

# Introductory Tutorial for APDL-Mode

A GNU Emacs programming mode for the APDL language,  
version 20.6.0

H. Dieter Wilhelm<sup>1</sup>

Time-stamp: <2021-08-31 Di>



---

<sup>1</sup>dieter@duenenhof-wilhelm.de

# Introducing APDL-Mode – an APDL Environment

This project supports your APDL workflows with the FEA suite Ansys.

It provides an editor mode for GNU Emacs for investigating and coding APDL. The mode offers also managing and communication capabilities for various Ansys processes, like interactive code debugging with the solver or inquiring the license manager status, etc. Some features are quite sophisticated but its documentation is accessible for Ansys users with little APDL and Emacs experience.

GNU-Emacs is a modern, powerful and extensible - yet free - editor. High quality software available for every operating system where Ansys is running.

# Download GNU-Emacs and APDL-Mode

There are no costs and restrictions even in commercial application. Please download **GNU Emacs** and the latest APDL-Mode from **Melpa** or an archive from the **APDL-Mode's releases page**.

dieter-wilhelm / apdl-mode

Watch 8 Star 21 Fork 7

Code Issues 0 Pull requests 0 Actions Projects 0 Wiki Security Insights

Releases Tags

Latest release

20.1.0  
5e29d7

Compare

APDL-Mode 20.1.0, first release working with GNU-Emacs' packaging system

dieter-wilhelm released this 8 days ago · 20 commits to master since this release

Download the tar file and install the package with `M-x package-install-file`

Assets 3

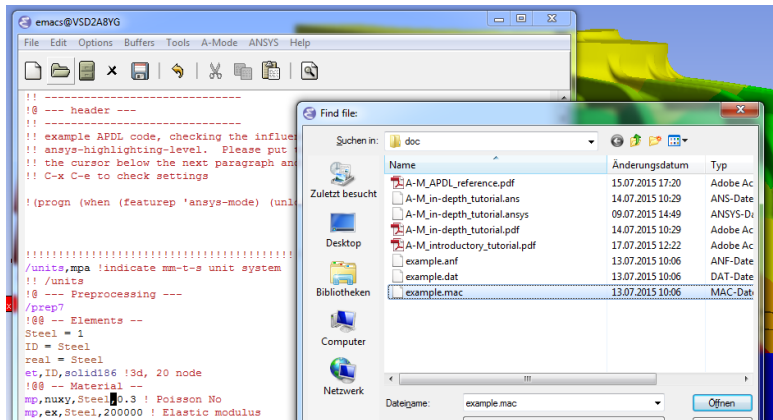
apdl-mode-20.1.0.tar 880 KB

Install APDL-Mode from Emacs: Please type: `M-x package-install-file` <RET> and select the downloaded tar archive.

## Open an Emacs Buffer under APDL-Mode

Start the editor and open *doc/example.mac* or any other APDL file (with the extensions *mac*, *ans*, *inp*, *dat* or *anf*, otherwise type M-x apdl and <RET>).

In the following **M-x** or **C-c** means typing <ALT> or <CTRL> key together with the 'x' or 'c' key.



# Explore the APDL-Mode Menus

Browsing the APDL help with **C-c C-b** is much faster than searching with the Ansys Help Viewer or online.

The screenshot displays the ANSYS APDL-Mode environment. The main window shows the command editor with the following text:

```
MP, Lab, MAT, C0, C1, C2, C3, C4
Defines a linear material property as a c
```

Below the command editor, a list of material properties is shown:

- Lab
- Valid material property label. Applicable to the [Material Reference](#) for more complete
- ALPD — Mass matrix multiplier for
- ALPX — Secant coefficients of ther
- BETD — Stiffness matrix multiplier
- Note: If used in an ex corresponds to 10%
- BETX — Coefficient of diffusion exp
- BVIS — Bulk viscosity
- C — Specific heat
- CREF — Reference concentration (n
- CSAT — Saturated concentration
- CTEX — Instantaneous coefficients
- CVH — Heat coefficient at constant
- DENS — Mass density.
- DMPR — Constant structural damping
- DXX — Diffusivity coefficients (also
- EMIS — Emissivity.
- ENTH — Enthalpy.
- EX — Elastic moduli (also EY, EZ)
- GYX — Shear moduli (also GYZ, GXZ)

The 'ANSYS Help' menu is open, showing the following options:

- Comment/Un- Region M-;
- Complete Symbol C-M-i
- Copy region or paragraph to clipboard C-c C-c
- Copy above Code C-c C-u
- Close Logical Block C-c ]
- Insert Parentheses M-(
- Preview Macro Template C-c C-s
- Align region or paragraph C-c C-a
- Show ANSYS Command Help M-?
- Display Variable Definitions C-c C-v
- Change Display a short help for the ANSYS command near the cursor with its parameters. M-x ansys-show-command-p
- Change the help text for a command in the help directory
- Open APDL help in Browser C-c C-b
- Start ANSYS help system C-c C-h
- Insert Template
- Navigate Code Lines
- Work with Logical Blocks
- Outline Minor Mode
- Show Paren Mode
- Delete Selection Mode
- ANSYS-Mode Online Documentation
- Help on ANSYS-Mode C-h m
- Customise ANSYS-Mode
- List Mode Abbreviations
- ANSYS Mode Bug Report
- Reload ANSYS-Mode
- Exit ANSYS-Mode Version: 161.1

On the right side of the screen, there is a link: [PREP7: Materials Product Restrictions](#). Below it, a note says: [Reference](#). See [Linear Material Properties](#) in

domain. For example, 0.1 roughly

# Easily Inspect WorkBench Solver Files (*doc/example.dat*)

APDL-Mode hides the normally uninteresting but usually very large number blocks. On the right hand side is the unhidden content. If APDL-Mode is not yet configured for *.dat* files please type:

