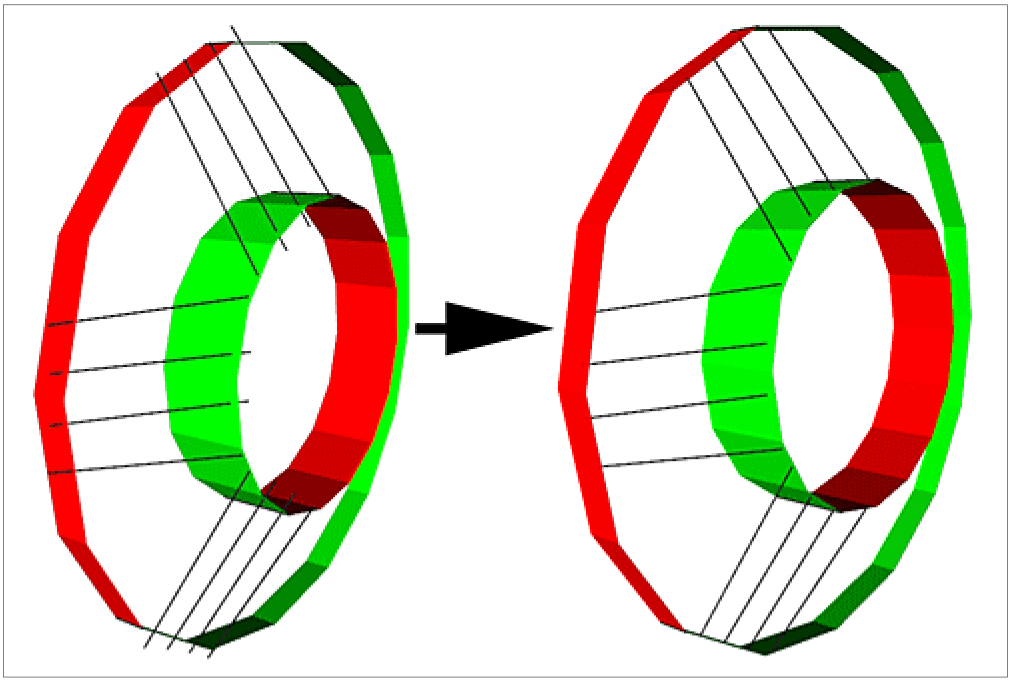
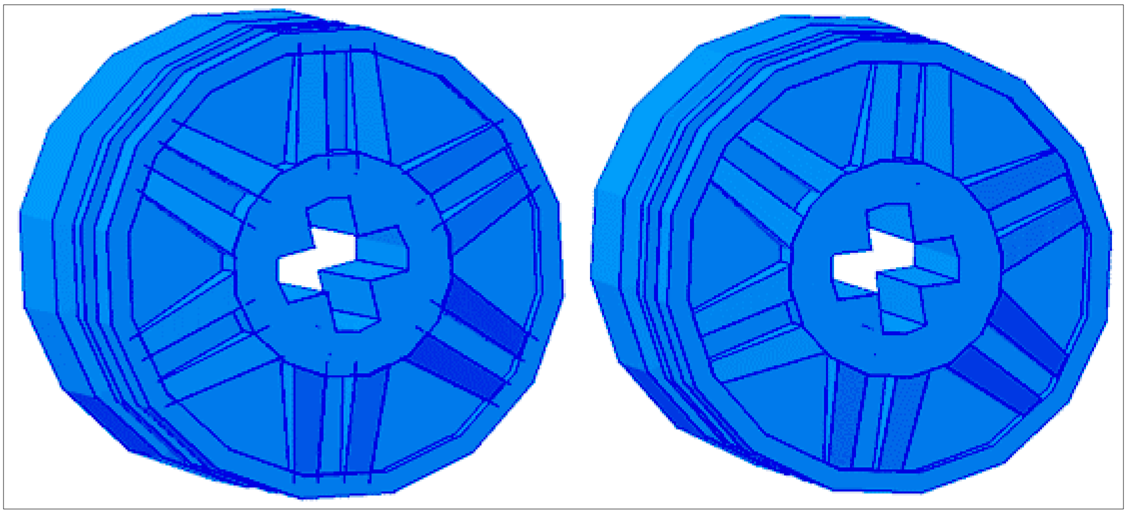
Linetrim, Line Trimmer



