The Focus

PADT

May 31, 2006 | A Publication for ANSYS Users | Issue 48

# Understanding DM: Freeze and Unfreeze

*By Eric Miller*


This article is the second in a series of articles looking into some of the unique capabilities in the ANSYS, Inc. tool DesignModeler, which the lazy amongst us refer to as DM. This tool has been written from the ground up with a powerful set of features that address the unique needs of analysts and what they need to do to their geometry. In this article we will discuss another important concept in DM – Freeze and Unfreeze of Parts.

Modern feature-based CAD tools, and you can consider DM a derivative of this class of tool, work on the concept that you define some sort of "operation" or feature and apply it to the current model as a whole. This makes sense and saves many steps over older methods. Analysts sometimes need to specify a feature but do not want it to affect the whole model. The traditional way of handling this is to make the user specify what objects to operate on for each operation, which can be a pain.

(Continued on Page 2)

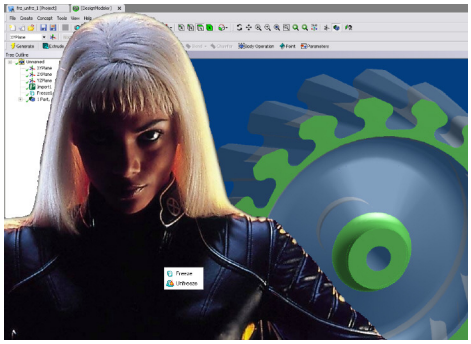


**Figure 1: Storm Understands Freeze and Unfreeze (Hey, any excuse to look at Ms. Berry)**

# Doing Prestressed PSD Studies in Workbench

*By Doug Oatis*

As a continuation to last month's article on command snippets in Workbench, we'll go over how to perform not only a PSD analysis, but a pre-stressed PSD analysis. It should be noted, however, that this article may become a piece of antiquity once Workbench v11 is released. At the 2006 User's Conference, Simulation supported PSD (along with many other features which will be discussed at another time).

Performing a PSD analysis in Workbench is easy if you understand what Workbench does when you insert a 'Frequency Tool' into the Solution branch. As soon as you do this, Workbench sets the ANSYS input file to perform a modal analysis, which is a precursor for the PSD. What we need to do is "hijack" ANSYS and have it perform multiple solves, and then point it back to the result file to pull back in the modal results.

This "hijacking" procedure is discussed in an excellent presentation by Krishna Raichur at the following link:

www.midwest-ansys-ug.com/102005_presentations/Advanced_WB_Techniques.zip

Krishna's process needs to be slightly modified due to differences in v9 and v10. You must now include an `OUTRES,ALL,ALL` in order to write out all modal results for the PSD solution portion. You can also add your own APDL to pull information such as the MCOEF's (mode coefficient value) and pull these back into Workbench using a 'Solution' Command Snippet.

This process gets even more complicated if you are already doing a dual-solve model, such as a thermal/structural or pre-stressed analysis. For each of these analyses, if you watch the solution information, you will see that two complete 'solves' are completed. For a thermal/structural, it first solves the thermal model, then switches the element types over, maps the temperatures, and then solves the structural side. For a pre-stressed analysis, the first load step solves for the displacement of the pretension elements, then locks that displacement in place.

To do the prestressed PSD, we must understand what happens to a command snippet that's inserted into the 'Environment' branch. In a dual-solve analysis, each solution run reads the command snippet. So if we were to do a simple pretension analysis, with a command snippet that contained "a=a+1", we would see that when Workbench is all done, a=2.

In order to do a pre-stressed PSD, we need Workbench to first solve for the pretension displacement, then we need to hijack and have our way with it once the analysis type is set to modal. This is done with a simple *get and *if statement: (Continued on Page 3)

Figure 1: Your intrepid author near the top of the Tallest Mountain in Arizona - Mt. Humpreys. This article has no cool pictures in ANSYS so we decided to show you what Doug did over the Memorial Day weekend.

## Contents

("Freeze/Unfreeze" cont.)