M-x apdl-mode

```
emacs@VSDTQB0D
File Edit Options Buffers Tools ANSYS Help

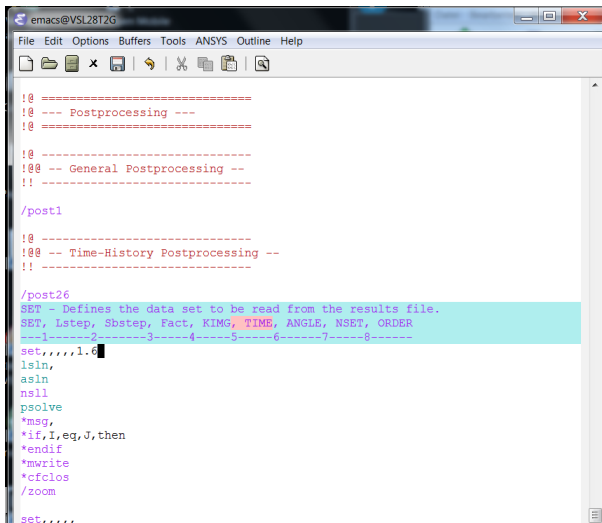
SHPP,OFF,,NOWARN
/nolist
etcon,set          ! allow ANSYS to choose best KEYOP's for 180x element
/com,***** Nodes for the whole assembly *****
nblock,3
(1i9,3e20.9e3)
  1      -7.500000000E-001    5.393061738E-001    1.363562179E+000
  ![ ... hidden region ... ]
    88596    1.994659086E+001    1.291219623E+001    2.154539857E-014
-1
/wb,elem,start      ! set before creation of elements
/com,***** Elements for Body 1 "40600024_CAD_102" *****
et,1,187
eblock,19,solid,,5350
(19i9)
  1      1      1      1      0      0      0      0
50  0      4929    23672    29579    26683    27434    43256    59546
52  43253
  ![ ... hidden region ... ]
    62622    63635
-1
!Material Id = (AA70AC48-F275-4CA7-BFD1-9FEE7F3E223D)
```

```
emacs@VSDTQB0D
File Edit Options Buffers Tools ANSYS Help

SHPP,OFF,,NOWARN
/nolist
etcon,set          ! allow ANSYS to choose best KEYOP's for 180x element
/com,***** Nodes for the whole assembly *****
nblock,3
(1i9,3e20.9e3)
  1      -7.500000000E-001    5.393061738E-001    1.363562179E+000    1.363562179E+000
  2      -5.000000000E-001    5.393061738E-001    1.363562179E+000    1.363562179E+000
  3      -2.500000000E-001    5.393061738E-001    1.363562179E+000    1.363562179E+000
  4      0.000000000E+000    5.393061738E-001    1.363562179E+000    1.363562179E+000
  5      2.500000000E-001    5.393061738E-001    1.363562179E+000    1.363562179E+000
  6      5.000000000E-001    5.393061738E-001    1.363562179E+000    1.363562179E+000
  7      7.500000000E-001    5.393061738E-001    1.363562179E+000    1.363562179E+000
  8      1.000000000E+000    5.393061738E-001    1.363562179E+000    1.363562179E+000
  9      1.250000000E+000    5.393061738E-001    1.363562179E+000    1.363562179E+000
 10     1.500000000E+000    5.393061738E-001    1.363562179E+000    1.363562179E+000
 11     1.750000000E+000    5.393061738E-001    1.363562179E+000    1.363562179E+000
 12     2.000000000E+000    5.393061738E-001    1.363562179E+000    1.363562179E+000
 13     2.250000000E+000    5.393061738E-001    1.363562179E+000    1.363562179E+000
 14     2.500000000E+000    5.393061738E-001    1.363562179E+000    1.363562179E+000
 15     2.750000000E+000    5.393061738E-001    1.363562179E+000    1.363562179E+000
 16     3.000000000E+000    5.393061738E-001    1.363562179E+000    1.363562179E+000
```

# Check Your Cursor Position in the Argument List

For commands with a large number of arguments it is cumbersome to count the arguments, **M-h** facilitates this for you and visualises dynamically at which argument position the cursor is.



```
emacs@VSL28T2G
File Edit Options Buffers Tools ANSYS Outline Help

!@ =====
!@ --- Postprocessing ---
!@ =====

!@ -----
!@@ -- General Postprocessing --
!! -----

/post1

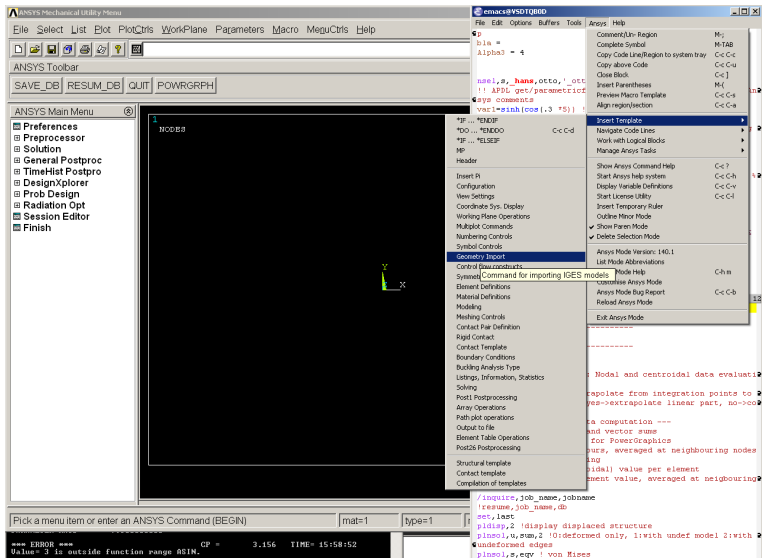
!@ -----
!@@ -- Time-History Postprocessing --
!! -----

/post26
SET - Defines the data set to be read from the results file.
SET, Lstep, Sbstep, Fact, KIMG, TIME, ANGLE, NSET, ORDER
--1-----2-----3-----4-----5-----6-----7-----8-----
set, , , , 1.6
lsln,
asln
nsl1
psolve
*msg,
*if, I, eq, J, then
*endif
*mwrite
*cfclos
/zoom

set, , , , ,
```

