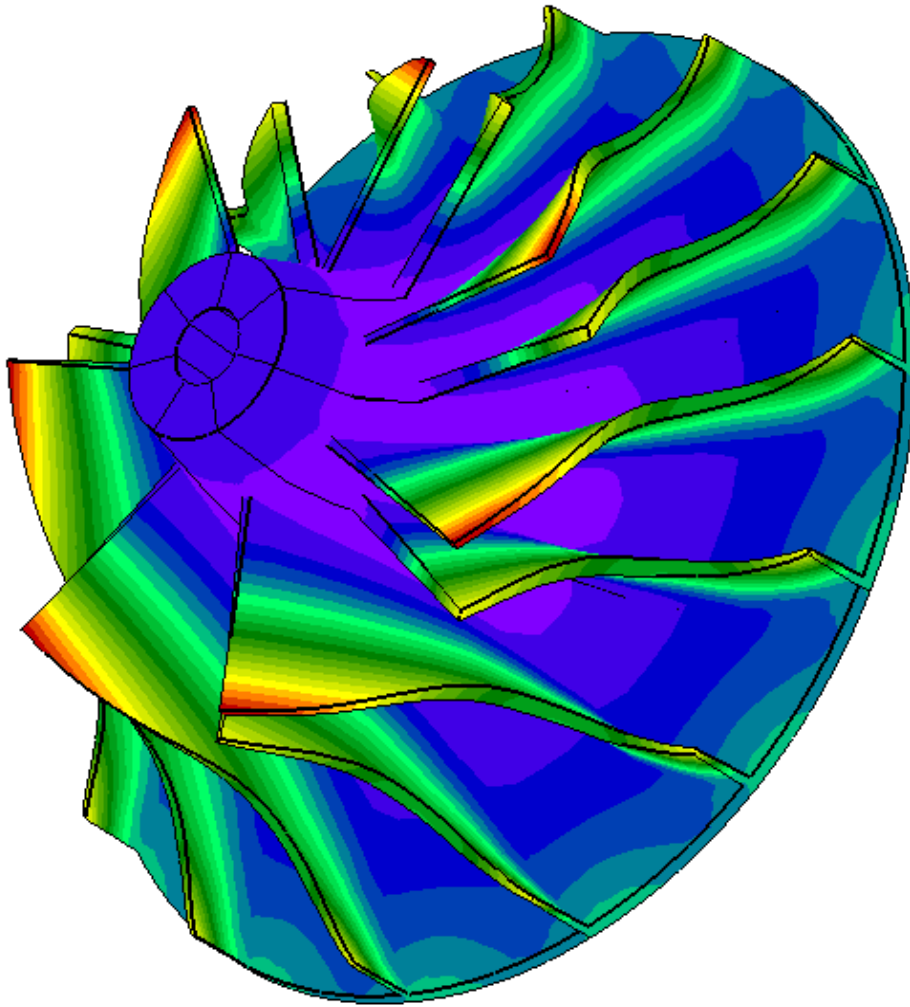# Getting Started with CalculiX

Jeff Baylor – Convergent Mechanical Solutions LLC
CGX Tutorial based on a tutorial by Guido Dhondt
Additional contributions by Chris Fisichella

Contents

Testing the Installation – Using the Command Line

On Windows, click on the CalculiX Command short cut in your Start Menu

[Start] [Programs] [bConverged][CalculiX] [CalculiX Command]

On UNIX/Linux, open a shell. At the command prompt type:

```
cgx -b test.fbd
```

Does the CalculiX Command window appear? The window should have usage information displayed.

There may be a warning message stating that the test.fbd file does not exist. This is ok. If the file does not exist, CGX will create it when you save the file for the first time.

Close the CGX window by clicking the close box.

If you prefer working at the command line, copy one of the CCX tests and view it in CGX by typing these commands (on Windows):

```
copy %CALCULIX_ROOT%\CalculiX\ccx\test\beampl.inp .
cgx -c beampl.inp
```

Does the CGX window appear with a beam rendered? Quit the CGX window.

As an alternative to the command line, copy the same file from the installation folder to your working folder using Window Explorer.
In Windows Explorer, double click on the beampl.inp file to open it in CGX. Does the CGX window appear with a beam rendered? Quit the CGX window.

2
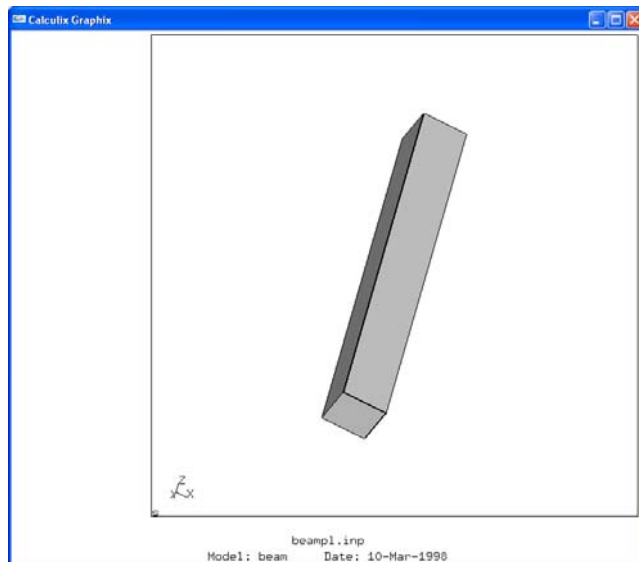
In Windows Explorer, right mouse click on the file beampl.inp and select the Edit option. This will open the file in a customized build of the text editor SciTE. Scroll to the bottom of the file and add the following lines just before the "*END STEP" line:

```
*NODE FILE
U
*EL FILE
S,E
```

These request the solver to write deformation, stress and strain data to the results file. Save the file using <ctrl>s or the menu option [File][Save]

Solve the model with CCX by pressing the keys <ctrl>F10 or use the menu option [Tools][Solve]. Alternatively, to run the CCX job from the CalculiX Command prompt, type:

```
ccx beampl
```

Is "Job finished" the last line printed?

To view the results, open them in CGX. From within the text editor, press the keys <shift>F10 or use the menu option [Tools][Post process]. To run CGX from the CalculiX Command prompt type:

```
cgx -v beampl.frd
```

Does the CGX window appear with the same beam rendered?

