#### ISO/IEC JTC 1/SC 22/OWG Linux N0005

Date: 5 October 2017

ISO/IEC 23360-1-5

Edition 1

ISO/IEC JTC 1/SC 22/OWG LSB

Secretariat: ANSI

Information Technology — Operating systems – Linux Standard Base imaging specification

Document type: International standard

Document subtype: if applicable

Document stage: (10) development stage

Document language: E

*Élément introductif — Élément principal — Partie n: Titre de la partie*

Warning

This document is not an ISO International Standard. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an International Standard.

Recipients of this draft are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

**Copyright notice**

This ISO document is a working draft or committee draft and is copyright-protected by ISO. While the reproduction of working drafts or committee drafts in any form for use by participants in the ISO standards development process is permitted without prior permission from ISO, neither this document nor any extract from it may be reproduced, stored or transmitted in any form for any other purpose without prior written permission from ISO.

Requests for permission to reproduce this document for the purpose of selling it should be addressed as shown below or to ISO’s member body in the country of the requester:

*ISO copyright office*

*Case postale 56, CH-1211 Geneva 20*

*Tel. + 41 22 749 01 11*

*Fax + 41 22 749 09 47*

*E-mail copyright@iso.org*

*Web www.iso.org*

Reproduction for sales purposes may be subject to royalty payments or a licensing agreement.

Violators may be prosecuted.

**Linux Foundation Copyright**

This specification is published under the terms of the GNU Free Documentation License, Version 1.1, March 2000

Copyright (C) 2000 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

See Annex A for the complete GNU Free Documentation License. GNU

**Linux Standard Base Imaging Specification**

LSB Imaging 5.0

Copyright © 2015 Linux Foundation

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1; with no Invariant Sections, with no Front-Cover Texts, and with no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

Portions of the text may be copyrighted by the following parties:

• The Regents of the University of California

• Free Software Foundation

• Ian F. Darwin

• Paul Vixie

• BSDI (now Wind River)

• Jean-loup Gailly and Mark Adler

• Massachusetts Institute of Technology

• Apple Inc.

• Easy Software Products

• artofcode LLC

• Till Kamppeter

• Manfred Wassman

• Python Software Foundation

These excerpts are being used in accordance with their respective licenses.

Linux is the registered trademark of Linus Torvalds in the U.S. and other countries.

UNIX is a registered trademark of The Open Group.

LSB is a trademark of the Linux Foundation in the United States and other countries.

AMD is a trademark of Advanced Micro Devices, Inc.

Intel and Itanium are registered trademarks and Intel386 is a trademark of Intel Corporation.

PowerPC is a registered trademark and PowerPC Architecture is a trademark of the IBM Corporation.

S/390 is a registered trademark of the IBM Corporation.

OpenGL is a registered trademark of Silicon Graphics, Inc.

PAM documentation is Copyright (C) Andrew G. Morgan 1996-9. All rights reserved. Used under the following conditions:

 1. Redistributions of source code must retain the above copyright notice, and the entire permission notice in its entirety, including the disclaimer of warranties.

 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

 3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

# **Contents**

[**Contents** iv](#_Toc495000938)

[**List of Tables** vi](#_Toc495000939)

[**Foreword** vii](#_Toc495000940)

[**Status of this Document** viii](#_Toc495000941)

[**Introduction** ix](#_Toc495000942)

[**I Introductory Elements** 1](#_Toc495000943)

[**1 Scope** 1](#_Toc495000944)

[**1.1 General** 1](#_Toc495000945)

[**1.2 Module Specific Scope** 1](#_Toc495000946)

[**2 Normative References** 2](#_Toc495000947)

[**3 Requirements** 3](#_Toc495000948)

[**3.1 Relevant Libraries** 3](#_Toc495000949)

[**4 Terms and Definitions** 4](#_Toc495000950)

[**5 Documentation Conventions** 5](#_Toc495000951)

[**6 PPD Format Extensions** 6](#_Toc495000952)

[**II Printing Libraries** 9](#_Toc495000953)

[**7 Libraries** 10](#_Toc495000954)

[**7.1 Interfaces for libcups** 10](#_Toc495000955)

[**7.1.1 CUPS Convenience ABI** 10](#_Toc495000956)

[**7.2 Data Definitions for libcups** 12](#_Toc495000957)

[**7.2.1 cups/cups.h** 12](#_Toc495000958)

[**7.2.2 cups/http.h** 16](#_Toc495000959)

[**7.2.3 cups/ipp.h** 18](#_Toc495000960)

[**7.2.4 cups/ppd.h** 23](#_Toc495000961)

[**7.3 Interface Definitions for libcups** 26](#_Toc495000962)

[**7.4 Interfaces for libcupsimage** 44](#_Toc495000963)

[**7.4.1 CUPS Raster ABI** 45](#_Toc495000964)

[**7.5 Data Definitions for libcupsimage** 45](#_Toc495000965)

[**7.5.1 cups/raster.h** 45](#_Toc495000966)

[**7.6 Interface Definitions for libcupsimage** 48](#_Toc495000967)

[**III Printing Commands** 51](#_Toc495000968)

[**8 Printing Commands** 52](#_Toc495000969)

[**8.1 Commands and Utilities** 52](#_Toc495000970)

[**8.2 Command Behavior** 52](#_Toc495000971)

[**IV Execution Environment** 58](#_Toc495000972)

[**9 File System Hierarchy** 59](#_Toc495000973)

[**V Scanning Libraries** 60](#_Toc495000974)

[**10 Libraries** 61](#_Toc495000975)

[**10.1 Interfaces for libsane** 61](#_Toc495000976)

[**10.1.1 libsane interfaces** 61](#_Toc495000977)

[**10.2 Data Definitions for libsane** 61](#_Toc495000978)

[**10.2.1 sane/sane.h** 61](#_Toc495000979)

[**10.2.2 sane/saneopts.h** 64](#_Toc495000980)

[**VI Package Format and Installation** 71](#_Toc495000981)

[**11 Software Installation** 72](#_Toc495000982)

[**11.1 Package Dependencies** 72](#_Toc495000983)

[**Annex A Alphabetical Listing of Interfaces by Library** 73](#_Toc495000984)

[**A.1 libsane** 73](#_Toc495000985)

[**A.2 libcups** 73](#_Toc495000986)

[**A.3 libcupsimage** 75](#_Toc495000987)

[**Annex B GNU Free Documentation License (Informative)** 76](#_Toc495000988)

[**B.1 PREAMBLE** 76](#_Toc495000989)

[**B.2 APPLICABILITY AND DEFINITIONS** 76](#_Toc495000990)

[**B.3 VERBATIM COPYING** 77](#_Toc495000991)

[**B.4 COPYING IN QUANTITY** 77](#_Toc495000992)

[**B.5 MODIFICATIONS** 78](#_Toc495000993)

[**B.6 COMBINING DOCUMENTS** 79](#_Toc495000994)

[**B.7 COLLECTIONS OF DOCUMENTS** 79](#_Toc495000995)

[**B.8 AGGREGATION WITH INDEPENDENT WORKS** 79](#_Toc495000996)

[**B.9 TRANSLATION** 80](#_Toc495000997)

[**B.10 TERMINATION** 80](#_Toc495000998)

[**B.11 FUTURE REVISIONS OF THIS LICENSE** 80](#_Toc495000999)

[**B.12 How to use this License for your documents** 80](#_Toc495001000)

# **List of Tables**

[2-1 Normative References](#_118) [2](#_118)

[3-1 Standard Library Names](#ID_TBL_45_IMG_45_STDLIB) [3](#ID_TBL_45_IMG_45_STDLIB)

[7-1 libcups Definition](#ID_LIB_45_LIBCUPS_45_DEF) [10](#ID_LIB_45_LIBCUPS_45_DEF)

[7-2 libcups - CUPS Convenience ABI Function Interfaces](#ID_TBL_45_LIBCUPS_45_CUPS_45_INTS) [10](#ID_TBL_45_LIBCUPS_45_CUPS_45_INTS)

[7-3 libcups - CUPS Convenience ABI Deprecated Function Interfaces](#ID_TBL_45_LIBCUPS_45_CUPS_45_DEPINTS) [12](#ID_TBL_45_LIBCUPS_45_CUPS_45_DEPINTS)

[7-4 libcupsimage Definition](#ID_LIB_45_LIBCUPSIMAGE_45_DEF) [44](#ID_LIB_45_LIBCUPSIMAGE_45_DEF)

[7-5 libcupsimage - CUPS Raster ABI Function Interfaces](#ID_TBL_45_LIBCUPSIMAGE_45_CUPS_45_INTS) [45](#ID_TBL_45_LIBCUPSIMAGE_45_CUPS_45_INTS)

[7-6 libcupsimage - CUPS Raster ABI Deprecated Function Interfaces](#ID_TBL_45_LIBCUPSIMAGE_45_CUPS_45_DEPINTS) [45](#ID_TBL_45_LIBCUPSIMAGE_45_CUPS_45_DEPINTS)

[8-1 Commands And Utilities](#ID_TBL_45_CMDS) [52](#ID_TBL_45_CMDS)

[10-1 libsane Definition](#ID_LIB_45_LIBSANE_45_DEF) [61](#ID_LIB_45_LIBSANE_45_DEF)

[10-2 libsane - libsane interfaces Function Interfaces](#ID_TBL_45_LIBSANE_45_LIBSA_45_INTS) [61](#ID_TBL_45_LIBSANE_45_LIBSA_45_INTS)

[A-1 libsane Function Interfaces](#_3117) [73](#_3117)

[A-2 libcups Function Interfaces](#_3164) [73](#_3164)

[A-3 libcupsimage Function Interfaces](#_3490) [75](#_3490)

# **Foreword**

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

The committee responsible for this document is Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 22, *Programming languages, their environments and system software interfaces*.

This document is a direct adoption of the Linux Standards Base (LSB) 5.0, issued by the Linux Foundation. ISO/IEC 23360-1 through -8:2005 were international standards published under the ISO/IEC/JTC 1 Publicly Available Specification process. This document, and others in the series, are published under the GNU Free Documentation License (See Annex B).

This is version 1.0 of the Linux Standard Base Imaging Specification. This standard replaces the Imaging portion of ISO/IEC 23360-1 Linux Standard Base, which is cancelled and replaced by ISO/IEC 23360-1-1 through -1-5. Other (processor specific) parts of the original Linux Standards Base are also subdivided as follows

* the Intel 32 bit architecture in ISO/IEC 23360-2-2 and ISO/IEC 23360-2-3,
* the Intel 64 bit architecture in ISO/IEC 23360-3-2 and ISO/IEC 23360-3-3,
* the PowerPC 32 bit architecture in ISO/IEC 23360-4-2 and ISO/IEC 23360-4-2,
* the PowerPC 64 bit architecture in ISO/IEC 23360-5-2 and ISO/IEC 23360-5-3,
* the IBM S390 architecture in ISO/IEC 23360-6-2 and ISO/IEC 23360-7-3,
* the IBM S390X architecture in ISO/IEC 23360-7-2 and ISO/IEC 23360-7-3, and

the AMD 64 bit architecture in ISO/IEC 23360-8-2 and ISO/IEC 23360-8-3

* the Intel 32 bit architecture in ISO/IEC 23360-2-2 and ISO/IEC 23360-2-3,
* the Intel 64 bit architecture in ISO/IEC 23360-3-2 and ISO/IEC 23360-3-3,
* the PowerPC 32 bit architecture in ISO/IEC 23360-4-2 and ISO/IEC 23360-4-2,
* the PowerPC 64 bit architecture in ISO/IEC 23360-5-2 and ISO/IEC 23360-5-3,
* the IBM S390 architecture in ISO/IEC 23360-6-2 and ISO/IEC 23360-7-3,
* the IBM S390X architecture in ISO/IEC 23360-7-2 and ISO/IEC 23360-7-3, and
* the AMD 64 bit architecture in ISO/IEC 23360-8-2 and ISO/IEC 23360-8-3

# **Status of this Document**

A list of current released Linux Standard Base (LSB) specifications is available at http://refspecs.linuxbase.org (http://refspecs.linuxbase.org/).

If you wish to make comments regarding this document in a manner that is tracked by the LSB project, please submit them using the Linux Foundation public bug database at http://bugs.linuxbase.org. Please enter your feedback, carefully indicating the title of the section for which you are submitting feedback, and the volume and version of the specification where you found the problem, quoting the incorrect text if appropriate. If you are suggesting a new feature, please indicate what the problem you are trying to solve is. That is more important than the solution, in fact.

If you do not have or wish to create a bug database account then you can also e-mail feedback to <lsb-discuss@lists.linuxfoundation.org> (subscribe (http://lists.linuxfoundation.org/mailman/listinfo/lsb-discuss), archives (http://lists.linuxfoundation.org/pipermail/lsb-discuss/)), and arrangements will be made to transpose the comments to our public bug database.

# **Introduction**

The LSB defines a binary interface for application programs that are compiled and packaged for LSB-conforming implementations on many different hardware architectures. A binary specification must include information specific to the computer processor architecture for which it is intended. To avoid the complexity of conditional descriptions, the specification has instead been divided into generic parts which are augmented by one of several architecture-specific parts, depending on the target processor architecture; the generic part will indicate when reference must be made to the architecture part, and vice versa.

This document should be used in conjunction with the documents it references. This document enumerates the system components it includes, but descriptions of those components may be included entirely or partly in this document, partly in other documents, or entirely in other reference documents. For example, the section that describes system service routines includes a list of the system routines supported in this interface, formal declarations of the data structures they use that are visible to applications, and a pointer to the underlying referenced specification for information about the syntax and semantics of each call. Only those routines not described in standards referenced by this document, or extensions to those standards, are described in the detail. Information referenced in this way is as much a part of this document as is the information explicitly included here.

The specification carries a version number of either the form *x.y* or *x.y.z*. This version number carries the following meaning:

 1. The first number (*x*) is the major version number. Versions sharing the same major version number shall be compatible in a backwards direction; that is, a newer version shall be compatible with an older version. Any deletion of a library results in a new major version number. Interfaces marked as deprecated may be removed from the specification at a major version change.

 2. The second number (*y*) is the minor version number. Libraries and individual interfaces may be added, but not removed. Interfaces may be marked as deprecated at a minor version change. Other minor changes may be permitted at the discretion of the LSB workgroup.

 3. The third number (*z*), if present, is the editorial level. Only editorial changes should be included in such versions.

Since this specification is a descriptive Application Binary Interface, and not a source level API specification, it is not possible to make a guarantee of 100% backward compatibility between major releases. However, it is the intent that those parts of the binary interface that are visible in the source level API will remain backward compatible from version to version, except where a feature marked as "Deprecated" in one release may be removed from a future release. Implementors are strongly encouraged to make use of symbol versioning to permit simultaneous support of applications conforming to different releases of this specification.

LSB is a trademark of the Linux Foundation. Developers of applications or implementations interested in using the trademark should see the Linux Foundation Certification Policy for details.

# **I Introductory Elements**

# **1 Scope**

## **1.1 General**

The Linux Standard Base (LSB) defines a system interface for compiled applications and a minimal environment for support of installation scripts. Its purpose is to enable a uniform industry standard environment for high-volume applications conforming to the LSB.

These specifications are composed of two basic parts: a common part describing those parts of the interface that remain constant across all implementations of the LSB, and an architecture-specific part describing the parts of the interface that vary by processor architecture. Together, the common part and the relevant architecture-specific part for a single hardware architecture provide a complete interface specification for compiled application programs on systems that share a common hardware architecture.

The LSB contains both a set of Application Program Interfaces (APIs) and Application Binary Interfaces (ABIs). APIs may appear in the source code of portable applications, while the compiled binary of that application may use the larger set of ABIs. A conforming implementation provides all of the ABIs listed here. The compilation system may replace (e.g. by macro definition) certain APIs with calls to one or more of the underlying binary interfaces, and may insert calls to binary interfaces as needed.

The LSB is primarily a binary interface definition. Not all of the source level APIs available to applications may be contained in this specification.

## **1.2 Module Specific Scope**

This is the Imaging module of the Linux Standard Base (LSB). This module provides the fundamental system interfaces, libraries, and runtime environment upon which conforming applications and libraries requiring the LSB Imaging module depend.

Interfaces described in LSB Imaging are mandatory except where explicitly listed otherwise. Interfaces described in the LSB Imaging module supplement those described in the LSB Core module. They do not depend on other LSB modules.

# **2 Normative References**

The specifications listed below are referenced in whole or in part by the LSB Imaging specification. Such references may be normative or informative; a reference to specification shall only be considered normative if it is explicitly cited as such. The LSB Imaging specification may make normative references to a portion of these specifications (that is, to define a specific function or group of functions); in such cases, only the explicitly referenced portion of the specification is to be considered normative.

**Table 2-1 Normative References**

| **Name** | **Title** | **URL** |
| --- | --- | --- |
| CUPS API Reference | CUPS 1.2 API Reference | http://www.cups.org/documentation.php/doc-1.2/ |
| Filesystem Hierarchy Standard | Filesystem Hierarchy Standard (FHS) 3.0 | http://refspecs.linuxbase.org/fhs |
| ISO C (1999) | ISO/IEC 9899:1999 - Programming Languages -- C |  |
| PPD Specification | PostScript Printer Description File Format Specification version 4.3 | http://partners.adobe.com/public/developer/en/ps/5003.PPD\_Spec\_v4.3.pdf |
| PPD Specification Update | Update to PPD Specification Version 4.3 | http://partners.adobe.com/public/developer/en/ps/5645.PPD\_Update.pdf |
| SANE Standard Version 1.04 | SANE Standard Version 1.04 | http://www.sane-project.org/html/ |

# **3 Requirements**

## **3.1 Relevant Libraries**

The libraries listed in [Table 3-1](#ID_TBL_45_IMG_45_STDLIB) shall be available on a Linux Standard Base system, with the specified runtime names. This list may be supplemented or amended by an architecture-specific specification.

**Table 3-1 Standard Library Names**

| **Library** | **Runtime Name** |
| --- | --- |
| libcups | libcups.so.2 |
| libcupsimage | libcupsimage.so.2 |
| libsane | libsane.so.1 |

These libraries will be in an implementation-defined directory which the dynamic linker shall search by default.

# **4 Terms and Definitions**

For the purposes of this document, the terms given in *ISO/IEC Directives, Part 2, Annex H* and the following apply.

archLSB

  Some LSB specification documents have both a generic, architecture-neutral part and an architecture-specific part. The latter describes elements whose definitions may be unique to a particular processor architecture. The term archLSB may be used in the generic part to refer to the corresponding section of the architecture-specific part.

Binary Standard, ABI

  The total set of interfaces that are available to be used in the compiled binary code of a conforming application, including the run-time details such as calling conventions, binary format, C++ name mangling, etc.

Implementation-defined

  Describes a value or behavior that is not defined by this document but is selected by an implementor. The value or behavior may vary among implementations that conform to this document. An application should not rely on the existence of the value or behavior. An application that relies on such a value or behavior cannot be assured to be portable across conforming implementations. The implementor shall document such a value or behavior so that it can be used correctly by an application.

Shell Script

  A file that is read by an interpreter (e.g., awk). The first line of the shell script includes a reference to its interpreter binary.

Source Standard, API

  The total set of interfaces that are available to be used in the source code of a conforming application. Due to translations, the Binary Standard and the Source Standard may contain some different interfaces.

Undefined

  Describes the nature of a value or behavior not defined by this document which results from use of an invalid program construct or invalid data input. The value or behavior may vary among implementations that conform to this document. An application should not rely on the existence or validity of the value or behavior. An application that relies on any particular value or behavior cannot be assured to be portable across conforming implementations.

Unspecified

  Describes the nature of a value or behavior not specified by this document which results from use of a valid program construct or valid data input. The value or behavior may vary among implementations that conform to this document. An application should not rely on the existence or validity of the value or behavior. An application that relies on any particular value or behavior cannot be assured to be portable across conforming implementations.

In addition, for the portions of this specification which build on IEEE Std 1003.1-2001, the definitions given in *IEEE Std 1003.1-2001, Base Definitions, Chapter 3* apply.

# **5 Documentation Conventions**

Throughout this document, the following typographic conventions are used:

function()

  the name of a function

**command**

  the name of a command or utility

CONSTANT

  a constant value

*parameter*

  a parameter

variable

  a variable

Throughout this specification, several tables of interfaces are presented. Each entry in these tables has the following format:

name

  the name of the interface

(symver)

  An optional symbol version identifier, if required.

[*refno*]

  A reference number indexing the table of referenced specifications that follows this table.

For example,

|  |
| --- |
| forkpty(GLIBC\_2.0) [SUSv4] |

refers to the interface named forkpty() with symbol version GLIBC\_2.0 that is defined in the reference indicated by the tag SUSv4.

**Note:** For symbols with versions which differ between architectures, the symbol versions are defined in the architecture specific parts of of this module specification only. In the generic part, they will appear without symbol versions.