# Select and Insert Templates from the Menu into Your Code

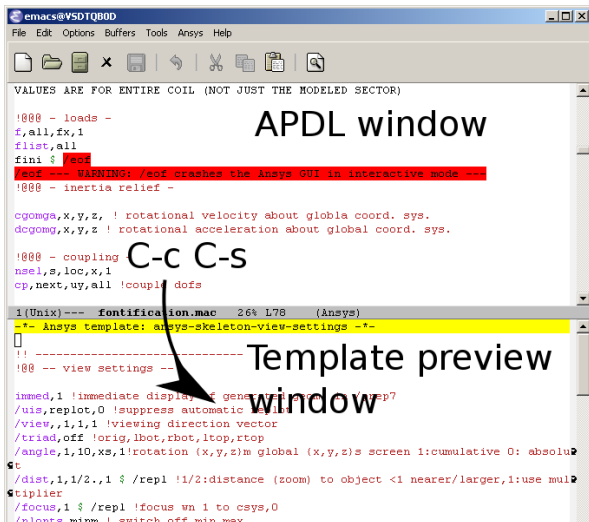
Screenshot with the Ansys Classics GUI on the left and Emacs on the right on Win64





# Preview the Extensible APDL Code Templates

Before inserting an entire template you are able to inspect its content in a preview window (C-c C-s) and might just copy the most relevant snippets, please see below and next slide.



The screenshot shows the Emacs editor window titled 'emacs@VSDTQB0D'. The menu bar includes 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Ansys', and 'Help'. The toolbar contains icons for file operations and editing. The main text area displays APDL code with the following content:

```
VALUES ARE FOR ENTIRE COIL (NOT JUST THE MODELED SECTOR)

!@@@ - loads -
f,all,fx,1
flist,all
fini $ /eof
/eof --- WARNING: /eof crashes the Ansys GUI in interactive mode ---
!@@@ - inertia relief -

cgonga,x,y,z, ! rotational velocity about globla coord. sys.
dcgomg,x,y,z ! rotational acceleration about global coord. sys.

!@@@ - coupling -
nse1,s,loc,x,1
cp,next,uy,all !couple dofs
```

Overlaid on the code are three labels with arrows pointing to specific parts:

- 'APDL window' points to the main code area.
- 'C-c C-s' points to the 'coupling' section of the code.
- 'Template preview window' points to a preview window at the bottom.

The preview window, titled 'fontification.mac 26% L78 (Ansys)', shows the following content:

```
!@ -- view settings --

immed,1 !immediate display of generated graphics w/ repl?
/uis,replot,0 !suppress automatic replot
/view,,1,1,1 !viewing direction vector
/triad,off !orig,lbot,rbot,ltop,rtop
/angle,1,10,xs,1!rotation (x,y,z)m global (x,y,z)s screen 1:cumulative 0: absolute
st
/dist,1,1/2.,1 $ /repl !1/2:distance (zoom) to object <1 nearer/larger,1:use multiplier
tiplier
/focus,1 $ /repl !focus wn 1 to csys,0
/plots,minmax,1,switch off min max
```

# Select an Interesting Template from a Completion Window

Type **C-c C-s** to choose a template name, use the **<TAB>** key to complete or to open the completion window of available items.

VALUES ARE FOR ENTIRE COIL (NOT JUST THE MODELED SECTOR)

```
!@@@ - loads -  
f,all,fx,1  
flist,all  
fini $ /eof  
/eof --- WARNING: /eof crashes the Ansys GUI in interactive mode ---  
!@@@ - inertia relief -  
  
!gong, x, y, z, ! rotational velocity about global coord. sys.  
dcgong, x, y, z ! rotational acceleration about global coord. sys.  
1((nix)--- fontification.mac 26% L78 (Ansys)  
Click <mouse-2> on a completion to select it.  
In this buffer, type RET to select the completion near point.
```

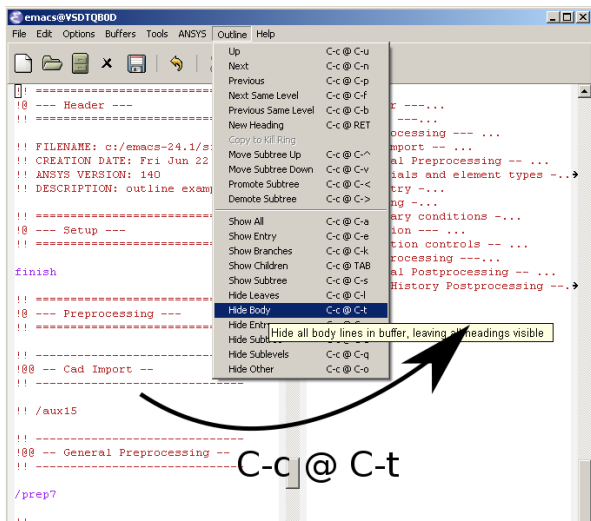
Possible completions are:

ansys-skeleton-array	ansys-skeleton-bc
ansys-skeleton-buckling	ansys-skeleton-compilation
ansys-skeleton-configuration	ansys-skeleton-contact-definition
ansys-skeleton-contact-rigid	ansys-skeleton-contact-template
ansys-skeleton-coordinates	ansys-skeleton-element-definition
ansys-skeleton-element-table	ansys-skeleton-expand
ansys-skeleton-function	ansys-skeleton-geometry
ansys-skeleton-header	ansys-skeleton-import
ansys-skeleton-information	ansys-skeleton-looping
ansys-skeleton-material-definition	ansys-skeleton-meshing
ansys-skeleton-multi-plot	ansys-skeleton-numbering-controls
ansys-skeleton-output-to-file	ansys-skeleton-path-plot
ansys-skeleton-post1	ansys-skeleton-post26
ansys-skeleton-select	ansys-skeleton-solve
ansys-skeleton-structural-template	ansys-skeleton-symbols
ansys-skeleton-view-settings	ansys-skeleton-working-plane

1\%- \*Completions\* All L1 [Completion List]

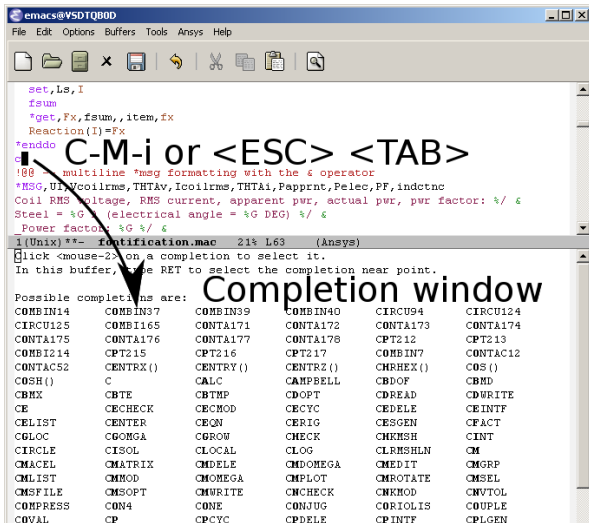
# Check Auto-Insertion and Outline Your Code (Tree View)

Create a new APDL file with the suffix `.mac` and let Emacs auto-insert a skeleton with 'outline' headings. Collapse the content to a tree view with `C-c @ C-t` and open all (`C-c @ C-a`) again.



# Utilise completions of all (~2000) APDL symbols

Move the cursor behind a character - here 'c' - or word fragment and type <ESC> <TAB> or **C-M-i** for completing up to date APDL command-, element- and function names.



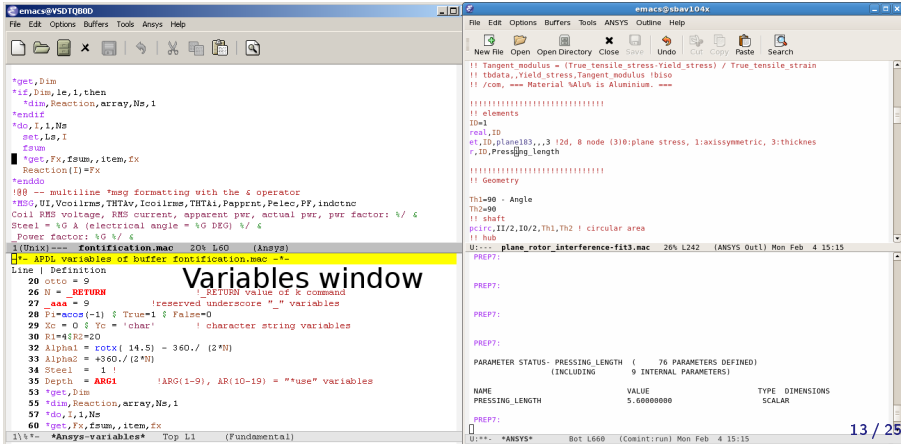
```
emacs@VSDTQB00
File Edit Options Buffers Tools Ansys Help

set,ls,I
fsum
*get,Fx,fsum,,item,fx
Reaction(I)=Fx
*enddo
c
![[ = multiline *msg formatting with the & operator
*MSG,UI,Vcoirms,THTAv,Icoirms,THTAi,Pappint,Pelec,PF,indctnc
Coil RMS voltage, RMS current, apparent pwr, actual pwr, pwr factor: %/ &
Steel = %G (electrical angle = %G DEG) %/ &
_Power factor: %G %/ &
l(Unix)**- fontification.mac 21% L63 (Ansys)
Click <mouse-2> on a completion to select it.
In this buffer, type RET to select the completion near point.

Possible completions are:
COMBIN14 COMBIN37 COMBIN39 COMBIN40 CIRC94 CIRC124
CIRC125 COMBI165 CONTA171 CONTA172 CONTA173 CONTA174
CONTA175 CONTA176 CONTA177 CONTA178 CPT212 CPT213
COMBI214 CPT215 CPT216 CPT217 COMBIN7 CONTA172
CONTA175 CENTRX() CENTRY() CENTRZ() CHRHEX() COS()
COSH() C CBDOF CBMD
CBMX CBTE CBTTP CDOPT CDREAD CDWRITE
CE CEHECK CECMOD CECYC CEDELE CEINTF
CELIST CENTER CEQN CERIG CESGEN CFACT
CGLOC CGOMGA CGROW CHECK CHKMSH CINT
CIRCLE CISOL CLOCAL CLOG CLRMASHLNM CM
CMACEL CMATRIX CMDELE CMDOMEGA CMEDIT CMGRP
CMLIST CMOD CMOMEGA CMPLT CMROTATE CMSEL
CMSFILE CMSOPT CMWRITE CMCHECK CMKMOD CMVTOL
COMPRESS CON4 CONE CONJUG CORIOLIS COUPLE
COVAL CP CPCYC CPDELE CPINTF CPLGEN
```

# Open a Summary Window of Your APDL Variables

Type **C-c C-v** to receive a summary window of all your variable definitions. With an argument (**C-u C-c C-v**) you will get the current value of your variable at the cursor (Linux only, right).