Hold the left mouse button down in the left margin and select from the menu:

[Data Sets] [-Entity-] [4 All]

Does a contour plot render?
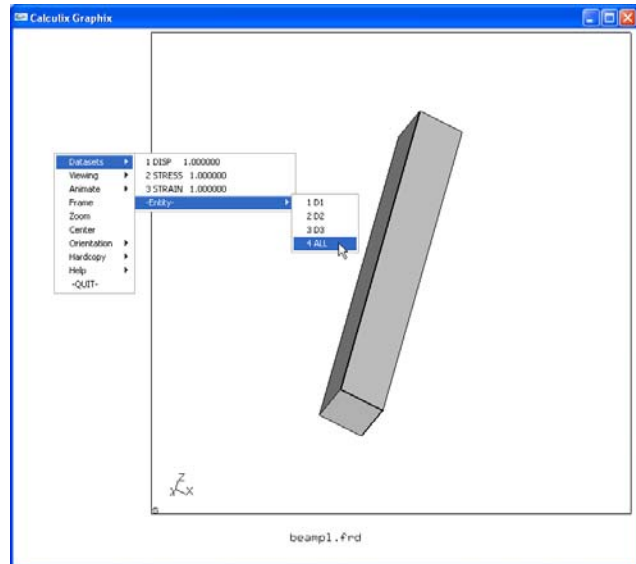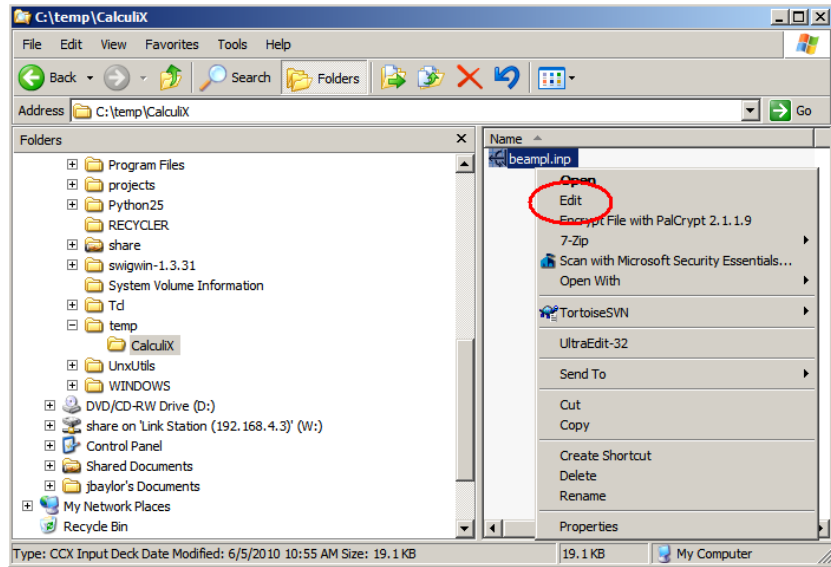
If all of these operations succeed, then the core components of CalculiX are installed properly. To test the solver in depth, run the test suite from your Start Menu.

      [Start] [Programs] [bConverged][CalculiX] [Test CalculiX 2.2]

This may take several minutes.
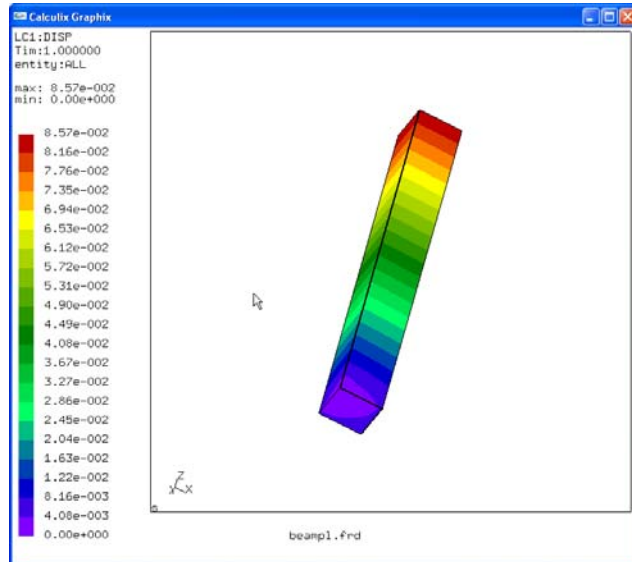
Tutorial Conventions

K: keyboard command
LMP: press left mouse button
LMR: release left mouse button
LMPR: press and release the left mouse button
MM: press middle mouse button
RM: press right mouse button
M: move mouse (no pressing or releasing)
V: visible result
E: press enter
Comment

CalculiX GraphiX (CGX) - Preprocessor Tutorial

This tutorial is based on a tutorial written by Guido Dhondt which is available at
http://www.dhondt.de/tutorial.txt.

In UNIX/Linux, start the tutorial by opening a shell. In Windows, either open the
CalculiX Command window [Start] [Programs] [CalculiX] [CalculiX Command] or
start the Launcher [Start] [Programs] [CalculiX] [Extended Version][CalculiX
Launcher]

K: If using a command line, start cgx by typing "cgx -b beam.fbd" (without the ""),
press enter (E). If using the Launcher, open a new file named "beam.fbd"
with CGX. The -b flag is the default flag when the Launcher opens an
FBD file.

V: A drawing window is created
with on top "CalculiX GraphiX No.
1" for Linux or "CalculiX GraphiX"
for Windows, on the bottom
"beam.fbd", a drawing area within
which there is a coordinate system
x-y-z and a mouse pointer.

The "-b" indicates build mode.
The file "beam.fbd" will be opened
if it exists, otherwise a new file will
be created.
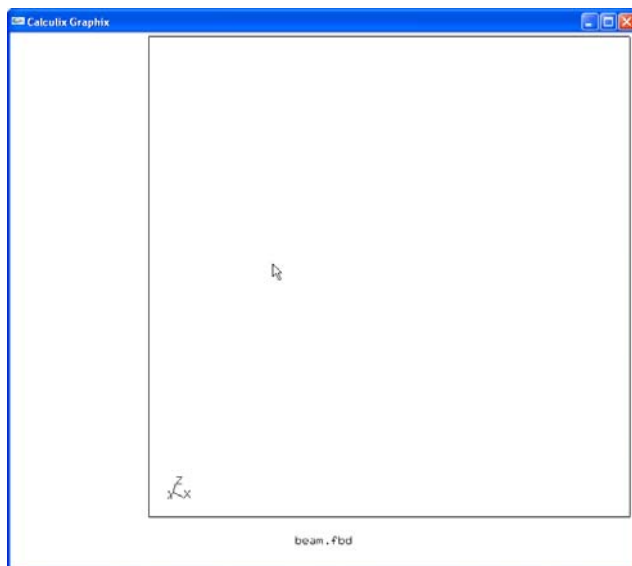
M: Move mouse into newly created window
K: Type "pnt p1 0 0 0", E

