RI

Conversion program

COLLABORATORS

	TITLE : RI	

ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY	Conversion program	October 9, 2022	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

RI

1.1 Overview of RI Shapes Lib V2.5

```
                                  Overview

The RIShapesLib is a library designed to fill the gaps left by inadequacies
in the Acid shapes system. The library introduces a new shapes file format,
called 'LEShapes'. These files can store compressed shapes and can also store
palette information.
```

1.2 RI Shapes Lib V2.5

```
Statement/Function : CludgeShapes
------------------------------------------------------------------------------
Modes  : Amiga/Blitz
Syntax : [success]=CludgeShapes(shape#,numshapes,address)

  This allows the creation of shapes through INCBIN statements. It allocates
chip memory for each shape and copies the data into this. It does the same
as LoadShapes except it grabs shapes from memory.

e.g.

  suc=BLoad("myshapes",0)
  suc=CludgeShapes(0,50,Start(0))
  MouseWait
  End
```

1.3 RI Shapes Lib V2.5

```
Statement : LESaveShapes
------------------------------------------------------------------------------
Modes  :  Amiga
Syntax : LESaveShapes shapenum#,shapenum#,filename$[,palette#]

  This saves shapes and a palette in an IFF type file (not an picture). The
```

palette can be saved along with the shape file. If no palette is passed or the passed palette is empty, no palette data will be saved.

1.4 RI Shapes Lib V2.5

```
Statement : LELoadShapes
-------------------------------------------------------------------------------
Modes  :  Amiga
Syntax : LELoadShapes shapenum#,[shapenum#,]filename$[,palette#]

  This attempts to load shapes from an LEShapes file, if there is a palette
saved in the shape file this will be loaded into the specified palette. You
can miss out an upper shape limit or a palette number or both.

IMPORTANT NOTE: Due to a limitation of the Blitz library system you cannot
                use the following form of the command:

    LELoadShapes 0,"shapesfile",0

You will get a "Can't convert types error". To get around this simply do:

    LELoadShapes 0,Maximum Shape,"shapesfile",0
```

1.5 RI Shapes Lib V2.5

```
Statement : LECludgeShapes
-------------------------------------------------------------------------------
Modes  : Amiga/Blitz
Syntax : LECludgeShapes shape#,shape#,address,palette#[,copyregardless]

    This command decodes a shape file (that may have a palette) saved by
LESaveShapes. It can cope with compresses or uncompressed data, and conforms
with Acids standards for indicating that a shape has been cludged. If you
wish to decompress as many shapes as are in the shapes file you may do:

  LECludgeShapes shape#,Maximum Shapes-1,address,palette#

This will decode all the shapes in the file with NO OVERRUN like ACID's
library.

IMPORTANT NOTE: There are some considerations with where in memory you want
                to place your LEShapes file to be Cludged. If you're shapes
                file is:

      1) Cached to CHIP MEM and
      2) UNCOMPRESSED

Then Cludge shapes will not create a second copy of the shapes data unless
the <copyregardless> parameter is set to -1.
```

1.6 RI Shapes Lib V2.5

```
Statement: LECompressShapes
------------------------------------------------------------------------------
Modes  : Amiga/Blitz
Syntax : LECompressShapes Boolean
```

By default LESaveShapes compresses shapes in a shape file. The compressor is quite intelligent in that if the compressed shape is larger (oxymoron any one?) than the orginal (this can happen, honest) it saves the full data from the old shape.

If you wish to turn shape compression on or off, call LECompressShapes with the correct parameter.

Below is a small table comparing the same shape files stored in 3 different ways. For very small shape files (1-3 shapes) you may find turning compression off results in the saving of a few bytes. The bigger the file, the larger the saving.

Shapes	Acid SaveShapes	LESaveShapes NO COMPRESS	LESaveShapes COMPRESSION
400	76912 bytes	68940 bytes	54091 bytes
223	43008 bytes	38576 bytes	35646 bytes

1.7 RI Shapes Lib V2.5

```
                .-------------------------------------------------------------------------
|                              RI Shapes Lib V2.5                                |
`-------------------------------------------------------------------------------'
```


Undocumented commands added by Toby Zuijdveld 02/03/1999
mailto: hotcakes@abacus.net.au

Overview

Command Index

CludgeShapes

LECludgeShapes

LECompressShapes

LELoadShapes

LESaveShapes

```
                                SafeToBlit       shape,x,y
                    LEInitShape      shape#,width,height,depth
                    LEScanShapes     startshape#,endshape#
                    LEMaxShapeWidth
                    LEMaxShapeHeight
                    LEMaxShapeDepth


          Examples
                                Main Document
                     Library Index
```

1.8 Example Programs

```
                          Example Programs

EXAMPLE 1 - An Acid Shapes -> LEShapes Library :

   Load Example 1
   Compile It!
```