The image shows two Emacs windows. The left window, titled 'emacs@VSDTQ80D', contains APDL code. The right window, titled 'emacs@sbav104x', shows the output of the 'C-c C-v' command, displaying a summary of all defined variables.

**Left Window (emacs@VSDTQ80D):**

```
*get,Dim
*if,Dim,le,1,then
  *dim,Reaction,array,Ns,1
*endif
*do,I,1,Ns
  set,Ls,I
  fsum
  *get,Fx,fsum,,item,fx
  Reaction(I)=Fx
*enddo
!@ -- multiline *msg formatting with the & operator
*MSG,UI,Vcoilrms,THTAv,Icoilrms,THTAi,Papprnt,Pelec,PF,indctnc
Coil RMS voltage, RMS current, apparent pwr, actual pwr, pwr factor: %/ %
Steel = %G A (electrical angle = %G DEG) %/ %
Power factor: %G %/ %
1(Unix)--- fontification.mac 20% L60 (Ansys)
!*- APDL variables of buffer fontification.mac -*-
Line | Definition
20 otto = 9
26 N = RETURN ! RETURN value of k command
27 _aaa = 9 !reserved underscore "_" variables
28 Pi=acos(-1) % True=1 % False=0
29 Xc = 0 % Yc = 'char' ! character string variables
30 R1=4%R2=20
32 Alpha1 = rotx( 14.5) - 360./ (2*N)
33 Alpha2 = +360./ (2*N)
34 Steel = 1 !
35 Depth = ARG1 !ARG(1-9), AR(10-19) = "use" variables
53 *get,Dim
55 *dim,Reaction,array,Ns,1
57 *do,I,1,Ns
60 *get,Fx,fsum,,item,fx
!*- *Ansys-variables* Top L1 (Fundamental)
```

**Right Window (emacs@sbav104x):**

```
!! Tangent_modulus = (True_tensile_stress-Yield_stress) / True_tensile_strain
!! tbddata,,Yield_stress,Tangent_modulus !biso
!! /com, --- Material %Alu% is Aluminium. ---

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!! elements
ID=1
real,ID
et,ID,plane183,,,3 !2d, 8 node (3D):plane stress, 1:axisymmetric, 3:thickness
r,ID,Pressing_length

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!! Geometry

Th1=90 - Angle
Th2=90
!! shaft
pcirc,II/2,IO/2,Th1,Th2 ! circular area
!! hub
U:--- plane_rotor_interference-fit3.mac 26% L242 (ANSYS Out1) Mon Feb 4 15:15
PREP7:

PREP7:

PREP7:

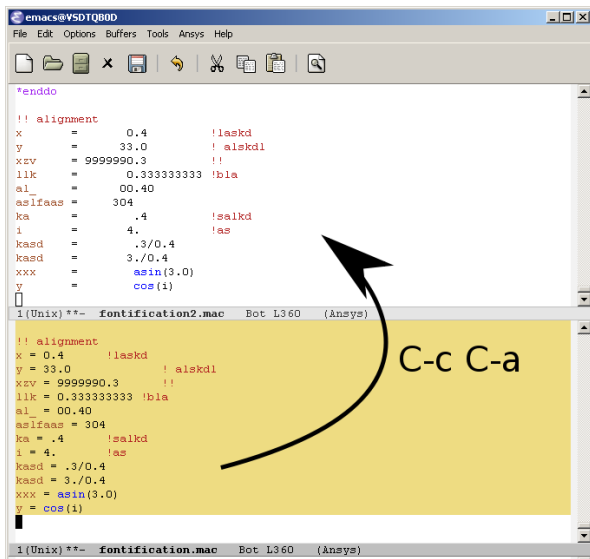
PARAMETER STATUS- PRESSING_LENGTH ( 76 PARAMETERS DEFINED)
(INCLUDING 9 INTERNAL PARAMETERS)

NAME VALUE TYPE DIMENSIONS
PRESSING_LENGTH 5.60808080 SCALAR

PREP7:
U:--- *ANSYS* Bot L660 (Comint:run) Mon Feb 4 15:15
```

# Structure Your Variable Assignments

Move the cursor to a variable definition paragraph or mark, here in yellow, some definitions and type **C-c C-a** to align them.



The screenshot shows the Emacs editor interface with a file named fontification2.mac. The code contains several variable assignments. A yellow highlight covers a block of code, and a large black arrow points from the text "C-c C-a" to the highlighted area.

