NAME

robjutil - operate on Wavefront .OBJ file

SYNOPSIS

robjutil [**+/-r**][**+/-v**][**+/-t**][**+/-n**][**+/-m mname**][**+/-g gname**][**-c epsilon**][**-x ’xf spec’**] [**input.obj ..**]

DESCRIPTION

Robjutil reads and manipulates a Wavefront .OBJ file, removing degenerate and duplicate faces. If the *+r* option is specified, a Radiance scene description will be produced on the standard output. If *-r* is given, a modified .OBJ file will be sent to the standard output. This is the default, and is typically more useful, since surface normals are better handled by the *obj2rad(1)* and *obj2mesh(1)* converters. The latter also preserves local texture coordinates for mapping materials.

The *+v* option turns on verbose progress reports.

The *−t* option tells *robjutil* to remove texture coordinates from the input. The *−n* option will cause all surface normals to be removed.

The *−m mname* option tells *robjutil* to remove all faces that use the material *mname*. Multiple *−m* options may be given, each with a single (unique) material name. Alternatively, the *+m mname* option may be used to specify a material one wishes to keep, discarding the rest. Multiple *+m* options will keep all the named materials. The *−m* and *+m* options are mutually exclusive.

Similarly, the *−g gname* option says to remove all faces belonging to the group *gname,* and multiple *−g* options may be given. There is also a *+g* option to save just the named group(s). Object statements are treated the same as group statements in the input, so *+/-g* option can also be applied to object names. Note that an object name will be overridden by a group name that follows it in the input, and vice versa.

The *−c epsilon* option tells *robjutil* to coalesce vertices that are within the given epsilon. This is useful to improve the connectivity of models that were produced in a sloppy conversion process from independent polygons. When producing a Radiance file via the *+r* option, there will be no savings in the output size, but there can be a benefit in reducing the number of "cracks" in the resulting geometry, at least for triangle mesh input. *Robjutil* does not attempt to break up non-planar polygons into triangles, which is usually not a problem for .OBJ output, but may be for Radiance scene descriptions (+r).

The *−x* option may be used to specify a final transform to apply to all coordinates in the file before output. This specification must be contained in a single quoted word and match the standard arguments used by *xform(1),* except that the *−a* option is not supported for creating arrays of objects. Also, any scaling factors do not affect the epsilon value given for coalescing vertices, as this operation is applied prior to the transform, even if *−c* appears later in the command line.

EXAMPLES

To remove degenerate and duplicate faces from "orig.obj" and write to "final.rad":

```
robjutil +r orig.obj > final.rad
```

To convert the same file, preserving surface normal interpolation:

```
robjutil orig.obj | obj2rad > withnormals.rad
```

To extract only the groups "cat" and "bell" from the file "kitty.obj" and write to a Radiance triangle mesh using materials from "kitty.mat":

```
robjutil +g cat +g bell kitty.obj | obj2mesh -a kitty.mat > kitty.rtm
```

To rotate the scene 90 degrees about the Y-axis and scale by 10:

```
robjutil -x "-ry 90 -s 10" orig.obj > transformed.obj
```

NOTES

The .OBJ output of *robjutil* uses relative indices, so files it produces may be concatenated on the input of other tools such as *obj2rad* and *obj2mesh* by piping from *cat(1)* or similar. This offers a convenient way to separate geometry from a single .OBJ input, manipulate it in some way, then recombine during conversion.

Unlike *obj2rad* and *obj2mesh, robjutil* with the *+r* option will not substitute the group name for a material id if no *usemtl* statements are found on the input. Instead, it will assign each surface the material *DEFAULT_MATERIAL*. Since it is possible to use the *+/-g* option to select groups to include or exclude on output, it is still possible to assign materials by group manually.

AUTHOR

Greg Ward

SEE ALSO

cat(1), obj2mesh(1), obj2rad(1) oconv(1), xform(1)