# **6 PPD Format Extensions**

The Postscript Printer Description (PPD) format is used in a text file to describe device capabilities for a printing device. PPD files shall conform to the format described by [PPD Specification](#ID_STD_46_PPD) and [PPD Specification Update](#ID_STD_46_PPDUPDATE). In addition, several extensions to the standard attribute list are recognized, as listed below. The "cupsVersion" attribute is required in a compliant PPD, while the other attributes are optional.

cupsColorProfile

  This string attribute specifies an sRGB-based color profile consisting of gamma and density controls and a 3x3 CMY color transform matrix.

The attribute has the following parameter usage:

\*cupsColorProfile Resolution/MediaType: "density gamma m00 m01 m02 m10 m11 m12 m20 m21 m22"

The Resolution and MediaType values may be "-" to act as a wildcard. Otherwise, they must match one of the Resolution or MediaType attributes defined in the PPD file.

The density and gamma values define the gamma and density adjustment function such that (in terms of C math):

f(x) = density \* pow(x, gamma)

The m00 through m22 values define a 3x3 transformation matrix for the CMY color values. The density function is applied after the CMY transformation:

| m00 m01 m02 | | m10 m11 m12 | | m20 m21 m22 |

cupsFax

  This boolean attribute specifies whether the PPD defines a facsimile device. The default is false.

cupsFilter

  The attribute has the following parameter usage:

\*cupsFilter: "source/type cost program"

This string attribute provides a conversion rule from the given source type to the printer's native format using the filter "program". A source type is specified according to the conventions of the MIME specification, using "type/subtype" nomenclature, and may refer to a standard MIME type or a CUPS-specific MIME type using the prefix "vnd.cups-" in the subtype. If a printer supports the source type directly, the special filter program "-" may be specified. The cost is an arbitrary positive integer, used to calculate the relative impact a print job has on system load.

cupsManualCopies

  This boolean attribute notifies the RIP filters that the destination printer does not support copy generation in hardware. The default value is false.

cupsModelNumber

  This integer attribute specifies a printer-specific model number. This number can be used by a filter program to adjust the output for a specific model of printer.

cupsVersion

  The attribute has the following parameter usage:

\*cupsVersion: "major.minor"

This required attribute describes which version of the CUPS PPD file extensions was used. Currently it must be the string "1.0" or "1.1". The strings "1.2" and "1.3" represent newer versions of the CUPS PPD API that are not covered in this version of the specification, and are currently not allowed, although they may be found in non-conforming PPDs which use a newer version of the CUPS PPD specification.

FoomaticIDs

  The attribute has the following parameter usage:

\*FoomaticIDs printer driver

The parameters correspond to the IDs in the Foomatic database for the printer and driver, respectively.

FoomaticNoPageAccounting

  This boolean attribute tells foomatic-rip whether or not to insert accounting information into the PostScript data stream. By default, foomatic-rip will insert this information.

FoomaticRIPCommandLine

  The attribute has the following parameter usage:

\*FoomaticRIPCommandLine "code"

This attribute defines the command line in the "code" parameter for the renderer that is called by foomatic-rip. The command must take PostScript on standard input and provide the job data stream in the printer's native language on standard output. The command must exit with status 0 if the conversion was successful and exit with another status if an error occurs. The "code" parameter may contain option setting wildcards, as described below under "FoomaticRIPOption".

FoomaticRIPDefault

  The attribute has the following parameter usage:

\*FoomaticRIPDefaultOptionName value

This attribute sets a default for a Foomatic option. The name of the attribute should contain the name of the option appended to "FoomaticRIPDefault", with the desired default value as the only parameter.

This option is only used to provide numeric options in the PPD, which are not supported by the Adobe spec, via enumerated options, and should not be used except for that purpose.

FoomaticRIPOption

  The attribute has the following parameter usage:

\*FoomaticRIPOption name: type style spot [order]

This attribute sets options for the command line specified in the "FoomaticRIPCommandLine" attribute. The "name" parameter specifies the option name, the "type" parameter specifies the option type, the "style" parameter specifies one of "CmdLine", "JCL", "PS", or "Composite", and the "spot" parameter specifies a letter, which is prepended with a "%" and used in the "FoomaticRIPCommandLine" attribute to indicate where the option should go in the command line. The optional "order" parameter indicates an order number for one-choice options.

FoomaticRIPOptionAllowedChars

  The attribute has the following parameter usage:

\*FoomaticRIPOptionAllowedChars name: "code"

This option sets a list of allowed characters in a string option. The "name" parameter identifies the option, while the "code" parameter is a list of allowed characters.

FoomaticRIPOptionAllowedRegExp

  The attribute has the following parameter usage:

\*FoomaticRIPOptionAllowedRegExp name: "code"

This option causes the option named by "name" to be validated by the Perl-style regular expression in "code".

FoomaticRIPOptionMaxLength

  The attribute has the following parameter usage:

\*FoomaticRIPOptionMaxLength name: length

For string or password options, this attribute sets a maximum length which can be returned. The "name" parameter identifies the option, and the "length" parameter is the maximum number of characters allowed.

FoomaticRIPOptionPrototype

  The attribute has the following parameter usage:

\*FoomaticRIPOptionPrototype name: "code"

For string, password, or simulated numeric options, this attribute sets a code prototype to be inserted into the output. This works for options where the FoomaticRIPOption "style" parameter is set to CmdLine, JCL, or PS. The value of the option can be represented with the string "%s" in the "code" parameter.

FoomaticRIPOptionRange

  The attribute has the following parameter usage:

\*FoomaticRIPOptionRange name: min max

This attribute adds a minimux and maximum limit to numeric options (that are simulated by Foomatic via emumerated options). The "name" parameter identifies the option, while the "min" and "max" parameters set the minumum and maximum allowed values, respectively, for the option.

FoomaticRIPOptionSetting

  The attribute has the following parameter usage:

\*FoomaticRIPOptionSetting name=choice: "code"

This attribute adds code to an option, identified by "name", with a FoomaticRIPOption "style" parameter set to Composite. It inserts options for other options that are members of the Composite option "name".

FoomaticRIPPostPipe

  The attribute has the following parameter usage:

\*FoomaticRIPPostPipe "code"

This attribute defines the command line in the "code" parameter for the job output command used by foomatic-rip in standalone mode. The command should take printer-native data on standard input. The "code" parameter should include the preceding shell pipe symbol ("|").

# **II Printing Libraries**

# **7 Libraries**

## **7.1 Interfaces for libcups**

[Table 7-1](#ID_LIB_45_LIBCUPS_45_DEF) defines the library name and shared object name for the libcups library

**Table 7-1 libcups Definition**

|  |  |
| --- | --- |
| Library: | libcups |
| SONAME: | libcups.so.2 |

The behavior of the interfaces in this library is specified by the following specifications:

|  |
| --- |
| [CUPS 1.2] [CUPS API Reference](#ID_STD_46_CUPS) |
| [LSB] [This Specification](#ID_STD_46_LSB) |

### **7.1.1 CUPS Convenience ABI**

#### 7.1.1.1 Interfaces for CUPS Convenience ABI

An LSB conforming implementation shall provide the generic functions for CUPS Convenience ABI specified in [Table 7-2](#ID_TBL_45_LIBCUPS_45_CUPS_45_INTS), with the full mandatory functionality as described in the referenced underlying specification.

**Table 7-2 libcups - CUPS Convenience ABI Function Interfaces**

|  |  |  |  |
| --- | --- | --- | --- |
| cupsAddDest [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | cupsAddOption [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | cupsCancelJob [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | cupsDoAuthentication [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) |
| cupsDoFileRequest [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | cupsEncodeOptions [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | cupsEncryption [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | cupsFreeDests [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) |
| cupsFreeJobs [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | cupsFreeOptions [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | cupsGetDefault [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | cupsGetDefault2 [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) |
| cupsGetDest [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | cupsGetDests [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | cupsGetDests2 [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | cupsGetFd [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) |
| cupsGetFile [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | cupsGetJobs [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | cupsGetJobs2 [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | cupsGetOption [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) |
| cupsGetPPD [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | cupsGetPPD2 [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | cupsGetPassword [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | cupsLangEncoding [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) |
| cupsLangFlush [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | cupsLangFree [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | cupsLangGet [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | cupsLastError [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) |
| cupsMarkOptions [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | cupsParseOptions [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | cupsPrintFile [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | cupsPrintFile2 [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) |
| cupsPrintFiles [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | cupsPrintFiles2 [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | cupsPutFd [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | cupsPutFile [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) |
| cupsServer [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | cupsSetDests [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | cupsSetDests2 [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | cupsSetEncryption [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) |
| cupsSetPasswordCB [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | cupsSetServer [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | cupsSetUser [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | cupsTempFd [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) |
| cupsUser [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | httpBlocking [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | httpCheck [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | httpClearCookie [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) |
| httpClearFields [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | httpClose [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | httpConnect [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | httpConnectEncrypt [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) |
| httpDecode64\_2 [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | httpDelete [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | httpEncode64\_2 [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | httpEncryption [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) |
| httpError [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | httpFlush [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | httpGet [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | httpGetCookie [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) |
| httpGetDateString [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | httpGetDateTime [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | httpGetField [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | httpGetHostByName [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) |
| httpGetSubField [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | httpGets [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | httpHead [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | httpInitialize [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) |
| httpMD5 [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | httpMD5Final [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | httpMD5String [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | httpOptions [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) |
| httpPost [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | httpPut [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | httpReconnect [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | httpSetCookie [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) |
| httpSetField [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | httpStatus [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | httpTrace [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | httpUpdate [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) |
| httpWait [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | ippAddBoolean [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | ippAddBooleans [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | ippAddCollection [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) |
| ippAddCollections [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | ippAddDate [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | ippAddInteger [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | ippAddIntegers [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) |
| ippAddRange [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | ippAddRanges [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | ippAddResolution [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | ippAddResolutions [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) |
| ippAddSeparator [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | ippAddString [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | ippAddStrings [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | ippDateToTime [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) |
| ippDelete [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | ippDeleteAttribute [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | ippErrorString [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | ippFindAttribute [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) |
| ippFindNextAttribute [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | ippLength [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | ippNew [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | ippPort [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) |
| ippRead [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | ippReadFile [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | ippReadIO [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | ippSetPort [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) |
| ippTimeToDate [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | ippWrite [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | ippWriteFile [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) | ippWriteIO [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) |
| ppdClose [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | ppdCollect [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | ppdConflicts [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | ppdEmit [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) |
| ppdEmitFd [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | ppdEmitJCL [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | ppdErrorString [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | ppdFindAttr [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) |
| ppdFindChoice [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | ppdFindMarkedChoice [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | ppdFindNextAttr [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | ppdFindOption [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) |
| ppdIsMarked [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | ppdLastError [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | ppdMarkDefaults [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | ppdMarkOption [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) |
| ppdOpen [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | ppdOpenFd [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | ppdOpenFile [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | ppdPageLength [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) |
| ppdPageSize [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | ppdPageWidth [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) | ppdSetConformance [[LSB]](#ID_REFSTD_46_LIBCUPS_46_2) |  |

An LSB conforming implementation shall provide the generic deprecated functions for CUPS Convenience ABI specified in [Table 7-3](#ID_TBL_45_LIBCUPS_45_CUPS_45_DEPINTS), with the full mandatory functionality as described in the referenced underlying specification.

**Note:** These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

**Table 7-3 libcups - CUPS Convenience ABI Deprecated Function Interfaces**

|  |  |  |  |
| --- | --- | --- | --- |
| httpConnect [[CUPS 1.2]](#ID_REFSTD_46_LIBCUPS_46_1) |  |  |  |

## **7.2 Data Definitions for libcups**

This section defines global identifiers and their values that are associated with interfaces contained in libcups. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the [ISO C (1999)](#ID_STD_46_ISOC99) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

### **7.2.1 cups/cups.h**

#define \_CUPS\_CUPS\_H\_

#define CUPS\_VERSION\_MAJOR 1

#define CUPS\_VERSION\_MINOR 1

#define CUPS\_VERSION 1.0123

#define CUPS\_VERSION\_PATCH 23

#define cupsLangDefault() cupsLangGet(NULL)

typedef enum {

 CUPS\_AUTO\_ENCODING = -1,

 CUPS\_US\_ASCII = 0,

 CUPS\_ISO8859\_1 = 1,

 CUPS\_ISO8859\_2 = 2,

 CUPS\_ISO8859\_3 = 3,

 CUPS\_ISO8859\_4 = 4,

 CUPS\_ISO8859\_5 = 5,

 CUPS\_ISO8859\_6 = 6,

 CUPS\_ISO8859\_7 = 7,

 CUPS\_ISO8859\_8 = 8,

 CUPS\_ISO8859\_9 = 9,

 CUPS\_ISO8859\_10 = 10,

 CUPS\_UTF8 = 11,

 CUPS\_ISO8859\_13 = 12,

 CUPS\_ISO8859\_14 = 13,

 CUPS\_ISO8859\_15 = 14,

 CUPS\_WINDOWS\_874 = 15,

 CUPS\_WINDOWS\_1250 = 16,

 CUPS\_WINDOWS\_1251 = 17,

 CUPS\_WINDOWS\_1252 = 18,

 CUPS\_WINDOWS\_1253 = 19,

 CUPS\_WINDOWS\_1254 = 20,

 CUPS\_WINDOWS\_1255 = 21,

 CUPS\_WINDOWS\_1256 = 22,

 CUPS\_WINDOWS\_1257 = 23,

 CUPS\_WINDOWS\_1258 = 24,

 CUPS\_KOI8\_R = 25,

 CUPS\_KOI8\_U = 26

} cups\_encoding\_t;

typedef struct cups\_lang\_s {

 struct cups\_lang\_s \*next;

 int used;

 cups\_encoding\_t encoding;

 char language[16];

 cups\_array\_t \*strings;

} cups\_lang\_t;

typedef enum {

 HTTP\_ENCRYPT\_IF\_REQUESTED = 0,

 HTTP\_ENCRYPT\_NEVER = 1,

 HTTP\_ENCRYPT\_REQUIRED = 2,

 HTTP\_ENCRYPT\_ALWAYS = 3

} http\_encryption\_t;

typedef struct {

 char \*name;

 char \*value;

} cups\_option\_t;

typedef struct {

 char \*name;

 char \*instance;

 int is\_default;

 int num\_options;

 cups\_option\_t \*options;

} cups\_dest\_t;

typedef enum {

 HTTP\_WAITING = 0,

 HTTP\_OPTIONS = 1,

 HTTP\_GET = 2,

 HTTP\_GET\_SEND = 3,

 HTTP\_HEAD = 4,

 HTTP\_POST = 5,

 HTTP\_POST\_RECV = 6,

 HTTP\_POST\_SEND = 7,

 HTTP\_PUT = 8,

 HTTP\_PUT\_RECV = 9,

 HTTP\_DELETE = 10,

 HTTP\_TRACE = 11,

 HTTP\_CLOSE = 12,

 HTTP\_STATUS = 13

} http\_state\_t;

typedef enum {

 HTTP\_ERROR = -1,

 HTTP\_CONTINUE = 100,

 HTTP\_SWITCHING\_PROTOCOLS = 101,

 HTTP\_OK = 200,

 HTTP\_CREATED = 201,

 HTTP\_ACCEPTED = 202,

 HTTP\_NOT\_AUTHORITATIVE = 203,

 HTTP\_NO\_CONTENT = 204,

 HTTP\_RESET\_CONTENT = 205,

 HTTP\_PARTIAL\_CONTENT = 206,

 HTTP\_MULTIPLE\_CHOICES = 300,

 HTTP\_MOVED\_PERMANENTLY = 301,

 HTTP\_MOVED\_TEMPORARILY = 302,

 HTTP\_SEE\_OTHER = 303,

 HTTP\_NOT\_MODIFIED = 304,

 HTTP\_USE\_PROXY = 305,

 HTTP\_BAD\_REQUEST = 400,

 HTTP\_UNAUTHORIZED = 401,

 HTTP\_PAYMENT\_REQUIRED = 402,

 HTTP\_FORBIDDEN = 403,

 HTTP\_NOT\_FOUND = 404,

 HTTP\_METHOD\_NOT\_ALLOWED = 405,

 HTTP\_NOT\_ACCEPTABLE = 406,

 HTTP\_PROXY\_AUTHENTICATION = 407,

 HTTP\_REQUEST\_TIMEOUT = 408,

 HTTP\_CONFLICT = 409,

 HTTP\_GONE = 410,

 HTTP\_LENGTH\_REQUIRED = 411,

 HTTP\_PRECONDITION = 412,

 HTTP\_REQUEST\_TOO\_LARGE = 413,

 HTTP\_URI\_TOO\_LONG = 414,

 HTTP\_UNSUPPORTED\_MEDIATYPE = 415,

 HTTP\_UPGRADE\_REQUIRED = 426,

 HTTP\_SERVER\_ERROR = 500,

 HTTP\_NOT\_IMPLEMENTED = 501,

 HTTP\_BAD\_GATEWAY = 502,

 HTTP\_SERVICE\_UNAVAILABLE = 503,

 HTTP\_GATEWAY\_TIMEOUT = 504,

 HTTP\_NOT\_SUPPORTED = 505

} http\_status\_t;

typedef enum {

 HTTP\_0\_9 = 9,

 HTTP\_1\_0 = 100,

 HTTP\_1\_1 = 101

} http\_version\_t;

typedef enum {

 HTTP\_KEEPALIVE\_OFF = 0,

 HTTP\_KEEPALIVE\_ON = 1

} http\_keepalive\_t;

typedef enum {

 HTTP\_ENCODE\_LENGTH = 0,

 HTTP\_ENCODE\_CHUNKED = 1

} http\_encoding\_t;

typedef enum {

 IPP\_JOB\_PENDING = 3,

 IPP\_JOB\_HELD = 4,

 IPP\_JOB\_PROCESSING = 5,

 IPP\_JOB\_STOPPED = 6,

 IPP\_JOB\_CANCELLED = 7,

 IPP\_JOB\_ABORTED = 8,

 IPP\_JOB\_COMPLETED = 9

} ipp\_jstate\_t;

typedef struct {

 int id;

 char \*dest;

 char \*title;

 char \*user;

 char \*format;

 ipp\_jstate\_t state;

 int size;

 int priority;

 time\_t completed\_time;

 time\_t creation\_time;

 time\_t processing\_time;

} cups\_job\_t;

typedef struct \_cups\_array\_s cups\_array\_t;

typedef struct \_http\_s http\_t;

extern int cupsAddDest(const char \*name, const char \*instance,

 int num\_dests, cups\_dest\_t \* \*dests);

extern int cupsAddOption(const char \*name, const char \*value,

 int num\_options, cups\_option\_t \* \*options);

extern int cupsCancelJob(const char \*printer, int job);

extern int cupsDoAuthentication(http\_t \* http, const char \*method,

 const char \*resource);

extern ipp\_t \*cupsDoFileRequest(http\_t \* http, ipp\_t \* request,

 const char \*resource,

 const char \*filename);

extern void cupsEncodeOptions(ipp\_t \* ipp, int num\_options,

 cups\_option\_t \* options);

extern http\_encryption\_t cupsEncryption(void);

extern void cupsFreeDests(int num\_dests, cups\_dest\_t \* dests);

extern void cupsFreeJobs(int num\_jobs, cups\_job\_t \* jobs);

extern void cupsFreeOptions(int num\_options, cups\_option\_t \* options);

extern const char \*cupsGetDefault(void);

extern const char \*cupsGetDefault2(http\_t \* http);

extern cups\_dest\_t \*cupsGetDest(const char \*name, const char \*instance,

 int num\_dests, cups\_dest\_t \* dests);

extern int cupsGetDests(cups\_dest\_t \* \*dests);

extern int cupsGetDests2(http\_t \* http, cups\_dest\_t \* \*dests);

extern http\_status\_t cupsGetFd(http\_t \* http, const char \*resource,

 int fd);

extern http\_status\_t cupsGetFile(http\_t \* http, const char \*resource,

 const char \*filename);

extern int cupsGetJobs(cups\_job\_t \* \*jobs, const char \*dest, int myjobs,

 int completed);

extern int cupsGetJobs2(http\_t \* http, cups\_job\_t \* \*jobs,

 const char \*dest, int myjobs, int completed);

extern const char \*cupsGetOption(const char \*name, int num\_options,

 cups\_option\_t \* options);

extern const char \*cupsGetPPD(const char \*printer);

extern const char \*cupsGetPPD2(http\_t \* http, const char \*printer);

extern const char \*cupsGetPassword(const char \*prompt);

extern const char \*cupsLangEncoding(cups\_lang\_t \* lang);

extern void cupsLangFlush(void);

extern void cupsLangFree(cups\_lang\_t \* lang);

extern cups\_lang\_t \*cupsLangGet(const char \*language);

extern ipp\_status\_t cupsLastError(void);

extern int cupsMarkOptions(ppd\_file\_t \* ppd, int num\_options,

 cups\_option\_t \* options);

extern int cupsParseOptions(const char \*arg, int num\_options,

 cups\_option\_t \* \*options);

extern int cupsPrintFile(const char \*printer, const char \*filename,

 const char \*title, int num\_options,

 cups\_option\_t \* options);

extern int cupsPrintFile2(http\_t \* http, const char \*printer,

 const char \*filename, const char \*title,

 int num\_options, cups\_option\_t \* options);

extern int cupsPrintFiles(const char \*printer, int num\_files,

 const char \*\*files, const char \*title,

 int num\_options, cups\_option\_t \* options);

extern int cupsPrintFiles2(http\_t \* http, const char \*printer,

 int num\_files, const char \*\*files,

 const char \*title, int num\_options,

 cups\_option\_t \* options);

extern http\_status\_t cupsPutFd(http\_t \* http, const char \*resource,

 int fd);

extern http\_status\_t cupsPutFile(http\_t \* http, const char \*resource,

 const char \*filename);

extern const char \*cupsServer(void);

extern void cupsSetDests(int num\_dests, cups\_dest\_t \* dests);

extern int cupsSetDests2(http\_t \* http, int num\_dests,

 cups\_dest\_t \* dests);

extern void cupsSetEncryption(http\_encryption\_t e);

extern void cupsSetPasswordCB(const char \*(\*cb) (const char \*));

extern void cupsSetServer(const char \*server);

extern void cupsSetUser(const char \*user);

extern int cupsTempFd(char \*filename, int len);

extern const char \*cupsUser(void);