Keyboard commands are accepted only if the CalculiX GraphiX window is active

V: The text "pnt p1 0 0 0" appears in the window you started cgx in.
M:
V: A blue dot appears in the drawing window

K: Type "pnt p2 1 0 0", E
M: Move mouse in the drawing window to the left of the drawing area or below
        the drawing area (this area is also called the menu area)
LMP
V: A grey menu appears, on top "Viewing", on the bottom "-QUIT-".
M: Move the mouse down to the item "Frame"
LMR
V: Two blue dots representing the points
appear in the drawing area.
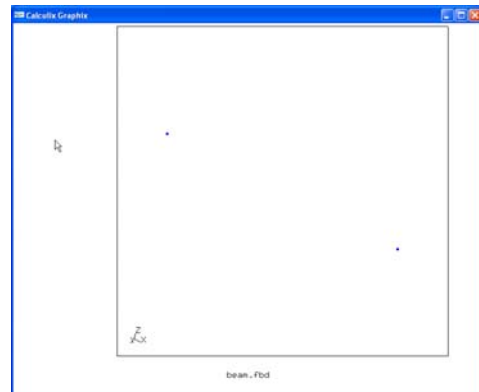
K: Type "plot pa all", E
V: Blue labels "p1" and "p2" appear next to the blue points

K: Type "plot pa all r", E
V: The dots and labels are red now.

g=green, y=yellow, k=black, b=blue m=magenta

If you mistype the first letter of a command, press enter, and type the command
again; else, if you mistype any other letter, move back with the back key and
retype the wrong letters

K: Type "qlin", E
M: Move the mouse inside the drawing area
K: Type "r"
M: Move the mouse pointer a little bit to the
right and down
K: Type "r"
M: Move the mouse a bit
V: The mouse pointer changes into a
rectangular pointer, the size of which is
        determined by the positions at which
        you typed the r's: the first "r" denoted
        the upper left corner, the second "r" the
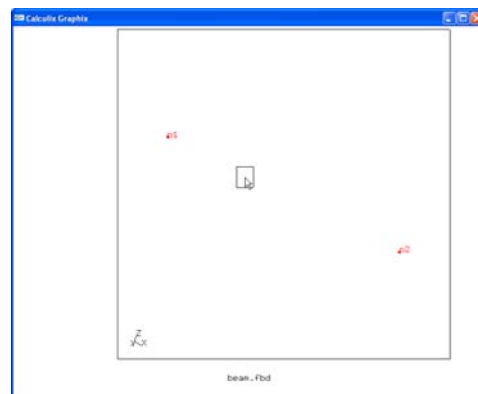
6

lower right corner

M: Move the new mouse pointer onto the dot symbolizing p1
K: Press "b" (for begin)
V: The marked point's definition will be printed to the typing window, unless you
        missed it.
M: Move the pointer onto the dot symbolizing p2
K: Press "g" (for generate)
V: The marked point's definition will be printed to the typing window as well as a
        description of the line you created

To create a circle, mark the first point with "b", then the center point with "c" and
finally the end point with "g"

K: Type "q" (to exit the qlin command)
V: The text "done" appears in the typing window
M
V: The mouse pointer returns to its original form

If you do not quit, you can continue to draw lines using repeatedly "g" to create
connected line segments, or interrupting with "b" to start a line from a new
starting point; with "l" an existing line can be picked and redefined using "b", "c"
and "g"

K: Type "plus la all" E
V: The newly created line appears together
with its label.

If you type "plus l all" E, no label appears

If you type "plot l all" the drawing starts from
scratch and only the line appears (the points
and point labels disappear)
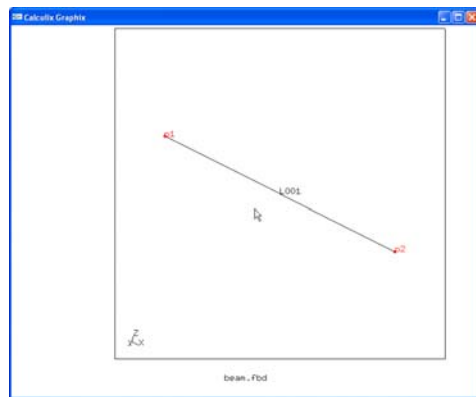
K: Type "qadd se1", E
K: Type "a"
V: After a fraction of a second the text "mode:a" appears in the typing window.
        Always wait for this text before proceeding.

Now create a window encompassing the line

M: Move the mouse pointer to the left upper corner of an imaginary rectangle

encompassing the line

T: Type "r"

M: Move the mouse pointer to the right lower corner of an imaginary rectangle encompassing the line

T: Type "r"

M

V: A rectangle appears with the size you just defined

If nothing happens, LMPR in the menu area and returning the mouse into the drawing area usually helps

M: move the rectangle such that the line is inside it

If the rectangle proves to be too small, restart the marking of the corners with "r"
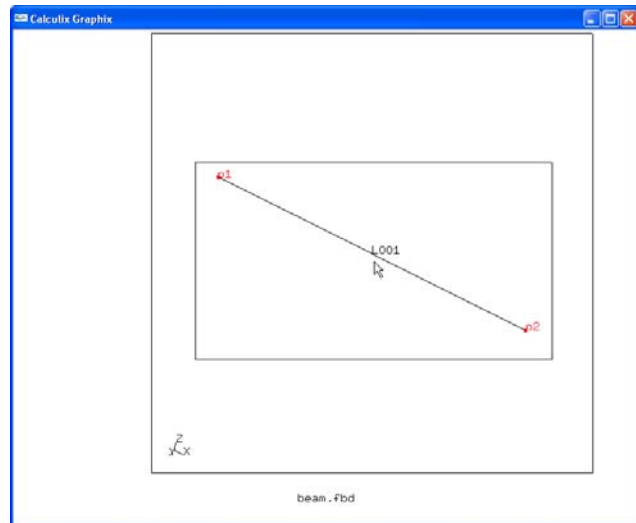
K: Type "l" (for lines)

V: In the text window the lines "caught" by your rectangle are listed (here only L1); in the drawing area the line is shown pink

K: Type "q" (to quit the qadd command)

By the previous command you created a set with name "se1" and containing one line with name "L1"