The developers and designers of DM came up with a better concept: the idea that you can flag a body or bodies as being "frozen" and that it should therefore not participate in any operations that are applied to the model. They then added an Unfreeze command to merge geometry together. This is a powerful concept, but like many powerful ideas, takes a little getting used to.

The obvious next step is to look at a simple example, but before we do that, you need to make sure you are familiar with the concept of bodies and parts in DM. If you are not, please take some time to read last month's article on the topic. OK, got your head around that? Let's do an example.

Imagine a block with a hole in the center (Figure 1). Now, if you create a boss as an Extrusion with Operation set to "Add Material" you will get the single body and part shown in Figure 2. For CAD, this is great, but for analysis simple is never good enough. You may want to: post-process only the boss, have uniform orthogonal elements, apply some load only to the boss, or make the boss a different material. Basically, you want the boss to be a different body, or volume in ANSYS speak, for meshing.

Now you could just slice the boss off the top of the block with a slice operation, but a simpler an easier way to achieve what you want is to switch the Operation to "Add Frozen" Figure 3 shows the results of doing so.

Notice two things. First, if you look at the close up of the model tree you will see that there are now two parts and two bodies and one of the bodies is a light blue (frozen) and the other is dark blue (active). Second, look at the solid model itself. Notice how the boss is transparent. By default DM shows frozen bodies as transparent.

So, what happens if you then apply an Unfreeze to the model? DM basically does a Boolean add and creates a single part, resulting in the same volume as when we used the "Add Material" Operation. (Of course, this is a silly example but it shows a point.). Something else to note is that when you apply an Unfreeze, you can tell DM to unfreeze a selected part and freeze all the active parts – kind of a "swap frozen" option.

Users will also find that many operations in Workbench only make sense when done to frozen parts. A good example of this is Slice. If you sliced an active part, the modeler would do the slice and then reconnect the resulting bodies. By freezing the part you avoid this. On the opposite side is the fact that you can not add a fillet blend or chamfer on frozen parts, because the resulting geometry would not attach to anything.

As with many of these new concepts, you have to use it to understand it. If you want to try it out, download a tutorial PADT created for cyclic-symmetry model preparation and get some exposure to using this powerful function:
ftp.padtinc.com/public/downloads/DMCyclic.zip

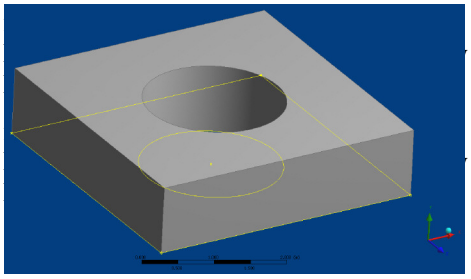


Figure 1: Basic Block with a Hole

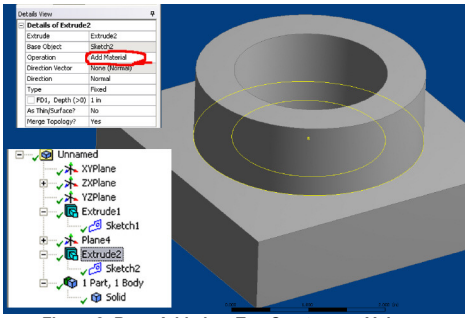



Figure 2: Boss Added on Top Creates one Volume

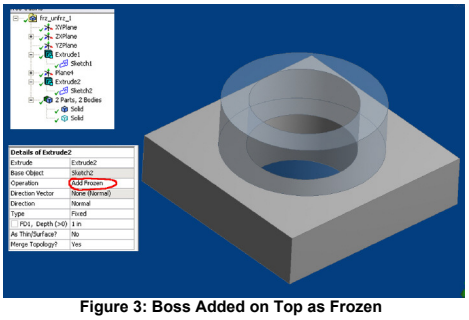



Figure 3: Boss Added on Top as Frozen

# Rare Sighting of "The Focus" Tribe

After almost 50 issues, a brave photographer famous for her collection of Yeti and Bigfoot images, finally captured members from the tribe that publishes The Focus engaged in a ritual social gathering in their communal tree. The large, elder member in the center is granting a blessing to the contributors still on the ground. Just after this photo was taken, the group engaged in ritual grooming and feces slinging, which we are unable to show due to our high decency standards

From left to right: Ted Harris (Manager of Technical Support and Training as well as a frequent contributor), David Mastel (IT miracle worker and the guy that gets each issue out the door and on the web), Eric Miller (Person most in need of a diet and a haircut), Rod Scholl (Expert Analyst, contributor and editor) and Doug Oatis (another Expert Analyst and a Frequent Contributor).

