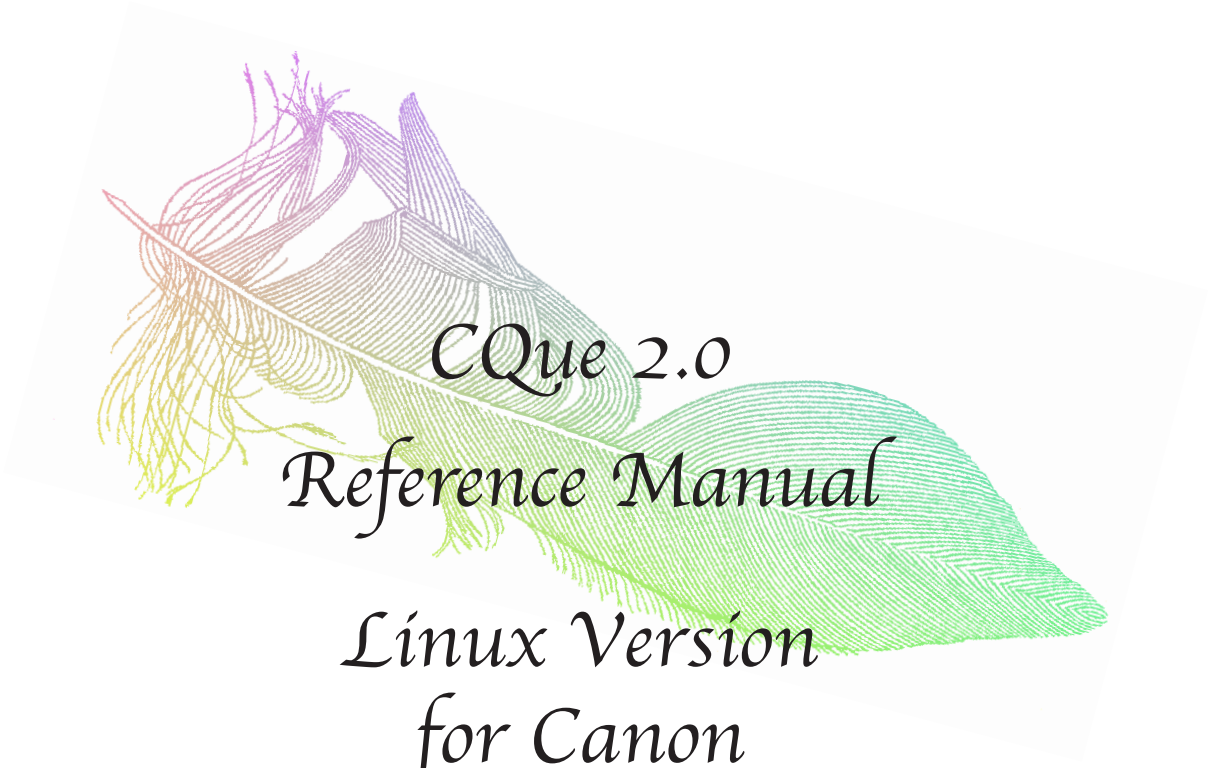
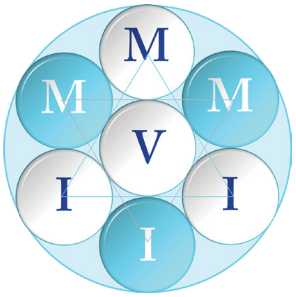


CQue 2.0
Reference Manual
Linux Version
for Canon



Mintze van der Velde Informatique
Chemin de la Vieille Fontaine 3
CH-1233 Bernex, Switzerland

Publication Ref. : 5410117C-3
Publication Date: June 2015
Version : CQue 2.0-9

©2007-2015 MV-Informatique
All rights reserved

For support, please contact your local Canon service technician/representative.

Contents

1 Introduction of CQue 1
1.1 Introduction 1
1.2 Compatibility 2

2 The General Concept 3
2.1 Some Terminology 3
2.2 The Modules of CQue 4
2.3 What Happens when Printing? 5
2.4 Before Installing CQue 7
2.4.1 Linux versus Windows Printing 7
2.4.2 Linux Printing: A Cook Book Approach 8
2.4.3 Linux Printing: When will CQue not work? 8
2.4.4 Linux Printing: CQue and OpenOffice, acroread, gimp, FireFox, GTK+ 8

3 Installing CQue 11
3.1 Obtaining CQue 2.0 11
3.1.1 Using the web 11
3.1.2 CD-ROM 11
3.1.3 Support 11
3.2 Starting the Installation CQue 2.0 12
3.2.1 Introduction 12
3.2.2 Installing a TGZ package 12
3.2.3 Installing an RPM package 13
3.2.4 Installing a DEB package 15
3.3 Running CQue 2.0 17
3.3.1 List of Queues 18
3.3.2 Main Menu 18
3.4 Setting up a Queue with CQue 2.0 22
3.4.1 Enter Creation Mode 22
3.4.2 Determine the Queue Type 23
3.4.2.1 A PostScript Queue 23
3.4.2.2 A PCL Queue 23
3.4.2.3 A PXL Queue 23
3.4.2.4 A Raw Queue 23
3.4.3 Enter the Queue Name 24
3.4.4 Select a Port 24
3.4.4.1 A Remote LPD Port 25
3.4.4.2 A Remote TCP/IP(9100) Port 26
3.4.4.3 A Remote IPP Port 26
3.4.4.4 A Local USB Port 27
3.4.4.5 A Local Port 28
3.4.4.6 A File Port 29
3.4.5 Select Printer Model 30
3.4.6 Device Options 31
3.4.7 Viewing the Printer Options 32
3.4.8 Creating the Queue 33
3.4.9 Advanced Options 34
3.4.9.1 How CQue Manages File Boxes, User IDs and Passwords for Different Linux Users 37
3.4.9.2 File Box Specification for UNIX Users 38
3.4.9.3 Accounting Printing Specification for UNIX Users 39
3.4.9.4 Force a User ID / Password 40
3.4.9.5 Secured Printing Specification for UNIX Users 40
3.4.9.6 Force a Password 41
3.4.9.7 Deleting an Entry of the Advanced Options Menu 42
3.4.10 Extra & Banner Options 43

3.4.11 Printing Multiple Copies of a Document with Stapling 44

4 Non-standard Installations of CQue 2.0 45
4.1 Upgrade Installation of CQue 1.0 to CQue 2.0 45
4.2 Upgrade Installation of SQue 9.X to CQue 2.0 45
4.3 Upgrade Installation of SQue 9.X to CQue 1.0 46
4.4 De-Installation of CQue 2.0 47

5 Miscellaneous Topics of CQue 2.0 49
5.1 Blob or Balloon Help of CQue 2.0 49
5.2 Filtering the list of Queues of CQue 2.0 50
5.3 Printing a Test Page with CQue 2.0 51
5.5 Reset of CQue 2.0 52
5.6 Printing to a File with CQue 2.0 53
5.7 About CQue 54

6 What to do with Problems of CQue 2.0 55
6.1 Introduction 55
6.2 After Installation of CQue 2.0 the Program Does Not Start 55
6.3 A Created Queue Does Not Print 57
6.4 Introduction to Debugging an Installation of CQue 60

7 Program Description of CQue 2.0 61
7.1 Introduction 61
7.2 The Program CQue.exe 61
7.3 The Program sicgsfilter 63
7.4 The Program setup 65
7.5 CQue and SELinux 66
7.6 The program sicnc 66

8 Configuration Files and Scripts of CQue 2.0 67
8.1 Introduction 67
8.2 Directory Layout of CQue 2.0 67
8.3 File Box and Accounting Configuration File 68
8.4 Adding a Printer Model to CQue 71
8.5 The Script cque 73
8.6 The file cqueInfo2.0 75

9 Using XPP for Passing of Printer Options 77
9.1 Introduction 77
9.2 Installing XPP 77
9.2 Creating an OpenOffice Printer for XPP 77
9.3 Printing from OpenOffice with XPP 80
9.4 An XPP Printer Queue 81
Appendix A: Accessing CQue from the Desktop 85
Appendix B: Command Line Options 86
Appendix C: A Note on PCL Printing 87
Appendix D: Release Notes 89

1 Introduction of CQue

1.1 Introduction

CQue 2.0 is a tool to create and manage printer queues on Linux platforms. It will be mostly used by system administrators, though once a queue is created, an ordinary user may benefit from *CQue 2.0* by the (optional) possibility to adjust 'last minute' printing paramaters. In a more classical sense, *CQue 2.0* allows for the creation of dedicated printer queues.

This manual provides details on all aspects of *CQue 2.0:* Installation, maintenance, scripts, etc. Parts of this document are rather technical and will not be required for a first approach to CQue. These pages may contain valuable information, however, for more advanced features of CQue, or for the system administrator who wants to customise CQue. These pages are indicated with a  sign on the bottom of the page, and may be skipped by the occasional reader.

The manual *CQue 2.0 Quick Installation and User Guide* for *Linux* provides enough details to enable a quick installation of *CQue 2.0.* Some parts of this guide will be present in this reference manual too.

The manual is organised as follows:

The *Introduction* gives general information.
The *General Concept* section gives information about the way CQue is designed and how it works. Even though this might look technical, it is recommended reading for an effective and efficient installation of CQue. It also gives some suggestions for various types of usage of CQue.
The *Installing CQue* section presents an example of an installation, much in the same way as the *CQue 2.0 Quick Installation and User Guide*.
The *Setting up a Queue* section presents various ways of how to use CQue.
The *What to do with Problems* section helps to answer questions that may arise if things do not work as expected.
The *Program Description s*ection gives detailed information about the various programs making up the CQue package.
The *Scripts and Configuration* section gives an outline of the various scripts of CQue and how to customise them.

The X11/Motif version of CQue 2.0 for Linux is shipped free of any license, i.e. there is no demonstration version.

To install CQue you must be logged on either as *super-user* or *root* (*System Administrator*). A user who is not super-user (i.e. a *normal user*) will not have the rights to add or delete printer queues. The setup program will not run if you are not super-user.

1.2 Compatibility

The package is compatible with and tested on the *Linux versions* shown in the following table.

Linux Workstation Type	32 Bits Operating System	64 Bits Operating System
RedHat (and Fedora Core)	7.2 (FC2) and newer	kernel 2.6-27 and newer
Mandriva	9.0 and newer	kernel 2.6-27 and newer
Suse (NOVELL)	8.0 and newer	kernel 2.6-27 and newer
Debian	2.0 and newer	kernel 2.6-27 and newer
Ubuntu	2.0 and newer	kernel 2.6-27 and newer
Oracle Linux	6.0 and newer	kernel 2.6-27 and newer
Knoppix	3.0 and newer	kernel 2.6-27 and newer
Linux (Intel) in general	kernel 2.4-7 and newer	kernel 2.6-27 and newer

Required packages CQue relies on are:

- CUPS
- foomatic-rip

The *Printer Manager* must be CUPS. As CUPS has become the standard printer manager for most of the current Linux distributions, CQue relies on the file and directory structure of Printer Description Files (PPD) as used by CUPS. Many main applications on Linux do the same, so that CQue and these main applications may share printer options through CUPS. For this reason no other printer managers, especially LPRng, are supported.

Standard CUPS installations will include the foomatic package by default. CQue explicitly relies on the foomatic-rip package for the transformation of one printer description language (PDL) to another.

We note that CQue is not a replacement of a printer manager. It is a separate layer fully compatible with the above mentioned CUPS printer manager. It does not modify any of the components of the native printer manager either.

CQue provides support for the following type of Canon devices:

Printer Description Language of Canon device
PostScript
PCL (PCL5e and/or PCL5c)
PXL (or PCL6)

Of course the Canon device itself would need the necessary hardware, if available, to support any of these PDLs.

2 The General Concept

2.1 Some Terminology

In this manual we will talk about different kinds of queues. For clarity we give the following description of how various terms will be used within this manual.

Physical Printer Queue

A *physical printer* queue corresponds with a printer queue, in the classical meaning of the word, assuring the connection of a computer with a printer. The files, sent to this queue, will be processed the one after the other. A physical printer queue may send the files as is, or in certain cases it is possible that the queue includes a *filter,* which processes the file passing through the queue. For example a transformation of the file from one format to another may be applied. The connection with the printer may be a network port, a parallel or USB port or yet another type of port. If the connection is through a network, then we speak about a *remote queue*. Otherwise we are talking about a *local queue*.

