NAME

stl2pov – converts an STL file to a povray mesh

SYNOPSIS

stl2pov [**-s**] [**-e** *distance*] *filename.stl >outfile.inc*

DESCRIPTION

This program reads an STL (Standard Triangulation Language) file (both in binary and ASCII format) and extracts the vertex data. From these vertices it generates a mesh object suitable for rendering with the **povray**(1) raytracer, and writes it to the standard output. This manual covers version 2.5.0, released on 2011-06-22.

OPTIONS

-s   Generate smooth triangles instead of normal triangles. See the **povray**(1) documentation for details on smooth triangles. If the STL file has relatively few triangles, the generated image can look faceted. Using this option can give a smoother rendered image. Although the program tries to avoid it, using this option may introduce small visual artifacts.

-e*dist*   Set the maximum distance between two points that are assumed to coincide. In previous versions this was fixed at 0.01 units, which is still the default, but you can change it now.

DIAGNOSTICS

The program will abort if one of the following fatal errors is detected:

Could not open *name***.**
          The program tried to open the file *name* using **fopen**(3) but failed. The program cannot continue because it cannot read the input file.

Could not determine the size of *name***.**
          The program tried to determine the size of the file *name* with **fseek**(3) and **ftell**(3) but failed. The program cannot continue because it doesn't know how much memory to allocate for the file.

Could not allocate *num* **bytes of memory for** *name***.**
          The program tried to allocate memory with **malloc**(3) but failed. The program cannot continue because it cannot read the data file into memory.

Could not read *name***.**
          The program tried to read the file *name* with **fread**(3) but this failed. The program cannot continue because it cannot read the input file.

Reading vertex[*num***] failed at "***string***".**
          The progam, using **strtod**(3), expected a floating point number but got *string* instead.

Reading normal[*num***] failed at "***string***".**
          The progam, using **strtod**(3), expected a floating point number but got *string* instead.

The following warnings do not abort the program, they just point out some minor problems with the command-line.

Unrecognized option "*string***" ignored.**
          The program saw an option that it didn't recognize, and ignored it.

Unreadable distance "*string***" ignored.**
          The program could not parse the *distance* argument to the **-s** option.

SEE ALSO

povray(1), **fopen**(3), **fseek**(3), **ftell**(3), **fread**(3), **strtod**(3), /usr/local/share/doc/stl2pov/README, /usr/local/share/doc/stl2pov/LICENSE

BUGS

Using the smoothing option may sometimes result in visual artefacts on the rendered pictures. Setting the allowed distance between coinciding points too large will produce unwanted results.

AUTHOR
  stl2pov was written and is maintained by Roland Smith.
  The latest version of this program is available at: **http://www.xs4all.nl/˜rsmith/software/**

LICENSE
  Copyright (C) 2004--2009,2011 R.F.Smith <rsmith@xs4all.nl>. All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided
  that the following conditions are met:

  1.   Redistributions of source code must retain the above copyright notice, this list of conditions and
          the following disclaimer.

  2.   Redistributions in binary form must reproduce the above copyright notice, this list of conditions
          and the following disclaimer in the documentation and/or other materials provided with the distri-
          bution.

  THIS SOFTWARE IS PROVIDED BY AUTHOR AND CONTRIBUTORS "AS IS" AND ANY
  EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WAR-
  RANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DIS-
  CLAIMED. IN NO EVENT SHALL AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT,
  INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUD-
  ING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
  USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
  THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
  NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE,
  EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.