Notice that the line name was given automatically. Names provided by the program always use CAPITAL letters
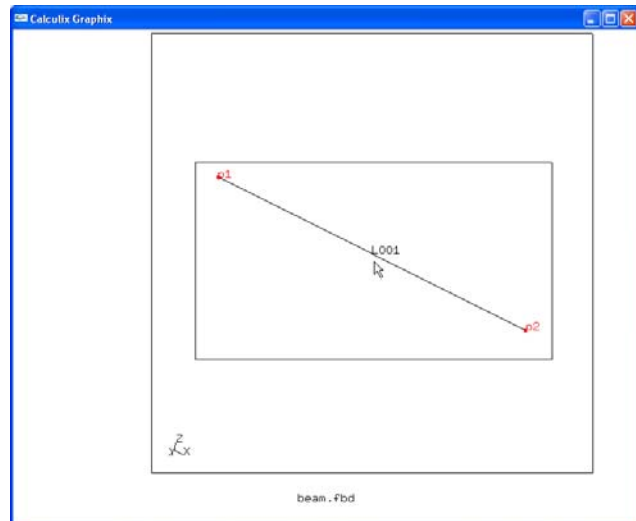
K: Type "plot la se1", E
V: The lines and their labels contained in set se1 appear in the drawing area, in
        this case just one line

K: Type "plot la all", E
V: Nothing really changes

The set "all" contains everything and always exists

K: Type "swep se1 se2 tra 0 1 0", E
M
V: Four lines and their labels
appear now in the drawing
window.

By the swep command you swept
the line using a translation with
direction vector 0 1 0; thereby
three new lines (L002, L003 and
L004) and a surface A001 were
created
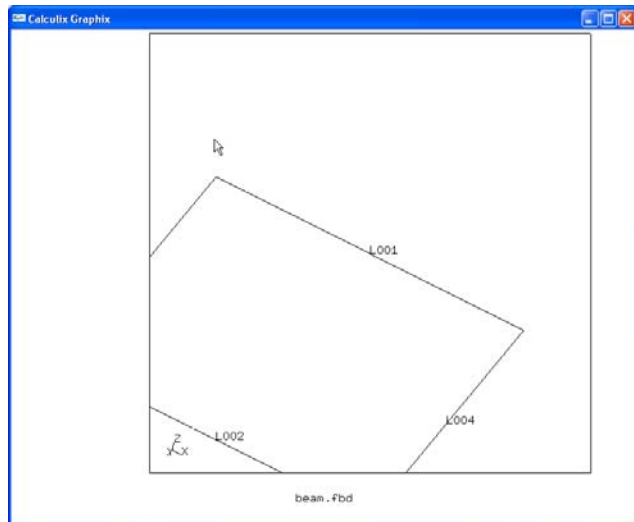
K: Type "plus sa all", E
V: The surface A001 and its label appear in magenta

K: Type "plus la se2 g", E
V: The line in set se2 (defined by the swep command) and its label appear in
        green

The swep command takes two set names: the first is the set being swept, the
second is the target set in which the swept entities are stored in their final
position

K: Type "prnt se", E
V: In the typing
window the names of
all existing sets
appear: se1, se2 and
all

9

Create set se3 which contains surface A001

K: Type "qadd se3", E
K: Type "a"
M: Move the mouse pointer to the upper left position of a rectangle containing
     A001
K: Type "r"
M: Lower right position of rectangle
K: Type "r"
M
V: Rectangle appears
M: Move such that the surface (or
at least its label) is in the rectangle
K: Type "s" (for surface)
V: The surfaces caught appear in
the typing window and acquire the
color pink
     in the drawing area
K: Type "q" (for quit)
M: The mouse pointer takes its
original form
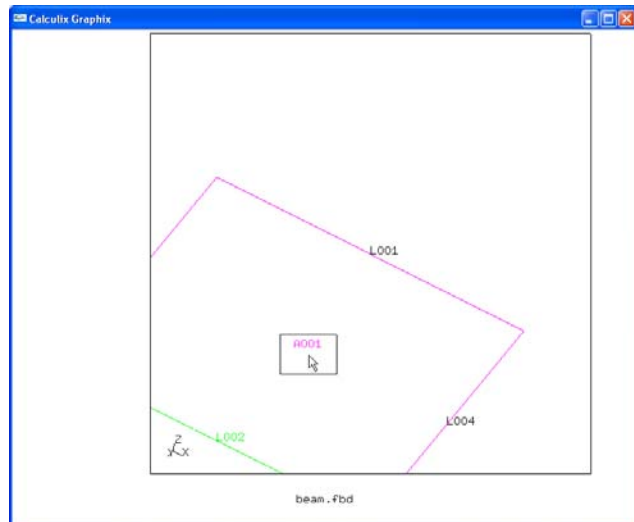
If you omit "a" in the above
procedure, maximum one surface
can be picked with each "s"
command; the "a" command
ensures that by typing "s", all surfaces whose labels lie within the pointing
rectangle are put into the set; no matter whether you use "a" or not, the "s" (or
equivalently for picking other entities: "p"=points, "l"=lines, "b"=bodies,
"n"=nodes, "e"=elements) command can be repeated after moving the mouse, in
order to pick other entities

LMP
M
V: The surface in the drawing window rotates
LMR
MMP
M
V: The surface increases or decreases in size
MMR
RMP
M
V: The surface is translated
RMR

The three mouse buttons are used to rotate the structure in the drawing window, to make it smaller or bigger, or to translate it

K: Type "swep se3 se4 tra 0 0 10", E
M
V: A total of six surfaces and 12 lines are drawn in the drawing window. You
     created a beam with a square cross section by translating surface A001.
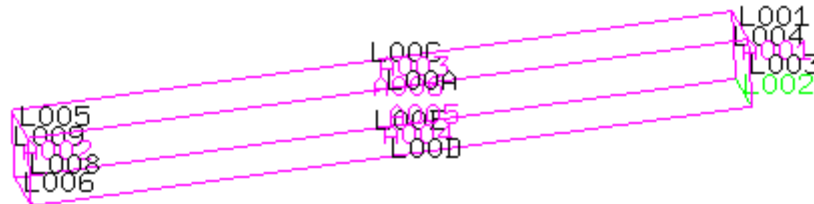
K: Type "plot ld all", E
V: All lines appear with their division

Each line has a default division of 4. This means that four linear elements or two quadratic elements can be created along each line