Logical Queue

A *logical queue* also has the property that the files will be processed the one after the other. But these files will not necessarily go directly to the printer. It is possible – as a matter of fact this is the task of a logical queue – to apply a specific processing of the files passing through this queue. After processing the files may be redirected to a physical printer queue in order to be output on the printer.

Filter

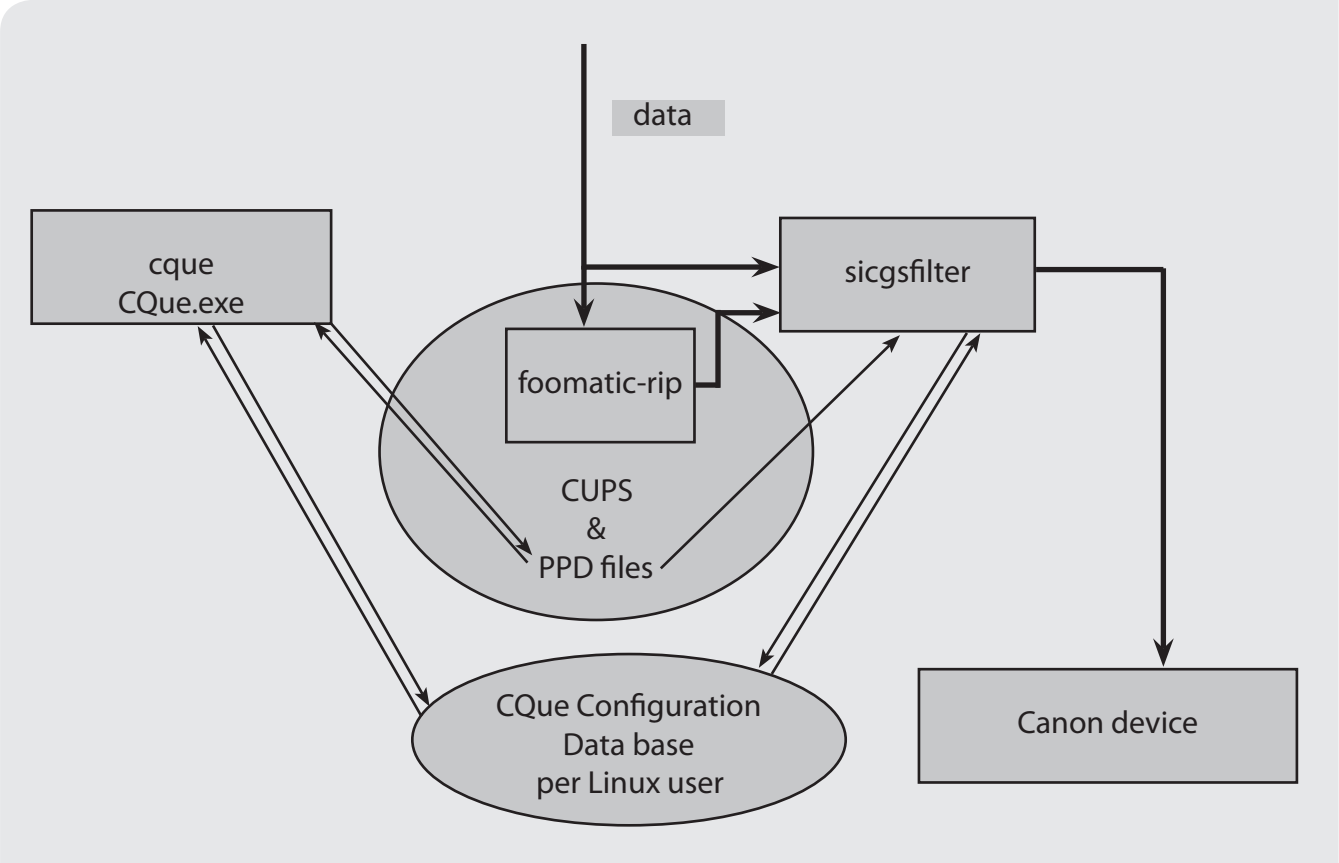
A *filter* is a program or executable file (script), which will be executed inside a printer queue. The syntax of such a filter is strictly defined by the print manager of the UNIX system and is therefore strongly system dependent. Filters of CUPS for example allow the user to pass user specific arguments to the filter. This is not possible (or very limited) for printer queues of the classical B.S.D. type of workstations.

CQue Queues

A queue of the *CQue type* is in principle a genuine CUPS physical queue with a filter that executes the CQue driver *sicgsfilter.* The program *sicgsfilter* will be described shortly. Most of the printer options may be managed either by CQue or by CUPS compatible tools. Some advanced options, however, may be managed by CQue only. These advanced options include mailbox printing, secured printing and printing with department accounting.

2.2 The Modules of CQue

The CQue package consists basically of the following modules:



- The program ***CQue.exe*** (or the script ***cque*** which runs *CQue.exe* including certain environment variables).
- The program ***sicgsfilter***, which is the driver taking care of the output of the queue.
- The ***foomatic*** package, which is not part of CQue itself. Yet CQue relies on the foomatic package especially for the transformation of printer description languages, if necessary. CUPS itself does the same.
- The ***database*** of CQue, which may include information per Linux login or user.

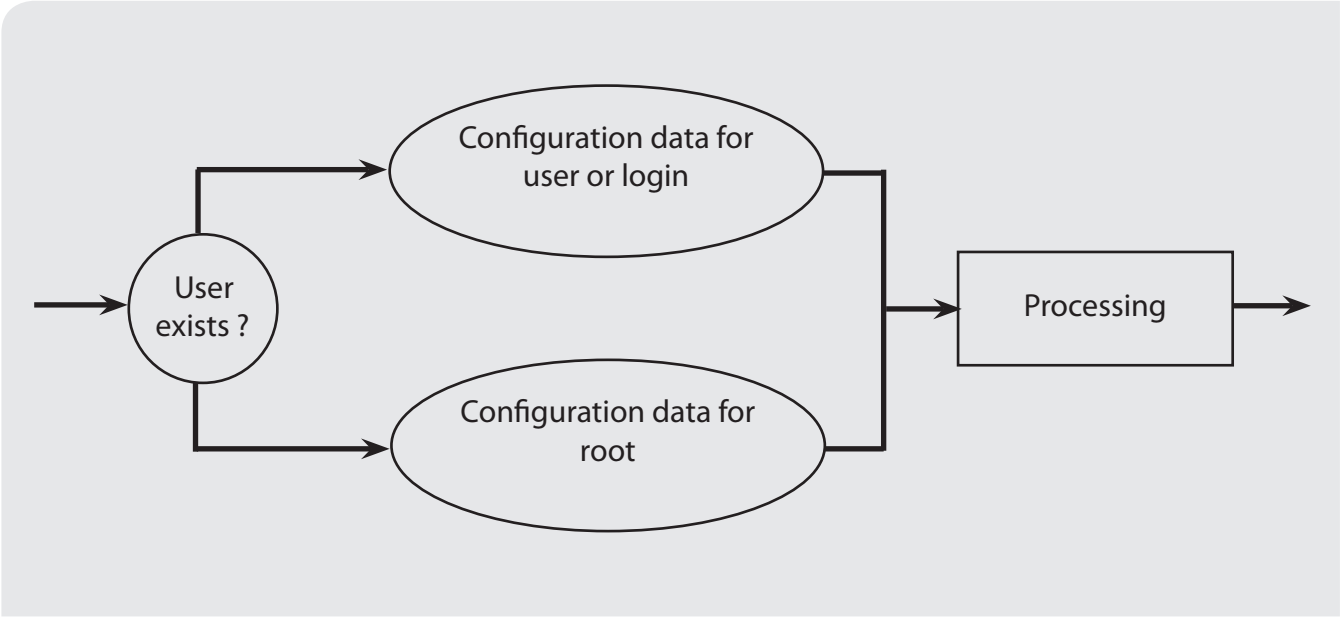
The print jobs of a CQue queue are processed by the program ***sicgsfilter***. They are never processed directly by *CQue.exe* itself. The program ***sicgsfilter***, however, is configured by CQue through the common database.

If no PDL transformation is necessary, then the data are directed straight to the ***sicgsfilter*** program. Otherwise CUPS will call foomatic-rip to perform the transformation and then send the resulting data to ***sicgsfilter***. ***sicgsfilter*** will always query the PPD information and the CQue Configuration Data base.

As of CQue 2.0 the GTK+ package may also read PPD information and thus format the data, for example for advanced options like mailbox and secured printing, etc.

2.3 What Happens when Printing?

When a CQue queue is configured, the finisher features of the queue will in general hold for all users. For advanced features like mailbox printing, secured printing and accounting, however, a distinction will be made according to the login or username. This is independent of the fact of whether these options are set by the GTK+ print interface or otherwise (command line, for example).
If a user is known to the data base of CQue, then those values will apply.
If a user is not known to the data base of CQue, then the values of root will apply.



CQue may process various file types, depending on the type of printer queue.

If a queue of CQue type has been configured for a ***PostScript Canon device***, then it will accept:

- ◊ *PostScript files*. This type of files is mainly considered as raw PostScript, i.e. without any printer specific PostScript code like duplex printing, stapling etc. It is the job of sicgsfilter to include these PostScript codes into the file. If the PostScript file itself includes already features of this kind, then the result may not be defined in a unique way.
- ◊ *Non-Postscript ASCII files*, which will automatically be transformed to PostScript through a filter. This filter will be a standard filter supplied by CUPS.
- ◊ *PDF files*. CQue will automatically convert the PDF data to PS. Thereafter it will insert any printer specific PostScript finishing code like duplex printing, stapling into the data stream. It will normally not sent the PDF file as is.
- ◊ *Image files* like TIFF, JPEG, X11, PNG, etc., which will also be transformed to a PostScript format internally. They will be printed 'as is' at standard printer resolution.

If a queue of CQue type has been configured for a *PCL Canon device*, then it will accept:

- ◊ *ASCII files*, to which the appropriate header with PCL/PJL data will be added. The ASCII data may be formatted so that the data will print more sophisticated than simply A4 and the default values of the copier or printer.
- ◊ *PostScript files*. CQue will automatically convert the PS data to PCL. Thereafter it will insert any printer specific PCL finishing code like duplex printing, stapling into the data stream. CQue can generate both black-white and colour PCL depending on the Canon device.
- ◊ *PDF files*. CQue will automatically convert the PDF data to PCL. Thereafter it will insert any printer specific PCL finishing code like duplex printing, stapling into the data stream. CQue can generate both black-white and colour PCL depending on the Canon device.
- ◊ *Image files* like TIFF, JPEG, X11, PNG, etc., which will also be transformed to a PCL format internally. They will be printed 'as is' at standard printer resolution.

Please note that ***genuine PCL files***, i.e. files including specific PCL and/or PJL code for page layout, fonts etc. ***are not supported by CQue***. They will be send "as is" to the Canon device, and no printer or finishing options may be set through CQue. For more advanced processing of PCL data we recommend using ***SQue 9.3 for Linux***, which is also available from the Canon web-site.

If a queue of CQue type has been configured for a *PXL Canon device*, then it will accept:

- ◊ *ASCII files*, to which the appropriate header with PXL/PJL data will be added. The ASCII data may be formatted so that the data will print more sophisticated than simply A4 and the default values of the copier or printer.
- ◊ *PostScript files*. CQue will automatically convert the PS data to PXL. Thereafter it will insert any printer specific PXL finishing code like duplex printing, stapling into the data stream. CQue can generate both black-white and colour PXL depending on the Canon device.
- ◊ *PDF files*. CQue will automatically convert the PDF data to PXL. Thereafter it will insert any printer specific PXL finishing code like duplex printing, stapling into the data stream. CQue can generate both black-white and colour PXL depending on the Canon device.
- ◊ *Image files* like TIFF, JPEG, X11, PNG, etc., which will also be transformed to a PXL format internally. They will be printed 'as is' at standard printer resolution.

