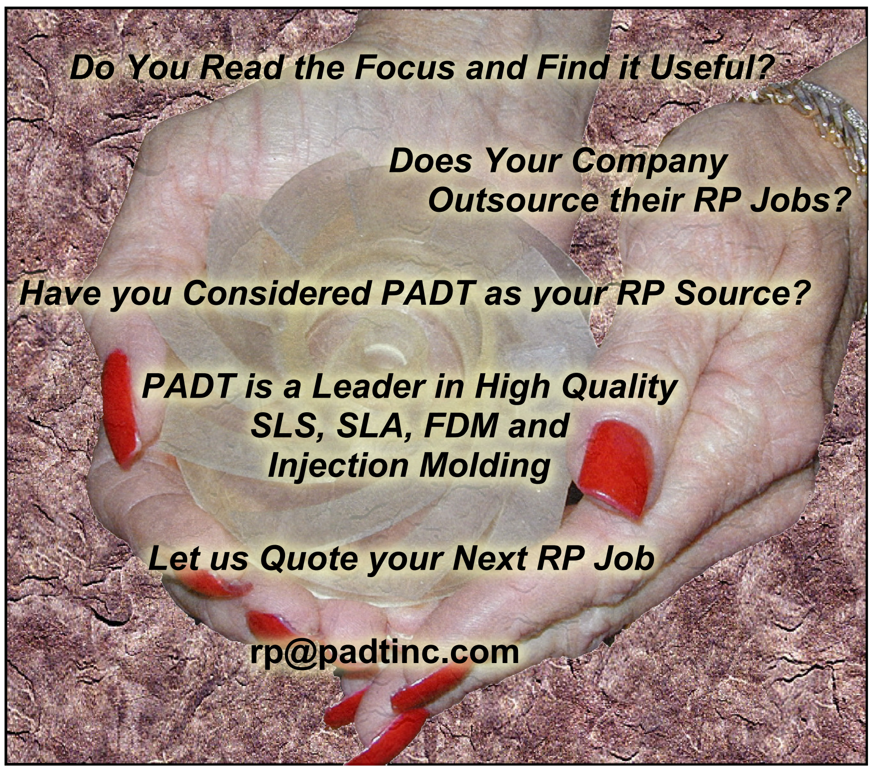
The Shameless Advertising Page

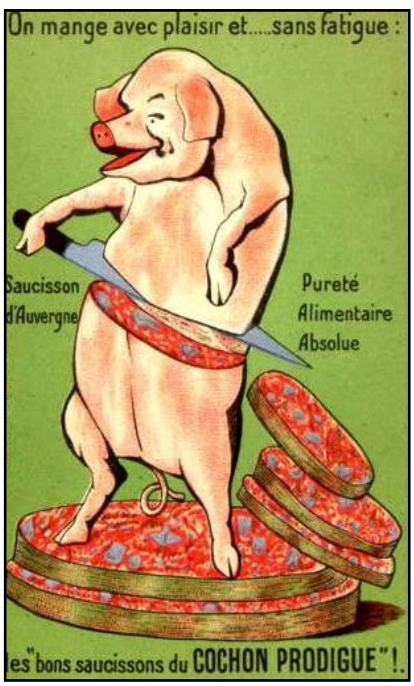
Outsourcing your Analysis to PADT Puts Food in this Child's Mouth

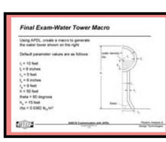
Call 1-800-293-PADT or e-mail info@padtinc.com Today

PADT is on a Pace to Sell over 20 Dimension 3D Printers in Arizona for 2006

Have you had your Demo yet?







Learn APDL the "Home Schooling" Way
Purchase PADT's ***Guide to ANSYS Customization with APDL***