### **7.2.2 cups/http.h**

#define HTTP\_MAX\_URI 1024

#define HTTP\_MAX\_BUFFER 2048

#define HTTP\_MAX\_HOST 256

#define HTTP\_MAX\_VALUE 256

typedef enum http\_auth\_e {

 HTTP\_AUTH\_NONE,

 HTTP\_AUTH\_BASIC,

 HTTP\_AUTH\_MD5,

 HTTP\_AUTH\_MD5\_SESS,

 HTTP\_AUTH\_MD5\_INT,

 HTTP\_AUTH\_MD5\_SESS\_INT,

 HTTP\_AUTH\_NEGOTIATE

} http\_auth\_t;

typedef enum http\_field\_e {

 HTTP\_FIELD\_UNKNOWN,

 HTTP\_FIELD\_ACCEPT\_LANGUAGE,

 HTTP\_FIELD\_ACCEPT\_RANGES,

 HTTP\_FIELD\_AUTHORIZATION,

 HTTP\_FIELD\_CONNECTION,

 HTTP\_FIELD\_CONTENT\_ENCODING,

 HTTP\_FIELD\_CONTENT\_LANGUAGE,

 HTTP\_FIELD\_CONTENT\_LENGTH,

 HTTP\_FIELD\_CONTENT\_LOCATION,

 HTTP\_FIELD\_CONTENT\_MD5,

 HTTP\_FIELD\_CONTENT\_RANGE,

 HTTP\_FIELD\_CONTENT\_TYPE,

 HTTP\_FIELD\_CONTENT\_VERSION,

 HTTP\_FIELD\_DATE,

 HTTP\_FIELD\_HOST,

 HTTP\_FIELD\_IF\_MODIFIED\_SINCE,

 HTTP\_FIELD\_IF\_UNMODIFIED\_SINCE,

 HTTP\_FIELD\_KEEP\_ALIVE,

 HTTP\_FIELD\_LAST\_MODIFIED,

 HTTP\_FIELD\_LINK,

 HTTP\_FIELD\_LOCATION,

 HTTP\_FIELD\_RANGE,

 HTTP\_FIELD\_REFERER,

 HTTP\_FIELD\_RETRY\_AFTER,

 HTTP\_FIELD\_TRANSFER\_ENCODING,

 HTTP\_FIELD\_UPGRADE,

 HTTP\_FIELD\_USER\_AGENT,

 HTTP\_FIELD\_WWW\_AUTHENTICATE,

 HTTP\_FIELD\_MAX

} http\_field\_t;

typedef enum http\_uri\_status\_e {

 HTTP\_URI\_OVERFLOW,

 HTTP\_URI\_BAD\_ARGUMENTS,

 HTTP\_URI\_BAD\_RESOURCE,

 HTTP\_URI\_BAD\_PORT,

 HTTP\_URI\_BAD\_HOSTNAME,

 HTTP\_URI\_BAD\_USERNAME,

 HTTP\_URI\_BAD\_SCHEME,

 HTTP\_URI\_BAD\_URI,

 HTTP\_URI\_OK,

 HTTP\_URI\_MISSING\_SCHEME,

 HTTP\_URI\_UNKNOWN\_SCHEME,

 HTTP\_URI\_MISSING\_RESOURCE

} http\_uri\_status\_t;

typedef enum http\_uri\_coding\_e {

 HTTP\_URI\_CODING\_NONE,

 HTTP\_URI\_CODING\_USERNAME,

 HTTP\_URI\_CODING\_HOSTNAME,

 HTTP\_URI\_CODING\_RESOURCE,

 HTTP\_URI\_CODING\_MOST,

 HTTP\_URI\_CODING\_QUERY,

 HTTP\_URI\_CODING\_ALL

} http\_uri\_coding\_t;

typedef union \_http\_addr\_u {

 struct sockaddr addr;

 struct sockaddr\_in ipv4;

 struct sockaddr\_in6 ipv6;

 struct sockaddr\_un un;

 char pad[256];

} http\_addr\_t;

typedef struct http\_addrlist\_s {

 struct http\_addrlist\_s \*next;

 http\_addr\_t addr;

} http\_addrlist\_t;

extern void httpBlocking(http\_t \* http, int b);

extern int httpCheck(http\_t \* http);

extern void httpClearCookie(http\_t \* http);

extern void httpClearFields(http\_t \* http);

extern void httpClose(http\_t \* http);

extern http\_t \*httpConnect(const char \*host, int port);

extern http\_t \*httpConnectEncrypt(const char \*host, int port,

 http\_encryption\_t encryption);

extern char \*httpDecode64\_2(char \*out, int \*outlen, const char \*in);

extern int httpDelete(http\_t \* http, const char \*uri);

extern char \*httpEncode64\_2(char \*out, int outlen, const char \*in,

 int inlen);

extern int httpEncryption(http\_t \* http, http\_encryption\_t e);

extern int httpError(http\_t \* http);

extern void httpFlush(http\_t \* http);

extern int httpGet(http\_t \* http, const char \*uri);

extern const char \*httpGetCookie(http\_t \* http);

extern const char \*httpGetDateString(time\_t t);

extern time\_t httpGetDateTime(const char \*s);

extern const char \*httpGetField(http\_t \* http, http\_field\_t field);

extern struct hostent \*httpGetHostByName(const char \*name);

extern char \*httpGetSubField(http\_t \* http, http\_field\_t field,

 const char \*name, char \*value);

extern char \*httpGets(char \*line, int length, http\_t \* http);

extern int httpHead(http\_t \* http, const char \*uri);

extern void httpInitialize(void);

extern char \*httpMD5(const char \*, const char \*, const char \*, char \*);

extern char \*httpMD5Final(const char \*, const char \*, const char \*,

 char \*);

extern char \*httpMD5String(const unsigned char \*, char \*);

extern int httpOptions(http\_t \* http, const char \*uri);

extern int httpPost(http\_t \* http, const char \*uri);

extern int httpPut(http\_t \* http, const char \*uri);

extern int httpReconnect(http\_t \* http);

extern void httpSetCookie(http\_t \* http, const char \*cookie);

extern void httpSetField(http\_t \* http, http\_field\_t field,

 const char \*value);

extern const char \*httpStatus(http\_status\_t status);

extern int httpTrace(http\_t \* http, const char \*uri);

extern http\_status\_t httpUpdate(http\_t \* http);

extern int httpWait(http\_t \* http, int msec);

### **7.2.3 cups/ipp.h**

#define IPP\_MAX\_NAME 256

#define IPP\_MAX\_LENGTH 32767

#define IPP\_PORT 631

#define IPP\_MAX\_VALUES 8

#define CUPS\_ADD\_CLASS CUPS\_ADD\_MODIFY\_CLASS

#define CUPS\_ADD\_PRINTER CUPS\_ADD\_MODIFY\_PRINTER

#define IPP\_ERROR\_JOB\_CANCELLED IPP\_ERROR\_JOB\_CANCELED

#define IPP\_JOB\_CANCELLED IPP\_JOB\_CANCELED

#define IPP\_VERSION "\001\001"

typedef enum {

 IPP\_OK = 0,

 IPP\_OK\_SUBST = 1,

 IPP\_OK\_CONFLICT = 2,

 IPP\_OK\_IGNORED\_SUBSCRIPTIONS = 3,

 IPP\_OK\_IGNORED\_NOTIFICATIONS = 4,

 IPP\_OK\_TOO\_MANY\_EVENTS = 5,

 IPP\_OK\_BUT\_CANCEL\_SUBSCRIPTION = 6,

 IPP\_REDIRECTION\_OTHER\_SITE = 768,

 IPP\_BAD\_REQUEST = 1024,

 IPP\_FORBIDDEN = 1025,

 IPP\_NOT\_AUTHENTICATED = 1026,

 IPP\_NOT\_AUTHORIZED = 1027,

 IPP\_NOT\_POSSIBLE = 1028,

 IPP\_TIMEOUT = 1029,

 IPP\_NOT\_FOUND = 1030,

 IPP\_GONE = 1031,

 IPP\_REQUEST\_ENTITY = 1032,

 IPP\_REQUEST\_VALUE = 1033,

 IPP\_DOCUMENT\_FORMAT = 1034,

 IPP\_ATTRIBUTES = 1035,

 IPP\_URI\_SCHEME = 1036,

 IPP\_CHARSET = 1037,

 IPP\_CONFLICT = 1038,

 IPP\_COMPRESSION\_NOT\_SUPPORTED = 1039,

 IPP\_COMPRESSION\_ERROR = 1040,

 IPP\_DOCUMENT\_FORMAT\_ERROR = 1041,

 IPP\_DOCUMENT\_ACCESS\_ERROR = 1042,

 IPP\_ATTRIBUTES\_NOT\_SETTABLE = 1043,

 IPP\_IGNORED\_ALL\_SUBSCRIPTIONS = 1044,

 IPP\_TOO\_MANY\_SUBSCRIPTIONS = 1045,

 IPP\_IGNORED\_ALL\_NOTIFICATIONS = 1046,

 IPP\_PRINT\_SUPPORT\_FILE\_NOT\_FOUND = 1047,

 IPP\_INTERNAL\_ERROR = 1280,

 IPP\_OPERATION\_NOT\_SUPPORTED = 1281,

 IPP\_SERVICE\_UNAVAILABLE = 1282,

 IPP\_VERSION\_NOT\_SUPPORTED = 1283,

 IPP\_DEVICE\_ERROR = 1284,

 IPP\_TEMPORARY\_ERROR = 1285,

 IPP\_NOT\_ACCEPTING = 1286,

 IPP\_PRINTER\_BUSY = 1287,

 IPP\_ERROR\_JOB\_CANCELLED = 1288,

 IPP\_MULTIPLE\_JOBS\_NOT\_SUPPORTED = 1289,

 IPP\_PRINTER\_IS\_DEACTIVATED = 1290

} ipp\_status\_t;

typedef enum ipp\_tag\_e {

 IPP\_TAG\_ZERO,

 IPP\_TAG\_OPERATION,

 IPP\_TAG\_JOB,

 IPP\_TAG\_END,

 IPP\_TAG\_PRINTER,

 IPP\_TAG\_UNSUPPORTED\_GROUP,

 IPP\_TAG\_SUBSCRIPTION,

 IPP\_TAG\_EVENT\_NOTIFICATION,

 IPP\_TAG\_UNSUPPORTED\_VALUE,

 IPP\_TAG\_DEFAULT,

 IPP\_TAG\_UNKNOWN,

 IPP\_TAG\_NOVALUE,

 IPP\_TAG\_NOTSETTABLE,

 IPP\_TAG\_DELETEATTR,

 IPP\_TAG\_ADMINDEFINE,

 IPP\_TAG\_INTEGER,

 IPP\_TAG\_BOOLEAN,

 IPP\_TAG\_ENUM,

 IPP\_TAG\_STRING,

 IPP\_TAG\_DATE,

 IPP\_TAG\_RESOLUTION,

 IPP\_TAG\_RANGE,

 IPP\_TAG\_BEGIN\_COLLECTION,

 IPP\_TAG\_TEXTLANG,

 IPP\_TAG\_NAMELANG,

 IPP\_TAG\_END\_COLLECTION,

 IPP\_TAG\_TEXT,

 IPP\_TAG\_NAME,

 IPP\_TAG\_KEYWORD,

 IPP\_TAG\_URI,

 IPP\_TAG\_URISCHEME,

 IPP\_TAG\_CHARSET,

 IPP\_TAG\_LANGUAGE,

 IPP\_TAG\_MIMETYPE,

 IPP\_TAG\_MEMBERNAME,

 IPP\_TAG\_MASK,

 IPP\_TAG\_COPY

} ipp\_tag\_t;

typedef enum ipp\_res\_e {

 IPP\_RES\_PER\_INCH,

 IPP\_RES\_PER\_CM

} ipp\_res\_t;

typedef enum ipp\_finish\_e {

 IPP\_FINISHINGS\_NONE,

 IPP\_FINISHINGS\_STAPLE,

 IPP\_FINISHINGS\_PUNCH,

 IPP\_FINISHINGS\_COVER,

 IPP\_FINISHINGS\_BIND,

 IPP\_FINISHINGS\_SADDLE\_STITCH,

 IPP\_FINISHINGS\_EDGE\_STITCH,

 IPP\_FINISHINGS\_FOLD,

 IPP\_FINISHINGS\_TRIM,

 IPP\_FINISHINGS\_BALE,

 IPP\_FINISHINGS\_BOOKLET\_MAKER,

 IPP\_FINISHINGS\_JOB\_OFFSET,

 IPP\_FINISHINGS\_STAPLE\_TOP\_LEFT,

 IPP\_FINISHINGS\_STAPLE\_BOTTOM\_LEFT,

 IPP\_FINISHINGS\_STAPLE\_TOP\_RIGHT,

 IPP\_FINISHINGS\_STAPLE\_BOTTOM\_RIGHT,

 IPP\_FINISHINGS\_EDGE\_STITCH\_LEFT,

 IPP\_FINISHINGS\_EDGE\_STITCH\_TOP,

 IPP\_FINISHINGS\_EDGE\_STITCH\_RIGHT,

 IPP\_FINISHINGS\_EDGE\_STITCH\_BOTTOM,

 IPP\_FINISHINGS\_STAPLE\_DUAL\_LEFT,

 IPP\_FINISHINGS\_STAPLE\_DUAL\_TOP,

 IPP\_FINISHINGS\_STAPLE\_DUAL\_RIGHT,

 IPP\_FINISHINGS\_STAPLE\_DUAL\_BOTTOM,

 IPP\_FINISHINGS\_BIND\_LEFT,

 IPP\_FINISHINGS\_BIND\_TOP,

 IPP\_FINISHINGS\_BIND\_RIGHT,

 IPP\_FINISHINGS\_BIND\_BOTTOM

} ipp\_finish\_t;

typedef enum ipp\_orient\_e {

 IPP\_PORTRAIT,

 IPP\_LANDSCAPE,

 IPP\_REVERSE\_LANDSCAPE,

 IPP\_REVERSE\_PORTRAIT

} ipp\_orient\_t;

typedef enum ipp\_quality\_e {

 IPP\_QUALITY\_DRAFT,

 IPP\_QUALITY\_NORMAL,

 IPP\_QUALITY\_HIGH

} ipp\_quality\_t;

typedef enum ipp\_pstate\_e {

 IPP\_PRINTER\_IDLE,

 IPP\_PRINTER\_PROCESSING,

 IPP\_PRINTER\_STOPPED

} ipp\_pstate\_t;

typedef enum ipp\_state\_e {

 IPP\_ERROR,

 IPP\_IDLE,

 IPP\_HEADER,

 IPP\_ATTRIBUTE,

 IPP\_DATA

} ipp\_state\_t;

typedef enum ipp\_op\_e {

 IPP\_PRINT\_JOB,

 IPP\_PRINT\_URI,

 IPP\_VALIDATE\_JOB,

 IPP\_CREATE\_JOB,

 IPP\_SEND\_DOCUMENT,

 IPP\_SEND\_URI,

 IPP\_CANCEL\_JOB,

 IPP\_GET\_JOB\_ATTRIBUTES,

 IPP\_GET\_JOBS,

 IPP\_GET\_PRINTER\_ATTRIBUTES,

 IPP\_HOLD\_JOB,

 IPP\_RELEASE\_JOB,

 IPP\_RESTART\_JOB,

 IPP\_PAUSE\_PRINTER,

 IPP\_RESUME\_PRINTER,

 IPP\_PURGE\_JOBS,

 IPP\_SET\_PRINTER\_ATTRIBUTES,

 IPP\_SET\_JOB\_ATTRIBUTES,

 IPP\_GET\_PRINTER\_SUPPORTED\_VALUES,

 IPP\_CREATE\_PRINTER\_SUBSCRIPTION,

 IPP\_CREATE\_JOB\_SUBSCRIPTION,

 IPP\_GET\_SUBSCRIPTION\_ATTRIBUTES,

 IPP\_GET\_SUBSCRIPTIONS,

 IPP\_RENEW\_SUBSCRIPTION,

 IPP\_CANCEL\_SUBSCRIPTION,

 IPP\_GET\_NOTIFICATIONS,

 IPP\_SEND\_NOTIFICATIONS,

 IPP\_GET\_PRINT\_SUPPORT\_FILES,

 IPP\_ENABLE\_PRINTER,

 IPP\_DISABLE\_PRINTER,

 IPP\_PAUSE\_PRINTER\_AFTER\_CURRENT\_JOB,

 IPP\_HOLD\_NEW\_JOBS,

 IPP\_RELEASE\_HELD\_NEW\_JOBS,

 IPP\_DEACTIVATE\_PRINTER,

 IPP\_ACTIVATE\_PRINTER,

 IPP\_RESTART\_PRINTER,

 IPP\_SHUTDOWN\_PRINTER,

 IPP\_STARTUP\_PRINTER,

 IPP\_REPROCESS\_JOB,

 IPP\_CANCEL\_CURRENT\_JOB,

 IPP\_SUSPEND\_CURRENT\_JOB,

 IPP\_RESUME\_JOB,

 IPP\_PROMOTE\_JOB,

 IPP\_SCHEDULE\_JOB\_AFTER,

 IPP\_PRIVATE,

 CUPS\_GET\_DEFAULT,

 CUPS\_GET\_PRINTERS,

 CUPS\_ADD\_MODIFY\_PRINTER,

 CUPS\_DELETE\_PRINTER,

 CUPS\_GET\_CLASSES,

 CUPS\_ADD\_MODIFY\_CLASS,

 CUPS\_DELETE\_CLASS,

 CUPS\_ACCEPT\_JOBS,

 CUPS\_REJECT\_JOBS,

 CUPS\_SET\_DEFAULT,

 CUPS\_GET\_DEVICES,

 CUPS\_GET\_PPDS,

 CUPS\_MOVE\_JOB,

 CUPS\_AUTHENTICATE\_JOB,

 CUPS\_GET\_PPD

} ipp\_op\_t;

typedef unsigned char ipp\_uchar\_t;

typedef ssize\_t(\*ipp\_iocb\_t) (void \*, ipp\_uchar\_t \*, size\_t);

typedef union ipp\_request\_u {

 struct {

 ipp\_uchar\_t version[2];

 int op\_status;

 int request\_id;

 } any;

 struct {

 ipp\_uchar\_t version[2];

 ipp\_op\_t operation\_id;

 int request\_id;

 } op;

 struct {

 ipp\_uchar\_t version[2];

 ipp\_status\_t status\_code;

 int request\_id;

 } status;

 struct {

 ipp\_uchar\_t version[2];

 ipp\_status\_t status\_code;

 int request\_id;

 } event;

} ipp\_request\_t;

typedef struct ipp\_s {

 ipp\_state\_t state;

 ipp\_request\_t request;

 ipp\_attribute\_t \*attrs;

 ipp\_attribute\_t \*last;

 ipp\_attribute\_t \*current;

 ipp\_tag\_t curtag;

} ipp\_t;

typedef union ipp\_value\_u {

 int integer;

 char boolean;

 ipp\_uchar\_t date[11];

 struct {

 int xres;

 int yres;

 ipp\_res\_t units;

 } resolution;

 struct {

 int lower;

 int upper;

 } range;

 struct {

 char \*charset;

 char \*text;

 } string;

 struct {

 int length;

 void \*data;

 } unknown;

 ipp\_t \*collection;

} ipp\_value\_t;

typedef struct ipp\_attribute\_s {

 struct ipp\_attribute\_s \*next;

 ipp\_tag\_t group\_tag;

 ipp\_tag\_t value\_tag;

 char \*name;

 int num\_values;

 ipp\_value\_t values[1];