Please note that ***genuine PXL files***, i.e. files including specific PXL and/or PJL code for page layout, fonts etc. ***are not supported by CQue***. They will be send "as is" to the Canon device, and no printer or finishing options may be set through CQue. (Also SQue 9.3 does not support PXL).
Please note that the copier should support the PDL: if you send PCL data to a copier which does not support PCL, then the job will not print.

2.4 Before Installing CQue

This section presents some concepts to be considered before installing CQue.

2.4.1 Linux versus Windows Printing

There is an important difference between the printing concept in a Linux environment and a MS Windows environment.

For UNIX each program has its own way of preparing and formatting the data to be printed. For Linux the situation is slightly better: many programs make use of the integrated print interface of GTK+, but unfortunately not all programs do so. For these latter a printer queue is basically a concept largely independent of the main program.

For MS Windows there exists a layer of Windows itself, that is highly involved in the printing. In this way each program (with a few exceptions) relies on the same driver with the same behaviour for each program. Thus, there is for each printer driver a *Properties* button, allowing for the modification of the printer parameters.

CQue tries to imitate the behaviour of Windows printing, if desired by the customer. Therefore, in a first approach, and independent of the PDL (Postscript, PCL or PXL), the UNIX printer queues as far as CQue is concerned can be divided into three categories:

1. *Raw printer queues*. This type of queue can be created by CQue in a homogeneous way on different UNIX platforms, though once created, CQue has no means to manage them. Such a queue would typically print A4 only.

2. *System (or foomatic) printer queues*. This type of queue will, when processing a job, take into account finishing options and advanced options specified at the time a job is submitted. It will also automatically transform the job data to the correct printer format (PostScript, PCL or PXL). If the GTK+ print interface is used even advanced options (mailbox, secured printing, printing with department account) will be taken into account properly.

3. *Dedicated printer queues*. This type of queue would, when processing a job, always apply the same finishing properties, without bothering the user. This type of queues is excellent when the user knows what he or she wants and needs only a limited number of different finishing options. Many UNIX users, for example, are quite happy with dedicated printer queues for: A4 (the default), A4 Duplex, A4 Duplex + Staple and possibly A3. The creation and configuration of these queues is very well possible with CQue.

The decision of the type of queue to create can always be made once CQue is installed. This is actually one of the first steps in using CQue.



2.4.2 Linux Printing: A Cook Book Approach

The following scenario may help to set up a printer queue before it will be created on most *Linux Systems*:

1 Activate ***creation mode*** on the first window of CQue.
2 Determine the ***type of queue*** required, i.e. will it be a raw, PostScript, PCL or PXL queue, and specify this on the first window of CQue.
3 Select a ***port***. For example you might select a remote LPD or a TCP/IP 9100 port.
4 Select the ***model*** of the printer. Depending on your choice of the type of queue under 1. the list of supported Canon devices may be different.
5 Specify the ***device options***.
6 Specify the ***printer options***.
7 Specify the advanced printer options, like file box printing, secured printing, printing with department accounting.

2.4.3 Linux Printing: When will CQue not work?

Even though the aim of this manual is to explain how CQue does work, there are some known situations where it will not be able to function correctly.

1. For Linux programs where the main program does highly sophisticated page formatting, especially for PCL printing. In that case the CQue formatting may conflict with the PCL formatting of the main program with undefined end results.
2. Linux environments with an alphanumerical terminal only. This configuration often consists of a server with a number of PCs running a terminal emulation on the Linux server. The current version of *CQue 2.0* requires an X11 graphical user interface.

2.4.4 Linux Printing: CQue and OpenOffice, acroread, gimp, FireFox, GTK+

Recent user interfaces of programs like OpenOffice, the gimp, acroread etc. on Linux support a printer interface allowing for the setting of printer options like paper format, duplex printing etc. CQue is compatible with these interfaces as long as these interfaces rely on the standard CUPS PPD files.
Some of these main applications rely on the printer tool of the GTK+ library, which is also compatible with the CUPS PPD mechanism, including the CUPS PPD extentions. There are exceptions, explained below.
Firefox, for example, uses its own printer interface, unfortunately bypassing much of the CUPS PPD options, and thus of CQue too.

Please note that the user interface structure for the passing of accounting user IDs and passwords currently is supported by print interfaces using the ***latest GTK+ release*** (version 2.2 and newer) only.
All programs making use of this GTK+ print interface benefit from this feature, bringing Linux printer a step closer to what one is used to on MS Windows. Some programs, like OpenOffice and Adobe acroread, however, maintain their own printer interface and are as yet not compatible with the CUPS PPD extensions. This is an unfortunate limitation of current printer dialogs on Linux. In section 9 we describe one work around for certain programs, making use of the XPP program.

If you are using CUPS on a server, including the broadcasting mechanism of CUPS, then any CQue queue (which is also a genuine CUPS queue) will be broadcast and thus be known on all CUPS clients.

3 Installing CQue

3.1 Obtaining CQue 2.0

CQue 2.0 can be obtained by:

- Downloading from the web.

or

- Obtaining a copy on CD-ROM.

There are three different archive formats:

- tar + gzip: a gzipped tar archive with extension: *.tgz.
- rpm: an rpm archive with extension: *.rpm.
- deb: a deb archive with extension: *.deb.

As there exist many different Linux distributions and various package management systems too, it may happen that your distribution does not support rpm nor deb packages by default. In that case we recommend using the *.tgz version of CQue, which should run on all known Linux systems.

In general we recommend to use either the rpm or deb package, as the installation of such a package is fully automatic, including the upgrading of any existing CQue 1.0 queues.

3.1.1 Using the web

You can download the driver from the Canon web site for your country or directly from http://software.canon-europe.com.

3.1.2 CD-ROM

To obtain a copy of CQue 2.0 on CD-ROM please contact your local Canon Sales Office.

On the CD-ROM the package will be available in both *.tgz, *.rpm and *.deb formats.

3.1.3 Support

For support, please contact your local Canon service technician/representative.
This chapter will describe the general installation of CQue for Linux environments.

✉ ***Both the RPM, DEB and the setup program will restart the CUPS printer manager after finishing.***

3.2 Starting the Installation CQue 2.0

3.2.1 Introduction

The following sections will describe an installation of CQue on a workstation which does not have any CQue installed. The same description holds for rpm and deb packages installation when a CQue 1.X versions is already installedd - upgrade installation. For a description of a manual upgrade installation of CQue or of a de-installation of CQue, please refer to the sections 4.2 and 4.4 respectively of this manual.

3.2.2 Installing a TGZ package

If you downloaded a copy of CQue 2.0 from the web in **.tgz* format the file will be in a compressed format. You will need to decompress the file, using an appropriate interactive package of gzip, and extract the contents to a designated directory on your system, for example ***/tmp***.
If you cannot decompress the file using this method you can extract the contents of the file using the following commands (we recommend doing this from the /tmp directory):

```
gzip  -d  cque-xx-2.0-9.tgz
tar  xvf   cque-xx-2.0-9.tar
```

or in one sweep (with the GNU version of tar):

```
tar  xzvf  cque-xx-2.0-9.tgz
```

where the 'x' depends on the language of the version of CQue.

The first line will decompress the archive, the second line will extract all files from it.

The extracted the files from a **.tgz* version you will need to start setup from the directory where you extracted the files to. Normally a subdirectory: cque-xx-2.0-0 will have been created.
The section XXX describes all options supported by setup. Here we only mention the -q option: If you run setup with the option ***-q1*** then setup will ask for permission to upgrade any previously CQue 1.0 printer queue.

To start a ***default*** installation, either double-click on the **setup** icon, or alternatively from a terminal window type:

```
./setup
```

The installation will silently start and will terminate with a restart of the CUPS printer

daemon. The start up may take a while when setup configures an SELinux module if necessary.

Otherwise, if you want to *control the upgrading of individual CQue 1.0* printer queues then from a terminal window type:

```
./setup -q1
```

The start up may take a while when setup configures an SELinux module if necessary.
If the setup program does not encounter any previous CQue 1.0 installation, it will behave as described above.

If it does encounter a previous CQue 1.0 installation, setup will ask for permission to continue the upgrade. You answer by typing ***y*** or ***yes*** if OK, or ***n*** or ***no*** if not OK, followed by the **<RETURN>** key.
For each CQue 1.0 printer queue encountered this procedure is repeated.

If you want to upgrade all following CQue 1.0 printer queues at once, you may also reply by ***a*** or ***all*** followed by the **<RETURN>** key, when asked permission for the upgrade of a printer queue.

Of course setup never modifies a non-CQue printer queue.

Please have a look at the log file: ***/var/log/CQue2.0_UpdateLog*** after running setup.

3.2.3 Installing an RPM package

The rpm package appears with its specific icon in a file browser as in fig. 1. The "look and feel" of the menus may differ among different Linux distributions, but the general way will resemble the example we present here (Fedora Core).

When you double click on the icon, the following menu opens up:

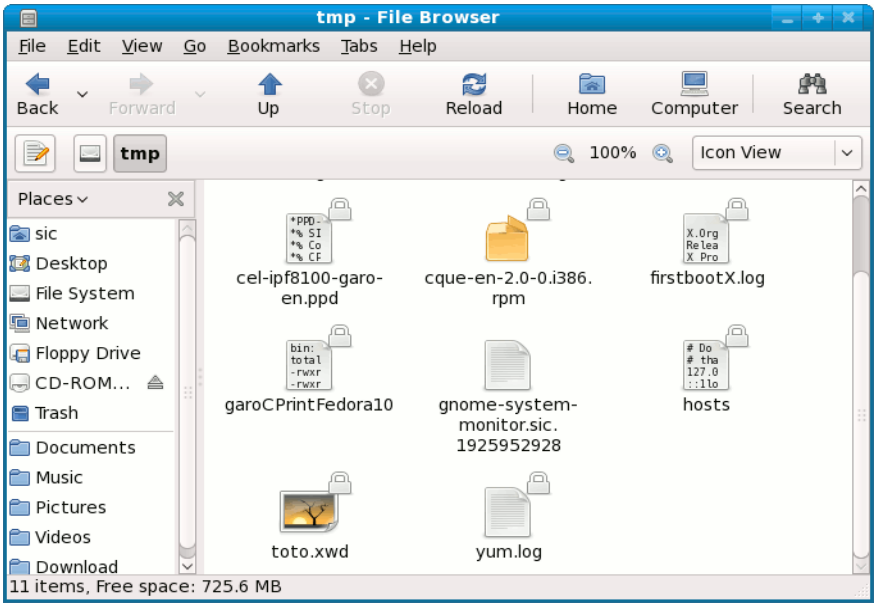


Figure 1 The CQue rpm package.

If you agree to continue with the installation, click on *Install*.
You may now be queried to identify yourself with the ***root*** password. Only the ***root*** has the right to add system software. Thereafter the following window displays:

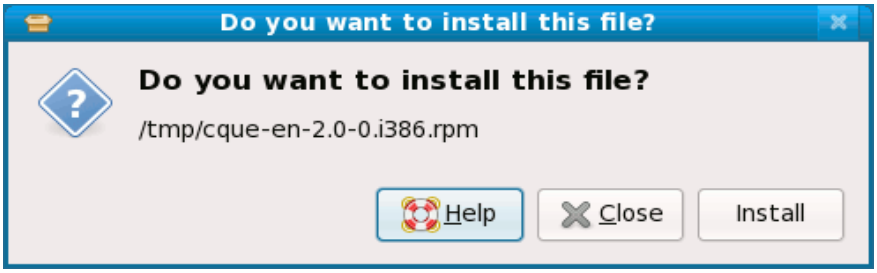


Figure 2 Confirm CQue rpm package installation.

As the CQue driver is not bound to a specific Linux distribution, the package has not been signed. It is, however, completely safe to continue with the installation.



Figure 3 Missing signature of the CQue rpm package.

The rpm manager checks any package dependencies.

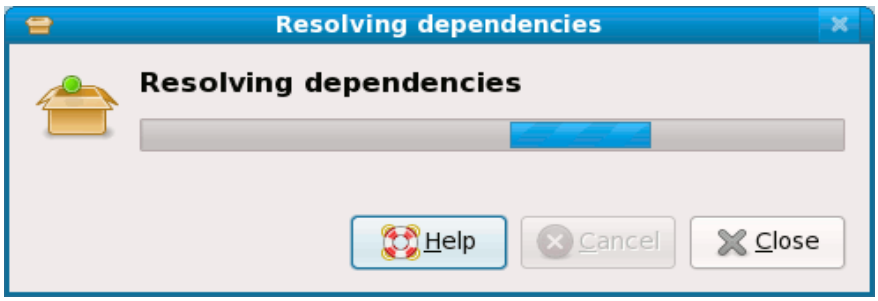


Figure 4 Resolving dependencies of the CQue rpm package.