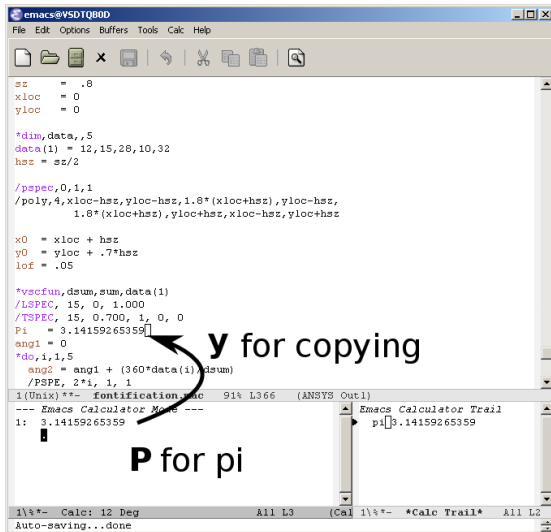
```
*enddo

!! alignment
x      =      0.4      !laskd
y      =      33.0     ! alskdl
xzv    = 9999990.3    !!
llk    =      0.333333333 !bla
al_    =      00.40
aslfaas = 304
ka     =      .4      !salkd
i      =      4.      !as
kasd  =      .3/0.4
kasd  =      3./0.4
xxx   =      asin(3.0)
y     =      cos(i)

!! alignment
x = 0.4      !laskd
y = 33.0     ! alskdl
xzv = 9999990.3    !!
llk = 0.333333333 !bla
al_ = 00.40
aslfaas = 304
ka = .4      !salkd
i = 4.      !as
kasd = .3/0.4
kasd = 3./0.4
xxx = asin(3.0)
y = cos(i)
```

# Use the Emacs Integrated, Programmable RPN Calculator

Type C-x \* \* to open the calculator, type y for pasting results directly into the APDL file. And q to quit the 'Calc' windows.



```
emacs@VSDTQ800
File Edit Options Buffers Tools Calc Help

sz = .8
xloc = 0
yloc = 0

*dim,data,,5
data(1) = 12,15,28,10,32
hsz = sz/2

/pspec,0,1,1
/poly,4,xloc-hsz,yloc-hsz,1.8*(xloc+hsz),yloc-hsz,
      1.8*(xloc+hsz),yloc+hsz,xloc-hsz,yloc+hsz

x0 = xloc + hsz
y0 = yloc + .7*hsz
lof = .05

*vscfun,dsum,sum,data(1)
/LSPEC, 15, 0, 1.000
/TSPEC, 15, 0.700, 1, 0, 0
Pi = 3.14159265359
ang1 = 0
*do,i,1,5
  ang2 = ang1 + (360*data(i)/dsum)
/PSP, 2*i, 1, 1

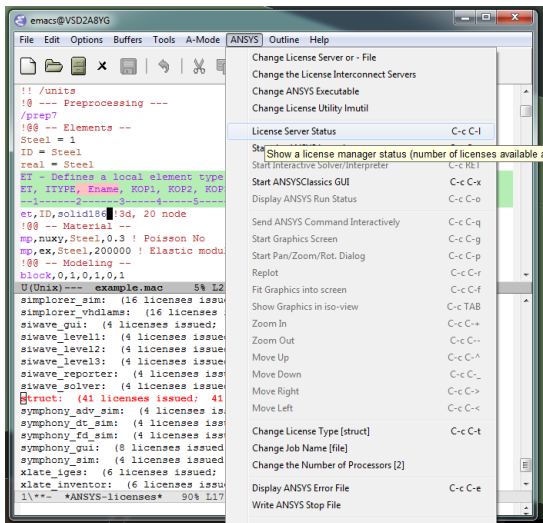
1(Unix) *- fontification.c 91% L366 (ANSYS Out1)
--- Emacs Calculator Mode ---
1: 3.14159265359

Emacs Calculator Trail
p1 3.14159265359

1) *- Calc: 12 Deg All L3 [Calc] 1) *- *Calc Trail* All L2
Auto-saving...done
```

# Display the License Status and Preview Images

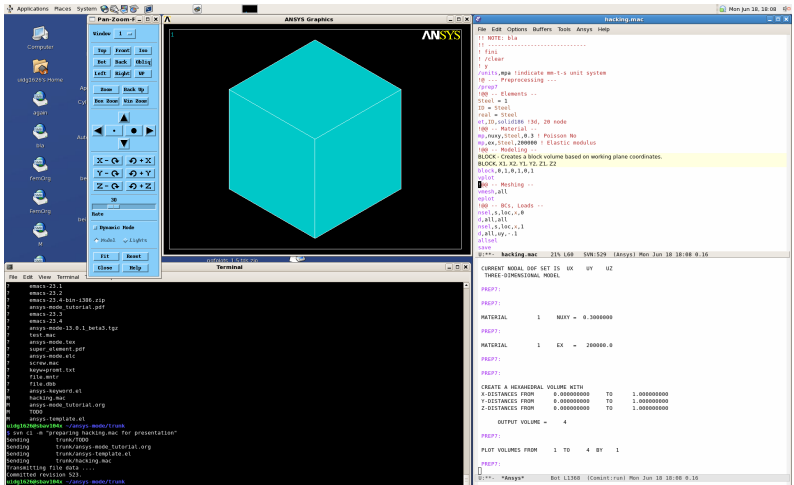
In Emacs' lower-half you see the license status (C-c C-l). All described features can be executed through the APDL-Mode menu or with keyboard shortcuts.