***Resources***

There are two great places to get material properties for ANSYS.
MATWEB is a huge online resource for almost every material, including properties for plastics and rare materials. www.matweb.com
The other resource is a growing database from one of the British Channel Partners, IDAC.
www.idac.co.uk/services/materialdatabase.htm

("Workbench PSD" cont.)

```
!Find out what analysis is being done
*get,d_antype,active,0,anty!
!If it is modal (antype=2), then...
*if,d_antype,EQ,2,then
```

Using Krishna's method, you can have Workbench import static images from ANSYS for various plots. This creates a problem if you don't know where your maximum stresses are going to be, not to mention that you have to figure out the `/VIEW`, `/ANG`, and other orientation commands to get the picture you want. There are two ways to get around this. The first is to launch ANSYS and post process as usual, then take the orientation commands from the log file and paste them into the command snippet. The other, involves a little more trickery.

If you set the option in the 'Solution' of "Save ANSYS Files?" to "Yes", you can create a copy of the result file (or use the `/SYS` command to copy out the .rst file). Next, duplicate the environment in which the PSD analysis was performed. Now, if you have set the Tools > Options>Solution setting to 'Use Project Directory', you will see a new folder created for the new Environment. Simply paste the result file into this directory, and for good measure rename it. Change the environment command snippet to only contain *abbr,solve,stat. Next, change the 'Solution' command snippet to be:

```
!use the renamed result file for
! all post processing
FILE,file-d,rst
!Tell Workbench to read result from
! the 3rd set in the result file
SET,3,1
```

You can now request plots, as you would any Workbench analysis, (σx, σy, etc.) and Workbench will plot these values, allowing you to pan/zoom/rotate the model. One warning, however, is that the von Mises Equivalent Stresses will not match up to results in ANSYS Classic. This is because Workbench does not perform the Segalman Reese method in calculating the von Mises stresses. Instead, it uses the component stresses and the standard von Mises equation. All of your directional stresses and displacements (contained in set,3,1) will match up with ANSYS Classic.

Hopefully this shed more light on how command snippets are treated in the 'Environment' and 'Solution' branch. Not only can you perform analyses not directly supported by Workbench, but you can post-process them as well.

You can download these command snippets from our ftp server at:

ftp.padtinc.com/public/downloads/Prestressed PSD in Workbench.zip

Figure 2: The view from near the top. Unfortunately the 90 MPH winds kept Doug and his climbing partner from making it to the top.

# *MWRITE: Writing Tables to Text Files

*By Carlos Shultz*

ANSYS tables are useful but sometimes confusing. For the most part, writing tables using `*MWRITE` is an exercise in brute force cryptography. In other words, you try every possible combination until you successfully input/output your data correctly.

This article will cover some common uses of tables and hopefully reduce the amount of time spent looking at the arguments of table indices while scratching your head.

First, we need to create a table to test with. The following APDL creates the table shown in Figure 1. The values placed into the table have a "hundreds digit" equal to the row, a "tens digit" equal to the column, and a "one digit" equal to the plane. This convention will allow us to quickly verify that our scripts are successful before tackling real data.

```
*dim,table1,table,5,3,2
*do,i,0,5
  *do,j,0,3
    *do,k,1,2
      table1(i,j,k)=k+10*j+100*i
    *enddo
  *enddo
*enddo
```

**Write out tables using *MWRITE**

The tables were written out to test files using a variety of formats to demonstrate

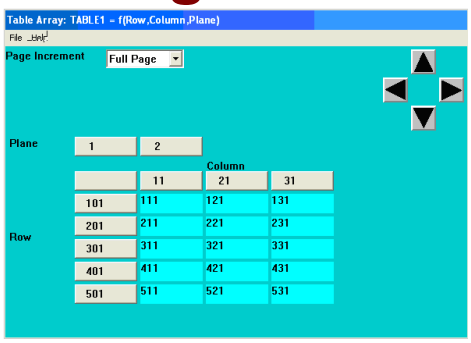



Figure 1: TABLE1 displayed in ANSYS

the usage of the various arguments. The documentation of `*MWRITE` is somewhat confusing but should be less so after checking a few permutations of the indices.