And finally the package will be installed.

A review of the full installation procedure will be found in the file: ***/var/log/CQue2.0_UpdateLog***.

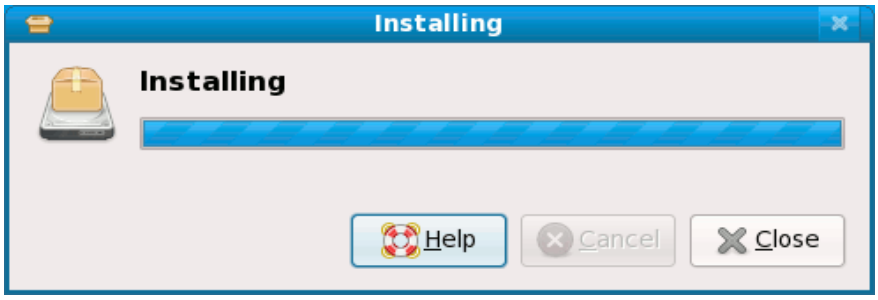


Figure 5 Installing the CQue rpm package.

The documentation will be installed into the directory ***/usr/share/doc/CQue2.0***.
Please note that if you upgrade from a CQue 1.0 installation an rpm installation will automatically update any CQue printer queue. You may check the file ***/var/log/CQue2.0_UpdateLog*** to verify which printer queues have been upgraded successfully.

If you do not want to extract the files and install them using this interactive method you can extract and install the contents of the file using the following command from a terminal window:

```
rpm -Uhv cque-xx-2.0-0-i386.rpm
```

If (obvious) dependency conflicts would arise (see release notes, appendix D), you may run:

```
rpm  -Uhv --nodeps  cque-xx-2.0-9-i386.rpm
```

The rpm program will automatically extract the files to the directory:

```
/opt/cel
```

and run the setup program in the same manner as described above. At the end de CUPS daemon will be restarted.

3.2.4 Installing a DEB package

The deb package appears with its specific icon in a file browser as in fig. 6. The "look and feel" of the menus may differ among different Linux distributions, but the general way will resemble the example we present here (Ubuntu).

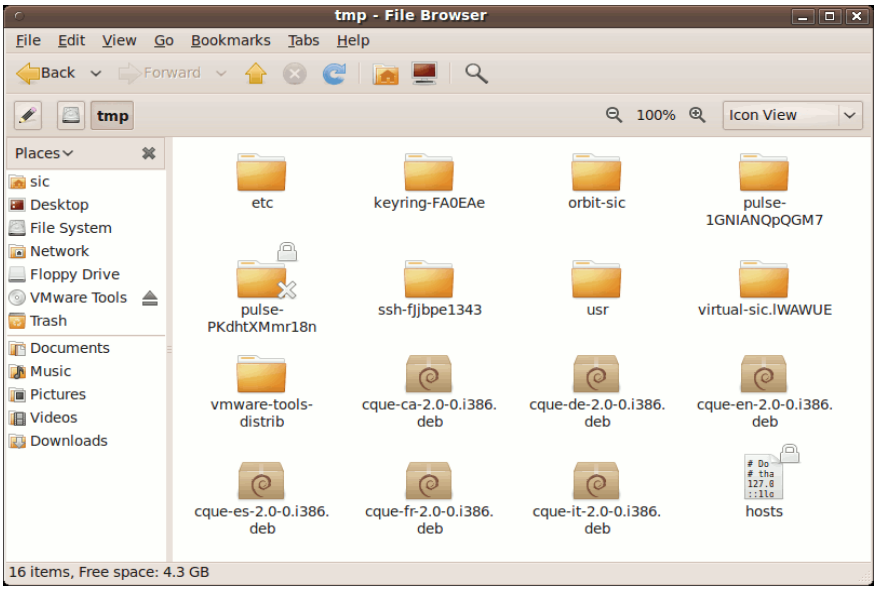


Figure 6 The CQue deb package.

When you double click on the icon, the following menu opens up:

A review of the package to install appears. To continue the installation, please click ***Install package***.

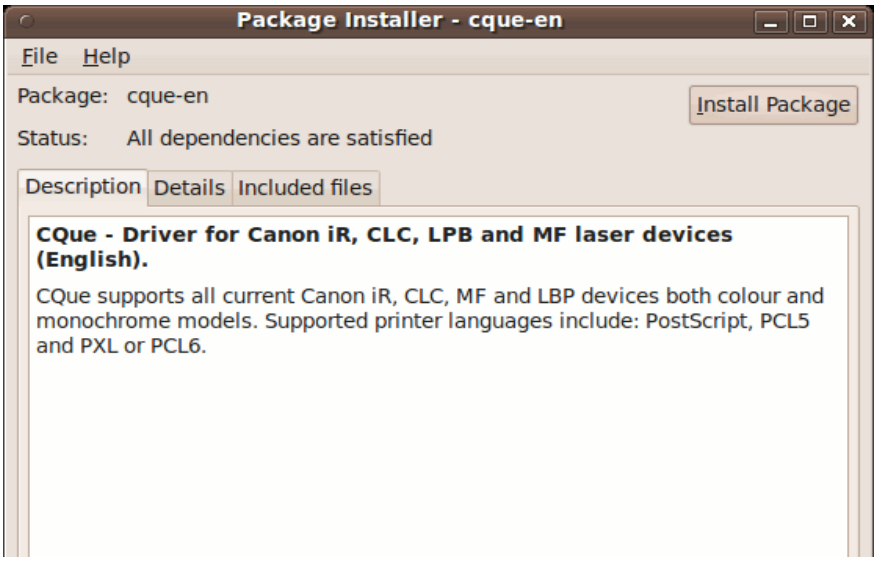


Figure 7 Confirm CQue deb package installation.

Then the installation of *CQue* will start.



Figure 8 Installing the CQue deb package.

At the end of the installation of *CQue* this window will display.



Figure 9 End of the installation of the CQue deb package.

If you do not want to install the package using this interactive method you can extract and install the contents of the file using the following command from a terminal window:

```
dpkg  -i  cque-xx-2.0-9-i386.deb
```

The *dpkg* program will automatically extract the files to the directory:

```
/opt/cel
```

and run the setup program in the same manner as described above.
At the end de CUPS daemon will be restarted.

On Debian you may also use the recommended program ***aptitude*** to install CQue, which is a more user friendly interface to ***dpkg***.

3.3 Running CQue 2.0

Once CQue is installed you, i.e. root, may run it directly from the setup procedure, or you may start it from a shell. If you were to run it from a shell, use the following command*:

```
cque
```

or

```
/opt/cel/bin/cque
```

Alternatively, you can use the Browser to the directory *'/opt/cel/bin'* and from there select the icon or file *cque*.
The file *cque* is a script, created during setup, which includes the necessary environment variables to run CQue.exe. *Never* try to run *CQue.exe* itself.
(Refer to *Appendix B* for further information on how to run *cque* from the "Start" menu of the desktop).

When you run *cque* the following window will display:

1 A list of existing Queues.
2 A Main Menu (File, Edit, Options, About).
3 A Menu of Tabs of which only one is visible, i.e. Queues.
4 Next and/or Previous buttons.
5 OK, Help and Cancel buttons.
6 A Menu for the Queue Type.
7 A text field for the queue name.

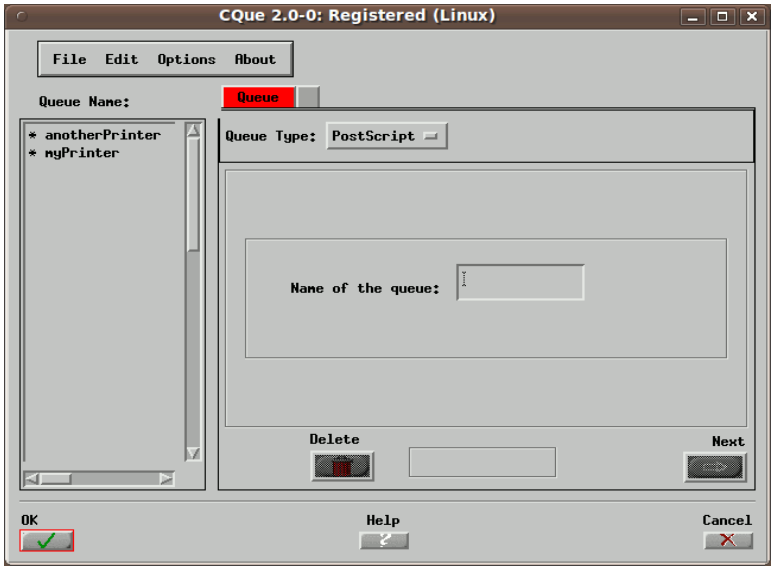


Figure 10 CQue Main Menu

*) A link will exist to /usr/bin/cque too allowing to run cque without any prefix.

3.3.1 List of Queues

In the *list of queues* there are two types of entries:

- If an entry has an asterisk (*) in front, then it is a printer queue, which is not managed by CQue. These could be local or remote queues generated with the native printer manager of your Linux system.

- The other queues are genuine CQue queues, which can be managed by CQue.

A CQue queue can also be managed by the native printer manager of your Linux system except for some advanced options (like forced mailbox printing, secure printing and department accounting).

3.3.2 Main Menu

The *main menu* contains the following sub menus:

1 File
2 Edit
3 Options
4 About

The details of these menus can be found in the *section 3.3.4.*

3.3.3 Main Menu of Tabs

The *menu of tabs* contains the following tabs:

1 Queues
2 Port
3 Model
4 Device Options
5 Print Options
6 Advanced
7 Extra

3.3.4 Buttons

The *Next button* navigates to the next menu or window. This button is initially greyed out as long as no valid queue name is specified.

Note: No *Previous button* shows on the first menu/window.

When you use the *Next* (and possibly) *Previous* buttons to navigate, the tab of the current menu will become red. You can also directly access a specific menu of the above list making use of the *tabs menu*. Finally you can navigate using the *Edit* menu.

The *Delete* button enables you to delete a selected queue. You must first select the queue from the list and then select *Delete*. A message will display asking you to confirm the deletion. If you confirm, the queue will be permanently deleted from the system.

The *OK button* and *Cancel button* are quite general and serve the obvious purposes. They can be used in many situations.

The *Help button* displays on-line help as follows:

1. Select *Help*.
2. The cursor changes into a question mark.
3. Select the object or window element you want information about.
4. A help window will display with the information on the requested object.

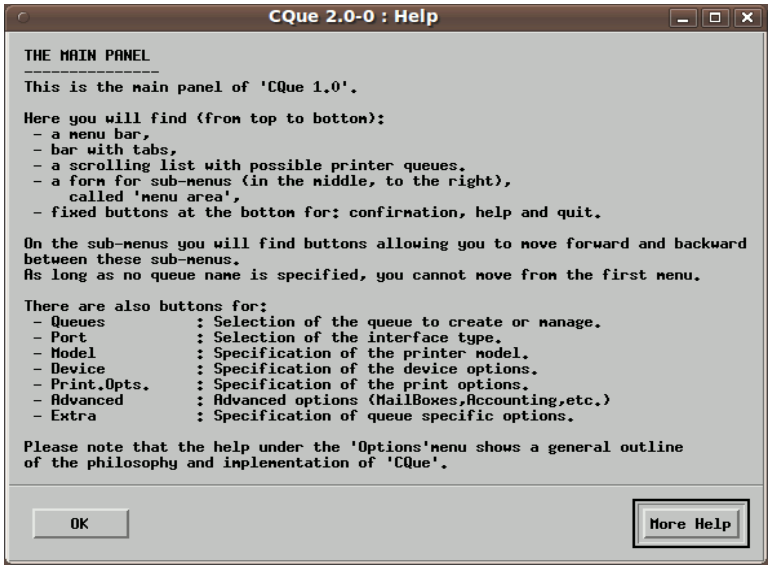


Figure 11 CQue Help Menu

The ***File Menu*** contains two buttons:

1. *Create*. A toggle button, when activated then CQue allows for the creation of a new queue. Otherwise existing queues can be updated/modified.
2. *Save*. When activated the configuration of the selected queue will be saved to disk.
2. *Cancel*. When activated CQue will abandon, after asking for confirmation.



