NAME
>   filter – cups file conversion filter interface

SYNOPSIS
>   **filter** *job user title num-copies options* [*filename*]
>
>   **#include <cups/cups.h>**
>
>   ssize_t **cupsBackChannelRead**(**char ***buffer*, **size_t** *bytes*,
>   **double** *timeout*);
>
>   cups_sc_status_t **cupsSideChannelDoRequest**(**cups_sc_command_t** *command*,
>   **char ***data*, **int ***datalen*,
>   **double** *timeout*);
>
>   **#include <cups/ppd.h>**
>
>   const char ***cupsGetOption**(**const char ***name*, **int** *num_options*,
>   **cups_option_t ***options*);
>
>   int **cupsMarkOptions**(**ppd_file_t ***ppd*, **int** *num_options*,
>   **cups_option_t ***options*);
>
>   int **cupsParseOptions**(**const char ***arg*, **int** *num_options*,
>   **cups_option_t ****options*);
>
>   ppd_choice_t ***ppdFindMarkedChoice**(**ppd_file_t ***ppd*, **const char ***keyword*);
>
>   void **ppdMarkDefaults**(**ppd_file_t ***ppd*);
>
>   ppd_file_t ***ppdOpenFile**(**const char ***filename*);

DESCRIPTION
>   The CUPS filter interface provides a standard method for adding support for new document types or print-
>   ers to CUPS. Each filter is capable of converting from one or more input formats to another format that can
>   either be printed directly or piped into another filter to get it to a printable format.
>
>   Filters **MUST** be capable of reading from a filename on the command-line or from the standard input,
>   copying the standard input to a temporary file as required by the file format. All output **MUST** be sent to
>   the standard output. Filters **MUST NOT** attempt to communicate directly with the printer, other processes,
>   or other services.
>
>   The command name (*argv[0]*) is set to the name of the destination printer but is also available in the
>   **PRINTER** *environment variable.*

OPTIONS
>   Options are passed in *argv[5]* and are encoded from the corresponding IPP attributes used when the job
>   was submitted. Use the **cupsParseOptions**() function to load the options into a **cups_option_t** array and
>   the **cupsGetOption**() function to get the value of a specific attribute. Be careful to look for common
>   aliases of IPP attributes such as "landscape" for the IPP "orientation-requested" attribute.
>
>   Options passed on the command-line typically do not include the default choices the printer's PPD file. Use
>   the **ppdMarkDefaults**() and **cupsMarkOptions**() functions in the CUPS library to apply the options to the
>   PPD defaults and map any IPP attributes to the corresponding PPD options. Use **ppdFindMarked-
>   Choice**() to get the user-selected choice for a PPD option. For example, a filter might use the following
>   code to determine the current value of the **Duplex** PPD option:
>
>   ppd_file_t *ppd = ppdOpenFile(getenv("PPD"));

```
cups_option_t *options = NULL;
int num_options = cupsParseOptions(argv[5], 0, &options);

ppdMarkDefaults(ppd);
cupsMarkOptions(ppd, num_options, options);

ppd_choice_t *choice = ppdFindMarkedChoice(ppd, "Duplex");
```

Raster filters should use option choices set through the raster page header, as those reflect the options in effect for a given page. Options specified on the command-line determine the default values for the entire job, which can be overridden on a per-page basis.

LOG MESSAGES

Messages sent to the standard error are generally stored in the printer's "printer-state-message" attribute and the current **ErrorLog** file. Each line begins with a standard prefix:

ALERT: *message*
> Sets the "printer-state-message" attribute and adds the specified message to the current **ErrorLog** using the "alert" log level.

ATTR: *attribute=value* [*... attribute=value*]
> Sets the named job or printer attribute(s). The following job attributes can be set: "job-media-progress". The following printer attributes can be set: "auth-info-required", "marker-colors", "marker-high-levels", "marker-levels", "marker-low-levels", "marker-message", "marker-names", "marker-types", "printer-alert", and "printer-alert-description".

CRIT: *message*
> Sets the "printer-state-message" attribute and adds the specified message to the current **ErrorLog** using the "critical" log level.