K: Type "elty all he8", E
K: Type "mesh all", E
K: Type "plot m all", E
V: The beam is smoothly colored with the color green
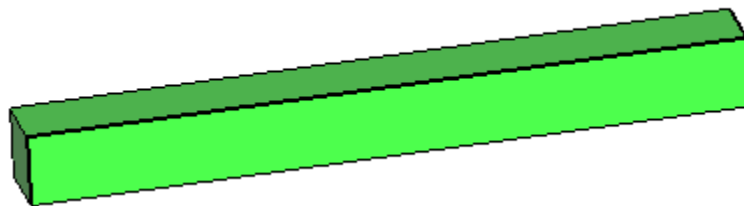
M: Move the pointer into the menu area
LMP
M: Select the menu viewing and the submenu "LINES"
LMR
V: The surface coloring disappears and green lines appear marking 64 elements

8-node brick elements are he8-elements; other element types are:
        tr3   3-node 2-D elements (linear)
        qu4   4-node 2-D elements (linear)
        qu8   8-node 2-D elements (quadratic)
        he20   20-node brick elements (quadratic)
tr3, qu4 and qu8 elements are used to mesh surfaces, he8 and he20 to mesh bodies

K: Type "plot e all", E

11

V: Nothing really changes
K: Type "plot ea all", E
V: The elements AND their labels appear (in green)
K: Type "plus na all", E
V: The nodes and their labels appear (in red)

From the above you can tell that a body must have been created. Indeed:

K: Type "plot ba all", E
V: The body B1 appears with its label
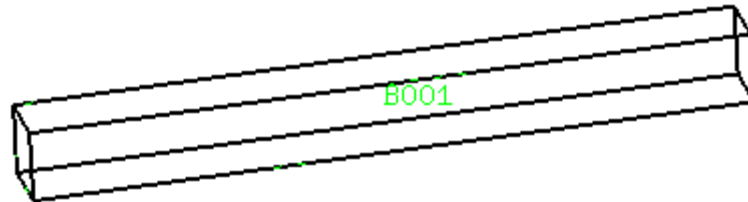
The black structure edges can be removed by selecting "viewing - toggle model edges" in the menu (move pointer into menu area)

K: Type "plot ld all", E
V: Lines and labels appear

K: Type "qdiv", E
M: Move the mouse pointer onto one of the long lines
K: Type "2"
M
V: The division of the line changed from 4 to 2
M: Move the pointer onto another of the long lines
K: Type " 12" (note the space before the 12!)
M
V: The division changed to 12
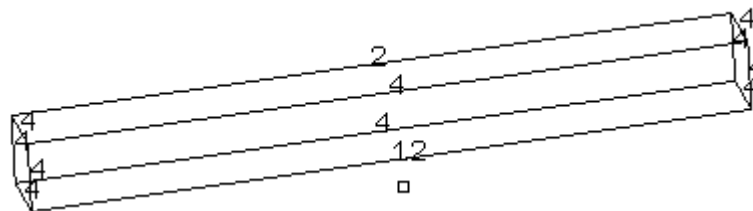
K: Type "a"

Wait till "mode:a" appears in the typing window, now create a rectangle by marking its corners with "r", and move it so that all the division numbers for the long lines are within the rectangle

K: Type " 12"

12

V: All lines have 12 divisions
K: Type "q" (to leave qdiv)

The divisions should be:
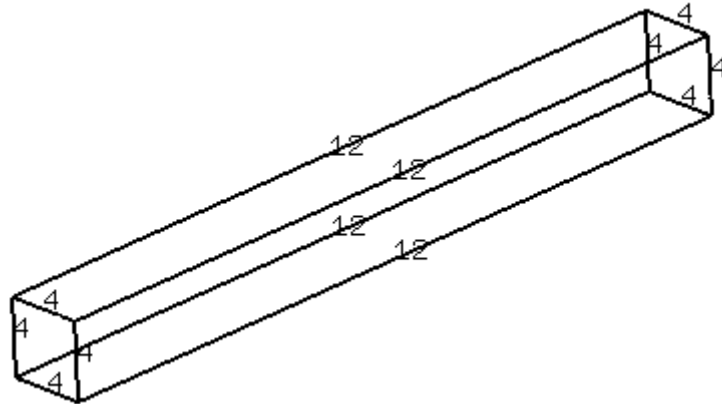
K: Type "del me all", E

Mesh is deleted

K: Type "elty all he20", E

New element type is assigned to set "all" (which contains all entities)

K: Type "mesh all", E
K: Type "plus e all", E
V: Now 8 quadratic elements were created (two along each line, since the
       division is 4)

Now select the "orientation, +x view" in the menu and move the mouse pointer
back into the drawing area

V: The structure is looked at along the x-axis in +x direction

K: Type "plot l all", E
K: Type "plus n all", E

Now pick all nodes with z-coordinate 0 and put them into a set with name fix

K: Type "qadd fix1", E
K: Type "a"
M: To left upper corner of rectangle
K: Type "r"
M: To left lower corner of rectangle
K: Type "r"
M: Move rectangle such that all nodes with z=0 lie within the rectangle

K: Type "n"

If the rectangle was too small, you can move it and repeat the command "n", nodes picked by consecutive "n" commands are all added to the set fix1

K: Type "q"

Now pick all nodes with z-coordinate 0 at the positive y surface of the beam and put them into a set with name fix2

K: Type "qadd fix2", E
K: Type "a"
M: To left upper corner of rectangle
K: Type "r"
M: To left lower corner of rectangle
K: Type "r"
M: Move rectangle such that only the nodes with z=0 and the are on the positive surface of the beam lie within the rectangle (in this view, all these nodes will be coincident)
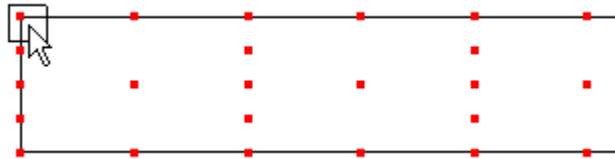
K: Type "n"

Five nodes should be printed to the output window

K: Type "q"

Now pick the node with the coordinates (0, 1, 0) and put it in the set fix3. Rotate the model slightly so the node of interest is not coincident with any other nodes.

K: Type "qadd fix3", E
M: Move the cursor so it is on the node

K: Type "n"

One node description should be printed to the output window, verify that the coordinates are correct.