**Fixed Columns**

The table is written out using the default arguments and 4 columns, see Figure 2. The number of columns is defined by how many formats are included in the line after `*MWRITE`.

```
*mwrite,table1(0,0),table1a,txt,,JIK
%10.4F %10.4F %10.4F %10.4F
```

To change the format to 5 columns, add a `%10.4F` to the format line, see Figure 3. You'll see that because the number of columns doesn't match the data, ANSYS rolls over

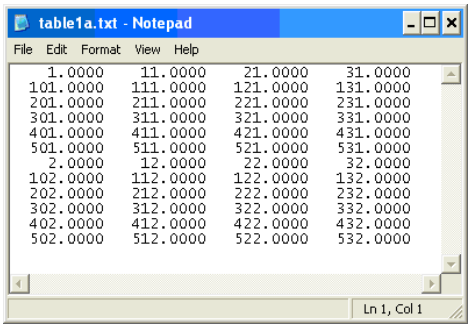



Figure 2: TABLE1 with 4 Fixed Columns

to the next line in the table to grab data. In the final row, there isn't another row to go to so ANSYS puts a zero there.

```
*mwrite,table1(0,0),table1aa,txt,,JIK,,,
%10.4F %10.4F %10.4F %10.4F %10.4F
```

**General Format, Comma Delimited**

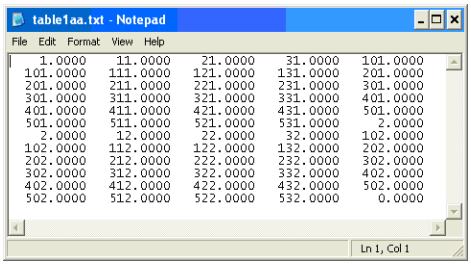



Figure 3: TABLE1 with 5 Fixed Columns

(Continued on Page 4)

("*MWRITE" cont.)

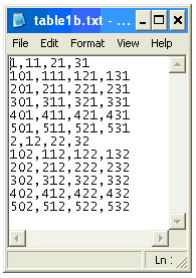



Figure 4: TABLE1 written as comma delimited

A variety of output options are available, in C and FORTRAN syntax, including a general comma delimited format.

```
*mwrite,table1(0,0),table1b,txt,,JIK,,,
%G,%G,%G,%G
```

To transpose a table, switch the indices from JIK to IJK. Don't forget to correct the formats to get the correct number of columns. Figure 5 shows the resulting text file. The digits of the values in the cells are now in order of column, row, plane.

```
*mwrite,table1(0,0),table1e,txt,,IJK,,,
%10.4F %10.4F %10.4F %10.4F %10.4F %10.4F
```

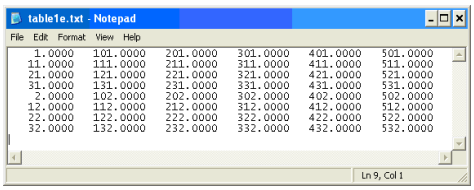



Figure 5: TABLE1 transposed with fixed columns

**Plane 1 Only**

To limit the plotting to plane 1, put a 1 in the last argument, see Figure 6.

```
*mwrite,table1(0,0),table1e,txt,,JIK,,,1
%10.4F %10.4F %10.4F %10.4F
```