} ipp\_attribute\_t;

extern ipp\_attribute\_t \*ippAddBoolean(ipp\_t \* ipp, ipp\_tag\_t group,

 const char \*name, char value);

extern ipp\_attribute\_t \*ippAddBooleans(ipp\_t \* ipp, ipp\_tag\_t group,

 const char \*name, int num\_values,

 const char \*values);

extern ipp\_attribute\_t \*ippAddCollection(ipp\_t \* ipp, ipp\_tag\_t group,

 const char \*name, ipp\_t \* value);

extern ipp\_attribute\_t \*ippAddCollections(ipp\_t \* ipp, ipp\_tag\_t group,

 const char \*name, int num\_values,

 const ipp\_t \* \*values);

extern ipp\_attribute\_t \*ippAddDate(ipp\_t \* ipp, ipp\_tag\_t group,

 const char \*name,

 const ipp\_uchar\_t \* value);

extern ipp\_attribute\_t \*ippAddInteger(ipp\_t \* ipp, ipp\_tag\_t group,

 ipp\_tag\_t type, const char \*name,

 int value);

extern ipp\_attribute\_t \*ippAddIntegers(ipp\_t \* ipp, ipp\_tag\_t group,

 ipp\_tag\_t type, const char \*name,

 int num\_values, const int \*values);

extern ipp\_attribute\_t \*ippAddRange(ipp\_t \* ipp, ipp\_tag\_t group,

 const char \*name, int lower,

 int upper);

extern ipp\_attribute\_t \*ippAddRanges(ipp\_t \* ipp, ipp\_tag\_t group,

 const char \*name, int num\_values,

 const int \*lower, const int \*upper);

extern ipp\_attribute\_t \*ippAddResolution(ipp\_t \* ipp, ipp\_tag\_t group,

 const char \*name, ipp\_res\_t units,

 int xres, int yres);

extern ipp\_attribute\_t \*ippAddResolutions(ipp\_t \* ipp, ipp\_tag\_t group,

 const char \*name, int num\_values,

 ipp\_res\_t units, const int \*xres,

 const int \*yres);

extern ipp\_attribute\_t \*ippAddSeparator(ipp\_t \* ipp);

extern ipp\_attribute\_t \*ippAddString(ipp\_t \* ipp, ipp\_tag\_t group,

 ipp\_tag\_t type, const char \*name,

 const char \*charset,

 const char \*value);

extern ipp\_attribute\_t \*ippAddStrings(ipp\_t \* ipp, ipp\_tag\_t group,

 ipp\_tag\_t type, const char \*name,

 int num\_values, const char \*charset,

 const char \*const \*values);

extern time\_t ippDateToTime(const ipp\_uchar\_t \* date);

extern void ippDelete(ipp\_t \* ipp);

extern void ippDeleteAttribute(ipp\_t \* ipp, ipp\_attribute\_t \* attr);

extern const char \*ippErrorString(ipp\_status\_t error);

extern ipp\_attribute\_t \*ippFindAttribute(ipp\_t \* ipp, const char \*name,

 ipp\_tag\_t type);

extern ipp\_attribute\_t \*ippFindNextAttribute(ipp\_t \* ipp, const char \*name,

 ipp\_tag\_t type);

extern size\_t ippLength(ipp\_t \* ipp);

extern ipp\_t \*ippNew(void);

extern int ippPort(void);

extern ipp\_state\_t ippRead(http\_t \* http, ipp\_t \* ipp);

extern ipp\_state\_t ippReadFile(int fd, ipp\_t \* ipp);

extern ipp\_state\_t ippReadIO(void \*src, ipp\_iocb\_t cb, int blocking,

 ipp\_t \* parent, ipp\_t \* ipp);

extern void ippSetPort(int p);

extern const ipp\_uchar\_t \*ippTimeToDate(time\_t t);

extern ipp\_state\_t ippWrite(http\_t \* http, ipp\_t \* ipp);

extern ipp\_state\_t ippWriteFile(int fd, ipp\_t \* ipp);

extern ipp\_state\_t ippWriteIO(void \*dst, ipp\_iocb\_t cb, int blocking,

 ipp\_t \* parent, ipp\_t \* ipp);

### **7.2.4 cups/ppd.h**

#define \_CUPS\_PPD\_H\_

#define PPD\_MAX\_LINE 256

#define PPD\_VERSION 4.3

#define PPD\_MAX\_NAME 41

#define PPD\_MAX\_TEXT 81

typedef enum {

 PPD\_CS\_CMYK = -4,

 PPD\_CS\_CMY = -3,

 PPD\_CS\_GRAY = 1,

 PPD\_CS\_RGB = 3,

 PPD\_CS\_RGBK = 4,

 PPD\_CS\_N = 5

} ppd\_cs\_t;

typedef struct {

 char name[41];

 char \*start;

 char \*stop;

} ppd\_emul\_t;

typedef enum {

 PPD\_UI\_BOOLEAN = 0,

 PPD\_UI\_PICKONE = 1,

 PPD\_UI\_PICKMANY = 2

} ppd\_ui\_t;

typedef enum {

 PPD\_ORDER\_ANY = 0,

 PPD\_ORDER\_DOCUMENT = 1,

 PPD\_ORDER\_EXIT = 2,

 PPD\_ORDER\_JCL = 3,

 PPD\_ORDER\_PAGE = 4,

 PPD\_ORDER\_PROLOG = 5

} ppd\_section\_t;

typedef struct {

 char marked;

 char choice[41];

 char text[81];

 char \*code;

 void \*option;

} ppd\_choice\_t;

typedef struct {

 char conflicted;

 char keyword[41];

 char defchoice[41];

 char text[81];

 ppd\_ui\_t ui;

 ppd\_section\_t section;

 float order;

 int num\_choices;

 ppd\_choice\_t \*choices;

} ppd\_option\_t;

typedef struct ppd\_group\_str {

 char text[40];

 char name[41];

 int num\_options;

 ppd\_option\_t \*options;

 int num\_subgroups;

 struct ppd\_group\_str \*subgroups;

} ppd\_group\_t;

typedef struct {

 int marked;

 char name[41];

 float width;

 float length;

 float left;

 float bottom;

 float right;

 float top;

} ppd\_size\_t;

typedef struct {

 char option1[41];

 char choice1[41];

 char option2[41];

 char choice2[41];

} ppd\_const\_t;

typedef struct {

 char resolution[41];

 char media\_type[41];

 float density;

 float gamma;

 float matrix[3][3];

} ppd\_profile\_t;

typedef struct {

 char name[41];

 char spec[41];

 char text[81];

 char \*value;

} ppd\_attr\_t;

typedef struct {

 int language\_level;

 int color\_device;

 int variable\_sizes;

 int accurate\_screens;

 int contone\_only;

 int landscape;

 int model\_number;

 int manual\_copies;

 int throughput;

 ppd\_cs\_t colorspace;

 char \*patches;

 int num\_emulations;

 ppd\_emul\_t \*emulations;

 char \*jcl\_begin;

 char \*jcl\_ps;

 char \*jcl\_end;

 char \*lang\_encoding;

 char \*lang\_version;

 char \*modelname;

 char \*ttrasterizer;

 char \*manufacturer;

 char \*product;

 char \*nickname;

 char \*shortnickname;

 int num\_groups;

 ppd\_group\_t \*groups;

 int num\_sizes;

 ppd\_size\_t \*sizes;

 float custom\_min[2];

 float custom\_max[2];

 float custom\_margins[4];

 int num\_consts;

 ppd\_const\_t \*consts;

 int num\_fonts;

 char \*\*fonts;

 int num\_profiles;

 ppd\_profile\_t \*profiles;

 int num\_filters;

 char \*\*filters;

 int flip\_duplex;

 char \*protocols;

 char \*pcfilename;

 int num\_attrs;

 int cur\_attr;

 ppd\_attr\_t \*\*attrs;

} ppd\_file\_t;

typedef enum {

 PPD\_OK = 0,

 PPD\_FILE\_OPEN\_ERROR = 1,

 PPD\_NULL\_FILE = 2,

 PPD\_ALLOC\_ERROR = 3,

 PPD\_MISSING\_PPDADOBE4 = 4,

 PPD\_MISSING\_VALUE = 5,

 PPD\_INTERNAL\_ERROR = 6,

 PPD\_BAD\_OPEN\_GROUP = 7,

 PPD\_NESTED\_OPEN\_GROUP = 8,

 PPD\_BAD\_OPEN\_UI = 9,

 PPD\_NESTED\_OPEN\_UI = 10,

 PPD\_BAD\_ORDER\_DEPENDENCY = 11,

 PPD\_BAD\_UI\_CONSTRAINTS = 12,

 PPD\_MISSING\_ASTERISK = 13,

 PPD\_LINE\_TOO\_LONG = 14,

 PPD\_ILLEGAL\_CHARACTER = 15,

 PPD\_ILLEGAL\_MAIN\_KEYWORD = 16,

 PPD\_ILLEGAL\_OPTION\_KEYWORD = 17,

 PPD\_ILLEGAL\_TRANSLATION = 18,

 PPD\_ILLEGAL\_WHITESPACE = 19

} ppd\_status\_t;

typedef enum {

 PPD\_CONFORM\_RELAXED = 0,

 PPD\_CONFORM\_STRICT = 1

} ppd\_conform\_t;

extern void ppdClose(ppd\_file\_t \* ppd);

extern int ppdCollect(ppd\_file\_t \* ppd, ppd\_section\_t section,

 ppd\_choice\_t \* \*\*choices);

extern int ppdConflicts(ppd\_file\_t \* ppd);

extern int ppdEmit(ppd\_file\_t \* ppd, FILE \* fp, ppd\_section\_t section);

extern int ppdEmitFd(ppd\_file\_t \* ppd, int fd, ppd\_section\_t section);

extern int ppdEmitJCL(ppd\_file\_t \* ppd, FILE \* fp, int job\_id,

 const char \*user, const char \*title);

extern const char \*ppdErrorString(ppd\_status\_t status);

extern ppd\_attr\_t \*ppdFindAttr(ppd\_file\_t \* ppd, const char \*name,

 const char \*spec);

extern ppd\_choice\_t \*ppdFindChoice(ppd\_option\_t \* o, const char \*option);

extern ppd\_choice\_t \*ppdFindMarkedChoice(ppd\_file\_t \* ppd,

 const char \*keyword);

extern ppd\_attr\_t \*ppdFindNextAttr(ppd\_file\_t \* ppd, const char \*name,

 const char \*spec);

extern ppd\_option\_t \*ppdFindOption(ppd\_file\_t \* ppd, const char \*keyword);

extern int ppdIsMarked(ppd\_file\_t \* ppd, const char \*keyword,

 const char \*option);

extern ppd\_status\_t ppdLastError(int \*line);

extern void ppdMarkDefaults(ppd\_file\_t \* ppd);

extern int ppdMarkOption(ppd\_file\_t \* ppd, const char \*keyword,

 const char \*option);

extern ppd\_file\_t \*ppdOpen(FILE \* fp);

extern ppd\_file\_t \*ppdOpenFd(int fd);

extern ppd\_file\_t \*ppdOpenFile(const char \*filename);

extern float ppdPageLength(ppd\_file\_t \* ppd, const char \*name);

extern ppd\_size\_t \*ppdPageSize(ppd\_file\_t \* ppd, const char \*name);

extern float ppdPageWidth(ppd\_file\_t \* ppd, const char \*name);

extern void ppdSetConformance(ppd\_conform\_t c);

## **7.3 Interface Definitions for libcups**

The interfaces defined on the following pages are included in libcups and are defined by this specification. Unless otherwise noted, these interfaces shall be included in the source standard.

Other interfaces listed in [Section 7.1](#ID_LIBCUPS) shall behave as described in the referenced base document.

#### cupsAddDest

#####  Name

cupsAddDest — Add a destination to the list of destinations.

#####  Synopsis

#include <cups/cups.h>

int cupsAddDest(const char \* *name*, const char \* *instance*, int *num\_dests*, cups\_dest\_t \*\* *dests*);

#####  Description

Add a destination to the list of destinations.

This function cannot be used to add a new class or printer queue, it only adds a new container of saved options for the named destination or instance.

If the named destination already exists, the destination list is returned unchanged. Adding a new instance of a destination creates a copy of that destination's options.

Use the cupsSaveDests() function to save the updated list of destinations to the user's lpoptions file.

##### Return Value

New number of destinations

#### cupsAddOption

#####  Name

cupsAddOption — Add an option to an option array.

#####  Synopsis

#include <cups/cups.h>

int cupsAddOption(const char \* *name*, const char \* *value*, int *num\_options*, cups\_option\_t \*\* *options*);

#####  Description

Add an option to an option array.

#####  Return Value

Number of options

#### cupsCancelJob

#####  Name

cupsCancelJob — Cancel a print job on the default server.

#####  Synopsis

#include <cups/cups.h>

int cupsCancelJob(const char \* *name*, int *job*);

#####  Description

Cancel a print job on the default server.

Use the cupsLastError() and cupsLastErrorString() functions to get the cause of any failure.

#####  Return Value

1 on success, 0 on failure

#### cupsEncryption

#####  Name

cupsEncryption — Get the default encryption settings.

#####  Synopsis

#include <cups/cups.h>

http\_encryption\_t cupsEncryption(void);

#####  Description

Get the default encryption settings.

The default encryption setting comes from the CUPS\_ENCRYPTION environment variable, then the ~/.cupsrc file, and finally the /etc/cups/client.conf file. If not set, the default is HTTP\_ENCRYPT\_IF\_REQUESTED.

#####  Return Value

Encryption settings

#### cupsFreeDests

#####  Name

cupsFreeDests — Free the memory used by the list of destinations.

#####  Synopsis

#include <cups/cups.h>

void cupsFreeDests(int *num\_dests*, cups\_dest\_t \* *dests*);

#####  Description

Free the memory used by the list of destinations.

#####  Return Value

This function does not return a value.

#### cupsFreeJobs

#####  Name

cupsFreeJobs — Free memory used by job data.

#####  Synopsis

#include <cups/cups.h>

void cupsFreeJobs(int *num\_jobs*, cups\_job\_t \* *jobs*);

#####  Description

Free memory used by job data.

#####  Return Value

This function does not return a value.

#### cupsFreeOptions

#####  Name

cupsFreeOptions — Free all memory used by options.

#####  Synopsis

#include <cups/cups.h>

void cupsFreeOptions(int *num\_options*, cups\_option\_t \* *options*);

#####  Description

Free all memory used by options.

#####  Return Value

This function does not return a value.

#### cupsGetDefault

#####  Name

cupsGetDefault — Get the default printer or class for the default server.

#####  Synopsis

#include <cups/cups.h>

const char \* cupsGetDefault(void);

#####  Description

Get the default printer or class for the default server.

This function returns the default printer or class as defined by the LPDEST or PRINTER environment variables. If these environment variables are not set, the server default destination is returned. Applications should use the cupsGetDests() and cupsGetDest() functions to get the user-defined default printer, as this function does not support the lpoptions-defined default printer.

#####  Return Value

Default printer or NULL

#### cupsGetDest

#####  Name

cupsGetDest — Get the named destination from the list.

#####  Synopsis

#include <cups/cups.h>

cups\_dest\_t \* cupsGetDest(const char \* *name*, const char \* *instance*, int *num\_dests*, cups\_dest\_t \* *dests*);

#####  Description

Get the named destination from the list.

Use the cupsGetDests() or cupsGetDests2() functions to get a list of supported destinations for the current user.

#####  Return Value

Destination pointer or NULL

#### cupsGetDests

#####  Name

cupsGetDests — Get the list of destinations from the default server.

#####  Synopsis

#include <cups/cups.h>

int cupsGetDests(cups\_dest\_t \*\* *dests*);

#####  Description

Get the list of destinations from the default server.

Starting with CUPS 1.2, the returned list of destinations include the printer-info, printer-is-accepting-jobs, printer-is-shared, printer-make-and-model, printer-state, printer-state-change-time, printer-state-reasons, and printer-type attributes as options.

Use the cupsFreeDests() function to free the destination list and the cupsGetDest() function to find a particular destination.

#####  Return Value

Number of destinations

#### cupsGetJobs

#####  Name

cupsGetJobs — Get the jobs from the default server.

#####  Synopsis

#include <cups/cups.h>

int cupsGetJobs(cups\_job\_t \*\* *jobs*, const char \* *mydest*, int *myjobs*, int *completed*);

#####  Description

Get the jobs from the default server.

#####  Return Value

Number of jobs

#### cupsGetOption

#####  Name

cupsGetOption — Get an option value.

#####  Synopsis

#include <cups/cups.h>

const char \* cupsGetOption(const char \* *name*, int *num\_options*, cups\_option\_t \* *options*);

#####  Description

Get an option value.

#####  Return Value

Option value or NULL

#### cupsGetPPD

#####  Name

cupsGetPPD — Get the PPD file for a printer on the default server.

#####  Synopsis

#include <cups/cups.h>

const char \* cupsGetPPD(const char \* *name*);

#####  Description

Get the PPD file for a printer on the default server.

For classes, cupsGetPPD() returns the PPD file for the first printer in the class.

#####  Return Value

Filename for PPD file

#### cupsGetPassword

#####  Name

cupsGetPassword — Get a password from the user.

#####  Synopsis

#include <cups/cups.h>

const char \* cupsGetPassword(const char \* *prompt*);

#####  Description

Get a password from the user.

Uses the current password callback function. Returns NULL if the user does not provide a password.

#####  Return Value

Password

#### cupsLangEncoding

#####  Name

cupsLangEncoding — Return the character encoding (us-ascii, etc.)

#####  Synopsis

#include <cups/cups.h>

const char \* cupsLangEncoding(cups\_lang\_t \* *lang*);

#####  Description

Return the character encoding (us-ascii, etc.) for the given language.

#####  Return Value

Character encoding

#### cupsLangFlush

#####  Name

cupsLangFlush — Flush all language data out of the cache.

#####  Synopsis

#include <cups/cups.h>

void cupsLangFlush(void);

#####  Description

Flush all language data out of the cache.

#####  Return Value

This function does not return a value.

#### cupsLangFree

#####  Name

cupsLangFree — Free language data.

#####  Synopsis

#include <cups/cups.h>

void cupsLangFree(cups\_lang\_t \* *lang*);

#####  Description

Free language data.

This does not actually free anything; use cupsLangFlush() for that.

#####  Return Value

This function does not return a value.

#### cupsLangGet

#####  Name

cupsLangGet — Get a language.

#####  Synopsis

#include <cups/cups.h>

cups\_lang\_t \* cupsLangGet(const char \* *language*);

#####  Description

Get a language.

#####  Return Value

Language data

#### cupsLastError

#####  Name

cupsLastError — Return the last IPP status code.

#####  Synopsis

#include <cups/cups.h>

ipp\_status\_t cupsLastError(void);

#####  Description

Return the last IPP status code.

#####  Return Value

IPP status code from last request

#### cupsMarkOptions

#####  Name

cupsMarkOptions — Mark command-line options in a PPD file.

#####  Synopsis

#include <cups/cups.h>

int cupsMarkOptions(ppd\_file\_t \* *ppd*, int *num\_options*, cups\_option\_t \* *options*);

#####  Description

Mark command-line options in a PPD file.

#####  Return Value

1 if conflicting

#### cupsParseOptions

#####  Name

cupsParseOptions — Parse options from a command-line argument.

#####  Synopsis

#include <cups/cups.h>

int cupsParseOptions(const char \* *arg*, int *num\_options*, cups\_option\_t \*\* *options*);

#####  Description

Parse options from a command-line argument.

This function converts space-delimited name/value pairs according to the PAPI text option ABNF specification. Collection values ("name={a=... b=... c=...}") are stored with the curley brackets intact - use cupsParseOptions() on the value to extract the collection attributes.

#####  Return Value

Number of options found

#### cupsPrintFile

#####  Name

cupsPrintFile — Print a file to a printer or class on the default server.

#####  Synopsis

#include <cups/cups.h>

int cupsPrintFile(const char \* *name*, const char \* *filename*, const char \* *title*, int *num\_options*, cups\_option\_t \* *options*);

#####  Description

Print a file to a printer or class on the default server.

#####  Return Value

Job ID

#### cupsPrintFiles

#####  Name

cupsPrintFiles — Print one or more files to a printer or class on the

#####  Synopsis

#include <cups/cups.h>

int cupsPrintFiles(const char \* *name*, int *num\_files*, const char \*\* *files*, const char \* *title*, int *num\_options*, cups\_option\_t \* *options*);

#####  Description

Print one or more files to a printer or class on the default server.

#####  Return Value

Job ID

#### cupsServer

#####  Name

cupsServer — Return the hostname/address of the default server.

#####  Synopsis

#include <cups/cups.h>

const char \* cupsServer(void);

#####  Description

Return the hostname/address of the default server.

The returned value can be a fully-qualified hostname, a numeric IPv4 or IPv6 address, or a domain socket pathname.

#####  Return Value

Server name

#### cupsSetDests

#####  Name

cupsSetDests — Save the list of destinations for the default server.