14
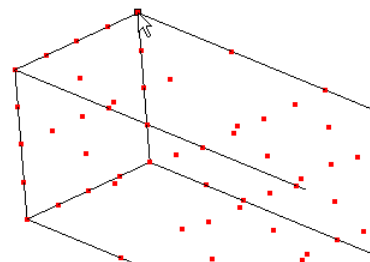
K: Type "q"

K: Type "plot n all", E
K: Type "plus n fix1 g", E
V: All nodes in set fix1 are green colored

K: Type "plus n fix2 b", E
V: All nodes in set fix2 are blue colored

K: Type "plus n fix3 m", E
V: All nodes in set fix3 are magenta colored
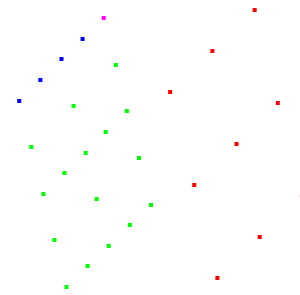
Now reselect the "orientation, +x view" in the menu and move the mouse pointer
back into the drawing area. Put all faces at y=1 into set "load":

K: Type "plot f all", E

Now select "viewing - FILL" and subsequently "viewing - Toggle Element Edges"
from the menu; all faces are shown yellow; faces are external sides of elements
and only exist after meshing

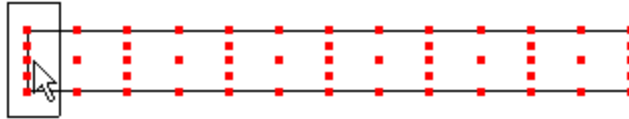K: Type "qadd load", E
K: Type "a"
M: To left upper corner of rectangle
K: Type "r"
M: To right upper corner of rectangle
K: Type "r"
M: Move rectangle such that all faces with y=1 lie within the rectangle
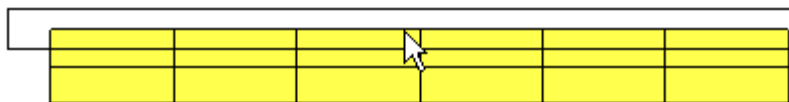
K: Type "f"
V: All faces caught get the color pink. It is impossible to pick only the faces on
        top, you will have caught some of the side as well. We will take care of this
                                              next.
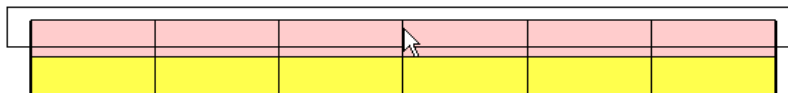
K: Type "q"
K: Type "plot f load", E
V: This shows all faces belonging to set "load"

Now we want to get rid of the faces not on top of the beam and belonging to set "load"

K: Type "qrem load", E

This command removes entities from set "load"

K: Type "a"
M: To left upper corner of rectangle
K: Type "r"
M: To right lower corner of rectangle
K: Type "r"
M: Move the rectangle such that all faces not on top of the beam lie within the
    rectangle
K: Type "f"
V: All faces caught get the color pink. They are removed from set "load"
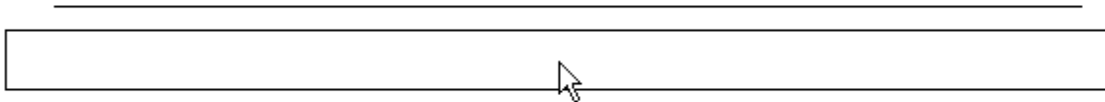
K: Type "q"

In a similar way one can also pick elements

K: Type "send all fbd", E
V: Some text in the typing window, ending with "ready"

By this action a file "all.fbd" was created containing the geometric entities of the model (points, lines, surfaces, bodies, sets.. but no nodes or elements); by renaming this file into "beam.fbd", the model can be reloaded into another cgx session by the same command you started this session with)

K: Type "send all frd", E
V: Some text in the typing window, ending with "ready"

By this action a file "all.frd" was created containing the nodes and elements of the model. It can be reopened in another session by the command cgx all.frd, similar to the command opening this session, but without "-b". However, no geometric entities are contained in this file.

K: Type "send all abq"
V: Some text in the typing window, ending with "ready"

By this action a file "all.msh" was created containing the nodes and elements in ABAQUS format; in addition an empty file "all.temp" was generated. If temperatures are defined this file is not empty.

K: Type "send load abq pres 1.0"
V: Some text in the typing window, ending with "ready"

By this action a file "load.dlo" was created containing the ABAQUS cards for a distributed pressure of magnitude 1.0 on all element sides to which the nodes in set load belong.

K: Type "send fix1 abq nam"
K: Type "send fix2 abq nam"
K: Type "send fix3 abq nam"
V: Some text in the typing window, ending with "ready"

By these actions files "fix1.nam", "fix2.nam" and "fix3.nam" were created containing the node labels collected into a set in ABAQUS convention. This can be useful to apply forces onto a node set, or assign a different material to an element subset.

K: Type "help"
V: A help menu appears in the typing window containing much more commands
      than the ones described here

Now select "-QUIT-" from the menu: you quit the session, and should find "all.fbd", "all.frd", etc. in your directory.

Solve With CCX and Post-process with CGX - Tutorial

Getting the mesh is only one part of a finite element analysis problem definition. You also need to:

- Describe the material
- Describe your constraints and loading with boundary conditions
- Define the type of solution
- Declare output requests

This chapter will discuss these steps and illustrate them by example.

In UNIX/Linux, start the tutorial by opening a shell. In Windows, open the CalculiX Command window [Start] [Programs] [CalculiX] [CalculiX Command], or from the Launcher [Run][CalculiX Command]

K: Type "mkdir beam", E
K: Type "move all.msh beam ", E
K: Type "move fix*.nam beam ", E
K: Type "move load.dlo beam ", E
K: Type "cd beam ", E

Open up a text editor by selecting [Start][Programs][bConverged][SciTE], or if you are using the Launcher [File][Open Editor], navigate to the beam folder, and type "beam.inp" for the file name.

```
*INCLUDE, INPUT=all.msh
*INCLUDE, INPUT=fix1.nam
*INCLUDE, INPUT=fix2.nam
*INCLUDE, INPUT=fix3.nam
*MATERIAL, Name=steel
*ELASTIC
  28000000, 0.3
*SOLID SECTION, Elset=Eall, Material=steel
*STEP
*STATIC
*BOUNDARY
  Nfix1,3,3,0
  Nfix2,2,3,0
  Nfix3,1,3,0
*DLOAD
*INCLUDE, INPUT=load.dlo
*NODE FILE
  U
*EL FILE
  S, E
*END STEP
```

An input file is commonly called an "input deck", the characters from the asterisk to the first comma are called a "keyword" and the keyword with associated data is called a "card". We will briefly discuss this input deck before continuing with the tutorial.

An input deck defines the finite element analysis problem to be solved. In general, it contains a mesh definition, material, analysis type, boundary conditions and output requests. The above deck is a very simplistic analysis; most will require significantly more cards to define the problem.