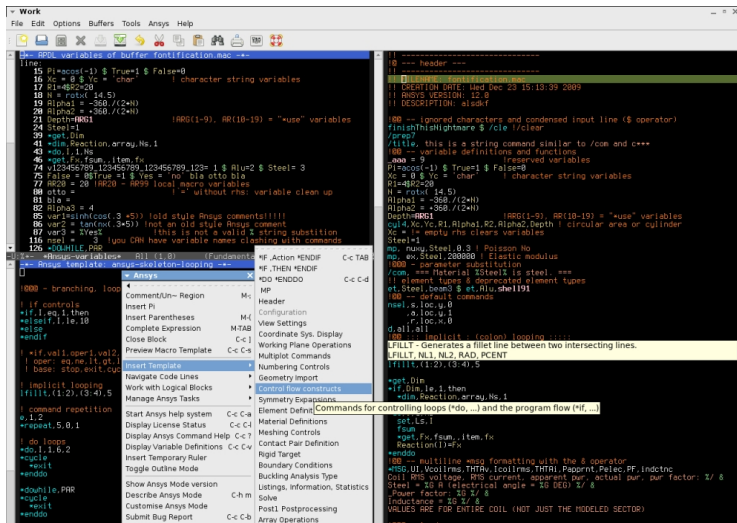
# Debug Your Code Interactively with the Solver (GNU-Linux)

You can run the Ansys solver under Emacs and send code lines from above APDL window with **C-c C-j** (**C-c C-c** for whole regions) directly to this process. Below you see the **interactive** solver output and on the left hand side the corresponding Ansys images.



## Arrange the APDL-Mode Windows to Your Needs

The image shows an Emacs 23.2 frame (in reversed colour mode and compiled with the GTK+ toolkit under GNU-Linux) with a ripped off APDL-Mode menu field

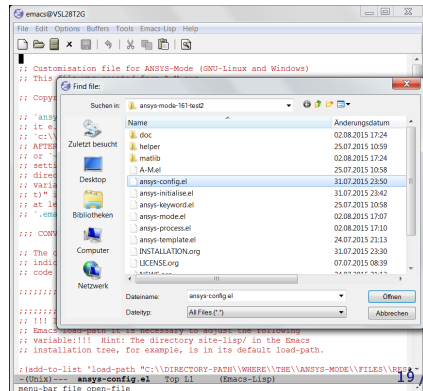
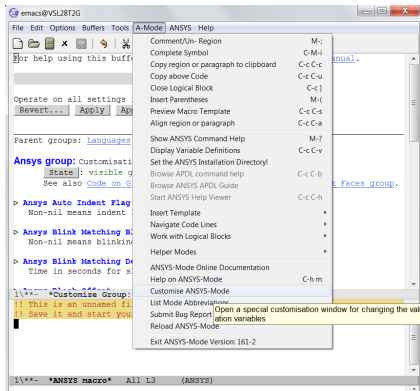


# Configure User Options and System Dependent Aspects

Please check the mode settings with Emacs' customisation system  
(no programming necessary) use the menu and

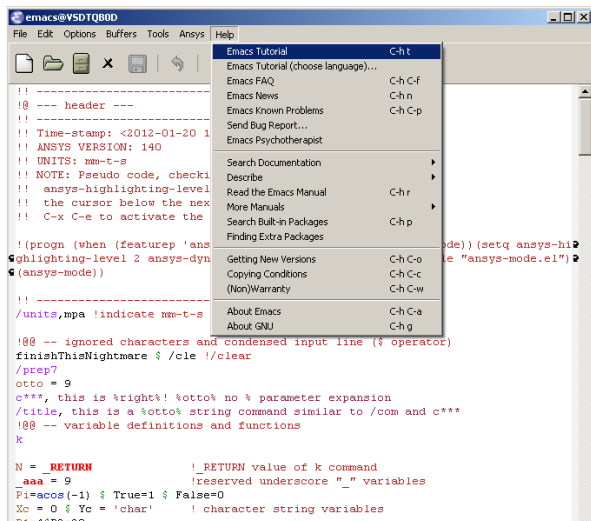
modify settings with Emacs' customisation system

or check the well commented file *apdl-config.el*



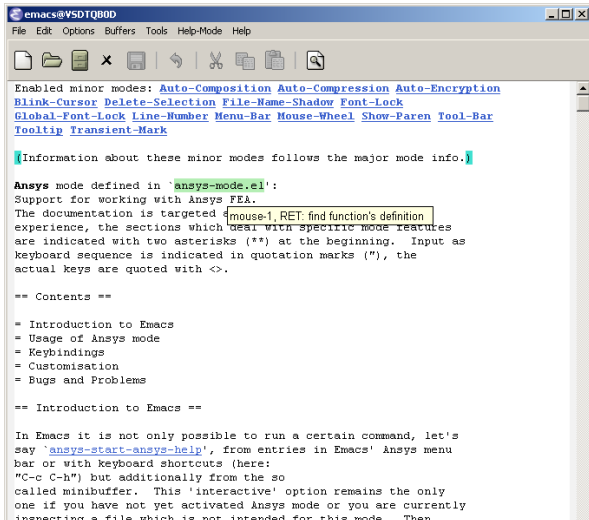
# Get to Know Emacs and Its Excellent Documentation

Newcomers to Emacs should take the **guided online tour** to get a background of its capabilities and fire up the interactive tutorial (**C-h t**) which is translated to various languages.



# Use the APDL-Mode Built-in Help

Please type **C-c C-h** for the mode manual (there is also a brief introduction of basic Emacs concepts) and **C-h m**, especially for APDL-Mode's keybindings.