DEBUG: *message*
> Adds the specified message to the current **ErrorLog** using the "debug" log level. **DEBUG** messages are never stored in the "printer-state-message" attribute.

DEBUG2: *message*
> Adds the specified message to the current **ErrorLog** using the "debug2" log level. **DEBUG2** messages are never stored in the "printer-state-message" attribute.

EMERG: *message*
> Sets the "printer-state-message" attribute and adds the specified message to the current **ErrorLog** using the "emergency" log level.

ERROR: *message*
> Sets the "printer-state-message" attribute and adds the specified message to the current **ErrorLog** using the "error" log level.

INFO: *message*
> Sets the "printer-state-message" attribute. If the current **LogLevel** is set to "debug2", also adds the specified message to the current **ErrorLog** using the "info" log level.

NOTICE: *message*
> Sets the "printer-state-message" attribute and adds the specified message to the current **ErrorLog** using the "notice" log level.

PAGE: *page-number #-copies*

PAGE: *total #-pages*
> Adds an entry to the current **PageLog**. The first form adds *#-copies* to the "job-media-sheets-completed" attribute. The second form sets the "job-media-sheets-completed" attribute to *#-pages*.

PPD: *Keyword=Value* [*... KeywordN=Value*]
> Sets the named keywords in the printer's PPD file. This is typically used to update default option keywords such as **DefaultPageSize** and the various installable options in the PPD file.

STATE: *printer-state-reason* [*... printer-state-reason*]

STATE: + *printer-state-reason* [*... printer-state-reason*]

STATE: – *printer-state-reason* [*... printer-state-reason*]
>   Sets, adds, or removes "printer-state-reason" keywords for the current queue. Typically this is used to indicate media, ink, and toner conditions on a printer.

WARNING: *message*
>   Sets the "printer-state-message" attribute and adds the specified message to the current **ErrorLog** using the "warning" log level.

ENVIRONMENT VARIABLES
The following environment variables are defined by the CUPS server when executing the filter:

CHARSET
>   The default text character set, typically "utf-8".

CLASS
>   When a job is submitted to a printer class, contains the name of the destination printer class. Otherwise this environment variable will not be set.

CONTENT_TYPE
>   The MIME media type associated with the submitted job file, for example "application/postscript".

CUPS_CACHEDIR
>   The directory where semi-persistent cache files can be found and stored.

CUPS_DATADIR
>   The directory where data files can be found.

CUPS_FILETYPE
>   The type of file being printed: "job-sheet" for a banner page and "document" for a regular print file.

CUPS_MAX_MESSAGE
>   The maximum size of a message sent to *stderr*, including any leading prefix and the trailing newline.

CUPS_SERVERROOT
>   The root directory of the server.

FINAL_CONTENT_TYPE
>   The MIME media type associated with the output destined for the printer, for example "application/vnd.cups-postscript".

LANG
>   The default language locale (typically C or en).

PATH
>   The standard execution path for external programs that may be run by the filter.

PPD The full pathname of the PostScript Printer Description (PPD) file for this printer.

PRINTER
>   The name of the printer.

RIP_CACHE
>   The recommended amount of memory to use for Raster Image Processors (RIPs).

SOFTWARE
>   The name and version number of the server (typically CUPS/*major.minor*).

TZ The timezone of the server.

USER
>   The user executing the filter, typically "lp" or "root"; consult the *cups-files.conf* file for the current setting.

CONFORMING TO

While the filter interface is compatible with System V interface scripts, CUPS does not support System V interface scripts.

NOTES

CUPS filters are not meant to be run directly by the user. Aside from the legacy System V interface issues (*argv[0]* is the printer name), CUPS filters also expect specific environment variables and file descriptors, and typically run in a user session that (on macOS) has additional restrictions that affect how it runs. Unless you are a developer and know what you are doing, please do not run filters directly. Instead, use the **cupsfilter**(8) program to use the appropriate filters to do the conversions you need.

SEE ALSO

backend(7), **cups**(1), **cups-files.conf**(5), **cupsd**(8), **cupsfilter**(8),
CUPS Online Help (http://localhost:631/help)

COPYRIGHT

Copyright © 2007-2017 by Apple Inc.