

NAME

rhcopy - copy ray information to/from a holodeck

SYNOPSIS

```
rhcopy dest_holo [ -u ][ -d ] -h src_holo ..
or
rhcopy dest_holo [ -u ][ -d ] -p src_hdr src_dpt ..
or
rhcopy dest_holo [ -f{alfld} ][ -u ][ -d ] -i[odpILv]
or
rhcopy src_holo [ -f{alfld} ] -o[odpILv]
```

DESCRIPTION

Rhcopy typically adds ray sample data to an existing holodeck. In the first form, the ray samples are taken from one or more holodeck files given after the *-h* option. In the second form, the ray samples are taken from one or more RADIANCE picture files and their corresponding depth buffers, which must be paired on the command line after the *-p* option. In the third form, ray samples are read from the standard input based on parameters specified by the *-i* option, which should match (for example) the corresponding *-o* option of an *rtrace(1)* command. Note that a header is not expected on input, and some parameters are required for correct operation. A *-f* option may be used to specify raw rather than ASCII data.

The final form reads rays from the given holodeck and writes them to the standard output according to the *-o* specification.

The *-u* option turns on duplicate ray checking. In some cases, the same ray may already exist in the destination holodeck, and it would be redundant to add it. By default, *rhcopy* does not check for duplicates, because it takes extra time, and in many invocations is not necessary, as when copying into an empty holodeck.

The *-d* option turns off depth checking. Normally, *rhcopy* checks the OBSTRUCTIONS variable of the destination holodeck, and if it is set to True, makes sure that all contributing rays start outside each section. If OBSTRUCTIONS is set to False, then *rhcopy* makes sure that any contributing rays end outside each section. If OBSTRUCTIONS is not set, then this option has no effect. (See the *rhola(1)* man page for a definition of the OBSTRUCTIONS variable.)

Rcopy cannot be used to create a holodeck -- use *rhola* for this purpose. For example, to create an empty holodeck, run *rhola* without either the *-n* or *-o* option. Whatever variables are set by *rhola* when the new holodeck is created are the ones that will affect later rendering or viewing. Since the ray sample data may be taken from any source, *rhola* and *rhcopy* may be used together to change certain unalterable holodeck parameters, such as the section grid geometry.

EXAMPLE

To take data from an existing holodeck after changing the section grid:

```
rhola new.hdk new.hif
rhcopy new.hdk -h old.hdk
```

A much less efficient way to copy the same data:

```
rhcopy old.hdk -oodlv | rhcopy new.hdk -iodlv
```

To add ray samples from two pictures to the new holodeck:

```
rhcopy new.hdk -p view1.hdr view1.zbf view2.hdr view2.zbf
```

NOTES

Rhcopy attempts to place the beams in the holodeck in a good order for quick access, but if the data comes from multiple sources, the results may not be optimal. For large holodecks, it is sometimes useful to run the *rhoptimize(1)* program once all runs of *rhcopy* are through.

AUTHOR

Greg Ward

ACKNOWLEDGMENT

This work was supported by Silicon Graphics, Inc.

SEE ALSO

getinfo(1), pfilt(1), rcode_depth(1), rcontrib(1), rhinfo(1), rholo(1), rhoptimize(1), rpict(1), rtrace(1),