Figure 12 The File Menu

The ***Edit Menu*** contains seven buttons, which will sequentially become available during the process of defining of the queue. They correspond with the various tabs and menus respectively:

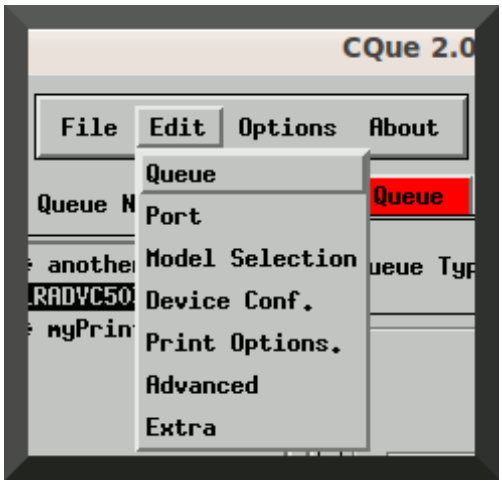


Figure 13 CQue Edit Menu

1 *Queue*. The tab for the specification of the name of the queue and its type (PS, PCL, PXL or raw).
2 *Port*. The tab for the specification of the port of the queue (LPD, TCP/IP 9100, USB, parallel, file, etc.).
3 *Model*. The tab for the specification of the model of the queue. This basically selects the PPD file to be associated with the queue.
4 *Device Options*. The tab for the specification of the device options of the queue (finisher modules, extra paper trays, etc.).
5 *Print Options*. The tab for the specification of the print options of the queue (Duplex printing, stapling, sorting, output bin, etc.).
6 *Advanced*. The tab for the specification of the advanced options of the queue (mailbox printing, secured printing, printing with accounting, etc.).
7 *Extra*. The tab for the specification of some extra, CUPS specific options of the queue (banner pages, error processing behaviour of the queue, etc.).

The ***Options Menu*** contains four buttons, with the following meaning:

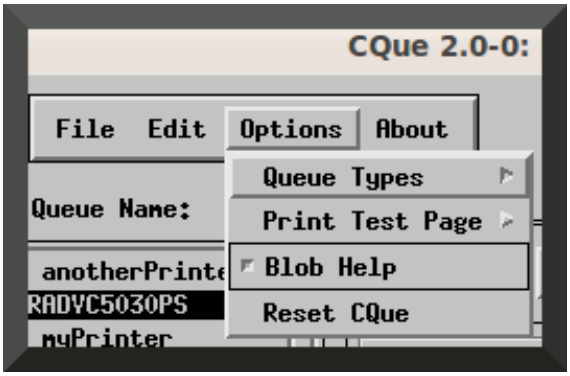


Figure 14 CQue Options Menu

1 *Queue Types*. This opens a sub-menu with which the type of queues shown in the list at the left of CQue can be modified. It is possible to show:

- all queues (CQue and system)
- system queues only.
- CQue queues only.

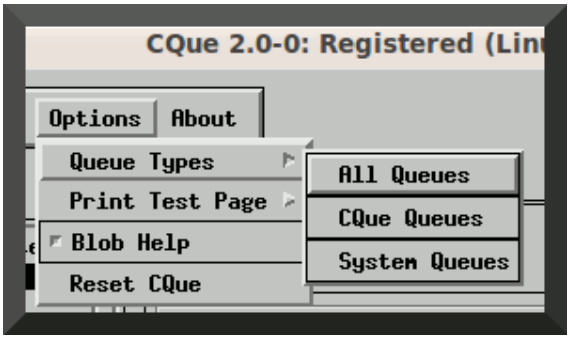


Figure 15 CQue Queue Types Menu

Note: if for one reason or the other the list of queues would be corrupted or perhaps empty, modifying the type of queues will refresh the list of queues.

2 *Print Test Page*. This opens a sub-menu with which one or more test pages may be send to the selected queue. The following options exist:

- 1 page ASCII.
- 2 pages ASCII.
- 4 pages ASCII.
- 1 page PostScript.
- 2 pages PostScript.
- 4 pages PostScript.

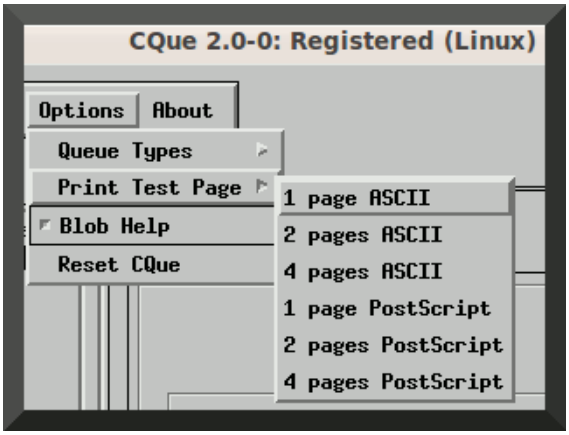


Figure 16 CQue Print Test Pages Menu

The ASCII page contains only pure ASCII text. The PostScript page contains both graphics and text. The options to send more pages allow for testing of duplex printing and printing with stapling. (See Section 5.3 for more details).

3 *Blob Help*. When the cursor is positioned over a menu or button a short message about its function or purpose may pop up in a so-called blob help window. This feature may be turned on or off with this menu. (See section 5.1 for more details).

4 *Reset CQue*. When a queue has been created and you want to create another queue this menu may be used to reset all the parameters of CQue to their default values. This is recommended.

The ***About Menu*** shows, when clicked upon, a window with version information about the origin of CQue. (See section 5.7 for more details).

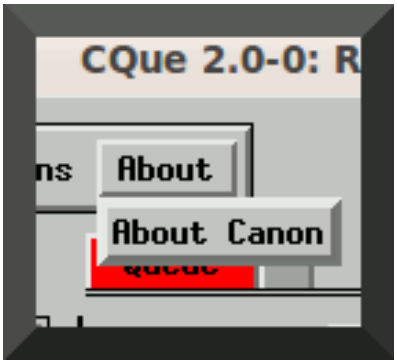


Figure 17 CQue About Menu

3.4 Setting up a Queue with CQue 2.0

The following procedure must be run to set up a printer queue before it can be created:

1 Enter creation mode.
2 Determine the type of queue required.
3 Enter the queue name.
4 Select a port.
5 Select the model of the printer.
6 Specify the device options.
7 Viewing the printer options.

3.4.1 Enter Creation Mode

(If you already created or managed other queues with CQue you may reset CQue first with "Reset CQue" under the Options menu).
Before creating a printer queue with CQue, you should first enter ***creation mode***. In this mode all actions of CQue apply to the new printer queue to be created.
You enter creation mode from the ***File*** menu. The Create button is a toggle button, which should be activated.



Figure 18 Enter Creation Mode

Once activated the main menu slightly changes, displaying a red message under the queue name text field as in figure 19.

Please note that creation mode stays active until you either de-activate it through the File menu, or when you finish creating the queue.

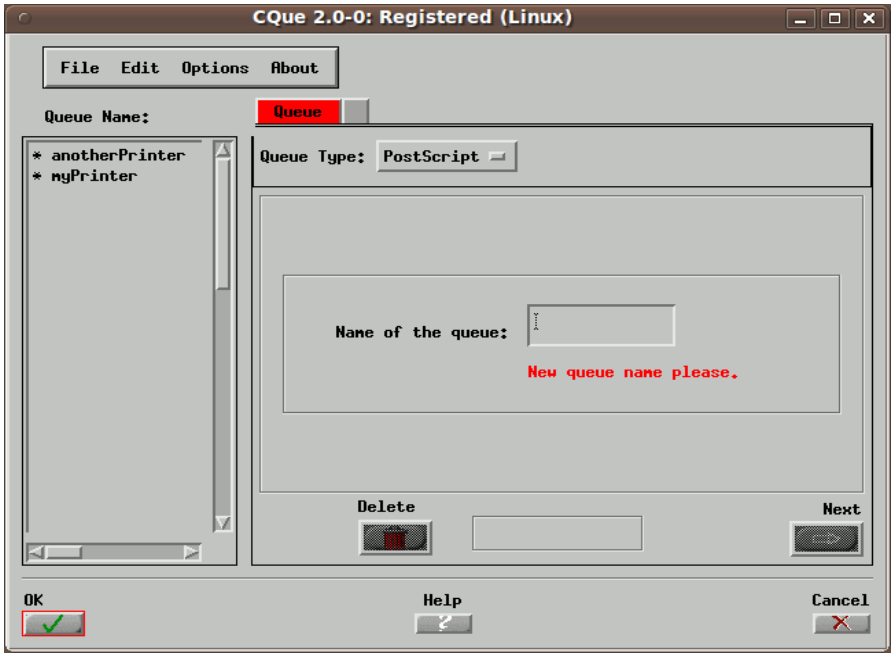


Figure 19 CQue in Creation Mode

When in creation mode the name of the queue you are working on will display in red.

3.4.2 Determine the Queue Type

Before you will create your first printer queue with CQue, you should first decide what the purpose of the queue will be.
There are four different types of queues:

1 PostScript
2 PCL (PCL5e or PCL5c)
3 PXL (PCL6)
4 raw

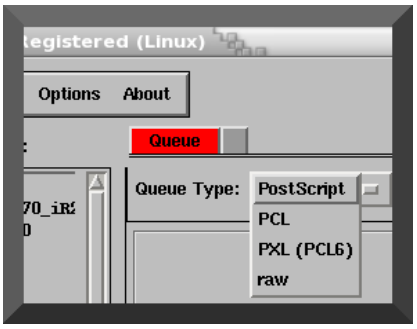


Figure 20 Queue Type Menu

3.4.2.1 A PostScript Queue

A *PostScript queue* handles PostScript data, but it will also try converting non-PostScript data to PostScript data, if necessary.

3.4.2.2 A PCL Queue

A *PCL queue* will print PCL data, including plain ASCII text. Input from applications generating PostScript will automatically be converted to PCL. In general PCL5e will be used. Most Canon colour devices, if supporting PCL, will be compatible with this type of PCL too.

3.4.2.3 A PXL Queue

A *PXL queue* will print PXL or PCL6 data, including plain ASCII text. Input from applications printing in PostScript will automatically be converted into PXL. Most recent Canon colour devices, if supporting PXL, will be compatible with this type of PXL. Note that older Canon colour devices may not support this type of PXL. If both PCL and PXL are supported please keep in mind that PXL generates more efficient data than PCL using maximum device quality.

3.4.2.4 A Raw Queue

A *raw queue* will not perform any processing and output the data 'as is' to the Canon device.

A raw queue can be created or deleted with *CQue*, but once it is created you can not modify any of its parameters.

3.4.3 Enter the Queue Name

Once you have determined the type of queue you require, enter a unique name for the queue in the *Name of the Queue* field. Please note that this name appears in red too at the bottom of the menu.

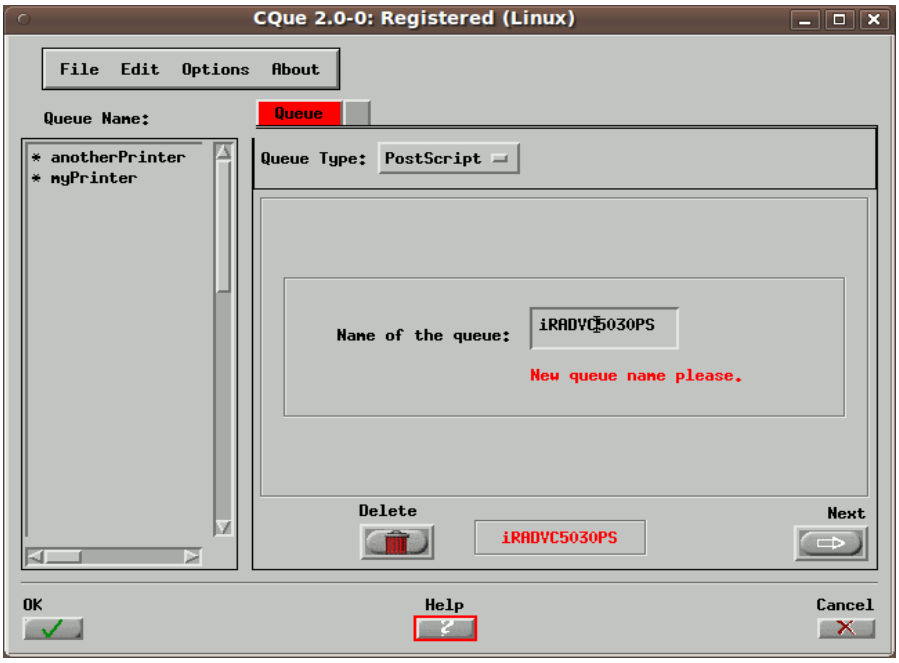


Figure 21 Entering a Queue Name

Select ***Next***. The menu for selecting a port will now display.