Mesh:

```
*INCLUDE, INPUT=all.msh
*INCLUDE, INPUT=fix1.nam
*INCLUDE, INPUT=fix2.nam
*INCLUDE, INPUT=fix3.nam
```

The *INCLUDE keyword directs CCX to read an additional file. The mesh definition for this model is defined in the "all.msh" file. The files with the ".nam" extension have node set definitions. These were generated from CGX in the previous tutorial.

Material:

```
*MATERIAL, Name=steel
*ELASTIC
  28000000, 0.3
*SOLID SECTION, Elset=Eall, Material=steel
```

The material model used for this analysis is a linear elastic model. The Young's Modulus and Poisson's Ratio are all that is required for a static linear elastic solution. This is the simplest material model. Along with the material definition is a SECTION card, in this case a *SOLID SECTION.

MatWeb is a great resource (http://www.matweb.com) for preliminary material data in the absence of a testing program or information from the material's manufacture.

Analysis type:

```
*STEP
*STATIC
...

*END STEP
```

An analysis must have at least one load step defined. A step is opened by the *STEP card and closed by an *END STEP. Within the step definition is a declaration of the analysis type and typically boundary conditions and output requests. The analysis in this case is a linear static analysis.

Boundary conditions:

```
*BOUNDARY
  Nfix1,3,3,0
  Nfix2,2,3,0
  Nfix3,1,3,0
*DLOAD
*INCLUDE, INPUT=load.dlo
```

The displacement boundary conditions for this model are defined by the *BOUNDARY card. The node set "Nfix1" is constrained from movement in the third degree of freedom (DOF). The second and third DOF of the nodes in "Nfix2" are constrained from movement. The first three DOFs of the nodes in "Nfix3" are also constrained to zero.

A distributed load is also applied (*DLOAD). This load is applied to element faces in the file "load.dlo"

Output requests:

```
*NODE FILE
  U
*EL FILE
  S, E
```

Nodal displacements and element stresses and strains are written to a file for post processing. This is an FRD formatted file.

More can be learned about each of these keywords and the many other options in the CCX documentation.

Run the solver.

K: Type "ccx beam ", E

The last line should be "Job finished"
Post process in CGX.

K: Type "cgx -v beam.frd", E

M: Move mouse in the drawing window to the left of the drawing area or below
       the drawing area (this area is also called the menu area)
LMP
V: A grey menu appears, on top "Datasets", on the bottom "-QUIT-".
M: Move the mouse down to the item "Datasets", pause
V: A grey submenu appears from "Datasets"
M: Move the mouse down to the item "1. Disp 1.00000"
LMR

The dataset is now displacement.

M: Move mouse in the drawing window to the left of the drawing area or below
       the drawing area
LMP
V: A grey menu appears, on top "Datasets", on the bottom "-QUIT-".
M: Move the mouse down to the item "Datasets", pause
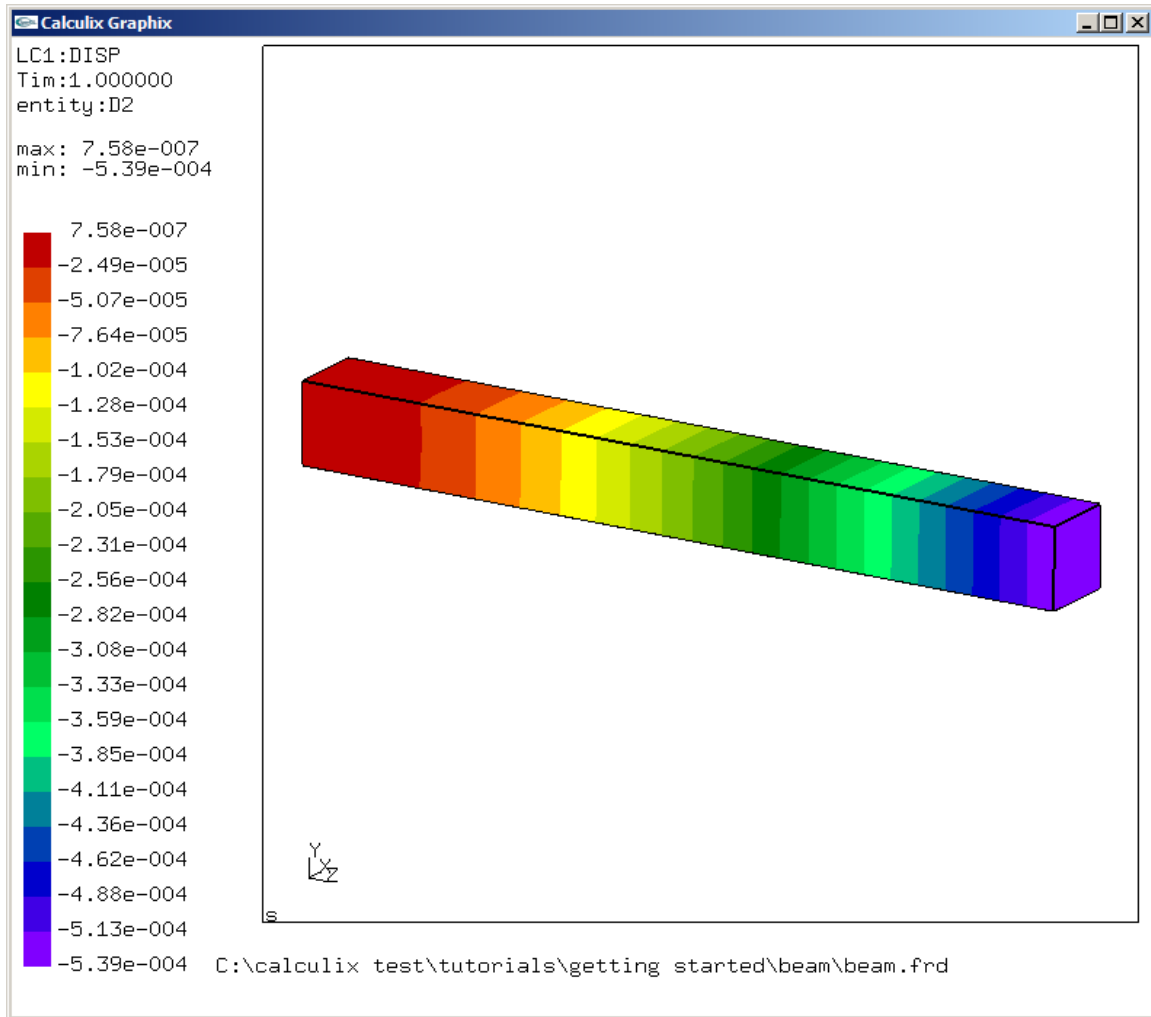V: A grey submenu appears from "Datasets"
M: Move the mouse down to the item "-Entity-", pause
V: A grey submenu appears from "-Entity-"
M: Move the mouse down to the item "D2"
LMR

