3D Gmsh Assembly -> Calculix (via msh)

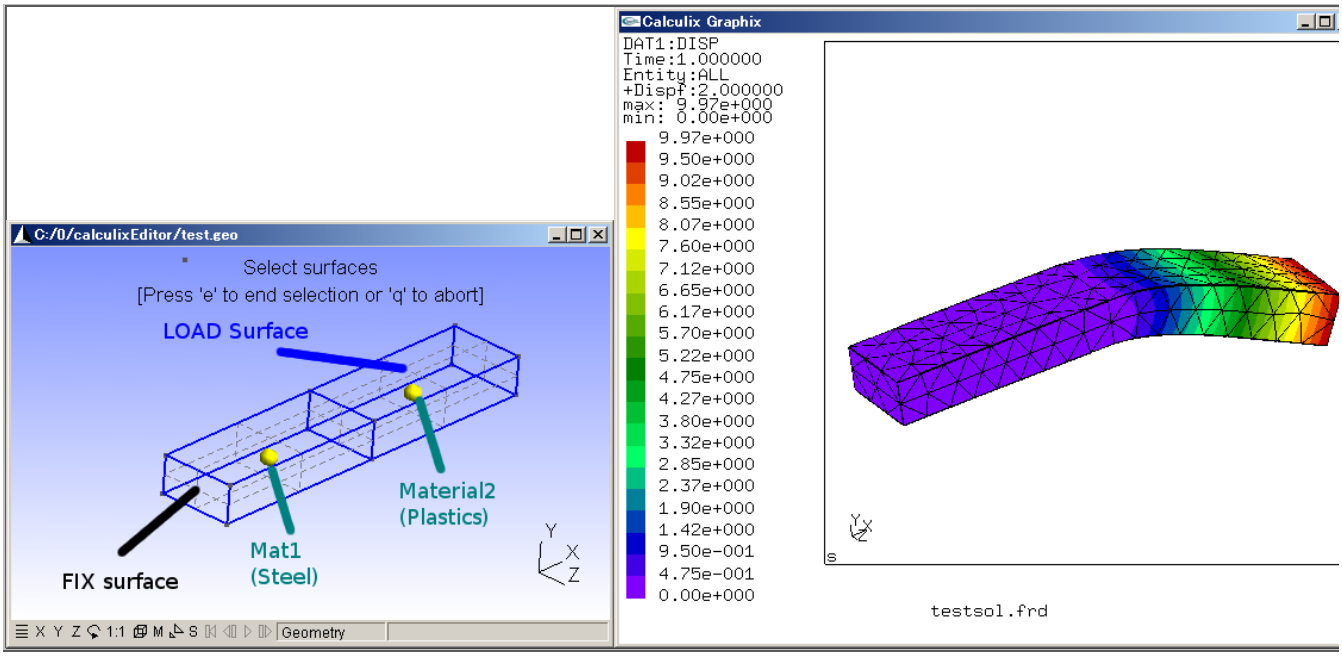
Tweet

---

g2c3.py ver 0.01 2011/Aug/19

*causion

- INCOMPLETION version
- 3D mesh only
- cannot convert Gmsh's Type18(18-node second order prism)
  cannot convert Gmsh's Type 12(27-node second order hexahedron)
  (Type18 and Type 12 will be made when using **Recombine** and mesh oder 2)



gmsh -- gmsh

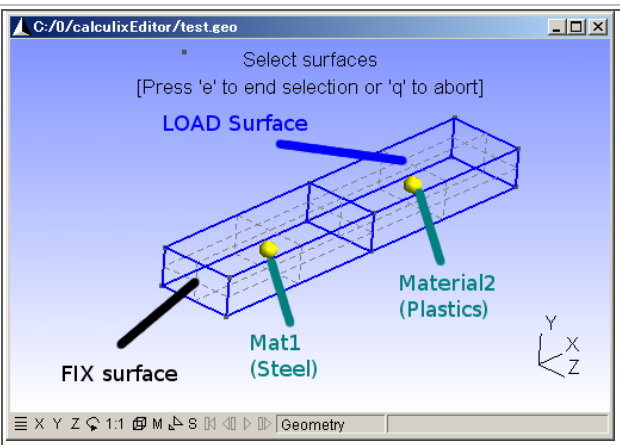
Calculix -- Calculix WindowsVersion

You need Python2.* . if you don't have it , please get and install it.
(I don't know the script will run with Python3.)