CUPS is rather free in the specification of queue names. Yet, classical UNIX machines would limit a queue name to a maximum of 14 characters and not allow for white spaces and other non-ASCII characters. We strongly recommend using this specification.

3.4.4 Select a Port

The *'port'* of the queue enables you to specify where the output of the queue will go to. There are the following possibilities:

1. Remote LPD
2. Remote TCP/IP(9100)
3. Remote IPP
4. Local USB
5. Local
6. File

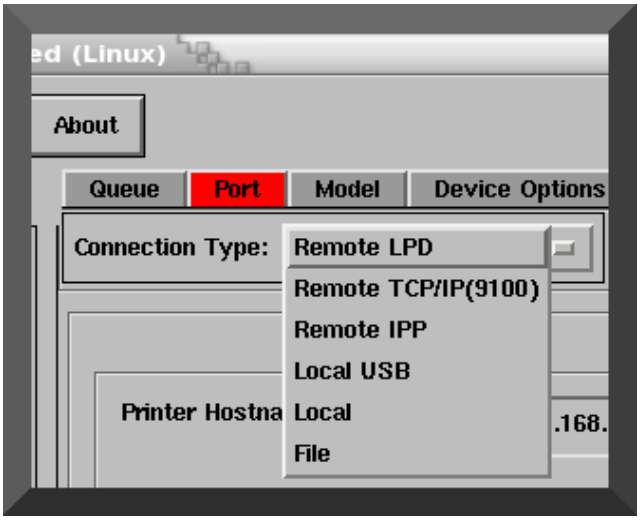


Figure 22 Port Type Menu

You may make use of any type of queues, but *we recommend using remote LPD queues* or possibly remote TCP/IP(9100).

Even though CUPS is using the IPP protocol itself for many internal processes, not all Canon devices support this protocol by default.

When connecting the Canon device through a USB port, please make sure the cable is connected to the Linux workstation before creating the queue.
When you want to print to a file on disk, please note that by default CUPS does not allow this. You would have to modify the cupsd.conf file. (See section 5.4 for more details).

3.4.4.1 A Remote LPD Port

When we select a remote LPD port, the following window will display:

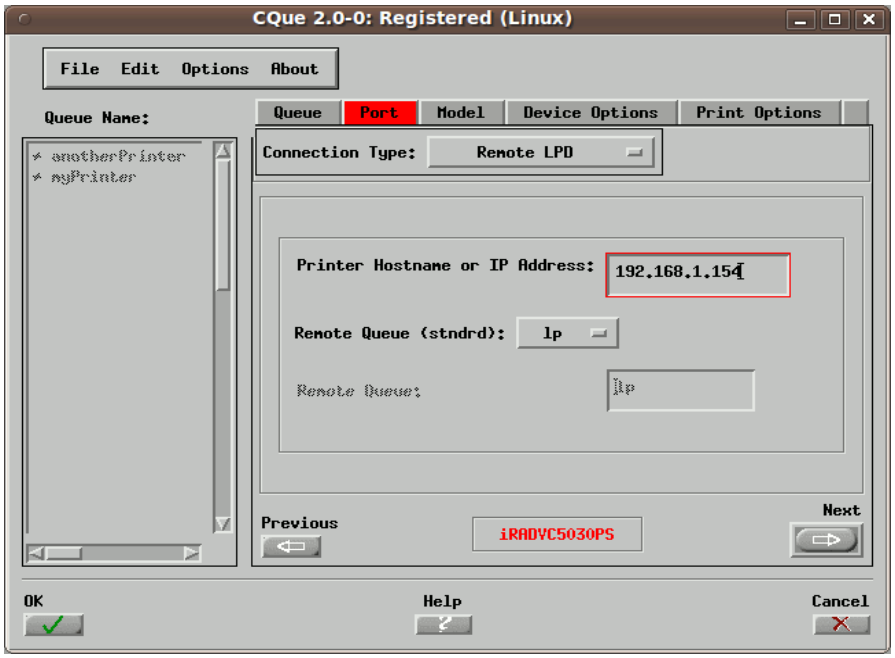


Figure 23 A Remote LPD Port

You must specify the *hostname* or the *IP address* of the Canon device.

And the name of the *remote LPD queue* on the Canon device together with the name of the Remote LPD Queue. A list of pre-defined names is provided; if you cannot find the name, then you can select "Other queue" and type the name in the text field called *Remote Queue*.

Please note that for Canon devices with EFI controllers the remote queue name ***must*** be either ***print*** or ***hold***.

This type of queue will be most common for Linux.

If the *'next button'* is not becoming available or not reacting, please verify on the preceding menu that the queue type is really other than *raw*.

3.4.4.2 A Remote TCP/IP(9100) Port

When you select a remote TCP/IP(9100) port, the following window will display:

You must specify the *IP address* or the *hostname* of the Canon device.

You may specify a port number too, but for most copiers/printers the port number should be 9100.

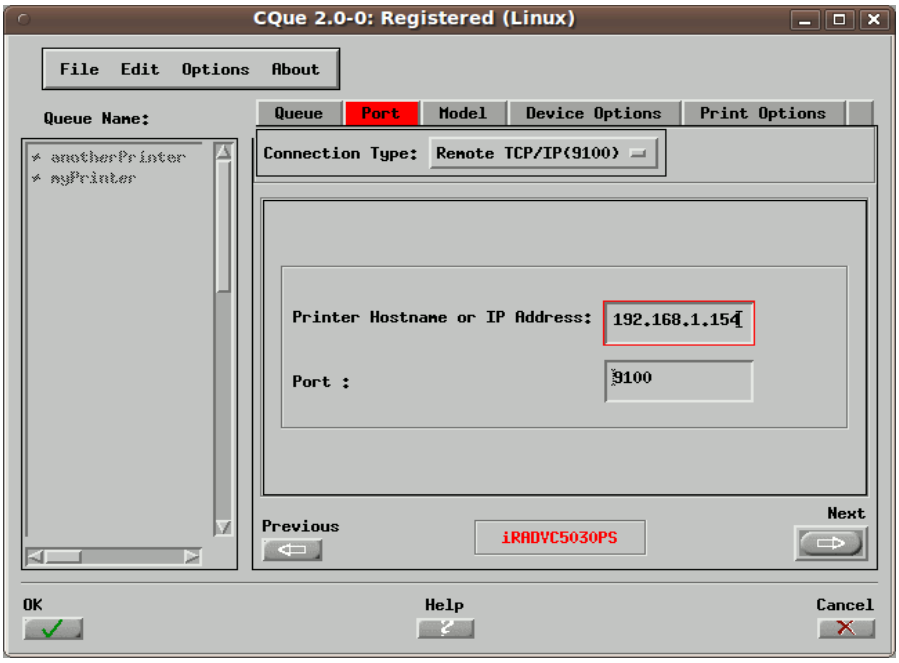


Figure 24 A TCP/IP (9100) Port

3.4.4.3 A Remote IPP Port

When you select a remote IPP port, the following window will display:

You must specify the *IP address* or the *hostname* of the Canon device.

You must specify a remote IPP printer too. For Canon devices you may leave ***ipp***. If printing through a MS Windows server, you might enter: ***printers/remotePrinterName*** where ***remotePrinterName*** corresponds with the printer on the server.

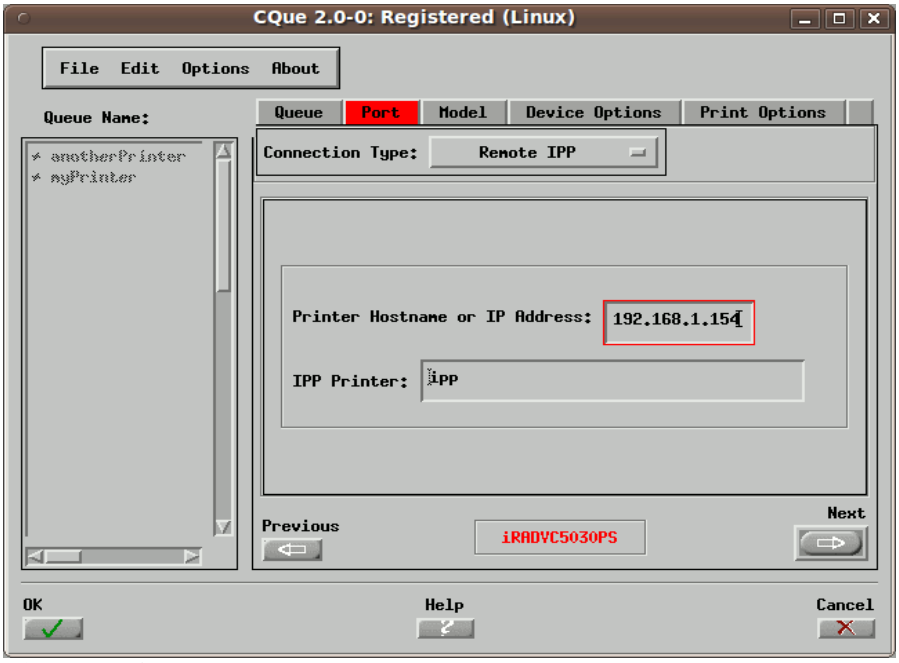


Figure 25 An IPP Port

Not all Canon devices may support this IPP protocol.

3.4.4.4 A Local USB Port

Please make sure before selecting this type of port that the Canon device is connected to your Linux system.
When you select a local USB port, the following window will display:

A list will display with all recognised USB devices. Select the line corresponding with your Canon device by double clicking on it.

If for some reason the device would not show up in the list, you may select manual USB and enter a device manually. In that case that would probably be something like:

/dev/usb/lp0

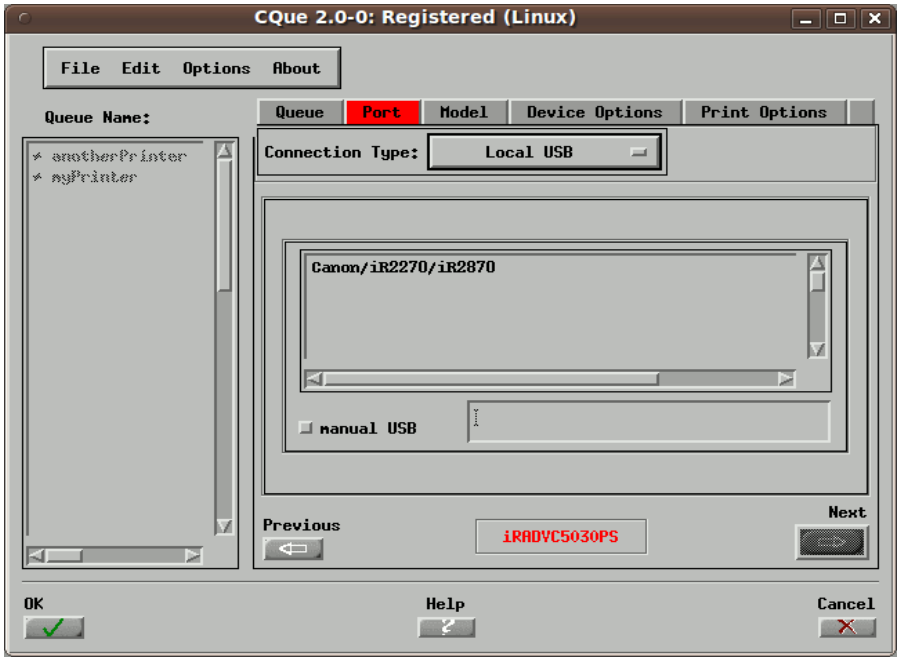


Figure 26 A local USB Port

The first method, i.e. double clicking on the line in the list, is the preferred one. In that case CQue will select the device the same way as CUPS does. If more than one USB devices are connected at the same time to your system, the order of connection does not matter in this case.
When a device is specified as */dev/usb/lp0* or so, and more than one USB devices are connected to the Linux system, than it does matter in what order the devices are connected.