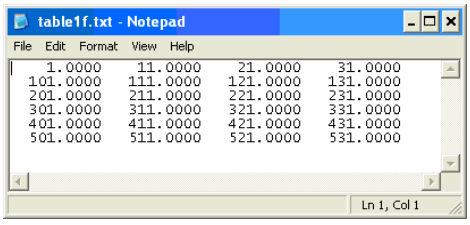



Figure 6: TABLE1 Plane 1 only, Written with Fixed Columns

**Subset Only**

To limit the writing to just a subset of the table, change the starting position and the rows and columns to retrieve. This example, see Figure 7, starts in the 2nd row and 2nd column and then retrieves 2 columns and 3 rows for 1 plane.

```
*mwrite,table1(2,2),table1g,txt,,JIK,2,3,1
%10.4F %10.4F
```

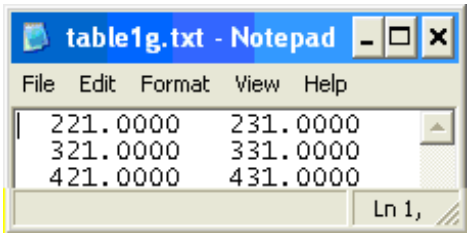



Figure 7: TABLE1, subset of Plane 1, written with fixed columns

**Other Permutations**

There are also several other, less useful permutations available. See Figures 8 through 11 for KIJ, KJI, IKJ, and JKI.

```
*mwrite,table1(0,0),table1h,txt,,KIJ,,,
%10.4F %10.4F %
```

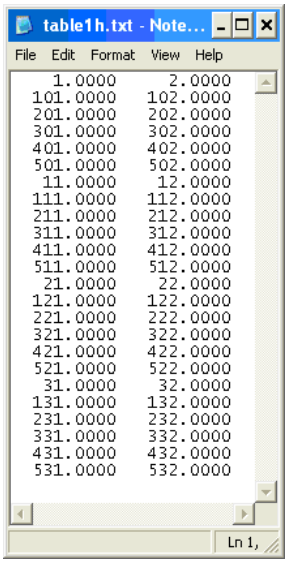



Figure 8: TABLE1 KIJ Format Used

```
*mwrite,table1(0,0),table1i,txt,,KJI,,,
%10.4F %10.4F
```

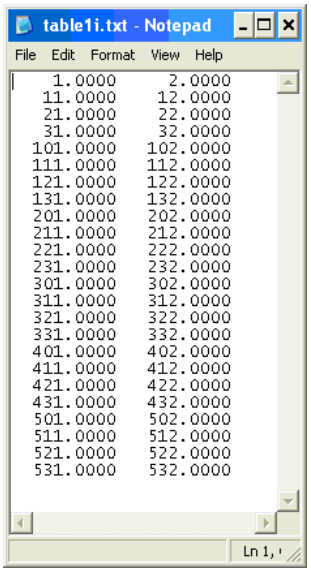



Figure 9: TABLE1 KJI Format Used

```
*mwrite,table1(0,0),table1j,txt,,IKJ,,,
%10.4F %10.4F %10.4F %10.4F %10.4F
```

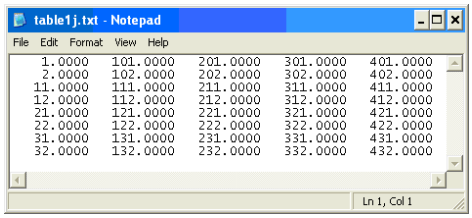



Figure 10: TABLE1 IKJ Format Used

```
*mwrite,table1(0,0),table1k,txt,,JKI,,,
%10.4F %10.4F %10.4F
```

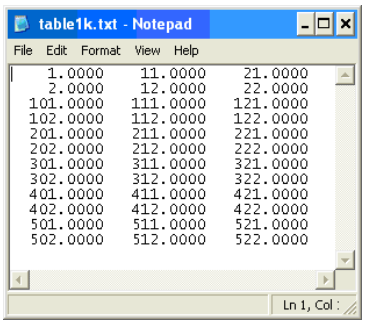



Figure 11: TABLE1 JKI Format Used

**Write out a Table Using *VWRITE**

You can also write tables using the `*VWRITE` command, see Figure 12.

```
*cfopen,test1c,txt
*VWRITE,table1(0,0),table1(0,1),table1(0,
2),table1(0,3)
%10.4F %10.4F %10.4F %10.4F
*cfclose
```

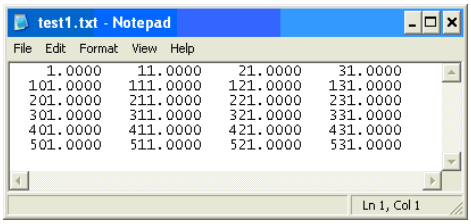



Figure 12: TABLE1 JKI Format Used

**The Command is Ignored**

All of the examples shown above will not work directly from the GUI. You must create an APDL macro to use these commands because of the formatting lines.