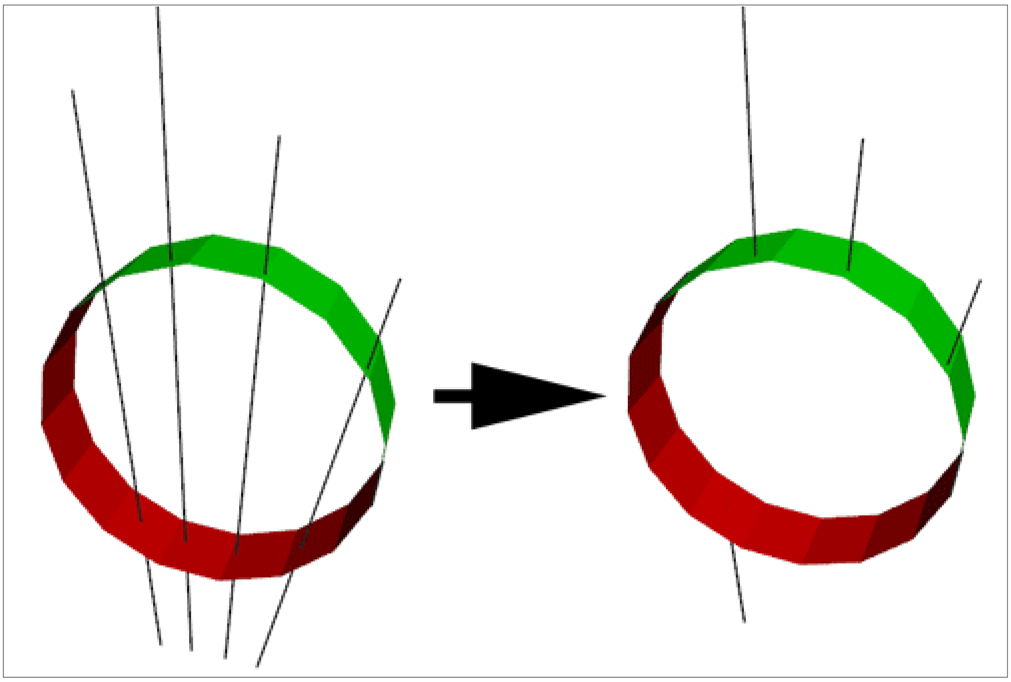
Linetrim cuts lines going through triangles or quads. Sets of lines and triangles are provided to the utility as separate LDraw files. A third file containing the cropped lines is created. Optionnally, a fourth file containing everything but the lines of the first file may be created.

It is a simple console application, source code is provided below to anyone willing to integrate it in a more palatable interface.



This image shows the purpose of line trimmer: even though the lines highlighting wheel spokes are placed below rim surface, they bleed through it (left image) with some renderers (here MLCad). Cutting the lines at their exact length provides a clean result (right image).

Caveat: LDraw parts are built from flat surfaces and have no clear inside. The only information available is local orientation of facets winding. **Linetrim** will remove line portion which is on CCW side of intersecting polygon (that is to say the side of triangle or quad where vertex order is seen counter clock wise). A side effect is that **Linetrim** is effective only when ends of lines are to be trimmed, it will fail when the center part of a line should be removed, as shown in the image below. Intended result could have be obtained by splitting lines in half prior to using **Linetrim.**



