

NAME

peiCof - perform operations on COF file sets

SYNOPSIS

peiCof [*options*] [*dir1 dir2 ...*]

DESCRIPTION

peiCof examines the COF file sets in the named directories, verifies their compliance with the COF specification, and optionally performs image extraction and other operations. If no directory is named, the current directory is used.

Default behavior is to examine each file set in detail, print progress reports, and generate error and warning messages whenever compliance violations are detected, but to abort the examination of any file or file set for which the error count is too high. The last line of the report for each file set contains the word **PASS** if the file set is in compliance, and **FAIL** otherwise.

The program attempts to interpret file sets even in the presence of significant violations and to provide error messages that assist in fixing problems in COF generation software.

OPTIONS

In case of conflicting options, those that appear later on the command line have precedence.

COF Version Option

peiCof can check for compliance with version 1.1, 1.2, or 1.3 of the COF specification. The target version can be set explicitly using the **-version** option; otherwise, the default behavior is to use the version that is given as the value of the keyword **COFVersion** in section **[General]** in the file **HEADER.TXT** of each file set, if that keyword is found and is valid. Otherwise, the latest version (currently 1.3) is used.

-version *vers* check against version *vers*, which may be **1.1**, **1.2**, or **1.3**.

Abort Control Options

Normally, **peiCof** aborts examination of a file whenever the number of errors encountered in the file exceeds a limit; there is also a similar limit for file sets. These limits can be set with the following options:

-flim *count* set the maximum number of errors tolerated in each file before aborting examination of the file and attempting to continue with the next file. *count* can be a non-negative number or the word **none**, in which case no such limit is enforced (default: 40).

-slim *count* set the maximum number of errors tolerated in each file set before aborting examination of the file set and attempting to continue with the next file set. *count* can be a non-negative number or the word **none**, in which case no such limit is enforced (default: 200).

-nolim remove limits on errors; equivalent to **-flim none -slim none**.

Output Options

Normally, **peiCof** prints progress reports, error messages, warning messages, and a results summary, giving error and warning counts, on the standard output. This behavior can be modified with the following options:

-noprog suppress the printing of progress reports

-noerr suppress the printing of error and warning messages; errors continue to be counted, but warnings do not.

-quiet suppress the printing of progress, error, warning, and results messages; this option implies **-noprog** and **-noerr**.

-boolean reduce the output to a single line of text that is "0" if every COF file set is in compliance, and "1" otherwise; this allows the output of **peiCof** to be used in a boolean expression. This option implies **-quiet**.

-wlim *count* set the maximum number of warnings that are printed for each file. *count* can be a non-negative number or the word **none**, in which case no such limit is enforced

(default: 20).

-dump also print a tabulation of the COF data structures that were successfully parsed.

Image Extraction Options

These options allow images embedded in the COF images files (.IMG files) in the first named directory to be extracted, producing a set of TIFF and/or IOCA files in a specified output directory. For purposes of image extraction, an image is referred to by its suite number, *suite*, which need not have leading zeros, its item number, *item*, and its view number, *view*, where item numbers are counted starting at the beginning of the .IMG file with 1, and view numbers are counted within each item starting with 1. Alternatively, the word **all** can appear when specifying the suite, item, and/or view number, in which case all qualified images are extracted. Extracted images have names of the form

COF*suite.item.view.suffix*,

where *suffix* is either **tif** or **ioc**.

-extract *suite item view dir* extract the specified image(s) and deposit the resulting TIFF and/or IOCA file(s) in the specified directory; *suite*, *item*, and *view* are numbers greater than zero or the word **all**.

-extall *dir* shorthand for **-extract all all all** *dir*

Other Options

-help print software version information and a summary of this manual page, and do nothing else.

EXAMPLES

The COF file set in the current directory can be checked with the command:

```
peiCof
```

The next example checks every file set in the current directory, suppressing warning messages and checking every file regardless of the number of errors found earlier in its file set, but aborting the check of files with more than 100 errors:

```
peiCof -slim none -flim 100 -wlim 0 *
```

This example produces the maximum output for the file set in the current directory:

```
peiCof -dump -nolim -wlim none
```

To see only a dump of the COF data structures of the file set in the directory **/data/cof/19990311**, use the command:

```
peiCof -dump -nolim -quiet /data/cof/19990311
```

The use of **-nolim** prevents the error count from aborting the operation, assuring that the dump will include all of the structures that the program is able to parse.

This example shows how to use **peiCof** within a C shell script to branch on the result of the COF verification:

```
if ('peiCof -boolean $1' == 0) then
    echo test passes
else
    echo test fails
endif
```

REFERENCES

This program and other COF resources are available from Picture Elements, Inc., by anonymous FTP at **ftp.picturel.com/pub/cof**. That directory holds several versions of the COF specification in files with names of the form **cofXX.pdf**, where *XX* is the version number, and this manual page is available in the file **coftest/peiCof/peiCof.pdf**. The program itself is written in ANSI C and is available as source code in the file **coftest/peiCof/peiCof-X.XXX.tar.gz**, where *X.XXX* is the program version number. Bug reports and suggestions are invited and should be sent to **rick@picturel.com**.