Below are the commands that you would use to create a command macro that executes some non-GUI commands and then deletes itself.

```
*cfopen,test_,mac
!!APDL commands to execute here!!!
*cfclose
test_
/del,test_,mac
```

**Bonus: *TREAD an Array?**

This procedure is useful when importing boundary condition data developed external to ANSYS.

If you need non-fixed format data put into an array, there isn't a single command to read it in since `*TREAD` works on tables only and `*VREAD` requires consistent formats. So use `*TREAD` to read the data into a table and then write it to an array using `*MFUN` like this:

```
*dim,table1,table,5,3,2
*tread,table1,table1,txt
*get,num_row,parm,table1,dim,x
*get,num_col,parm,table1,dim,y
*get,num_plane,parm,table1,dim,z
*dim,array1,array,num_row+1,num_col+1,num
_plane
*MFUN,array1(1,1),COPY,table1(0,0)
```

## Guest Article:

# Getting DOS to Speak Unix

*By: Dave Foster*
*fosterd@result-tech.com*

I ran into a problem the other day-maybe you have it too. I was leaning into the monitor puzzling over a busted batch file when Larry stopped by my desk. "We're ordering Chinese food. Want anything?" He dropped a menu crowded with hundreds of tiny listings. I squinted, turning over the menu twice.

"What are you looking for?" asked Larry.

"Almond chicken, but it doesn't look like they have it." I muttered.

"#73, they call it Commodore Chow's Chicken. They made up cute names for all of their plates. It's their gimmick." he offered helpfully. "Great," I thought, "now I gotta remember another stupid name for something that's practically a standard. Where do these guys get off?" Anyway, I gave Larry a couple of bucks and he shuffled along.

Going back to my problem about the DOS batch file, I tried it again and this time actually bothered to read the error message.

```
'cp' is not recognized as an
internal or external command,
operable program or batch file.
```

Yep, it was that same old DOS versus Unix command line hassle that pops up every time I put my scripts on a new computer. They're similar enough to almost work but not quite. Luckily I had the solution for this one. A Google search and download later my batch file was purring like a kitten.

Lots of people have run into this problem over the years and there are several decent options. For instance, you can write a bunch of batch file aliases for your favorite Unix commands, but that does not handle command line switches.

You can also try the heavy duty solution, installing the Cygwin Unix emulator. It has all the commands you could want and more but hogs RAM, runs slowly, and has been reported to fight with anti-virus software. It's like swatting a fly with a sledgehammer.

My favorite solution takes the middle ground. A few years ago a band of mysterious free software pixies created Windows-compatible versions of the command line trinkets that make Unix so handy: ls, mv, cp, rm, awk, sed, grep, and over a hundred others. There's no installation fuss, just unzip the tiny executables and go. They're fast and run directly under all versions of Windows I've tried. Heck, they don't even care whether you use forward or backslashes with file paths. Imagine: the same batch file you use on Unix runs unchanged on Windows.

You get them for free at unxutils.sourceforge.net. Unzip the programs someplace in your PATH (such as your C:\Windows\System32 directory). That's it. Just so that you know, these things aren't perfect. Because Windows handles things like file security and linked files differently, some of the commands can't behave exactly the same. A few of them are just plain dummies that actually do nothing. Even with these limits, I call them a big win.

I'm still eating Commodore Chow's Chicken, but at least my batch file blues are long gone.

For a Full list of command-line goodies visit the web site.