#####  Synopsis

#include <cups/cups.h>

void cupsSetDests(int *num\_dests*, cups\_dest\_t \* *dests*);

#####  Description

Save the list of destinations for the default server.

This function saves the destinations to /etc/cups/lpoptions when run as root and ~/.cups/lpoptions when run as a normal user.

#####  Return Value

This function does not return a value.

#### cupsSetEncryption

#####  Name

cupsSetEncryption — Set the encryption preference.

#####  Synopsis

#include <cups/cups.h>

void cupsSetEncryption(http\_encryption\_t *e*);

#####  Description

Set the encryption preference.

#####  Return Value

This function does not return a value.

#### cupsSetPasswordCB

#####  Name

cupsSetPasswordCB — Set the password callback for CUPS.

#####  Synopsis

#include <cups/cups.h>

void cupsSetPasswordCB(cups\_password\_cb\_t *cb*);

#####  Description

Set the password callback for CUPS.

Pass NULL to restore the default (console) password callback.

#####  Return Value

This function does not return a value.

#### cupsSetServer

#####  Name

cupsSetServer — Set the default server name.

#####  Synopsis

#include <cups/cups.h>

void cupsSetServer(const char \* *server*);

#####  Description

Set the default server name.

The "server" string can be a fully-qualified hostname, a numeric IPv4 or IPv6 address, or a domain socket pathname. Pass NULL to restore the default server name.

#####  Return Value

This function does not return a value.

#### cupsSetUser

#####  Name

cupsSetUser — Set the default user name.

#####  Synopsis

#include <cups/cups.h>

void cupsSetUser(const char \* *user*);

#####  Description

Set the default user name.

Pass NULL to restore the default user name.

#####  Return Value

This function does not return a value.

#### cupsTempFd

#####  Name

cupsTempFd — Creates a temporary file.

#####  Synopsis

#include <cups/cups.h>

int cupsTempFd(char \* *filename*, int *len*);

#####  Description

Creates a temporary file.

The temporary filename is returned in the filename buffer. The temporary file is opened for reading and writing.

#####  Return Value

New file descriptor or -1 on error

#### cupsUser

#####  Name

cupsUser — Return the current user's name.

#####  Synopsis

#include <cups/cups.h>

const char \* cupsUser(void);

#####  Description

Return the current user's name.

#####  Return Value

User name

#### ppdClose

#####  Name

ppdClose — Free all memory used by the PPD file.

#####  Synopsis

#include <cups/cups.h>

void ppdClose(ppd\_file\_t \* *ppd*);

#####  Description

Free all memory used by the PPD file.

#####  Return Value

This function does not return a value.

#### ppdCollect

#####  Name

ppdCollect — Collect all marked options that reside in the specified

#####  Synopsis

#include <cups/cups.h>

int ppdCollect(ppd\_file\_t \* *ppd*, ppd\_section\_t *section*, ppd\_choice\_t \*\*\* *choices*);

#####  Description

Collect all marked options that reside in the specified section.

#####  Return Value

Number of options marked

#### ppdConflicts

#####  Name

ppdConflicts — Check to see if there are any conflicts.

#####  Synopsis

#include <cups/cups.h>

int ppdConflicts(ppd\_file\_t \* *ppd*);

#####  Description

Check to see if there are any conflicts.

#####  Return Value

Number of conflicts found

#### ppdEmit

#####  Name

ppdEmit — Emit code for marked options to a file.

#####  Synopsis

#include <cups/cups.h>

int ppdEmit(ppd\_file\_t \* *ppd*, FILE \* *fp*, ppd\_section\_t *section*);

#####  Description

Emit code for marked options to a file.

#####  Return Value

0 on success, -1 on failure

#### ppdEmitFd

#####  Name

ppdEmitFd — Emit code for marked options to a file.

#####  Synopsis

#include <cups/cups.h>

int ppdEmitFd(ppd\_file\_t \* *ppd*, int *fd*, ppd\_section\_t *section*);

#####  Description

Emit code for marked options to a file.

#####  Return Value

0 on success, -1 on failure

#### ppdEmitJCL

#####  Name

ppdEmitJCL — Emit code for JCL options to a file.

#####  Synopsis

#include <cups/cups.h>

int ppdEmitJCL(ppd\_file\_t \* *ppd*, FILE \* *fp*, int *job\_id*, const char \* *user*, const char \* *title*);

#####  Description

Emit code for JCL options to a file.

#####  Return Value

0 on success, -1 on failure

#### ppdErrorString

#####  Name

ppdErrorString — Returns the text assocated with a status.

#####  Synopsis

#include <cups/cups.h>

const char \* ppdErrorString(ppd\_status\_t *status*);

#####  Description

Returns the text assocated with a status.

#####  Return Value

Status string

#### ppdFindAttr

#####  Name

ppdFindAttr — Find the first matching attribute...

#####  Synopsis

#include <cups/cups.h>

ppd\_attr\_t \* ppdFindAttr(ppd\_file\_t \* *ppd*, const char \* *name*, const char \* *spec*);

#####  Description

Find the first matching attribute...

#####  Return Value

Attribute or NULL if not found

#### ppdFindChoice

#####  Name

ppdFindChoice — Return a pointer to an option choice.

#####  Synopsis

#include <cups/cups.h>

ppd\_choice\_t \* ppdFindChoice(ppd\_option\_t \* *o*, const char \* *choice*);

#####  Description

Return a pointer to an option choice.

#####  Return Value

Choice pointer or NULL

#### ppdFindMarkedChoice

#####  Name

ppdFindMarkedChoice — Return the marked choice for the specified option.

#####  Synopsis

#include <cups/cups.h>

ppd\_choice\_t \* ppdFindMarkedChoice(ppd\_file\_t \* *ppd*, const char \* *option*);

#####  Description

Return the marked choice for the specified option.

#####  Return Value

Pointer to choice or NULL

#### ppdFindNextAttr

#####  Name

ppdFindNextAttr — Find the next matching attribute...

#####  Synopsis

#include <cups/cups.h>

ppd\_attr\_t \* ppdFindNextAttr(ppd\_file\_t \* *ppd*, const char \* *name*, const char \* *spec*);

#####  Description

Find the next matching attribute...

#####  Return Value

Attribute or NULL if not found

#### ppdFindOption

#####  Name

ppdFindOption — Return a pointer to the specified option.

#####  Synopsis

#include <cups/cups.h>

ppd\_option\_t \* ppdFindOption(ppd\_file\_t \* *ppd*, const char \* *option*);

#####  Description

Return a pointer to the specified option.

#####  Return Value

Pointer to option or NULL

#### ppdIsMarked

#####  Name

ppdIsMarked — Check to see if an option is marked...

#####  Synopsis

#include <cups/cups.h>

int ppdIsMarked(ppd\_file\_t \* *ppd*, const char \* *option*, const char \* *choice*);

#####  Description

Check to see if an option is marked...

#####  Return Value

Non-zero if option is marked

#### ppdLastError

#####  Name

ppdLastError — Return the status from the last ppdOpen\*().

#####  Synopsis

#include <cups/cups.h>

ppd\_status\_t ppdLastError(int \* *line*);

#####  Description

Return the status from the last ppdOpen\*().

#####  Return Value

Status code

#### ppdMarkDefaults

#####  Name

ppdMarkDefaults — Mark all default options in the PPD file.

#####  Synopsis

#include <cups/cups.h>

void ppdMarkDefaults(ppd\_file\_t \* *ppd*);

#####  Description

Mark all default options in the PPD file.

#####  Return Value

This function does not return a value.

#### ppdMarkOption

#####  Name

ppdMarkOption — Mark an option in a PPD file.

#####  Synopsis

#include <cups/cups.h>

int ppdMarkOption(ppd\_file\_t \* *ppd*, const char \* *option*, const char \* *choice*);

#####  Description

Mark an option in a PPD file.

Notes:

-1 is returned if the given option would conflict with any currently selected option.

#####  Return Value

Number of conflicts

#### ppdOpen

#####  Name

ppdOpen — Read a PPD file into memory.

#####  Synopsis

#include <cups/cups.h>

ppd\_file\_t \* ppdOpen(FILE \* *fp*);

#####  Description

Read a PPD file into memory.

#####  Return Value

PPD file record

#### ppdOpenFd

#####  Name

ppdOpenFd — Read a PPD file into memory.

#####  Synopsis

#include <cups/cups.h>

ppd\_file\_t \* ppdOpenFd(int *fd*);

#####  Description

Read a PPD file into memory.

#####  Return Value

PPD file record

#### ppdOpenFile

#####  Name

ppdOpenFile — Read a PPD file into memory.

#####  Synopsis

#include <cups/cups.h>

ppd\_file\_t \* ppdOpenFile(const char \* *filename*);

#####  Description

Read a PPD file into memory.

#####  Return Value

PPD file record

#### ppdPageLength

#####  Name

ppdPageLength — Get the page length for the given size.

#####  Synopsis

#include <cups/cups.h>

float ppdPageLength(ppd\_file\_t \* *ppd*, const char \* *name*);

#####  Description

Get the page length for the given size.

#####  Return Value

Length of page in points or 0.0

#### ppdPageSize

#####  Name

ppdPageSize — Get the page size record for the given size.

#####  Synopsis

#include <cups/cups.h>

ppd\_size\_t \* ppdPageSize(ppd\_file\_t \* *ppd*, const char \* *name*);

#####  Description

Get the page size record for the given size.

#####  Return Value

Size record for page or NULL

#### ppdPageWidth

#####  Name

ppdPageWidth — Get the page width for the given size.

#####  Synopsis

#include <cups/cups.h>

float ppdPageWidth(ppd\_file\_t \* *ppd*, const char \* *name*);

#####  Description

Get the page width for the given size.

#####  Return Value

Width of page in points or 0.0

#### ppdSetConformance

#####  Name

ppdSetConformance — Set the conformance level for PPD files.

#####  Synopsis

#include <cups/cups.h>

void ppdSetConformance(ppd\_conform\_t *c*);

#####  Description

Set the conformance level for PPD files.

#####  Return Value

This function does not return a value.

## **7.4 Interfaces for libcupsimage**

[Table 7-4](#ID_LIB_45_LIBCUPSIMAGE_45_DEF) defines the library name and shared object name for the libcupsimage library

**Table 7-4 libcupsimage Definition**

|  |  |
| --- | --- |
| Library: | libcupsimage |
| SONAME: | libcupsimage.so.2 |

The behavior of the interfaces in this library is specified by the following specifications:

|  |
| --- |
| [LSB] [This Specification](#ID_STD_46_LSB) |

### **7.4.1 CUPS Raster ABI**

#### 7.4.1.1 Interfaces for CUPS Raster ABI

An LSB conforming implementation shall provide the generic functions for CUPS Raster ABI specified in [Table 7-5](#ID_TBL_45_LIBCUPSIMAGE_45_CUPS_45_INTS), with the full mandatory functionality as described in the referenced underlying specification.

**Table 7-5 libcupsimage - CUPS Raster ABI Function Interfaces**

|  |  |  |  |
| --- | --- | --- | --- |
| cupsRasterClose [[LSB]](#ID_REFSTD_46_LIBCUPSIMAGE_46_1) | cupsRasterOpen [[LSB]](#ID_REFSTD_46_LIBCUPSIMAGE_46_1) | cupsRasterReadHeader [[LSB]](#ID_REFSTD_46_LIBCUPSIMAGE_46_1) | cupsRasterReadPixels [[LSB]](#ID_REFSTD_46_LIBCUPSIMAGE_46_1) |
| cupsRasterWriteHeader [[LSB]](#ID_REFSTD_46_LIBCUPSIMAGE_46_1) | cupsRasterWritePixels [[LSB]](#ID_REFSTD_46_LIBCUPSIMAGE_46_1) |  |  |

An LSB conforming implementation shall provide the generic deprecated functions for CUPS Raster ABI specified in [Table 7-6](#ID_TBL_45_LIBCUPSIMAGE_45_CUPS_45_DEPINTS), with the full mandatory functionality as described in the referenced underlying specification.

**Note:** These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

**Table 7-6 libcupsimage - CUPS Raster ABI Deprecated Function Interfaces**

|  |  |  |  |
| --- | --- | --- | --- |
| cupsRasterReadHeader [[LSB]](#ID_REFSTD_46_LIBCUPSIMAGE_46_1) | cupsRasterWriteHeader [[LSB]](#ID_REFSTD_46_LIBCUPSIMAGE_46_1) |  |  |

## **7.5 Data Definitions for libcupsimage**

This section defines global identifiers and their values that are associated with interfaces contained in libcupsimage. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the [ISO C (1999)](#ID_STD_46_ISOC99) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

### **7.5.1 cups/raster.h**

#define \_CUPS\_RASTER\_H\_

#define CUPS\_RASTER\_SYNC 0x52615374

#define CUPS\_RASTER\_REVSYNC 0x74536152

#define CUPS\_RASTER\_HAVE\_COLORIMETRIC 1

typedef enum {

 CUPS\_RASTER\_READ = 0,

 CUPS\_RASTER\_WRITE = 1

} cups\_mode\_t;

typedef struct \_cups\_raster\_s cups\_raster\_t;

typedef enum {

 CUPS\_ADVANCE\_NONE = 0,

 CUPS\_ADVANCE\_FILE = 1,

 CUPS\_ADVANCE\_JOB = 2,

 CUPS\_ADVANCE\_SET = 3,

 CUPS\_ADVANCE\_PAGE = 4

} cups\_adv\_t;

typedef enum {

 CUPS\_FALSE = 0,

 CUPS\_TRUE = 1

} cups\_bool\_t;

typedef enum {

 CUPS\_CUT\_NONE = 0,

 CUPS\_CUT\_FILE = 1,

 CUPS\_CUT\_JOB = 2,

 CUPS\_CUT\_SET = 3,

 CUPS\_CUT\_PAGE = 4

} cups\_cut\_t;

typedef enum {

 CUPS\_JOG\_NONE = 0,

 CUPS\_JOG\_FILE = 1,

 CUPS\_JOG\_JOB = 2,

 CUPS\_JOG\_SET = 3

} cups\_jog\_t;

typedef enum {

 CUPS\_EDGE\_TOP = 0,

 CUPS\_EDGE\_RIGHT = 1,

 CUPS\_EDGE\_BOTTOM = 2,

 CUPS\_EDGE\_LEFT = 3

} cups\_edge\_t;

typedef enum {

 CUPS\_ORIENT\_0 = 0,

 CUPS\_ORIENT\_90 = 1,

 CUPS\_ORIENT\_180 = 2,

 CUPS\_ORIENT\_270 = 3

} cups\_orient\_t;

typedef enum {

 CUPS\_ORDER\_CHUNKED = 0,

 CUPS\_ORDER\_BANDED = 1,

 CUPS\_ORDER\_PLANAR = 2

} cups\_order\_t;

typedef enum {

 CUPS\_CSPACE\_W = 0,

 CUPS\_CSPACE\_RGB = 1,

 CUPS\_CSPACE\_RGBA = 2,

 CUPS\_CSPACE\_K = 3,

 CUPS\_CSPACE\_CMY = 4,

 CUPS\_CSPACE\_YMC = 5,

 CUPS\_CSPACE\_CMYK = 6,

 CUPS\_CSPACE\_YMCK = 7,

 CUPS\_CSPACE\_KCMY = 8,

 CUPS\_CSPACE\_KCMYcm = 9,

 CUPS\_CSPACE\_GMCK = 10,

 CUPS\_CSPACE\_GMCS = 11,

 CUPS\_CSPACE\_WHITE = 12,

 CUPS\_CSPACE\_GOLD = 13,

 CUPS\_CSPACE\_SILVER = 14,

 CUPS\_CSPACE\_CIEXYZ = 15,

 CUPS\_CSPACE\_CIELab = 16,

 CUPS\_CSPACE\_ICC1 = 32,

 CUPS\_CSPACE\_ICC2 = 33,

 CUPS\_CSPACE\_ICC3 = 34,

 CUPS\_CSPACE\_ICC4 = 35,

 CUPS\_CSPACE\_ICC5 = 36,

 CUPS\_CSPACE\_ICC6 = 37,

 CUPS\_CSPACE\_ICC7 = 38,

 CUPS\_CSPACE\_ICC8 = 39,

 CUPS\_CSPACE\_ICC9 = 40,

 CUPS\_CSPACE\_ICCA = 41,

 CUPS\_CSPACE\_ICCB = 42,

 CUPS\_CSPACE\_ICCC = 43,

 CUPS\_CSPACE\_ICCD = 44,

 CUPS\_CSPACE\_ICCE = 45,

 CUPS\_CSPACE\_ICCF = 46

} cups\_cspace\_t;

typedef struct {

 char MediaClass[64];

 char MediaColor[64];

 char MediaType[64];

 char OutputType[64];

 unsigned int AdvanceDistance;

 cups\_adv\_t AdvanceMedia;

 cups\_bool\_t Collate;

 cups\_cut\_t CutMedia;

 cups\_bool\_t Duplex;

 unsigned int HWResolution[2];

 unsigned int ImagingBoundingBox[4];

 cups\_bool\_t InsertSheet;

 cups\_jog\_t Jog;

 cups\_edge\_t LeadingEdge;

 unsigned int Margins[2];

 cups\_bool\_t ManualFeed;

 unsigned int MediaPosition;

 unsigned int MediaWeight;

 cups\_bool\_t MirrorPrint;

 cups\_bool\_t NegativePrint;

 unsigned int NumCopies;

 cups\_orient\_t Orientation;

 cups\_bool\_t OutputFaceUp;

 unsigned int PageSize[2];

 cups\_bool\_t Separations;

 cups\_bool\_t TraySwitch;

 cups\_bool\_t Tumble;

 unsigned int cupsWidth;

 unsigned int cupsHeight;

 unsigned int cupsMediaType;

 unsigned int cupsBitsPerColor;

 unsigned int cupsBitsPerPixel;

 unsigned int cupsBytesPerLine;

 cups\_order\_t cupsColorOrder;

 cups\_cspace\_t cupsColorSpace;

 unsigned int cupsCompression;

 unsigned int cupsRowCount;

 unsigned int cupsRowFeed;

 unsigned int cupsRowStep;

} cups\_page\_header\_t;

typedef struct cups\_page\_header2\_s {

 char MediaClass[64];

 char MediaColor[64];

 char MediaType[64];

 char OutputType[64];

 unsigned int AdvanceDistance;

 cups\_adv\_t AdvanceMedia;

 cups\_bool\_t Collate;

 cups\_cut\_t CutMedia;

 cups\_bool\_t Duplex;

 unsigned int HWResolution[3];

 unsigned int ImagingBoundingBox[4];

 cups\_bool\_t InsertSheet;

 cups\_jog\_t Jog;

 cups\_edge\_t LeadingEdge;

 unsigned int Margins[3];

 cups\_bool\_t ManualFeed;

 unsigned int MediaPosition;

 unsigned int MediaWeight;

 cups\_bool\_t MirrorPrint;

 cups\_bool\_t NegativePrint;

 unsigned int NumCopies;

 cups\_orient\_t Orientation;

 cups\_bool\_t OutputFaceUp;

 unsigned int PageSize[3];

 cups\_bool\_t Separations;

 cups\_bool\_t TraySwitch;

 cups\_bool\_t Tumble;

 unsigned int cupsWidth;

 unsigned int cupsHeight;

 unsigned int cupsMediaType;

 unsigned int cupsBitsPerColor;

 unsigned int cupsBitsPerPixel;

 unsigned int cupsBytesPerLine;

 cups\_order\_t cupsColorOrder;

 cups\_cspace\_t cupsColorSpace;

 unsigned int cupsCompression;

 unsigned int cupsRowCount;

 unsigned int cupsRowFeed;

 unsigned int cupsRowStep;

 unsigned int cupsNumColors;

 float cupsBorderlessScalingFactor;

 float cupsPageSize[2];

 float cupsImagingBBox[4];

 unsigned int cupsInteger[16];

 float cupsReal[16];

 char cupsString[16][64];

 char cupsMarkerType[64];

 char cupsRenderingIntent[64];

 char cupsPageSizeName[64];

} cups\_page\_header2\_t;

typedef int (\*cups\_interpret\_cb\_t) (cups\_page\_header2\_t \*, int);

extern void cupsRasterClose(cups\_raster\_t \* r);

extern cups\_raster\_t \*cupsRasterOpen(int fd, cups\_mode\_t mode);

extern unsigned int cupsRasterReadHeader(cups\_raster\_t \* r,

 cups\_page\_header\_t \* h);

extern unsigned int cupsRasterReadHeader2(cups\_raster\_t \* r,

 cups\_page\_header2\_t \* h);

extern unsigned int cupsRasterReadPixels(cups\_raster\_t \* r,

 unsigned char \*p,

 unsigned int len);

extern unsigned int cupsRasterWriteHeader(cups\_raster\_t \* r,

 cups\_page\_header\_t \* h);

extern unsigned int cupsRasterWriteHeader2(cups\_raster\_t \* r,

 cups\_page\_header2\_t \* h);

extern unsigned int cupsRasterWritePixels(cups\_raster\_t \* r,

 unsigned char \*p,

 unsigned int len);

## **7.6 Interface Definitions for libcupsimage**

The interfaces defined on the following pages are included in libcupsimage and are defined by this specification. Unless otherwise noted, these interfaces shall be included in the source standard.