Not all Canon devices may support this USB protocol.

3.4.4.5 A Local Port

When you select a local port, the following window will display:

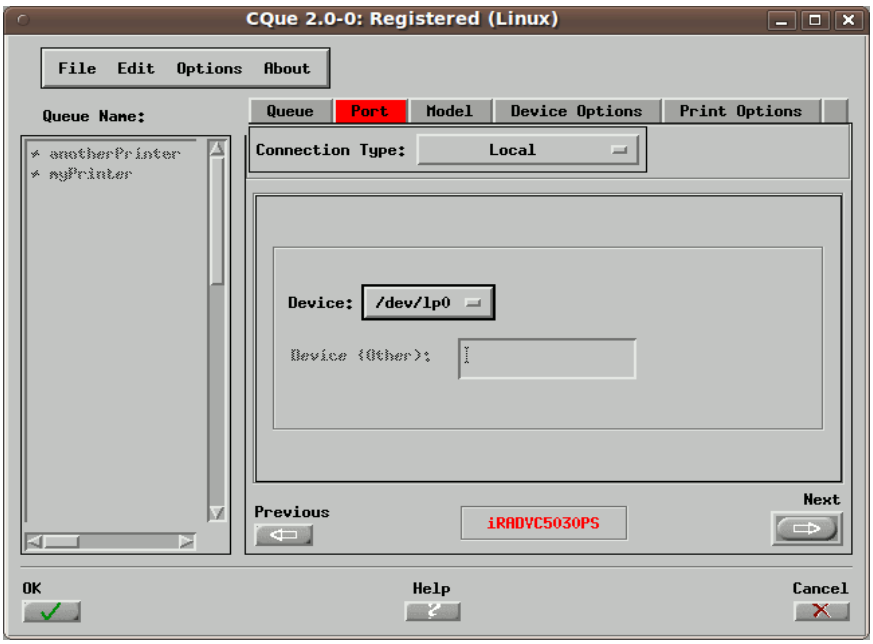


Figure 27 A Local Port

To specify the device of the local port you may click on the button for the devices.

A list with predefined devices will display corresponding to parallel or serial port(s) on your system. If the device you want to use is not listed, you must select *Other* and enter the device name into the text field called "Device (Other)".

The device */dev/null* is the so-called null device, which would be of use for logical queues only.

CQue will rely on the printer manager for the processing of the local device.

Many recent Canon devices do not support parallel or serial ports anymore.

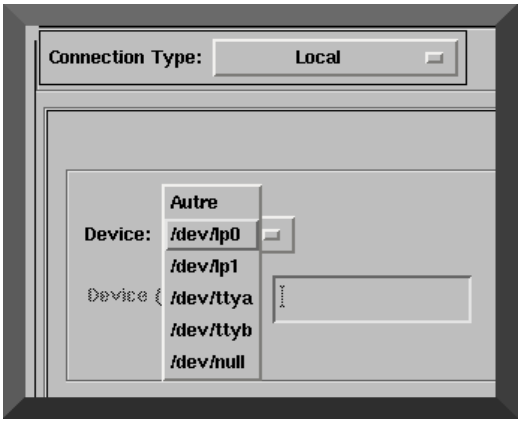


Figure 28 Supported Local Ports

3.4.4.6 A File Port

When you select a file port the following menu will display:

You must specify the name of a file into which the output will be saved.

With a file browser you can also browse the file system of your workstation, to search for an appropriate location.

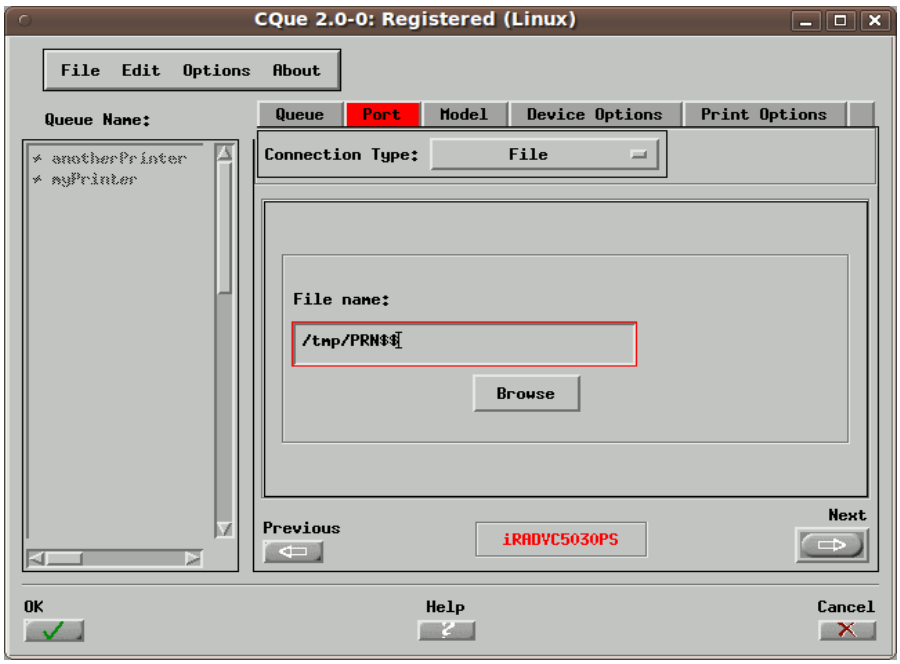


Figure 29 A File Port

A technical note:
If you append two dollar signs to the specified name of the file, each print job will be printed into a separate file, the name of each file consisting of that name extended with the process id. Thus each job will be printed in a different file.
For example, */tmp/PRN$$* will print to a file in the */tmp* directory with name *PRNXXXX* where *XXXX* corresponds with the process id of the print job (this is *not* the id attributed by the print manager to the job).

Please note: The CUPS printer manager does not allow for printing to file by default. You would need to modify the ***cupsd.conf*** file. See section 5.4 for more details.

3.4.5 Select Printer Model

The model of the Canon device includes all the features of the printer such as the installed finisher modules and the options related to these finisher modules.

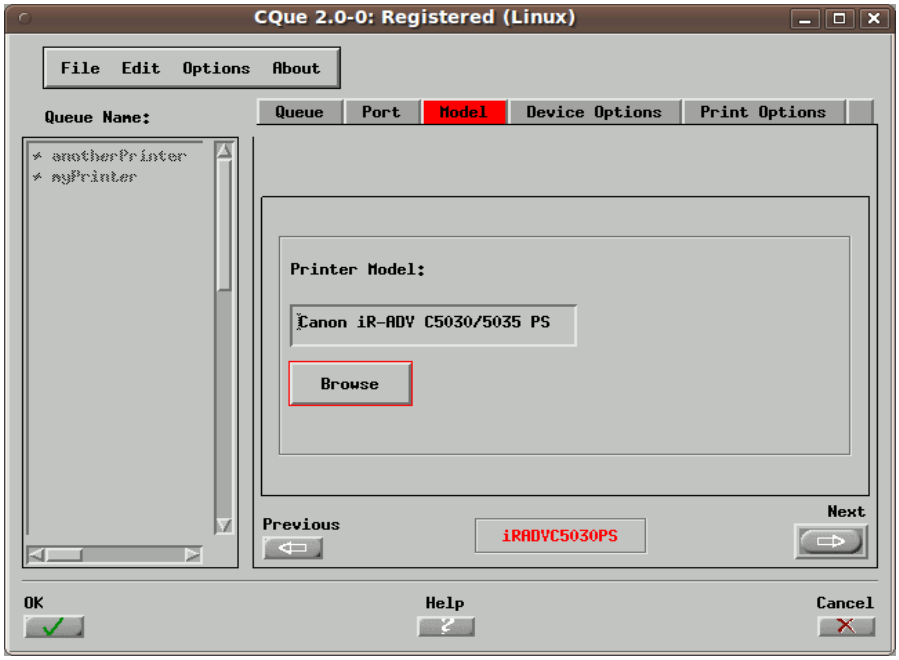


Figure 30 The Printer Model

CQue uses a number of pre-installed printer model files. Use the Browse feature to find and select the printer model you require. Please note that the list may be different depending on the type of queue you want to create (PS, PCL or PXL).

If your printer model is not in the list, please refer to the Section 5.5 for details on how to add a printer model.

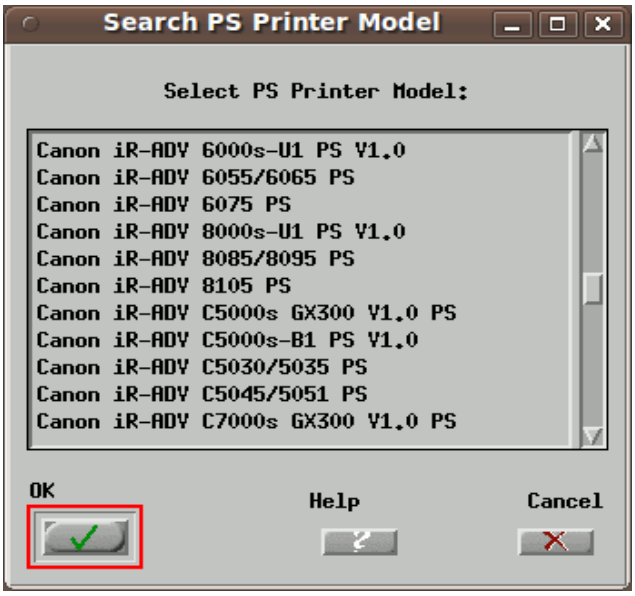


Figure 31 Model Browser

3.4.6 Device Options

Device options define the hardware and software components of the Canon device. They must be specified on the window under the “Device Options” tab. The current version of CQue does not detect the various finisher modules of the copier automatically. For the correct specification of the printer options (section 3.4.7), however, it is important to know what the exact capabilities of the Canon device are.

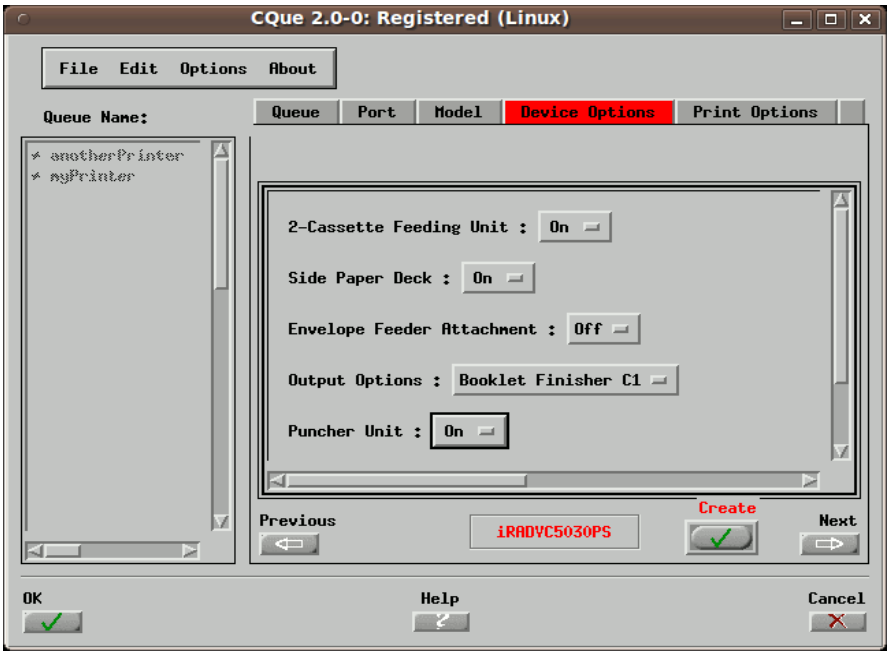


Figure 32 Device Options

On this menu you must select those features of the Canon device corresponding with the installed hardware and or software modules of the Canon device.