## Upcoming Training Classes

| Month | Start | End | # | Title | Location |
|---|---|---|---|---|---|
| Jun '06 | 12-Jun | 14-Jun | 101 | Introduction to ANSYS, Part 1 | Tempe, AZ |
| | 19-Jun | 20-Jun | 201 | Basic Structural Nonlinearities | Albq, NM |
| | 21-Jun | 22-Jun | 204 | Advanced Contact & Bolts | Albq, NM |
| | 26-Jun | 27-Jun | 301 | Heat Transfer | Irvine, CA |
| | 28-Jun | 29-Jun | 107 | ANSYS WB DesignModeler | Irvine, CA |
| Jul '06 | 10-Jul | 12-Jul | 104 | ANSYS WB Simulation - Intro | Albq, NM |
| | 13-Jul | 13-Jul | 105 | ANSYS WB Sim - Struct NL | Albq, NM |
| | 20-Jul | 21-Jul | 801 | ANSYS Customization APDL | Tempe,AZ |
| | 24-Jul | 25-Jul | 203 | Dynamics | Albq, NM |
| | 26-Jul | 28-Jul | 101 | Introduction to ANSYS, Part 1 | Irvine, CA |
| Aug '06 | 01-Aug | 03-Aug | 104 | ANSYS WB Simulation - Intro | LV, NV |
| | 04-Aug | 04-Aug | 105 | ANSYS WB Sim Struct NL | LV, NV |
| | 07-Aug | 09-Aug | 152 | AI*Environment | Tempe, AZ |
| | 10-Aug | 11-Aug | 202 | Advanced Structural NL | Tempe, AZ |
| | 14-Aug | 15-Aug | 502 | ANSYS/LS-DYNA | Tempe, AZ |
| | 17-Aug | 18-Aug | 604 | Introduction to CFX | Tempe, AZ |
| | 23-Aug | 25-Aug | 502 | High Frequency EMAG | Tempe, AZ |

### Links

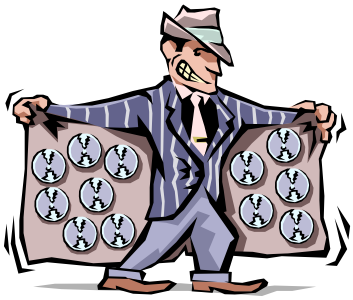
Customers who are up-to-date on their TECS can download all the latest ANSYS manuals in PDF format from the customer portal:
www1.ansys.com/customer under
"Product Information" -> "Product Documentation"

This includes the very important programming manual set which is not available with the install:
Interfacing, UIDL and UPF

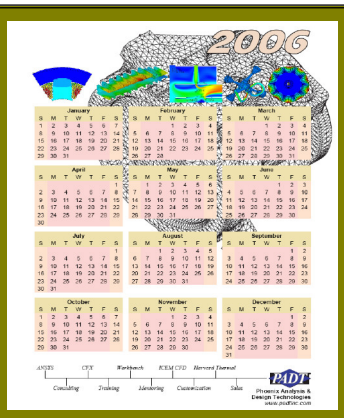
### News

- ANSYS, Inc. Improves Relationship with MATWEB link
- ANSYS, Inc. Officially Completes Acquisition of FLUENT link
- PADT Adds ANSYS Training Classes in Las Vegas, Nevada 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




Download the 2006 PADT ANSYS Calendar


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

## Affordable, effective Dimension 3D Printers

Nothing communicates design ideas faster than a three-dimensional part or model. With a Dimension 3D Printer, you can bring CAD files and design ideas to life – right from your desktop. Test form, fit and function – and as many design iterations as you like – with functional ABS parts. For added hands-free convenience, consider the Dimension SST with a water soluble support removal system. Starting at $24,900,* Dimension 3D printers represent a major price breakthrough – with leading 3D printing technology.

Simply connect a Dimension 3D printer to your network. Load Catalyst software. Insert the self-loading material cartridges and follow the display panel prompts. That's it. 3D printing with Dimension or Dimension SST is as easy as clicking print. See why a Dimension 3D Printer is the fastest selling office-friendly machine of its kind – and start shortening your design and development cycle.

dimension™