Download

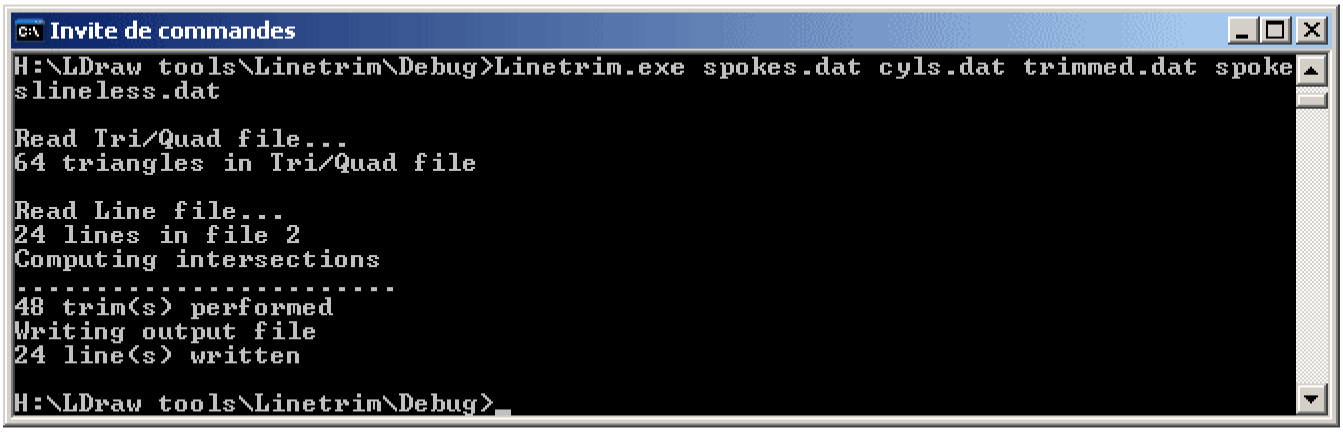
- **Linetrim** program for Windows
- **Linetrim** source files (Visual C++ 6.0)
- Sample files
  - spokes.dat contains lines to trim outlining a few quads.
  - cyls.dat contains triangles/quads against which lines will be trimmed
  - trimmed.dat is the resulting files with lines cropped, output of **Linetrim.**
  - spokeslineless.dat is the optional file created by **Linetrim**, equal to spokes.dat stripped from all lines.
  - spokestrimmed.dat is the combination of the two previous files
- Or get the full package.

Usage

- Prepare the input LDraw files. **Linetrim** calculates trims lines against triangles or quads. Other LDraw line types are ignored. Parts or primitives must be inlined down to triangles and quads (LDDesignPad does this very conveniently).
- While **Linetrim** does not require input files to be BFC certified, it does need the right winding of all trimming triangle and quads. The simplest way is to add a 0 BFC CERTIFY CW at the beginning of the trimming file and check using LDview with BFC option + red and green faces coloring. Lines on red side of polygons will be trimmed. If some polygons have the red face on the wrong side, use LDDesignPad to reverse windings of these polygons.

- Launch a command prompt
- Type the command line: Linetrim LdrawTriQuadFile LdrawLineFile LdrawTrimmedLineFileOut [LdrawLineRemovedFile]. **Linetrim** will create LdrawTrimmedLineFileOut containing the trimmed lines. Note that if file LdrawTrimmedLineFileOut exists it will be overwritten without warning. Optionally, LdrawLineRemovedFile with the same contents as LdrawLineFile with all lines removed will be created.
- **Linetrim** output file with 6 digits after decimal point, this precision is excessive for most usages and values should be rounded. Here again, LDDesignPad does that very well.

Here is a screen shot of a sample run:



```
H:\LDraw tools\Linetrim\Debug>Linetrim.exe spokes.dat cyls.dat trimmed.dat spoke
slineless.dat

Read Tri/Quad file...
64 triangles in Tri/Quad file

Read Line file...
24 lines in file 2
Computing intersections
........................
48 trim(s) performed
Writing output file
24 line(s) written

H:\LDraw tools\Linetrim\Debug>_
```

How Linetrim works

- Both input files are read and parsed. Quads are split into 2 triangles. Quads with bad winding ("bow-tie") will not be properly processed. Lines from the first file are stored in an array, triangles from second files are stored in another.
- All lines from the first set are tested for intersection with all triangle of the second set. If an intersection is found, the end of the line that is on CCW side of the triangle is replaced with this intersection. The line to triangle intersection program is adapted from www.softsurfer.com triangle intersection code.
- Output file is created.