A contour of the vertical displacement is now displayed.

```
Calculix Graphix                                          _ □ X
LC1:DISP
Tim:1.000000
entity:D2

max: 7.58e-007
min: -5.39e-004

    7.58e-007
   -2.49e-005
   -5.07e-005
   -7.64e-005
   -1.02e-004
   -1.28e-004
   -1.53e-004
   -1.79e-004
   -2.05e-004
   -2.31e-004
   -2.56e-004
   -2.82e-004
   -3.08e-004
   -3.33e-004
   -3.59e-004
   -3.85e-004
   -4.11e-004
   -4.36e-004
   -4.62e-004
   -4.88e-004
   -5.13e-004
   -5.39e-004    C:\calculix test\tutorials\getting started\beam\beam.frd
```

The vertical displacement of the end of an Euler beam is: -0.000536 assuming:

$$y_{max} = \frac{-wl^4}{8EI}$$

The resultant -0.000539 is sufficiently close given the assumptions.
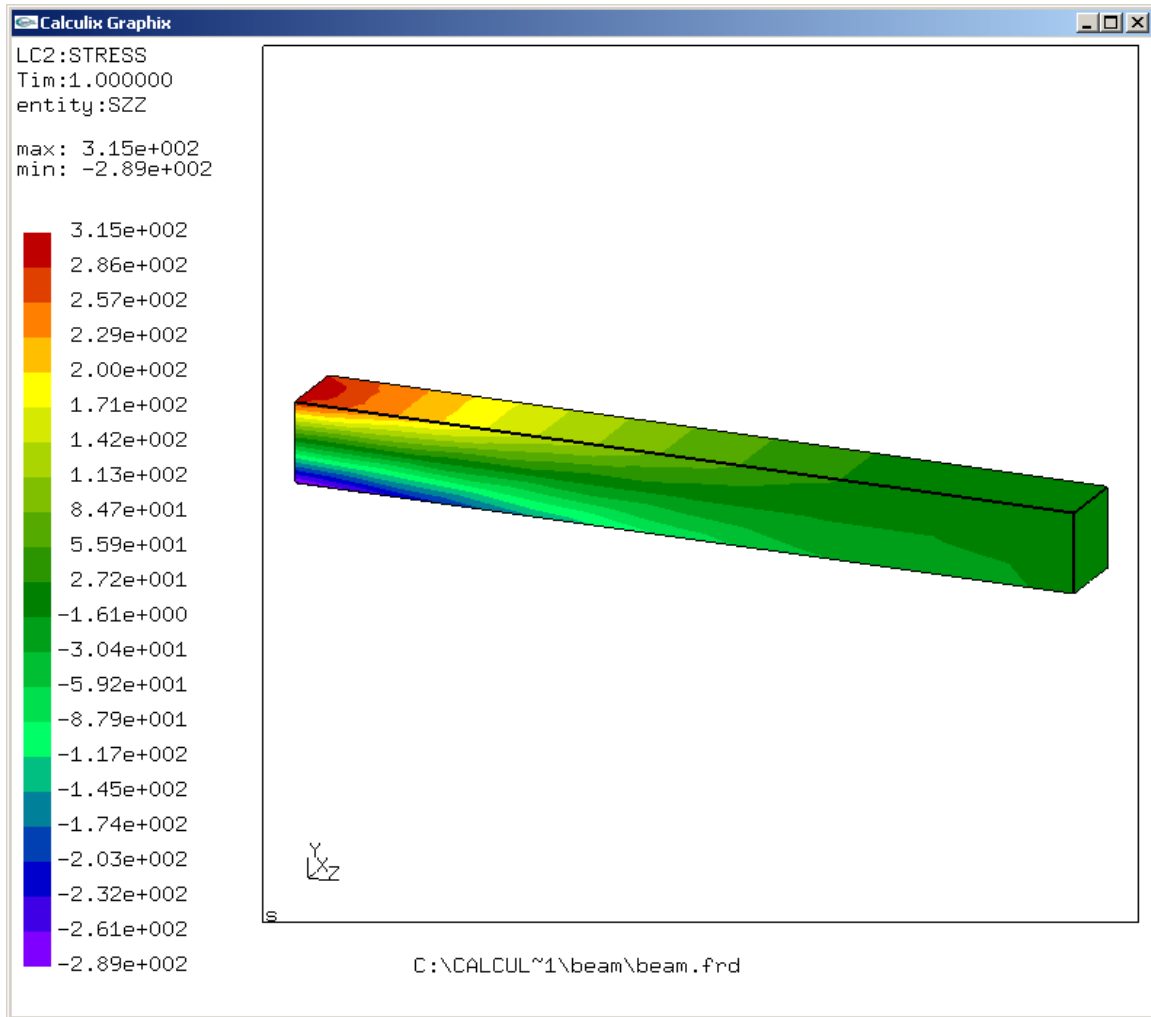
Select the dataset "Stress" and the entity SZZ

The beam bending stress is: 300 assuming:

$$\sigma_{yy} = \frac{Mc}{I}$$

The results from the FEM are 315 and -289.

A beam domain that closer resembles the assumptions in beam theory with a more refined mesh would yield results with closer agreement, but these results sufficiently demonstrate the problem.

Notice the lack of units through out this discussion. Units must be consistent throughout the model. Otherwise, they do not play a role.

Getting More Help

There are quite a few additional resources available to help.

For help with the **CGX** commands:

1. The documentation for CGX was written by the application's developer, Klaus Wittig, and is a good place to start. Reading the first few sections (at least through "Getting Started") will give you a quick overview. There is also a command reference and a section with some example models.

2. The Yahoo! Group for CalculiX (http://groups.yahoo.com/group/calculix/) is a great place to get help for CGX. It has been in use for six years now. You can search the archives and post questions. The developers and many active users contribute.

Some **CCX** references are:

1. The documentation is a good place to start. It was written by the CCX's developer, Guido Dhondt.

2. While the CCX input is not exactly the same as the ABAQUS syntax, it is sufficiently close that ABAQUS documentation will be helpful.

3. There are quite a few test files in the CCX build. These were also provided by Guido Dhondt.

4. The Yahoo! Group is also a great for CCX questions (Guido is the one who maintains forum).