NAME

ico2 – animate an icosahedron or other polyhedron

SYNOPSIS

ico2 [**-r**] [**-d** *pattern*] [**-i]** [**-dbl**] [**-faces**] [**-noedges**] [**-sleep** *n*] [**-obj** *object*] [**-objhelp**] [**-colors** *color-list*]

DESCRIPTION

Ico2 displays a wire-frame rotating polyhedron, with hidden lines removed, or a solid-fill polyhedron with hidden faces removed. There are a number of different polyhedra available; adding a new polyhedron to the program is quite simple.

OPTIONS

-r Display on the root window instead of creating a new window.

-d *pattern*
Specify a bit pattern for drawing dashed lines for wire frames.

-i Use inverted colors for wire frames.

-dbl Use double buffering on the display. This works for either wire frame or solid fill drawings. For solid fill drawings, using this switch results in substantially smoother movement. Note that this requires twice as many bit planes as without double buffering. Since some colors are typically allocated by other programs, most eight-bit-plane displays will probably be limited to eight colors when using double buffering.

-faces Draw filled faces instead of wire frames.

-noedges
Don't draw the wire frames. Typically used only when -faces is used.

-sleep *n*
Sleep n seconds between each move of the object.

-obj *object*
Specify what object to draw. If no object is specified, an icosahedron is drawn.

-objhelp
Print out a list of the available objects, along with information about each object.

-colors *colorlist*
Specify what colors should be used to draw the filled faces of the object. If less colors than faces are given, the colors are reused.

ADDING POLYHEDRA

If you have the source to ico, it is very easy to add more polyhedra. Each polyhedron is defined in an include file by the name of obj*name*.h, where *name* is something related to the name of the polyhedron. The format of the include file is defined in the file polyinfo.h. Look at the file objcube.h to see what the exact format of an obj*name*.h file should be, then create your obj*name*.h file in that format.

After making the new obj*name*.h file (or copying in a new one from elsewhere), simply do a 'make depend'. This will recreate the file allobjs.h, which lists all of the obj*name*.h files. Doing a 'make' after this will rebuild ico with the new object information.

BUGS

If *-dbl* or *-colors* is specified on a monochrome screen, or if the number of bit planes needed is more than what is available, the server hangs (this is on a Sun 3/60, patches up to 80 applied).

A separate color cell is allocated for each name in the *-colors* list, even when the same name may be specified twice.

SEE ALSO

ico(6)