Other interfaces listed in [Section 7.4](#ID_LIBCUPSIMAGE) shall behave as described in the referenced base document.

#### cupsRasterClose

#####  Name

cupsRasterClose — Close a raster stream.

#####  Synopsis

#include <cups/raster.h>

void cupsRasterClose(cups\_raster\_t \* *r*);

#####  Description

Close a raster stream.

#####  Return Value

This function does not return a value.

#### cupsRasterOpen

#####  Name

cupsRasterOpen — Open a raster stream.

#####  Synopsis

#include <cups/raster.h>

cups\_raster\_t \* cupsRasterOpen(int *fd*, cups\_mode\_t *mode*);

#####  Description

Open a raster stream.

#####  Return Value

New stream

#### cupsRasterReadHeader

#####  Name

cupsRasterReadHeader — Read a raster page header and store it in a

#####  Synopsis

#include <cups/raster.h>

unsigned cupsRasterReadHeader(cups\_raster\_t \* *r*, cups\_page\_header\_t \* *h*);

#####  Description

Read a raster page header and store it in a V1 page header structure.

#####  Return Value

1 on success, 0 on fail

#### cupsRasterReadPixels

#####  Name

cupsRasterReadPixels — Read raster pixels.

#####  Synopsis

#include <cups/raster.h>

unsigned cupsRasterReadPixels(cups\_raster\_t \* *r*, unsigned char \* *p*, unsigned *len*);

#####  Description

Read raster pixels.

#####  Return Value

Number of bytes read

#### cupsRasterWriteHeader

#####  Name

cupsRasterWriteHeader — Write a raster page header from a V1 page

#####  Synopsis

#include <cups/raster.h>

unsigned cupsRasterWriteHeader(cups\_raster\_t \* *r*, cups\_page\_header\_t \* *h*);

#####  Description

Write a raster page header from a V1 page header structure.

#####  Return Value

1 on success, 0 on failure

#### cupsRasterWritePixels

#####  Name

cupsRasterWritePixels — Write raster pixels.

#####  Synopsis

#include <cups/raster.h>

unsigned cupsRasterWritePixels(cups\_raster\_t \* *r*, unsigned char \* *p*, unsigned *len*);

#####  Description

Write raster pixels.

#####  Return Value

Number of bytes written

# **III Printing Commands**

# **8 Printing Commands**

## **8.1 Commands and Utilities**

An LSB conforming implementation shall provide the commands and utilities as described in [Table 8-1](#ID_TBL_45_CMDS), with at least the behavior described as mandatory in the referenced underlying specification, with the following exceptions:

 1. If any operand (except one which follows --) starts with a hyphen, the behavior is unspecified.

**Rationale (Informative):** Applications should place options before operands, or use --, as needed. This text is needed because, by default, GNU option parsing differs from POSIX, unless the environment variable POSIXLY\_CORRECT is set. For example, **ls . -a** in GNU **ls** means to list the current directory, showing all files (that is, "." is an operand and -a is an option). In POSIX, "." and -a are both operands, and the command means to list the current directory, and also the file named -a. Suggesting that applications rely on the setting of the POSIXLY\_CORRECT environment variable, or try to set it, seems worse than just asking the applications to invoke commands in ways which work with either the POSIX or GNU behaviors.

**Table 8-1 Commands And Utilities**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| foomatic-rip [[1]](#ID_STD_45_CMD_45_10) | gs [[1]](#ID_STD_45_CMD_45_10) |  |  |  |

*Referenced Specification(s)*

**[1].** [This Specification](#ID_STD_46_LSB)

## **8.2 Command Behavior**

This section contains descriptions for commands and utilities whose specified behavior in the LSB contradicts or extends the standards referenced. It also contains commands and utilities only required by the LSB and not specified by other standards.

#### foomatic-rip

2002-11-26

#####  Name

foomatic-rip — Universal print filter/RIP wrapper

##### SYNOPSIS

##### Standalone Mode

foomatic-rip [-v] [-q] [-d] [ --ppd ppdfile ] [ -J jobtitle ] [ -o option=value [...] ] [ files ]

##### CUPS Mode

foomatic-rip jobid user jobtitle numcopies options [file]

##### DESCRIPTION

foomatic-rip is a universal print filter which works with every known free software printer spooler.

This page describes the facilities of foomatic-rip when used as a CUPS filter and when used outside of a print system. While implementations of foomatic-rip may support other print systems, such use is not documented here.

When run as a CUPS filter, foomatic-rip reads the job from the specified file, or from standard input if no file is specified. It renders the file into a printer-specific format, and writes the result to standard output.

When run standalone, foomatic-rip will read the job from the specified files, or from standard input if no files are given. The files are rendered into a printer-specific format, which is then output according to the PPD option "FoomaticRIPPostPipe", documented in the LSB.

Printer capabilities are described to foomatic-rip via PPD files, as described (with extensions used by foomatic-rip) in the LSB. The method foomatic-rip uses to determine the proper PPD file for the printer in question is defined by the implementation of both the spooler and foomatic-rip.

##### CUPS OPTIONS

Unless otherwise noted, all parameters are required when running foomatic-rip as a CUPS filter.

*jobid*

  The internal Job ID from CUPS.

*username*

  The username of the user who submitted the job.

*jobtitle*

  The job's title, as submitted by the user.

*numcopies*

  The number of copies of the job requested.

*options*

  A series of printer options, separated by spaces, each of which take the form *name* or *name=value*. The specific list of options supported is dependent on the printer and spooler, and is usually documented in the PPD file for the printer.

An option may be preceded by a page specification, describing the pages to which the option should apply. A page specification consists of one or more items, separated by commas, and separated from the option name by a colon. Valid items include the words "even" and "odd", a single page number, and a page range. Page ranges are described with a starting page, a dash ("-"), and an ending page. If omitted, the starting and ending pages are the first and last page, respectively, but only one of the ends of the range may be omitted.

*file*

  The full path to the file containing the job. This parameter is optional; if it is not supplied, the job is read from standard input.

##### SPOOLER-LESS OPTIONS

-v

  Verbose mode. Intended for debugging and testing purposes only.

-q

  Quiet mode - minimal information output.

-d

  Identical to the 'opts' option, but option information is left in text format. The PPD file will need to be specified using the --ppd option.

--ppd *ppdfile*

  The PPD file ppdfile should be applied for processing this job.

-J *jobtitle*

  Print the given job title in the header of every page of a plain text job.

-o *option=value*

  Set an option setting for this job.

##### EXIT STATUS

*foomatic-rip* returns 0 unless something unexpected happens.

##### AUTHOR

Till Kamppeter <*till.kamppeter@gmail.com*> with parts of Manfred Wassmanns's <*manolo@NCC-1701.B.Shuttle.de*> man pages for the Foomatic 2.0.x filters.

Jeff Licquia <*licquia@linux-foundation.org*> adapted the original man page for the LSB.

#### gs

2007-11-29

#####  Name

gs — GhostScript (PostScript and PDF language interpreter)

##### SYNOPSIS

gs -h | --help

gs [ options ] ps-file [ [ options ] ps-file2 ] ...

##### DESCRIPTION

The gs command invokes Ghostscript, an interpreter of Adobe Systems' PostScript(tm) and Portable Document Format (PDF) languages. gs reads the files named by ps-file in sequence and executes them as Ghostscript programs. After doing this, it reads further input from the standard input stream (normally the keyboard), interpreting each line separately. The interpreter exits gracefully when it encounters the "quit" command (either in a file or from the keyboard), at end-of-file, or at an interrupt signal (such as Control-C at the keyboard).

Some of GhostScript's options are set via command-line options; others are set as processing parameters, each of which consists of a name and a value.

##### OPTIONS

*-h --help*

  Show GhostScript's help, as well as lists of the supported input formats, supported devices, and the search path for gs components.

*-q*

  Suppress normal startup messages, and also set the processing parameter QUIET.

*-c*

  Begin interpreting arguments as PostScript code. All following arguments are sent to the interpreter up to the next argument beginning with "-" followed by a non-digit, or with "@". This code is interpolated with the file list, so files specified before *-c* are interpreted beforehand, and files after *-c* are interpreted afterwards.

*-f*

  Specifies a PostScript file to run as its argument. This is equivalent to the ps-file arguments, but is useful for terminating PostScript code as passed via *-c*.

*-d -D*

  Set a processing parameter. The "name=value" pair follows immediately after the option, as in "-Dfoo=bar". The values here must be integers or the values "true" or "false". The equals sign and value may be omitted; this is assumed to set the name to "true".

*-s -S*

  Set a processing parameter to a string value. The "name=value" pair follows immediately after the option, as in "-Sfoo=bar".

*-u*

  Unset a processing parameter. The name to be unset follows immediately after the option, as in "-ufoo".

*-o*

  Write rendered output to the named file, and also inhibit pauses and the interactive shell. This is equivalent to setting the processing parameters BATCH and NOPAUSE to true, and OutputFile to the parameter after -o.

*-r*

  Set the device resolution. The resolution is specified as two numbers separated with an "x", as in "300x150", corresponding to the X and Y axis resolutions, respectively. If a single number is given without an "x", it is treated as the value for both resolutions.

This is equivalent to setting DEVICEXRESOLUTION and DEVICEYRESOLUTION in systemdict.

*-g*

  Set the device size, in pixels. The size is specified as two numbers separated with an "x", as in "640x480", corresponding to the width and height, respectively.

This is equivalent to setting DEVICEWIDTH and DEVICEHEIGHT in systemdict.

#### RECOGNIZED PROCESSING PARAMETERS

Processing parameters may have arbitrary names; no limits are placed on the settings that may be made. However, certain settings have meaning to the gs interpreter, and drivers may use other settings. Below is a list of recognized settings that the gs interpreter must respect.

*BATCH*

  If set to true, do not enter an interactive shell after processing all command-line files.

*DEVICE*

  Contains the name of the device used to render the page, as a string.

The list of available devices can be discovered with the -h parameter, as described above. At least the following devices must be present: cups (CUPS Raster), ijs, pxlmono, pxlcolor, and opvp (OpenPrinting Vector).

*DEVICEHEIGHT*

  Contains the height, in pixels, of the output device.

The effect of this setting when the current driver is a vector driver is undefined.

*DEVICEHEIGHTPOINTS*

  Sets the initial page height, in units of 1/72 of an inch.

*DEVICEWIDTH*

  Contains the width, in pixels, of the output device.

The effect of this setting when the current driver is a vector driver is undefined.

*DEVICEWIDTHPOINTS*

  Sets the initial page width, in units of 1/72 of an inch.

*DEVICEXRESOLUTION*

  Contains the resolution, in pixels per inch, of the X dimension (horizontal) of the output device.

*DEVICEYRESOLUTION*

  Contains the resolution, in pixels per inch, of the Y dimension (vertical) of the output device.

*NOPAUSE*

  If set to true, disable the prompt and pause normally displayed after rendering a page.

*OutputFile*

  Contains the path to the file to which gs should write its output, as a string. This parameter may be set to '-', in which case gs's output is sent to standard output.

*PAPERSIZE*

  Contains the string representation of the paper size. The ISO paper sizes a0-a10 (plus a4small), isob0-isob6, and c0-c6 are recognized, as are jisb0-jisb6 (JIS standard sizes) and the US paper sizes 11x17, ledger, legal, letter, lettersmall, and archA-archE.

*QUIET*

  If set to true, suppress routine information comments on standard output.

*SAFER*

  If set to true, disable several unsafe PostScript features: the deletefile and renamefile operators, piped commands, reading or writing to general files, and changing of certain system settings.

*STRICT*

  If set to true, disable as many extensions to the Adobe PostScript specification as possible.

##### EXIT STATUS

gs returns 0 on successful execution. Any other return value indicates an error.

##### AUTHOR

Jeff Licquia (licquia@linux-foundation.org) wrote this man page for the LSB specification.

Portions of this page were taken from the GhostScript documentation. The maintainer and rights holder for GhostScript is Artifex Software, Inc.

# **IV Execution Environment**

# **9 File System Hierarchy**

In addition to the requirements for /usr/share in the [Filesystem Hierarchy Standard](#ID_STD_46_FHS), an LSB conforming system shall also provide the following directories or symbolic links to directories:

/usr/share/ppd

#####   PostScript Printer Description (ppd) files

# **V Scanning Libraries**

# **10 Libraries**

## **10.1 Interfaces for libsane**

[Table 10-1](#ID_LIB_45_LIBSANE_45_DEF) defines the library name and shared object name for the libsane library

**Table 10-1 libsane Definition**

|  |  |
| --- | --- |
| Library: | libsane |
| SONAME: | libsane.so.1 |

The behavior of the interfaces in this library is specified by the following specifications:

|  |
| --- |
| [SANE] [SANE Standard Version 1.04](#ID_STD_46_SANE) |

### **10.1.1 libsane interfaces**

#### 10.1.1.1 Interfaces for libsane interfaces

An LSB conforming implementation shall provide the generic functions for libsane interfaces specified in [Table 10-2](#ID_TBL_45_LIBSANE_45_LIBSA_45_INTS), with the full mandatory functionality as described in the referenced underlying specification.

**Table 10-2 libsane - libsane interfaces Function Interfaces**

|  |  |  |  |
| --- | --- | --- | --- |
| sane\_cancel [[SANE]](#ID_REFSTD_46_LIBSANE_46_1) | sane\_close [[SANE]](#ID_REFSTD_46_LIBSANE_46_1) | sane\_control\_option [[SANE]](#ID_REFSTD_46_LIBSANE_46_1) | sane\_exit [[SANE]](#ID_REFSTD_46_LIBSANE_46_1) |
| sane\_get\_devices [[SANE]](#ID_REFSTD_46_LIBSANE_46_1) | sane\_get\_option\_descriptor [[SANE]](#ID_REFSTD_46_LIBSANE_46_1) | sane\_get\_parameters [[SANE]](#ID_REFSTD_46_LIBSANE_46_1) | sane\_get\_select\_fd [[SANE]](#ID_REFSTD_46_LIBSANE_46_1) |
| sane\_init [[SANE]](#ID_REFSTD_46_LIBSANE_46_1) | sane\_open [[SANE]](#ID_REFSTD_46_LIBSANE_46_1) | sane\_read [[SANE]](#ID_REFSTD_46_LIBSANE_46_1) | sane\_set\_io\_mode [[SANE]](#ID_REFSTD_46_LIBSANE_46_1) |
| sane\_start [[SANE]](#ID_REFSTD_46_LIBSANE_46_1) | sane\_strstatus [[SANE]](#ID_REFSTD_46_LIBSANE_46_1) |  |  |

## **10.2 Data Definitions for libsane**

This section defines global identifiers and their values that are associated with interfaces contained in libsane. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the [ISO C (1999)](#ID_STD_46_ISOC99) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

### **10.2.1 sane/sane.h**

#define sane\_h

#define SANE\_VERSION\_CODE(major,minor,build) ( (((SANE\_Word) (major) & 0xff) << 24) | (((SANE\_Word) (minor) & 0xff) << 16) | (((SANE\_Word) (build) & 0xffff) << 0))

#define SANE\_VERSION\_BUILD(code) ((((SANE\_Word)(code)) >> 0) & 0xffff)

#define SANE\_VERSION\_MINOR(code) ((((SANE\_Word)(code)) >> 16) & 0xff)

#define SANE\_VERSION\_MAJOR(code) ((((SANE\_Word)(code)) >> 24) & 0xff)

#define SANE\_OPTION\_IS\_ACTIVE(cap) (((cap) & SANE\_CAP\_INACTIVE) == 0)

#define SANE\_OPTION\_IS\_SETTABLE(cap) (((cap) & SANE\_CAP\_SOFT\_SELECT) != 0)

#define SANE\_UNFIX(v) ((double)(v) / (1 << SANE\_FIXED\_SCALE\_SHIFT))

#define SANE\_FIX(v) ((SANE\_Word) ((v) \* (1 << SANE\_FIXED\_SCALE\_SHIFT)))

#define SANE\_CAP\_SOFT\_SELECT (1 << 0)

#define SANE\_INFO\_INEXACT (1 << 0)

#define SANE\_CAP\_HARD\_SELECT (1 << 1)

#define SANE\_INFO\_RELOAD\_OPTIONS (1 << 1)

#define SANE\_CAP\_SOFT\_DETECT (1 << 2)

#define SANE\_INFO\_RELOAD\_PARAMS (1 << 2)

#define SANE\_CAP\_EMULATED (1 << 3)

#define SANE\_CAP\_AUTOMATIC (1 << 4)

#define SANE\_CAP\_INACTIVE (1 << 5)

#define SANE\_CAP\_ADVANCED (1 << 6)

#define SANE\_CURRENT\_MINOR 0

#define SANE\_FALSE 0

#define SANE\_CURRENT\_MAJOR 1

#define SANE\_TRUE 1

#define SANE\_MAX\_PASSWORD\_LEN 128

#define SANE\_MAX\_USERNAME\_LEN 128

#define SANE\_FIXED\_SCALE\_SHIFT 16

typedef unsigned char SANE\_Byte;

typedef int SANE\_Word;

typedef int SANE\_Bool;

typedef int SANE\_Int;

typedef char SANE\_Char;

typedef SANE\_Char \*SANE\_String;

typedef const SANE\_Char \*SANE\_String\_Const;

typedef void \*SANE\_Handle;

typedef int SANE\_Fixed;

typedef enum {

 SANE\_STATUS\_GOOD,

 SANE\_STATUS\_UNSUPPORTED,

 SANE\_STATUS\_CANCELLED,

 SANE\_STATUS\_DEVICE\_BUSY,

 SANE\_STATUS\_INVAL,

 SANE\_STATUS\_EOF,

 SANE\_STATUS\_JAMMED,

 SANE\_STATUS\_NO\_DOCS,

 SANE\_STATUS\_COVER\_OPEN,

 SANE\_STATUS\_IO\_ERROR,

 SANE\_STATUS\_NO\_MEM,

 SANE\_STATUS\_ACCESS\_DENIED

} SANE\_Status;

typedef enum {

 SANE\_TYPE\_BOOL,

 SANE\_TYPE\_INT,

 SANE\_TYPE\_FIXED,

 SANE\_TYPE\_STRING,

 SANE\_TYPE\_BUTTON,

 SANE\_TYPE\_GROUP

} SANE\_Value\_Type;

typedef enum {

 SANE\_UNIT\_NONE,

 SANE\_UNIT\_PIXEL,

 SANE\_UNIT\_BIT,

 SANE\_UNIT\_MM,

 SANE\_UNIT\_DPI,

 SANE\_UNIT\_PERCENT,

 SANE\_UNIT\_MICROSECOND

} SANE\_Unit;

typedef struct {

 SANE\_String\_Const name;

 SANE\_String\_Const vendor;

 SANE\_String\_Const model;

 SANE\_String\_Const type;

} SANE\_Device;

typedef enum {

 SANE\_CONSTRAINT\_NONE,

 SANE\_CONSTRAINT\_RANGE,

 SANE\_CONSTRAINT\_WORD\_LIST,

 SANE\_CONSTRAINT\_STRING\_LIST

} SANE\_Constraint\_Type;

typedef struct {

 SANE\_Word min;

 SANE\_Word max;

 SANE\_Word quant;

} SANE\_Range;

typedef struct {

 SANE\_String\_Const name;

 SANE\_String\_Const title;

 SANE\_String\_Const desc;

 SANE\_Value\_Type type;

 SANE\_Unit unit;

 SANE\_Int size;

 SANE\_Int cap;

 SANE\_Constraint\_Type constraint\_type;

 union {

 const SANE\_String\_Const \*string\_list;

 const SANE\_Word \*word\_list;

 const SANE\_Range \*range;

 } constraint;

} SANE\_Option\_Descriptor;

typedef enum {

 SANE\_ACTION\_GET\_VALUE,

 SANE\_ACTION\_SET\_VALUE,

 SANE\_ACTION\_SET\_AUTO

} SANE\_Action;

typedef enum {

 SANE\_FRAME\_GRAY,

 SANE\_FRAME\_RGB,

 SANE\_FRAME\_RED,

 SANE\_FRAME\_GREEN,

 SANE\_FRAME\_BLUE

} SANE\_Frame;

typedef struct {

 SANE\_Frame format;

 SANE\_Bool last\_frame;

 SANE\_Int bytes\_per\_line;

 SANE\_Int pixels\_per\_line;

 SANE\_Int lines;

 SANE\_Int depth;

} SANE\_Parameters;

typedef void (\*SANE\_Auth\_Callback) (SANE\_String\_Const, SANE\_Char \*,

 SANE\_Char \*);

extern void sane\_cancel(SANE\_Handle handle);

extern void sane\_close(SANE\_Handle handle);

extern SANE\_Status sane\_control\_option(SANE\_Handle handle, SANE\_Int option,

 SANE\_Action action, void \*value,

 SANE\_Int \* info);

extern void sane\_exit(void);

extern SANE\_Status sane\_get\_devices(const SANE\_Device \* \*\*device\_list,

 SANE\_Bool local\_only);

extern const SANE\_Option\_Descriptor \*sane\_get\_option\_descriptor(SANE\_Handle

 handle,

 SANE\_Int

 option);

extern SANE\_Status sane\_get\_parameters(SANE\_Handle handle,

 SANE\_Parameters \* params);

extern SANE\_Status sane\_get\_select\_fd(SANE\_Handle handle, SANE\_Int \* fd);

extern SANE\_Status sane\_init(SANE\_Int \* version\_code,

 SANE\_Auth\_Callback authorize);

extern SANE\_Status sane\_open(SANE\_String\_Const devicename,

 SANE\_Handle \* handle);

extern SANE\_Status sane\_read(SANE\_Handle handle, SANE\_Byte \* data,

 SANE\_Int max\_length, SANE\_Int \* length);

extern SANE\_Status sane\_set\_io\_mode(SANE\_Handle handle,

 SANE\_Bool non\_blocking);

extern SANE\_Status sane\_start(SANE\_Handle handle);

extern SANE\_String\_Const sane\_strstatus(SANE\_Status status);

### **10.2.2 sane/saneopts.h**

#define SANE\_NAME\_NUM\_OPTIONS ""

#define saneopts\_h

#define SANE\_NAME\_ADVANCED "advanced"

#define SANE\_NAME\_ANALOG\_GAMMA "analog-gamma"

#define SANE\_NAME\_ANALOG\_GAMMA\_B "analog-gamma-b"

#define SANE\_NAME\_ANALOG\_GAMMA\_BIND "analog-gamma-bind"

#define SANE\_NAME\_ANALOG\_GAMMA\_G "analog-gamma-g"

#define SANE\_NAME\_ANALOG\_GAMMA\_R "analog-gamma-r"

#define SANE\_NAME\_BACKTRACK "backtrack"

#define SANE\_NAME\_BLACK\_LEVEL "black-level"

#define SANE\_NAME\_GAMMA\_VECTOR\_B "blue-gamma-table"

#define SANE\_NAME\_SCAN\_BR\_X "br-x"

#define SANE\_NAME\_SCAN\_BR\_Y "br-y"

#define SANE\_NAME\_BRIGHTNESS "brightness"

#define SANE\_NAME\_CAL\_EXPOS\_TIME "cal-exposure-time"

#define SANE\_NAME\_CAL\_EXPOS\_TIME\_B "cal-exposure-time-b"

#define SANE\_NAME\_CAL\_EXPOS\_TIME\_G "cal-exposure-time-g"

#define SANE\_NAME\_CAL\_EXPOS\_TIME\_R "cal-exposure-time-r"

#define SANE\_NAME\_CAL\_LAMP\_DEN "cal-lamp-density"

#define SANE\_NAME\_CANCEL "cancel"

#define SANE\_TITLE\_CANCEL "Cancel button"

#define SANE\_NAME\_CONTRAST "contrast"

#define SANE\_NAME\_COPY "copy"

#define SANE\_TITLE\_COPY "Copy button"

#define SANE\_TITLE\_COVER\_OPEN "Cover open"

#define SANE\_NAME\_COVER\_OPEN "cover-open"

#define SANE\_NAME\_CUSTOM\_GAMMA "custom-gamma"

#define SANE\_NAME\_BIT\_DEPTH "depth"

#define SANE\_NAME\_DOR "double-res"

#define SANE\_NAME\_EMAIL "email"

#define SANE\_TITLE\_EMAIL "Email button"

#define SANE\_NAME\_ENHANCEMENT "enhancement"

#define SANE\_NAME\_FAX "fax"

#define SANE\_TITLE\_FAX "Fax button"

#define SANE\_NAME\_FILE "filename"

#define SANE\_NAME\_GAMMA\_VECTOR "gamma-table"

#define SANE\_NAME\_GEOMETRY "geometry"

#define SANE\_NAME\_GRAIN\_SIZE "grain"

#define SANE\_NAME\_GAMMA\_VECTOR\_G "green-gamma-table"

#define SANE\_NAME\_HALFTONE\_PATTERN "halftone-pattern"

#define SANE\_NAME\_HALFTONE\_DIMENSION "halftone-size"

#define SANE\_NAME\_HALFTONE "halftoning"

#define SANE\_NAME\_HIGHLIGHT "highlight"

#define SANE\_NAME\_HIGHLIGHT\_B "highlight-b"

#define SANE\_NAME\_HIGHLIGHT\_G "highlight-g"

#define SANE\_NAME\_HIGHLIGHT\_R "highlight-r"

#define SANE\_NAME\_HUE "hue"

#define SANE\_NAME\_LAMP\_OFF\_AT\_EXIT "lamp-off-at-exit"

#define SANE\_NAME\_SCAN\_MODE "mode"

#define SANE\_NAME\_NEGATIVE "negative"

#define SANE\_TITLE\_PAGE\_LOADED "Page loaded"

#define SANE\_NAME\_PAGE\_HEIGHT "page-height"

#define SANE\_NAME\_PAGE\_LOADED "page-loaded"

#define SANE\_NAME\_PAGE\_WIDTH "page-width"

#define SANE\_NAME\_PDF "pdf"

#define SANE\_TITLE\_PDF "PDF button"

#define SANE\_NAME\_PREVIEW "preview"

#define SANE\_NAME\_GRAY\_PREVIEW "preview-in-gray"

#define SANE\_NAME\_QUALITY\_CAL "quality-cal"

#define SANE\_NAME\_GAMMA\_VECTOR\_R "red-gamma-table"

#define SANE\_NAME\_SCAN\_RESOLUTION "resolution"

#define SANE\_NAME\_RESOLUTION\_BIND "resolution-bind"

#define SANE\_NAME\_RGB\_BIND "rgb-bind"

#define SANE\_TITLE\_ADVANCED SANE\_I18N("Advanced")

#define SANE\_TITLE\_ANALOG\_GAMMA\_B SANE\_I18N("Analog gamma blue")

#define SANE\_TITLE\_ANALOG\_GAMMA SANE\_I18N("Analog gamma correction")

#define SANE\_TITLE\_ANALOG\_GAMMA\_G SANE\_I18N("Analog gamma green")

#define SANE\_TITLE\_ANALOG\_GAMMA\_R SANE\_I18N("Analog gamma red")

#define SANE\_DESC\_ANALOG\_GAMMA\_B SANE\_I18N("Analog gamma-correction for blue")

#define SANE\_DESC\_ANALOG\_GAMMA\_G SANE\_I18N("Analog gamma-correction for green")

#define SANE\_DESC\_ANALOG\_GAMMA\_R SANE\_I18N("Analog gamma-correction for red")

#define SANE\_DESC\_ANALOG\_GAMMA SANE\_I18N("Analog gamma-correction")

#define SANE\_TITLE\_ANALOG\_GAMMA\_BIND SANE\_I18N("Bind analog gamma")

#define SANE\_TITLE\_RGB\_BIND SANE\_I18N("Bind RGB")

#define SANE\_TITLE\_RESOLUTION\_BIND SANE\_I18N("Bind X and Y resolution")

#define SANE\_TITLE\_BIT\_DEPTH SANE\_I18N("Bit depth")

#define SANE\_TITLE\_BLACK\_LEVEL SANE\_I18N("Black level")

#define SANE\_TITLE\_GAMMA\_VECTOR\_B SANE\_I18N("Blue intensity")

#define SANE\_DESC\_SCAN\_BR\_X SANE\_I18N("Bottom-right x position of scan area.")

#define SANE\_TITLE\_SCAN\_BR\_X SANE\_I18N("Bottom-right x")

#define SANE\_DESC\_SCAN\_BR\_Y SANE\_I18N("Bottom-right y position of scan area.")

#define SANE\_TITLE\_SCAN\_BR\_Y SANE\_I18N("Bottom-right y")

#define SANE\_TITLE\_BRIGHTNESS SANE\_I18N("Brightness")

#define SANE\_TITLE\_CAL\_EXPOS\_TIME\_G SANE\_I18N("Cal. exposure-time for " "green")

#define SANE\_TITLE\_CAL\_EXPOS\_TIME\_B SANE\_I18N("Cal. exposure-time for blue")

#define SANE\_TITLE\_CAL\_EXPOS\_TIME\_R SANE\_I18N("Cal. exposure-time for red")

#define SANE\_TITLE\_CAL\_EXPOS\_TIME SANE\_I18N("Cal. exposure-time")

#define SANE\_TITLE\_CAL\_LAMP\_DEN SANE\_I18N("Cal. lamp density")

#define SANE\_DESC\_CANCEL SANE\_I18N("Cancel button")

#define SANE\_VALUE\_SCAN\_MODE\_COLOR SANE\_I18N("Color")

#define SANE\_TITLE\_CONTRAST SANE\_I18N("Contrast")

#define SANE\_DESC\_BRIGHTNESS SANE\_I18N("Controls the brightness of the acquired image.")

#define SANE\_DESC\_CONTRAST SANE\_I18N("Controls the contrast of the acquired image.")

#define SANE\_DESC\_HUE SANE\_I18N("Controls the \"hue\" (blue-level) of the acquired image.")

#define SANE\_DESC\_BACKTRACK SANE\_I18N("Controls whether backtracking is forced.")

#define SANE\_DESC\_COPY SANE\_I18N("Copy button")

#define SANE\_DESC\_COVER\_OPEN SANE\_I18N("Cover open")

#define SANE\_DESC\_CAL\_EXPOS\_TIME\_B SANE\_I18N("Define exposure-time for blue calibration")

#define SANE\_DESC\_SCAN\_EXPOS\_TIME\_B SANE\_I18N("Define exposure-time for blue scan")

#define SANE\_DESC\_CAL\_EXPOS\_TIME SANE\_I18N("Define exposure-time for calibration")

#define SANE\_DESC\_CAL\_EXPOS\_TIME\_G SANE\_I18N("Define exposure-time for green calibration")

#define SANE\_DESC\_SCAN\_EXPOS\_TIME\_G SANE\_I18N("Define exposure-time for green scan")

#define SANE\_DESC\_CAL\_EXPOS\_TIME\_R SANE\_I18N("Define exposure-time for red calibration")

#define SANE\_DESC\_SCAN\_EXPOS\_TIME\_R SANE\_I18N("Define exposure-time for red scan")

#define SANE\_DESC\_SCAN\_EXPOS\_TIME SANE\_I18N("Define exposure-time for scan")

#define SANE\_DESC\_CAL\_LAMP\_DEN SANE\_I18N("Define lamp density for calibration")

#define SANE\_DESC\_SCAN\_LAMP\_DEN SANE\_I18N("Define lamp density for scan")

#define SANE\_DESC\_HALFTONE\_PATTERN SANE\_I18N("Defines the halftoning (dithering) pattern for scanning " "halftoned images.")

#define SANE\_DESC\_SCAN\_SPEED SANE\_I18N("Determines the speed at which the scan proceeds.")

#define SANE\_DESC\_CUSTOM\_GAMMA SANE\_I18N("Determines whether a builtin or a custom gamma-table should be " "used.")

#define SANE\_DESC\_QUALITY\_CAL SANE\_I18N("Do a quality white-calibration")

#define SANE\_TITLE\_DOR SANE\_I18N("Double Optical Resolution")

#define SANE\_DESC\_EMAIL SANE\_I18N("Email button")

#define SANE\_DESC\_SELECT\_EXPOSURE\_TIME SANE\_I18N("Enable selection of exposure-time")

#define SANE\_DESC\_SELECT\_LAMP\_DENSITY SANE\_I18N("Enable selection of lamp density")

#define SANE\_TITLE\_ENHANCEMENT SANE\_I18N("Enhancement")

#define SANE\_DESC\_FAX SANE\_I18N("Fax button")

#define SANE\_TITLE\_FILE SANE\_I18N("Filename")

#define SANE\_TITLE\_BACKTRACK SANE\_I18N("Force backtracking")

#define SANE\_TITLE\_GRAY\_PREVIEW SANE\_I18N("Force monochrome preview")

#define SANE\_DESC\_GAMMA\_VECTOR\_B SANE\_I18N("Gamma-correction table for the blue band.")

#define SANE\_DESC\_GAMMA\_VECTOR\_G SANE\_I18N("Gamma-correction table for the green band.")

#define SANE\_DESC\_GAMMA\_VECTOR\_R SANE\_I18N("Gamma-correction table for the red band.")

#define SANE\_DESC\_GAMMA\_VECTOR SANE\_I18N("Gamma-correction table. In color mode this option equally " "affects the red, green, and blue channels simultaneously (i.e., it is an " "intensity gamma table).")

#define SANE\_TITLE\_GEOMETRY SANE\_I18N("Geometry")

#define SANE\_TITLE\_GRAIN\_SIZE SANE\_I18N("Grain size")

#define SANE\_VALUE\_SCAN\_MODE\_GRAY SANE\_I18N("Gray")

#define SANE\_TITLE\_GAMMA\_VECTOR\_G SANE\_I18N("Green intensity")

#define SANE\_TITLE\_HALFTONE\_DIMENSION SANE\_I18N("Halftone pattern size")

#define SANE\_TITLE\_HALFTONE\_PATTERN SANE\_I18N("Halftone pattern")

#define SANE\_TITLE\_HALFTONE SANE\_I18N("Halftoning")

#define SANE\_DESC\_ADVANCED SANE\_I18N("Hardware specific options")

#define SANE\_TITLE\_HIGHLIGHT\_B SANE\_I18N("Highlight for blue")

#define SANE\_TITLE\_HIGHLIGHT\_G SANE\_I18N("Highlight for green")

#define SANE\_TITLE\_HIGHLIGHT\_R SANE\_I18N("Highlight for red")

#define SANE\_TITLE\_HIGHLIGHT SANE\_I18N("Highlight")

#define SANE\_TITLE\_HUE SANE\_I18N("Hue")

#define SANE\_TITLE\_GAMMA\_VECTOR SANE\_I18N("Image intensity")

#define SANE\_DESC\_ENHANCEMENT SANE\_I18N("Image modification options")

#define SANE\_DESC\_ANALOG\_GAMMA\_BIND SANE\_I18N("In RGB-mode use same values for each color")

#define SANE\_DESC\_RGB\_BIND SANE\_I18N("In RGB-mode use same values for each color")

#define SANE\_TITLE\_LAMP\_OFF\_AT\_EXIT SANE\_I18N("Lamp off at exit")

#define SANE\_VALUE\_SCAN\_MODE\_LINEART SANE\_I18N("Lineart")

#define SANE\_TITLE\_NEGATIVE SANE\_I18N("Negative")

#define SANE\_DESC\_BIT\_DEPTH SANE\_I18N("Number of bits per sample, typical values are 1 for \"line-art\" " "and 8 for multibit scans.")

#define SANE\_TITLE\_NUM\_OPTIONS SANE\_I18N("Number of options")

#define SANE\_TITLE\_PAGE\_HEIGHT SANE\_I18N("Page height")

#define SANE\_DESC\_PAGE\_LOADED SANE\_I18N("Page loaded")

#define SANE\_TITLE\_PAGE\_WIDTH SANE\_I18N("Page width")

#define SANE\_DESC\_PDF SANE\_I18N("PDF button")

#define SANE\_TITLE\_PREVIEW SANE\_I18N("Preview")

#define SANE\_TITLE\_QUALITY\_CAL SANE\_I18N("Quality calibration")

#define SANE\_DESC\_NUM\_OPTIONS SANE\_I18N("Read-only option that specifies how many options a specific " "devices supports.")

#define SANE\_TITLE\_GAMMA\_VECTOR\_R SANE\_I18N("Red intensity")

#define SANE\_DESC\_PREVIEW SANE\_I18N("Request a preview-quality scan.")

#define SANE\_DESC\_GRAY\_PREVIEW SANE\_I18N("Request that all previews are done in monochrome mode. On a " "three-pass scanner this cuts down the number of passes to one and on a " "one-pass scanner, it reduces the memory requirements and scan-time of the " "preview.")

#define SANE\_TITLE\_SATURATION SANE\_I18N("Saturation")

#define SANE\_DESC\_GEOMETRY SANE\_I18N("Scan area and media size options")

#define SANE\_DESC\_SCAN SANE\_I18N("Scan button")

#define SANE\_TITLE\_SCAN\_EXPOS\_TIME\_G SANE\_I18N("Scan exposure-time for " "green")

#define SANE\_TITLE\_SCAN\_EXPOS\_TIME\_B SANE\_I18N("Scan exposure-time for blue")

#define SANE\_TITLE\_SCAN\_EXPOS\_TIME\_R SANE\_I18N("Scan exposure-time for red")

#define SANE\_TITLE\_SCAN\_EXPOS\_TIME SANE\_I18N("Scan exposure-time")

#define SANE\_TITLE\_SCAN\_LAMP\_DEN SANE\_I18N("Scan lamp density")

#define SANE\_TITLE\_SCAN\_MODE SANE\_I18N("Scan mode")

#define SANE\_TITLE\_SCAN\_RESOLUTION SANE\_I18N("Scan resolution")

#define SANE\_TITLE\_SCAN\_SOURCE SANE\_I18N("Scan source")

#define SANE\_TITLE\_SCAN\_SPEED SANE\_I18N("Scan speed")

#define SANE\_DESC\_SENSORS SANE\_I18N("Scanner sensors and buttons")

#define SANE\_DESC\_THRESHOLD SANE\_I18N("Select minimum-brightness to get a white point")

#define SANE\_DESC\_SCAN\_MODE SANE\_I18N("Selects the scan mode (e.g., lineart, monochrome, or color).")

#define SANE\_DESC\_SCAN\_SOURCE SANE\_I18N("Selects the scan source (such as a document-feeder).")

#define SANE\_DESC\_GRAIN\_SIZE SANE\_I18N("Selects the \"graininess\" of the acquired image. Smaller values " "result in sharper images.")

#define SANE\_DESC\_SHADOW\_B SANE\_I18N("Selects what blue radiance level should be considered \"black\".")

#define SANE\_DESC\_HIGHLIGHT\_B SANE\_I18N("Selects what blue radiance level should be considered \"full " "blue\".")

#define SANE\_DESC\_WHITE\_LEVEL\_B SANE\_I18N("Selects what blue radiance level should be considered \"white\".")

#define SANE\_DESC\_SHADOW\_G SANE\_I18N("Selects what green radiance level should be considered \"black\".")

#define SANE\_DESC\_HIGHLIGHT\_G SANE\_I18N("Selects what green radiance level should be considered \"full " "green\".")

#define SANE\_DESC\_WHITE\_LEVEL\_G SANE\_I18N("Selects what green radiance level should be considered \"white\".")

#define SANE\_DESC\_BLACK\_LEVEL SANE\_I18N("Selects what radiance level should be considered \"black\".")

#define SANE\_DESC\_SHADOW SANE\_I18N("Selects what radiance level should be considered \"black\".")

#define SANE\_DESC\_HIGHLIGHT SANE\_I18N("Selects what radiance level should be considered \"white\".")

#define SANE\_DESC\_WHITE\_LEVEL SANE\_I18N("Selects what radiance level should be considered \"white\".")

#define SANE\_DESC\_SHADOW\_R SANE\_I18N("Selects what red radiance level should be considered \"black\".")

#define SANE\_DESC\_HIGHLIGHT\_R SANE\_I18N("Selects what red radiance level should be considered \"full red\".")

#define SANE\_DESC\_WHITE\_LEVEL\_R SANE\_I18N("Selects what red radiance level should be considered \"white\".")

#define SANE\_DESC\_HALFTONE SANE\_I18N("Selects whether the acquired image should be halftoned (dithered).")

#define SANE\_TITLE\_SENSORS SANE\_I18N("Sensors")

#define SANE\_TITLE\_SELECT\_EXPOSURE\_TIME SANE\_I18N("Set exposure-time")

#define SANE\_TITLE\_SELECT\_LAMP\_DENSITY SANE\_I18N("Set lamp density")

#define SANE\_DESC\_SCAN\_X\_RESOLUTION SANE\_I18N("Sets the horizontal resolution of the scanned image.")

#define SANE\_DESC\_SCAN\_RESOLUTION SANE\_I18N("Sets the resolution of the scanned image.")

#define SANE\_DESC\_HALFTONE\_DIMENSION SANE\_I18N("Sets the size of the halftoning (dithering) pattern used when " "scanning halftoned images.")

#define SANE\_DESC\_SCAN\_Y\_RESOLUTION SANE\_I18N("Sets the vertical resolution of the scanned image.")

#define SANE\_TITLE\_SHADOW\_B SANE\_I18N("Shadow for blue")

#define SANE\_TITLE\_SHADOW\_G SANE\_I18N("Shadow for green")

#define SANE\_TITLE\_SHADOW\_R SANE\_I18N("Shadow for red")

#define SANE\_TITLE\_SHADOW SANE\_I18N("Shadow")

#define SANE\_DESC\_STANDARD SANE\_I18N("Source, mode and resolution options")

#define SANE\_DESC\_PAGE\_HEIGHT SANE\_I18N("Specifies the height of the media.")

#define SANE\_DESC\_PAGE\_WIDTH SANE\_I18N("Specifies the width of the media. Required for automatic " "centering of sheet-fed scans.")

#define SANE\_TITLE\_STANDARD SANE\_I18N("Standard")

#define SANE\_DESC\_NEGATIVE SANE\_I18N("Swap black and white")

#define SANE\_DESC\_FILE SANE\_I18N("The filename of the image to be loaded.")

#define SANE\_DESC\_SATURATION SANE\_I18N("The saturation level controls the amount of \"blooming\" that " "occurs when acquiring an image with a camera. Larger values cause more " "blooming.")

#define SANE\_TITLE\_THRESHOLD SANE\_I18N("Threshold")

#define SANE\_DESC\_SCAN\_TL\_X SANE\_I18N("Top-left x position of scan area.")

#define SANE\_TITLE\_SCAN\_TL\_X SANE\_I18N("Top-left x")

#define SANE\_DESC\_SCAN\_TL\_Y SANE\_I18N("Top-left y position of scan area.")

#define SANE\_TITLE\_SCAN\_TL\_Y SANE\_I18N("Top-left y")

#define SANE\_DESC\_LAMP\_OFF\_AT\_EXIT SANE\_I18N("Turn off lamp when program exits")

#define SANE\_TITLE\_CUSTOM\_GAMMA SANE\_I18N("Use custom gamma table")

#define SANE\_DESC\_DOR SANE\_I18N("Use lens that doubles optical resolution")

#define SANE\_DESC\_RESOLUTION\_BIND SANE\_I18N("Use same values for X and Y resolution")

#define SANE\_DESC\_WARMUP SANE\_I18N("Warmup lamp before scanning")

#define SANE\_TITLE\_WARMUP SANE\_I18N("Warmup lamp")

#define SANE\_TITLE\_WHITE\_LEVEL\_B SANE\_I18N("White level for blue")

#define SANE\_TITLE\_WHITE\_LEVEL\_G SANE\_I18N("White level for green")

#define SANE\_TITLE\_WHITE\_LEVEL\_R SANE\_I18N("White level for red")

#define SANE\_TITLE\_WHITE\_LEVEL SANE\_I18N("White level")

#define SANE\_TITLE\_SCAN\_X\_RESOLUTION SANE\_I18N("X-resolution")

#define SANE\_TITLE\_SCAN\_Y\_RESOLUTION SANE\_I18N("Y-resolution")

#define SANE\_NAME\_SATURATION "saturation"

#define SANE\_NAME\_SCAN "scan"

#define SANE\_TITLE\_SCAN "Scan button"

#define SANE\_NAME\_SCAN\_EXPOS\_TIME "scan-exposure-time"

#define SANE\_NAME\_SCAN\_EXPOS\_TIME\_B "scan-exposure-time-b"

#define SANE\_NAME\_SCAN\_EXPOS\_TIME\_G "scan-exposure-time-g"

#define SANE\_NAME\_SCAN\_EXPOS\_TIME\_R "scan-exposure-time-r"

#define SANE\_NAME\_SCAN\_LAMP\_DEN "scan-lamp-density"

#define SANE\_NAME\_SELECT\_EXPOSURE\_TIME "select-exposure-time"

#define SANE\_NAME\_SELECT\_LAMP\_DENSITY "select-lamp-density"

#define SANE\_NAME\_SENSORS "sensors"

#define SANE\_NAME\_SHADOW "shadow"

#define SANE\_NAME\_SHADOW\_B "shadow-b"

#define SANE\_NAME\_SHADOW\_G "shadow-g"

#define SANE\_NAME\_SHADOW\_R "shadow-r"

#define SANE\_NAME\_SCAN\_SOURCE "source"

#define SANE\_NAME\_SCAN\_SPEED "speed"

#define SANE\_NAME\_STANDARD "standard"

#define SANE\_I18N(text) text

#define SANE\_NAME\_THRESHOLD "threshold"

#define SANE\_NAME\_SCAN\_TL\_X "tl-x"

#define SANE\_NAME\_SCAN\_TL\_Y "tl-y"

#define SANE\_NAME\_WARMUP "warmup"

#define SANE\_NAME\_WHITE\_LEVEL "white-level"

#define SANE\_NAME\_WHITE\_LEVEL\_B "white-level-b"

#define SANE\_NAME\_WHITE\_LEVEL\_G "white-level-g"

#define SANE\_NAME\_WHITE\_LEVEL\_R "white-level-r"

#define SANE\_NAME\_SCAN\_X\_RESOLUTION "x-resolution"

#define SANE\_NAME\_SCAN\_Y\_RESOLUTION "y-resolution"

# **VI Package Format and Installation**

# **11 Software Installation**

## **11.1 Package Dependencies**

The LSB runtime environment shall provide the following dependencies.

lsb-imaging

  This dependency is used to indicate that the application is dependent on features contained in the LSB Imaging module specification.

These dependencies shall have a version of 5.0.

# **Annex A Alphabetical Listing of Interfaces by Library**

## **A.1 libsane**

The behavior of the interfaces in this library is specified by the following Standards.

|  |
| --- |
| [SANE Standard Version 1.04](#ID_STD_46_SANE) [SANE] |

**Table A-1 libsane Function Interfaces**

|  |  |  |
| --- | --- | --- |
| sane\_cancel[[SANE]](#ID_STD_46_SANE) | sane\_get\_option\_descriptor[[SANE]](#ID_STD_46_SANE) | sane\_read[[SANE]](#ID_STD_46_SANE) |
| sane\_close[[SANE]](#ID_STD_46_SANE) | sane\_get\_parameters[[SANE]](#ID_STD_46_SANE) | sane\_set\_io\_mode[[SANE]](#ID_STD_46_SANE) |
| sane\_control\_option[[SANE]](#ID_STD_46_SANE) | sane\_get\_select\_fd[[SANE]](#ID_STD_46_SANE) | sane\_start[[SANE]](#ID_STD_46_SANE) |
| sane\_exit[[SANE]](#ID_STD_46_SANE) | sane\_init[[SANE]](#ID_STD_46_SANE) | sane\_strstatus[[SANE]](#ID_STD_46_SANE) |
| sane\_get\_devices[[SANE]](#ID_STD_46_SANE) | sane\_open[[SANE]](#ID_STD_46_SANE) |  |

## **A.2 libcups**

The behavior of the interfaces in this library is specified by the following Standards.

|  |
| --- |
| [CUPS API Reference](#ID_STD_46_CUPS) [CUPS 1.2] |
| [This Specification](#ID_STD_46_LSB) [LSB] |

**Table A-2 libcups Function Interfaces**

|  |  |  |
| --- | --- | --- |
| cupsAddDest[[LSB]](#ID_STD_46_LSB) | httpBlocking[[CUPS 1.2]](#ID_STD_46_CUPS) | ippAddResolution[[CUPS 1.2]](#ID_STD_46_CUPS) |
| cupsAddOption[[LSB]](#ID_STD_46_LSB) | httpCheck[[CUPS 1.2]](#ID_STD_46_CUPS) | ippAddResolutions[[CUPS 1.2]](#ID_STD_46_CUPS) |
| cupsCancelJob[[LSB]](#ID_STD_46_LSB) | httpClearCookie[[CUPS 1.2]](#ID_STD_46_CUPS) | ippAddSeparator[[CUPS 1.2]](#ID_STD_46_CUPS) |
| cupsDoAuthentication[[CUPS 1.2]](#ID_STD_46_CUPS) | httpClearFields[[CUPS 1.2]](#ID_STD_46_CUPS) | ippAddString[[CUPS 1.2]](#ID_STD_46_CUPS) |
| cupsDoFileRequest[[CUPS 1.2]](#ID_STD_46_CUPS) | httpClose[[CUPS 1.2]](#ID_STD_46_CUPS) | ippAddStrings[[CUPS 1.2]](#ID_STD_46_CUPS) |
| cupsEncodeOptions[[CUPS 1.2]](#ID_STD_46_CUPS) | httpConnect[[CUPS 1.2]](#ID_STD_46_CUPS) | ippDateToTime[[CUPS 1.2]](#ID_STD_46_CUPS) |
| cupsEncryption[[LSB]](#ID_STD_46_LSB) | httpConnectEncrypt[[CUPS 1.2]](#ID_STD_46_CUPS) | ippDelete[[CUPS 1.2]](#ID_STD_46_CUPS) |
| cupsFreeDests[[LSB]](#ID_STD_46_LSB) | httpDecode64\_2[[CUPS 1.2]](#ID_STD_46_CUPS) | ippDeleteAttribute[[CUPS 1.2]](#ID_STD_46_CUPS) |
| cupsFreeJobs[[LSB]](#ID_STD_46_LSB) | httpDelete[[CUPS 1.2]](#ID_STD_46_CUPS) | ippErrorString[[CUPS 1.2]](#ID_STD_46_CUPS) |
| cupsFreeOptions[[LSB]](#ID_STD_46_LSB) | httpEncode64\_2[[CUPS 1.2]](#ID_STD_46_CUPS) | ippFindAttribute[[CUPS 1.2]](#ID_STD_46_CUPS) |
| cupsGetDefault[[LSB]](#ID_STD_46_LSB) | httpEncryption[[CUPS 1.2]](#ID_STD_46_CUPS) | ippFindNextAttribute[[CUPS 1.2]](#ID_STD_46_CUPS) |
| cupsGetDefault2[[CUPS 1.2]](#ID_STD_46_CUPS) | httpError[[CUPS 1.2]](#ID_STD_46_CUPS) | ippLength[[CUPS 1.2]](#ID_STD_46_CUPS) |
| cupsGetDest[[LSB]](#ID_STD_46_LSB) | httpFlush[[CUPS 1.2]](#ID_STD_46_CUPS) | ippNew[[CUPS 1.2]](#ID_STD_46_CUPS) |
| cupsGetDests[[LSB]](#ID_STD_46_LSB) | httpGet[[CUPS 1.2]](#ID_STD_46_CUPS) | ippPort[[CUPS 1.2]](#ID_STD_46_CUPS) |
| cupsGetDests2[[CUPS 1.2]](#ID_STD_46_CUPS) | httpGetCookie[[CUPS 1.2]](#ID_STD_46_CUPS) | ippRead[[CUPS 1.2]](#ID_STD_46_CUPS) |
| cupsGetFd[[CUPS 1.2]](#ID_STD_46_CUPS) | httpGetDateString[[CUPS 1.2]](#ID_STD_46_CUPS) | ippReadFile[[CUPS 1.2]](#ID_STD_46_CUPS) |
| cupsGetFile[[CUPS 1.2]](#ID_STD_46_CUPS) | httpGetDateTime[[CUPS 1.2]](#ID_STD_46_CUPS) | ippReadIO[[CUPS 1.2]](#ID_STD_46_CUPS) |
| cupsGetJobs[[LSB]](#ID_STD_46_LSB) | httpGetField[[CUPS 1.2]](#ID_STD_46_CUPS) | ippSetPort[[CUPS 1.2]](#ID_STD_46_CUPS) |
| cupsGetJobs2[[CUPS 1.2]](#ID_STD_46_CUPS) | httpGetHostByName[[CUPS 1.2]](#ID_STD_46_CUPS) | ippTimeToDate[[CUPS 1.2]](#ID_STD_46_CUPS) |
| cupsGetOption[[LSB]](#ID_STD_46_LSB) | httpGetSubField[[CUPS 1.2]](#ID_STD_46_CUPS) | ippWrite[[CUPS 1.2]](#ID_STD_46_CUPS) |
| cupsGetPPD[[LSB]](#ID_STD_46_LSB) | httpGets[[CUPS 1.2]](#ID_STD_46_CUPS) | ippWriteFile[[CUPS 1.2]](#ID_STD_46_CUPS) |
| cupsGetPPD2[[CUPS 1.2]](#ID_STD_46_CUPS) | httpHead[[CUPS 1.2]](#ID_STD_46_CUPS) | ippWriteIO[[CUPS 1.2]](#ID_STD_46_CUPS) |
| cupsGetPassword[[LSB]](#ID_STD_46_LSB) | httpInitialize[[CUPS 1.2]](#ID_STD_46_CUPS) | ppdClose[[LSB]](#ID_STD_46_LSB) |
| cupsLangEncoding[[LSB]](#ID_STD_46_LSB) | httpMD5[[CUPS 1.2]](#ID_STD_46_CUPS) | ppdCollect[[LSB]](#ID_STD_46_LSB) |
| cupsLangFlush[[LSB]](#ID_STD_46_LSB) | httpMD5Final[[CUPS 1.2]](#ID_STD_46_CUPS) | ppdConflicts[[LSB]](#ID_STD_46_LSB) |
| cupsLangFree[[LSB]](#ID_STD_46_LSB) | httpMD5String[[CUPS 1.2]](#ID_STD_46_CUPS) | ppdEmit[[LSB]](#ID_STD_46_LSB) |
| cupsLangGet[[LSB]](#ID_STD_46_LSB) | httpOptions[[CUPS 1.2]](#ID_STD_46_CUPS) | ppdEmitFd[[LSB]](#ID_STD_46_LSB) |
| cupsLastError[[LSB]](#ID_STD_46_LSB) | httpPost[[CUPS 1.2]](#ID_STD_46_CUPS) | ppdEmitJCL[[LSB]](#ID_STD_46_LSB) |
| cupsMarkOptions[[LSB]](#ID_STD_46_LSB) | httpPut[[CUPS 1.2]](#ID_STD_46_CUPS) | ppdErrorString[[LSB]](#ID_STD_46_LSB) |
| cupsParseOptions[[LSB]](#ID_STD_46_LSB) | httpReconnect[[CUPS 1.2]](#ID_STD_46_CUPS) | ppdFindAttr[[LSB]](#ID_STD_46_LSB) |
| cupsPrintFile[[LSB]](#ID_STD_46_LSB) | httpSetCookie[[CUPS 1.2]](#ID_STD_46_CUPS) | ppdFindChoice[[LSB]](#ID_STD_46_LSB) |
| cupsPrintFile2[[CUPS 1.2]](#ID_STD_46_CUPS) | httpSetField[[CUPS 1.2]](#ID_STD_46_CUPS) | ppdFindMarkedChoice[[LSB]](#ID_STD_46_LSB) |
| cupsPrintFiles[[LSB]](#ID_STD_46_LSB) | httpStatus[[CUPS 1.2]](#ID_STD_46_CUPS) | ppdFindNextAttr[[LSB]](#ID_STD_46_LSB) |
| cupsPrintFiles2[[CUPS 1.2]](#ID_STD_46_CUPS) | httpTrace[[CUPS 1.2]](#ID_STD_46_CUPS) | ppdFindOption[[LSB]](#ID_STD_46_LSB) |
| cupsPutFd[[CUPS 1.2]](#ID_STD_46_CUPS) | httpUpdate[[CUPS 1.2]](#ID_STD_46_CUPS) | ppdIsMarked[[LSB]](#ID_STD_46_LSB) |
| cupsPutFile[[CUPS 1.2]](#ID_STD_46_CUPS) | httpWait[[CUPS 1.2]](#ID_STD_46_CUPS) | ppdLastError[[LSB]](#ID_STD_46_LSB) |
| cupsServer[[LSB]](#ID_STD_46_LSB) | ippAddBoolean[[CUPS 1.2]](#ID_STD_46_CUPS) | ppdMarkDefaults[[LSB]](#ID_STD_46_LSB) |
| cupsSetDests[[LSB]](#ID_STD_46_LSB) | ippAddBooleans[[CUPS 1.2]](#ID_STD_46_CUPS) | ppdMarkOption[[LSB]](#ID_STD_46_LSB) |
| cupsSetDests2[[CUPS 1.2]](#ID_STD_46_CUPS) | ippAddCollection[[CUPS 1.2]](#ID_STD_46_CUPS) | ppdOpen[[LSB]](#ID_STD_46_LSB) |
| cupsSetEncryption[[LSB]](#ID_STD_46_LSB) | ippAddCollections[[CUPS 1.2]](#ID_STD_46_CUPS) | ppdOpenFd[[LSB]](#ID_STD_46_LSB) |
| cupsSetPasswordCB[[LSB]](#ID_STD_46_LSB) | ippAddDate[[CUPS 1.2]](#ID_STD_46_CUPS) | ppdOpenFile[[LSB]](#ID_STD_46_LSB) |
| cupsSetServer[[LSB]](#ID_STD_46_LSB) | ippAddInteger[[CUPS 1.2]](#ID_STD_46_CUPS) | ppdPageLength[[LSB]](#ID_STD_46_LSB) |
| cupsSetUser[[LSB]](#ID_STD_46_LSB) | ippAddIntegers[[CUPS 1.2]](#ID_STD_46_CUPS) | ppdPageSize[[LSB]](#ID_STD_46_LSB) |
| cupsTempFd[[LSB]](#ID_STD_46_LSB) | ippAddRange[[CUPS 1.2]](#ID_STD_46_CUPS) | ppdPageWidth[[LSB]](#ID_STD_46_LSB) |
| cupsUser[[LSB]](#ID_STD_46_LSB) | ippAddRanges[[CUPS 1.2]](#ID_STD_46_CUPS) | ppdSetConformance[[LSB]](#ID_STD_46_LSB) |

## **A.3 libcupsimage**

The behavior of the interfaces in this library is specified by the following Standards.

|  |
| --- |
| [This Specification](#ID_STD_46_LSB) [LSB] |

**Table A-3 libcupsimage Function Interfaces**

|  |  |  |
| --- | --- | --- |
| cupsRasterClose[[LSB]](#ID_STD_46_LSB) | cupsRasterReadHeader[[LSB]](#ID_STD_46_LSB) | cupsRasterWriteHeader[[LSB]](#ID_STD_46_LSB) |
| cupsRasterOpen[[LSB]](#ID_STD_46_LSB) | cupsRasterReadPixels[[LSB]](#ID_STD_46_LSB) | cupsRasterWritePixels[[LSB]](#ID_STD_46_LSB) |

# **Annex B GNU Free Documentation License (Informative)**

This specification is published under the terms of the GNU Free Documentation License, Version 1.1, March 2000

Copyright (C) 2000 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

## **B.1 PREAMBLE**

The purpose of this License is to make a manual, textbook, or other written document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

## **B.2 APPLICABILITY AND DEFINITIONS**

This License applies to any manual or other work that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you".

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (For example, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, whose contents can be viewed and edited directly and straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup has been designed to thwart or discourage subsequent modification by readers is not Transparent. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML designed for human modification. Opaque formats include PostScript, PDF, proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

## **B.3 VERBATIM COPYING**

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

## **B.4 COPYING IN QUANTITY**

If you publish printed copies of the Document numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a publicly-accessible computer-network location containing a complete Transparent copy of the Document, free of added material, which the general network-using public has access to download anonymously at no charge using public-standard network protocols. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

## **B.5 MODIFICATIONS**

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

 A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

 B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has less than five).

 C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

 D. Preserve all the copyright notices of the Document.

 E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

 F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

 G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

 H. Include an unaltered copy of this License.

 I. Preserve the section entitled "History", and its title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

 J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

 K. In any section entitled "Acknowledgements" or "Dedications", preserve the section's title, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

 L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

 M. Delete any section entitled "Endorsements". Such a section may not be included in the Modified Version.

 N. Do not retitle any existing section as "Endorsements" or to conflict in title with any Invariant Section.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

## **B.6 COMBINING DOCUMENTS**

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections entitled "History" in the various original documents, forming one section entitled "History"; likewise combine any sections entitled "Acknowledgements", and any sections entitled "Dedications". You must delete all sections entitled "Endorsements."

## **B.7 COLLECTIONS OF DOCUMENTS**

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

## **B.8 AGGREGATION WITH INDEPENDENT WORKS**

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, does not as a whole count as a Modified Version of the Document, provided no compilation copyright is claimed for the compilation. Such a compilation is called an "aggregate", and this License does not apply to the other self-contained works thus compiled with the Document, on account of their being thus compiled, if they are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one quarter of the entire aggregate, the Document's Cover Texts may be placed on covers that surround only the Document within the aggregate. Otherwise they must appear on covers around the whole aggregate.

## **B.9 TRANSLATION**

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License provided that you also include the original English version of this License. In case of a disagreement between the translation and the original English version of this License, the original English version will prevail.

## **B.10 TERMINATION**

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

## **B.11 FUTURE REVISIONS OF THIS LICENSE**

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

## **B.12 How to use this License for your documents**

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (c) YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have no Invariant Sections, write "with no Invariant Sections" instead of saying which ones are invariant. If you have no Front-Cover Texts, write "no Front-Cover Texts" instead of "Front-Cover Texts being LIST"; likewise for Back-Cover Texts.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.