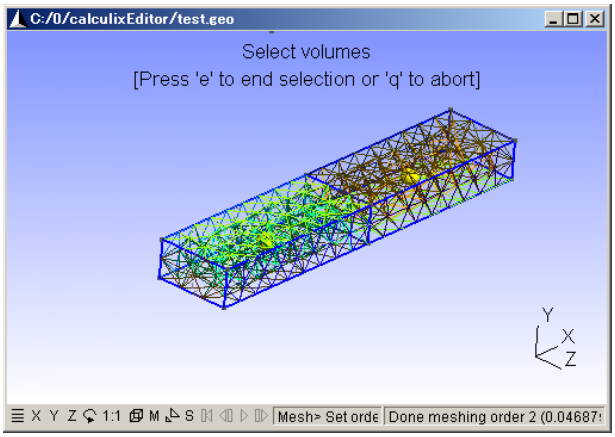
- get g2c3.py from **here**
- extract zip file

Make Assy with Gmsh

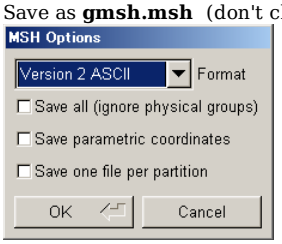
```
// Gmsh project created on Wed Aug 17 15:55:39 2011
lc = 5;
Point(1) = {0, 0, 0, lc};
Point(2) = {50, 0, 0, lc};
Point(3) = {50, 10, 0, lc};
Point(4) = {0, 10, 0, lc};
Point(5) = {100, 0, 0, lc};
Point(6) = {100, 10, 0, lc};
Line(1) = {1, 2};
Line(2) = {2, 3};
Line(3) = {3, 4};
Line(4) = {4, 1};
Line(5) = {2, 5};
Line(6) = {5, 6};
Line(7) = {6, 3};
Line Loop(8) = {4, 1, 2, 3};
Plane Surface(9) = {8};
Line Loop(10) = {2, -7, -6, -5};
Plane Surface(11) = {10};
Extrude {0, 0, 20} {
  Surface{9, 11};
}
Physical Surface(101) = {20};
Physical Surface(201) = {46};
Physical Volume(301) = {1};
Physical Volume(302) = {2};
```



- **Mesh 3D** and **set order 2**



- Save as **gmsh.msh** (don't check **any option**)



-
- Run convert script

```
>g2c3.py gmsh.msh sol.inp C3
g2c3.py : INCOMPLETION Version 0.01  2011/Aug/19
----------------------------------
*NSET,NSET=N101

*NSET,NSET=N201

*Element, type=C3D10, ELSET=E301

*Element, type=C3D10, ELSET=E302
```

```
----------------------------------
gmsh.msh -> sol.inp      Done
```

gmsh.msh is input file name , **sol.inp** is output inp file name

g2c3.py gmsh.msh sol.inp C3 ---> C3D10

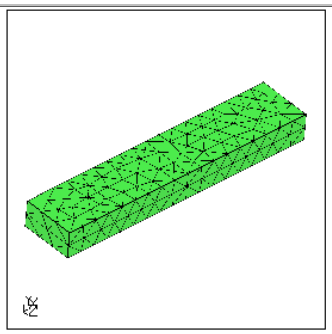
g2c3.py gmsh.msh sol.inp F3 ---> F3D10

g2c3.py gmsh.msh sol.inp F3 R ---> F3D10R (although nonsense)

- Run calculix

```
cgx -c sol.inp
```

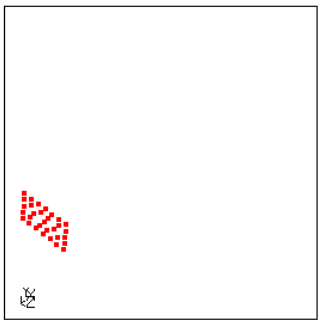
-
- display mesh

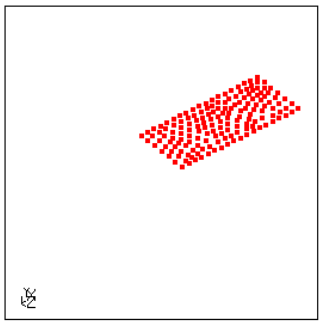
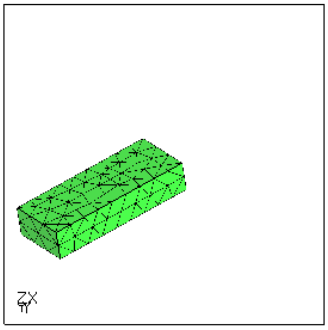
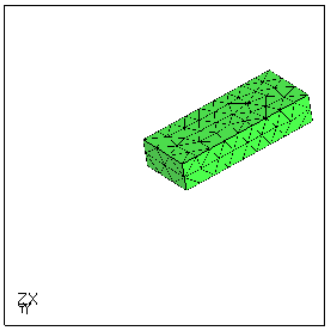
plot elem all **view elem on** **view edge off** **frame**	