The screenshot shows an Emacs window titled 'emacs@VSDTQB0D'. The menu bar includes 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Help-Mode', and 'Help'. The toolbar contains icons for file operations and editing. The main text area displays the 'ansys-mode' help text. It starts with a list of enabled minor modes: 'Auto-Composition', 'Auto-Compression', 'Auto-Encryption', 'Blink-Cursor', 'Delete-Selection', 'File-Name-Shadow', 'Font-Lock', 'Global-Font-Lock', 'Line-Number', 'Menu-Bar', 'Mouse-Wheel', 'Show-Paren', 'Tool-Bar', 'Tooltip', and 'Transient-Mark'. A line of text states: 'Information about these minor modes follows the major mode info.' This is followed by the definition of 'ansys-mode' and instructions on how to use the help, including a note about the 'mouse-1' key binding. A table of contents is listed, and the text concludes with instructions on how to access the help from the Emacs menu or keyboard shortcuts.

```
emacs@VSDTQB0D
File Edit Options Buffers Tools Help-Mode Help

Enabled minor modes: Auto-Composition Auto-Compression Auto-Encryption
Blink-Cursor Delete-Selection File-Name-Shadow Font-Lock
Global-Font-Lock Line-Number Menu-Bar Mouse-Wheel Show-Paren Tool-Bar
Tooltip Transient-Mark

Information about these minor modes follows the major mode info.

Ansyes mode defined in `ansys-mode.el':
Support for working with Ansys FEA.
The documentation is targeted at mouse-1, RET: find function's definition
experience, the sections which deal with specific mode features
are indicated with two asterisks (**) at the beginning. Input as
keyboard sequence is indicated in quotation marks ("), the
actual keys are quoted with <>.

== Contents ==

= Introduction to Emacs
= Usage of Ansys mode
= Keybindings
= Customisation
= Bugs and Problems

== Introduction to Emacs ==

In Emacs it is not only possible to run a certain command, let's
say `ansys-start-ansys-help', from entries in Emacs' Ansys menu
bar or with keyboard shortcuts (here:
"C-c C-h") but additionally from the so
called minibuffer. This 'interactive' option remains the only
one if you have not yet activated Ansys mode or you are currently
inspecting a file which is not intended for this mode. Then
```

# You Might Read Further APDL-Mode Documentation

**Licensing and costs:** This is free and open software, there are **no costs** and effectively **no restrictions** for you using Emacs and APDL-Mode - even - commercially. Both are under the **GPL, the Gnu Public License** copied in the *LICENSE.org* file.

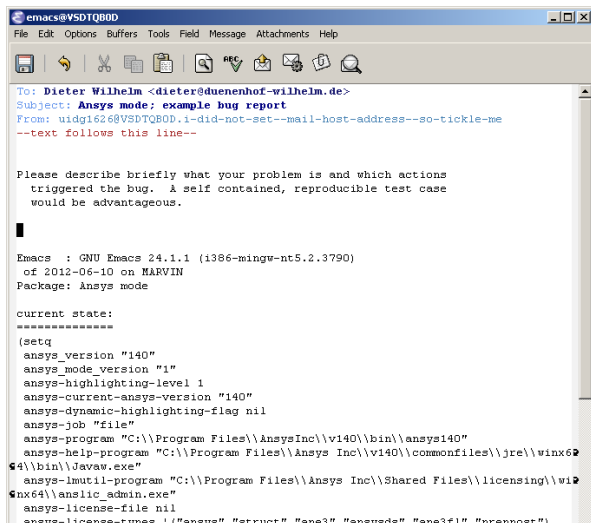
**Configuration:** More detailed instructions might be necessary if you are not using default *Ansys* installations. Please have a look in the *INSTALLATION.org* file. The online documentation is at **APDL-Modes's documentation site**.

**Hands-on tutorial and reference:** You will find these in-depth documentation included in the mode's archives on **GitHub's releases page** or read **online**.

**News and project history:** They are placed in the mode's accompanying *NEWS.org* file

# Search for Help, Report Bugs and Issues

Besides the documentation, have a look in the [GitHub's issues site](#) or send an [email to the maintainer](#). Please use the APDL-Mode bug report functionality, which might provide helpful status information.



```
emacs@VSDTQB0D
File Edit Options Buffers Tools Field Message Attachments Help

To: Dieter Wilhelm <dieter@duenenhof-wilhelm.de>
Subject: Ansys mode; example bug report
From: uidg16268@VSDTQB0D.i-did-not-set--mail-host-address--so-tickle-me
--text follows this line--

Please describe briefly what your problem is and which actions
triggered the bug. A self contained, reproducible test case
would be advantageous.

Emacs : GNU Emacs 24.1.1 (i386-mingw-nt5.2.3790)
of 2012-06-10 on MARVIN
Package: Ansys mode

current state:
=====
(setq
 ansys_version "140"
 ansys_mode_version "1"
 ansys-highlighting-level 1
 ansys-current-ansys-version "140"
 ansys-dynamic-highlighting-flag nil
 ansys-job "file"
 ansys-program "C:\\Program Files\\AnsysInc\\v140\\bin\\ansys140"
 ansys-help-program "C:\\Program Files\\Ansys Inc\\v140\\commonfiles\\jre\\winx64\\bin\\Javaw.exe"
 ansys-lmutil-program "C:\\Program Files\\Ansys Inc\\Shared Files\\licensing\\wi
nx64\\anslic_admin.exe"
 ansys-license-file nil
 ansys-license-times '("ansys" "struct" "ans3" "ansvds" "ans3fl" "ansvpost")
```

# Use APDL-Mode Appropriate to Your Needs

The relevance of APDL remains: 'WorkBench' and 'AIM' operate **exclusively** the Ansys solver with it! For a **true understanding** the study of APDL is **prerequisite**. Furthermore, code APDL only for **repetitive** tasks or WB/AIM snippets.

## Basic APDL Viewer

Navigating in WB solver input files, discerning relevant information through highlighting, quickly analysing APDL commands with the built-in help or pin-pointing the reference help in a browser.

## Earnest APDL Editor

Specific shortcut keybindings, outlining, code templates, completions, auto-indentation, abbreviations, auto-insertion.

## Advanced APDL Environment

Solver communication/feedback - hybrid between coding and debugging (GNU-Linux only), retrieving license states, error file viewing, abort file handling, extending APDL templates, ...



# Last Slide of the APDL-Mode Tutorial

Hint for the curious:

