

Outils logiciels pour la 3D

(page créée le 28https://www.viewstl.com/plugin/examples/simple_example.html fév 2022) ===== Modélisation en ligne ===== [svg2stl](https://svg2stl.com/) <https://svg2stl.com/>

Conversion de dessins vectoriels en fichier STL. [imagerSTL](https://imagerstl.com/) <https://imagerstl.com/>

Conversion d'images en niveau de gris / noir et blanc en fichier STL. [sculptGL](https://stephaneginier.com/sculptgl/) <https://stephaneginier.com/sculptgl/>

Modelage à la souris de modèles 3D (exporte en STL) [3Dslash](https://www.3dslash.net/) <https://www.3dslash.net/>

Modelage sur un bloc «à la minecraft» (l'export en STL nécessite un compte) [BlocksCAD](https://www.blockscad3d.com/) <https://www.blockscad3d.com/>

Langage à bloc (type scratch, blockly) pour la modélisation 3D [Tinkercad](https://www.tinkercad.com/) : cf. page dédiée à [Tinkercad STL viewer](https://stl-viewer.dualbox.com/) <https://stl-viewer.dualbox.com/>

Permet d'afficher des objets STL dans le navigateur (sans installation de logiciel). ===== Banques d'objets ===== [thingiverse](https://www.thingiverse.com/) <https://www.thingiverse.com/>

Plateforme d'objets 3D sous licence libre [printables](https://www.printables.com/) <https://www.printables.com/>

Plateforme d'objets 3D sous licence libre [cults3d](https://cults3d.com/fr/) <https://cults3d.com/fr/>

Objets 3D à imprimer, gratuits ou payants sous diverses licences [grabcad](https://grabcad.com/library) <https://grabcad.com/library>

Objets 3D en différents formats de fichier, pas tous destinés à l'impression. <http://3dwarehouse.sketchup.com/>

<https://www.youmagine.com/>

<https://pinshape.com/>

<https://www.myminifactory.com/>

<https://sketchfab.com/>

<https://www.turbosquid.com/>

===== Logiciels ===== ===== Utilitaires ===== STLviewer (à compiler) : <https://github.com/cravesoft/stlviewer> (voir aussi GMSH ci-dessous) ===== Blender ===== Voir page [Blender](#) ===== openSCAD ===== Voir [OpenSCAD](#) ===== Wings3D ===== Voir [Wings3D](#) ===== FreeCAD ===== <https://www.freecadweb.org/> ===== GMSH ===== **Visualiser des objets STL** et bien plus (édition de mesh). <http://gmsh.info/> Pour afficher les surfaces : Tools → Options → Mesh → Visibility → Surface faces ===== MeshLab ===== <https://www.meshlab.net/> ===== Scripting ===== ===== Dimensions d'un objet STL ===== Ce script en python3 donne les dimensions de la boîte englobante de l'objet :

stldim.py (cliquer pour afficher le code)

[stldim.py](#)

```
#!/usr/bin/python

# Script python pour trouver les dimensions de la boîte englobante d'un objet STL
# nb : les fichiers STL n'ont pas d'unité réelle, mais en général les logiciels qui ont
#       besoin d'une unité, attribuent 1mm pour 1 en unité STL
# source : https://www.reddit.com/r/3Dprinting/comments/7ehlfc/python\_script\_to\_find\_stl\_dimensions/
#
# Testé avec Python 3.5.3 + numpy-stl 2.16.3 @ kirin / Debian 9.5
#
# Requirements: sudo pip install numpy-stl

import math
import stl
from stl import mesh
import numpy

import os
import sys

if len(sys.argv) < 2:
    sys.exit('Usage: %s [stl file]' % sys.argv[0])

if not os.path.exists(sys.argv[1]):
    sys.exit('ERROR: file %s was not found!' % sys.argv[1])

# this stolen from numpy-stl documentation
# https://pypi.python.org/pypi/numpy-stl

# find the max dimensions, so we can know the bounding box, getting the height,
# width, length (because these are the step size)...
def find_mins_maxs(obj):
    minx = maxx = miny = maxy = minz = maxz = None
    for p in obj.points:
        # p contains (x, y, z)
        if minx is None:
            minx = p[stl.Dimension.X]
            maxx = p[stl.Dimension.X]
```

```

        miny = p[stl.Dimension.Y]
        maxy = p[stl.Dimension.Y]
        minz = p[stl.Dimension.Z]
        maxz = p[stl.Dimension.Z]
    else:
        maxx = max(p[stl.Dimension.X], maxx)
        minx = min(p[stl.Dimension.X], minx)
        maxy = max(p[stl.Dimension.Y], maxy)
        miny = min(p[stl.Dimension.Y], miny)
        maxz = max(p[stl.Dimension.Z], maxz)
        minz = min(p[stl.Dimension.Z], minz)
    return minx, maxx, miny, maxy, minz, maxz

main_body = mesh.Mesh.from_file(sys.argv[1])

minx, maxx, miny, maxy, minz, maxz = find_mins_maxs(main_body)

# the logic is easy from there

print ("File:", sys.argv[1])
print ("X:", maxx - minx)
print ("Y:", maxy - miny)
print ("Z:", maxz - minz)

```

==== Autres ==== **stlviewer plugin** pour dokuwiki

Plugin dokuwiki permettant d'afficher les objets STL de manière interactive. * <https://www.dokuwiki.org/plugin:stlviewer> *
 source : <https://github.com/babs/dokuwiki-plugin-stlviewer> * basé sur viewstl : <https://www.viewstl.com/plugin/#intro> Pour
 aller plus loin sur l'intégration d'objets dans le navigateur, voir <https://modelviewer.dev/>

Article extrait de : <http://www.lesporteslogiques.net/wiki/> - **WIKI Les Portes Logiques**
 Adresse : http://www.lesporteslogiques.net/wiki/ressource/logiciel/outils_3d?rev=1750077113
 Article mis à jour: **2025/06/16 14:31**