- check set

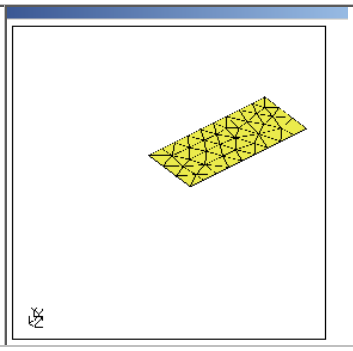
```
prnt set
1    all stat:o n:1595 e:838 f:440 p:0 l:0 s:0 b:0 L:0 S:0 se:0 sh:0
2    N101 stat:c n:31 e:0 f:0 p:0 l:0 s:0 b:0 L:0 S:0 se:0 sh:0
3    N201 stat:c n:165 e:0 f:0 p:0 l:0 s:0 b:0 L:0 S:0 se:0 sh:0
4    E301 stat:c n:0 e:420 f:0 p:0 l:0 s:0 b:0 L:0 S:0 se:0 sh:0
5    +C3D10 stat:c n:0 e:838 f:0 p:0 l:0 s:0 b:0 L:0 S:0 se:0 sh:0
6    E302 stat:c n:0 e:418 f:0 p:0 l:0 s:0 b:0 L:0 S:0 se:0 sh:0
```

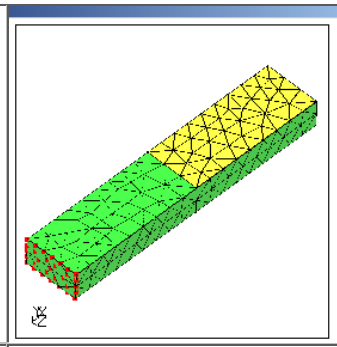
display and check each sets

plot n N101 **(fix nodes)**	

plot n N201 **(load nodes)**	
plot e E301 **(Material1)**	
plot e E302 **(Material2)**	

- then make surface set and element sets form these node sets

comp N201 down **plot f N201** (LOAD is face sets)	

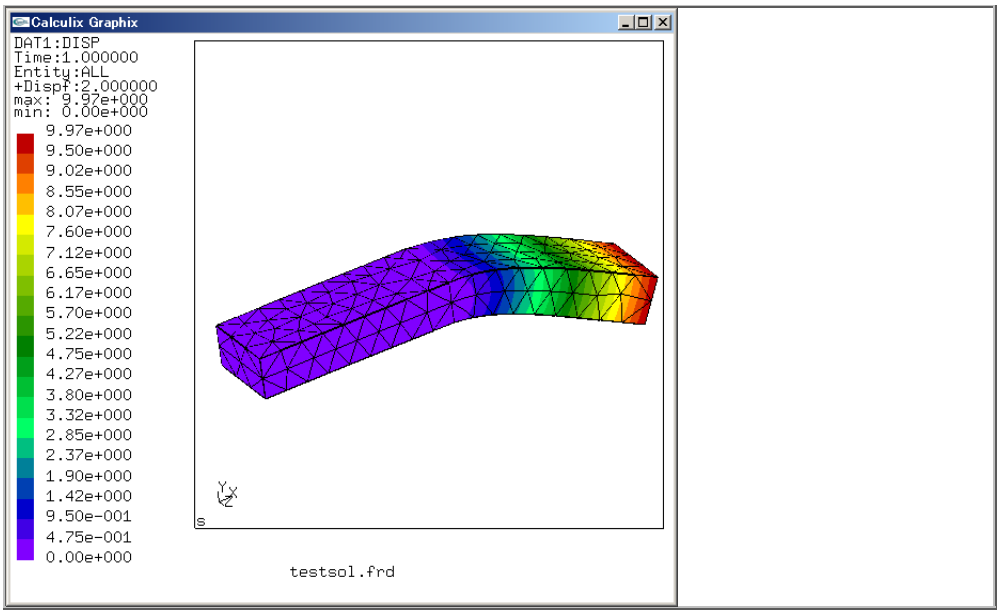
plot e E301 **plus e E302** **plus n N101** **plus f N201** (confirm each sets)	
send all abq names **send N201 abq pres 1**	

- solsol.inp is like this

```
*INCLUDE, INPUT=sol.inp
*INCLUDE, INPUT=all.nam

*MATERIAL, NAME=STEEL
*ELASTIC
 200000  ,  0.3

*MATERIAL, NAME=PLA
*ELASTIC
 1000  ,  0.35

*SOLID SECTION , Elset=E301 , Material=STEEL
*SOLID SECTION , Elset=E302 , Material=PLA
*STEP
*STATIC
*BOUNDARY
N101,1,3,0
*DLOAD
*include,input=N201.dlo
*NODE PRINT,NSET=Nall
U
*EL PRINT,ELSET=Eall
S
*NODE FILE
U
*EL FILE
S
*END STEP
```

ccx solsol

cgx solsol.frd



---

return

///