3.4.7 Viewing the Printer Options

Finally you should specify the so called printer options on the window under the "Print Options" tab.

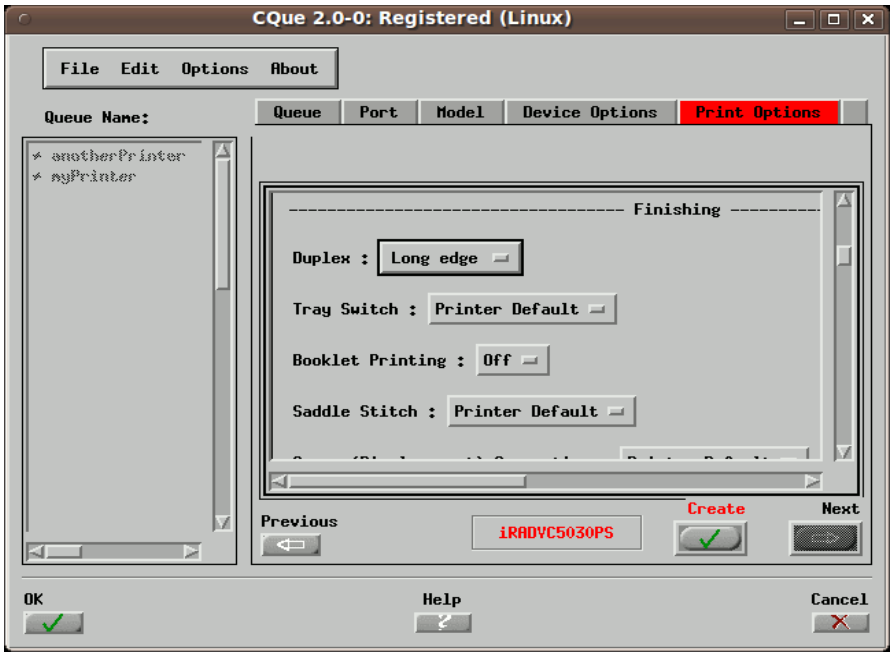


Figure 33 Printer Options

You can ***update*** or add the data of the printer setting with the ***Update*** button, but only if the queue is already created. (fig 34).
If you are in *creation mode* then there is no update button. Instead there is a red ***Create*** button (fig 35). The creation of the queue is described in the next section.

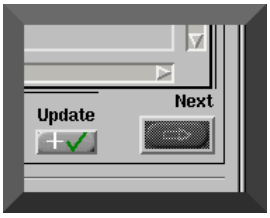


Figure 34 Printer Options Update

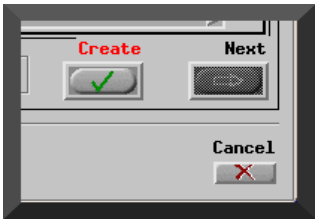


Figure 35 Printer Options Create

This window also allows for the setting of some Canon device specific printer options, like destination, mailbox number, user ID and password as well as secured printing password.
Some main applications like OpenOffice, acroread etc. may not allow for the setting of password and/ or user ID strings. This is a feature due to the different types of printer interfaces used by these applications.

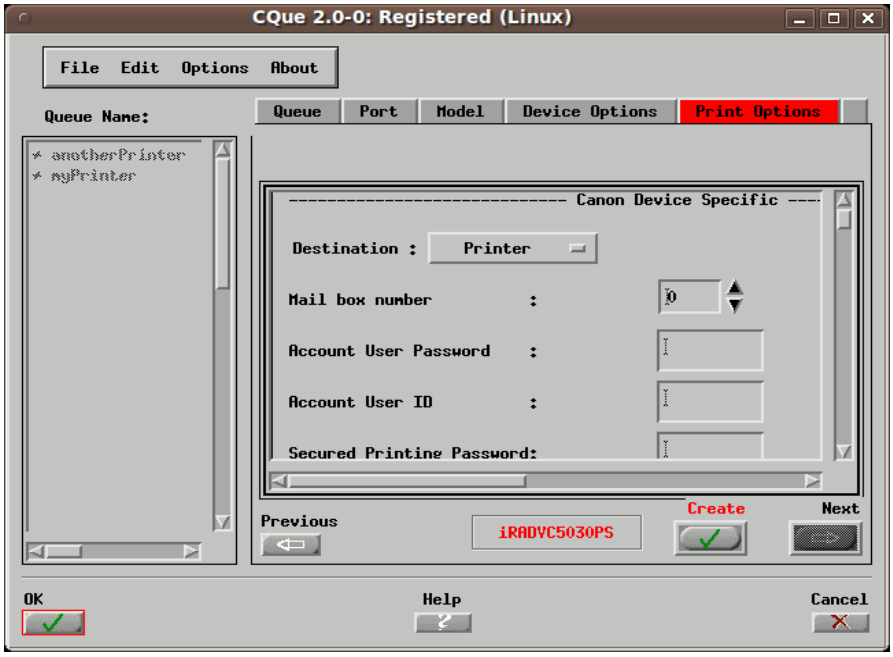


Figure 36 Canon Device Specific Printer Options

Section 3.4.9 describes how to deal with these Advanced Options in more detail.

3.4.8 Creating the Queue

After setting up the required data for the queue, i.e. its name, type, port, device options and printer options, you can now create the queue.

To create the queue, select *Create* from the lower right of the *CQue main window* (see fig. 33, 35 or 36 on the previous page).
The following window will display:

This window shows a review of the data for the queue to be created. If the data are correct select *OK* to create the queue.

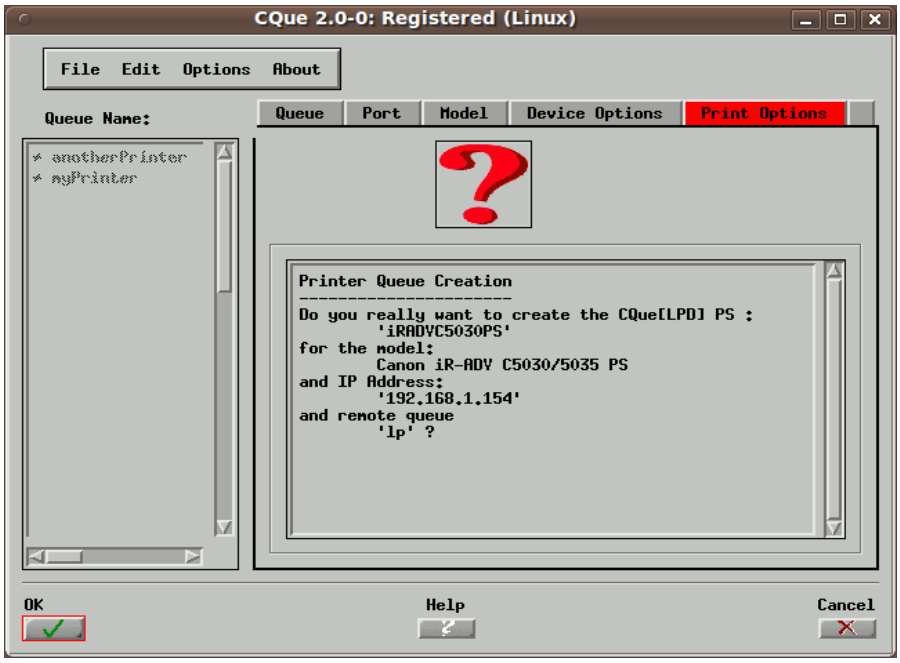


Figure 37 Confirmation of Creation of the Queue

Please note that once the queue has been created the creation mode ceases to be active. The name of the queue now appears in black instead of in red. You may add advanced data, such as mail box or secured print information and update these data using the *Update button* (see fig. 34 on the previous page) on the various menus or the *OK button* on the main menu.

3.4.9 Advanced Options

The following advanced options are available:

1. *Mail box (or file box) printing*: The job will be sent to one of the file boxes of the copier and will stay there to be printed at a later time. A mailbox may contain a limited amount of documents (most Canon devices support a maximum of 100 documents). There is also a limit to the number of total pages being stored (please refer to the documentation of your Canon device). An advantage of printing from a mailbox is that most of the finishing features may still be modified once the document is received in the mailbox. A mailbox may be protected by a password.
   Documents stored in a mailbox will automatically be purged after a time period to be specified on the Canon device.
2. *Printing with (department) accounting.* In this case the user should have a valid user ID and password on the Canon device, on which department accounting should be activated. The job to be printed will be credited on the account of the user.
   The user ID and password have to be registered on the Canon device. For each account a maximum quota of pages (sometimes different for black/white and colour pages) may be specified.
   Accounting feed back is only possible through the Canon CPCA protocol. For the PostScript, PCL and PXL queues we therefore cannot give any user friendly feedback. If a job is not accepted on the Canon device, for example because of exceeded quota, the job will be silently ignored.
3. *Secured printing.* The job will be sent to a public mail box (with a password) to be printed at a later time. The user will be required to type in the password on the Canon device panel to start the printing of the document.
   In contrast to habits in MS Windows environments no pop-op window is used to confirm the password. If XPP is used it is possible to modify the password through XPP (see chapter 9). If XPP is not available it is possible to use a secured password on the command line. Otherwise the password is fixed by CQue (see section 3.4.9.3).

The following table shows which combinations of options are allowed (these properties are typical of the device and do not depend on the operating system):

Combination	**Supported**
Mailbox + Accounting	Yes
Mailbox + Secured Printing	No
Accounting + Secured Printing	Yes

In many cases the above operations require user intervention locally on the Canon device, for example to specify user id's and passwords, or to print a job once it arrived in a mailbox, etc. Yet most of these operations can also be performed through a web browser having access to the Canon device, i.e. knowing the required passwords.

You may find more information in the documentation of the Canon device itself.

The latest GTK+ interface supports advanced options directly from the PPD. Hence applications using the GTK+ interface will allow for setting of advanced options directly from an application when printing. Fig. 38 shows an example of setting such print options.

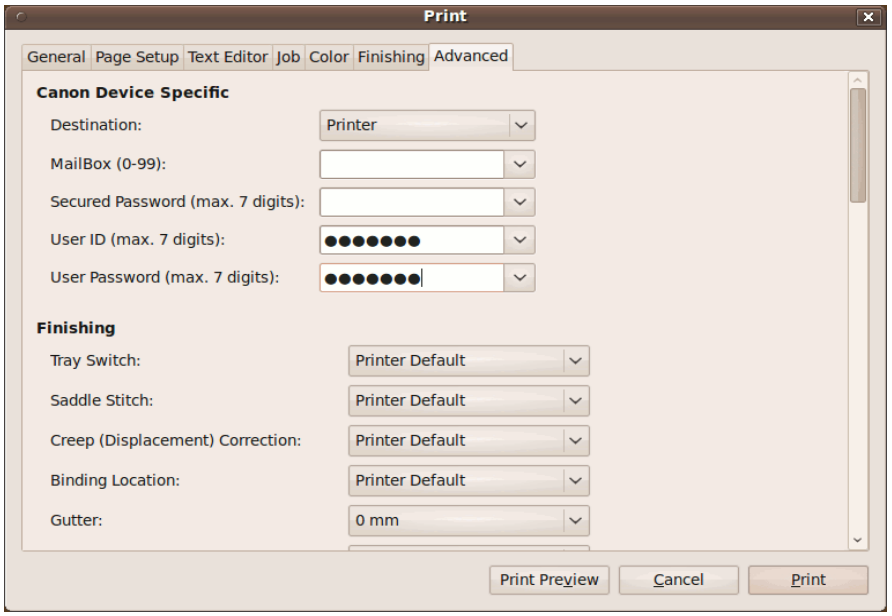


Figure 38 Advanced Options: GTK+ user interface

Making use of these advanced options is one of the strong features of CQue. Even if some of the main applications may not provide mechanisms to pass user identifiers or passwords, the mechanism used by CQue allows a system administrator to impose department accounting in a way, which is transparent for the ordinary user. In that case no password passing is required at all, as CQue will take care of this automatically.

When re-printing or re-opening an older session and then printing, the GTK+ interface does not always show advanced options in a correct way. If, for example in the text editor ***gedit***, like in fig. 38 you would have activated department accounting with a user ID and a user password, these would not be shown again by the GTK+ interface! But

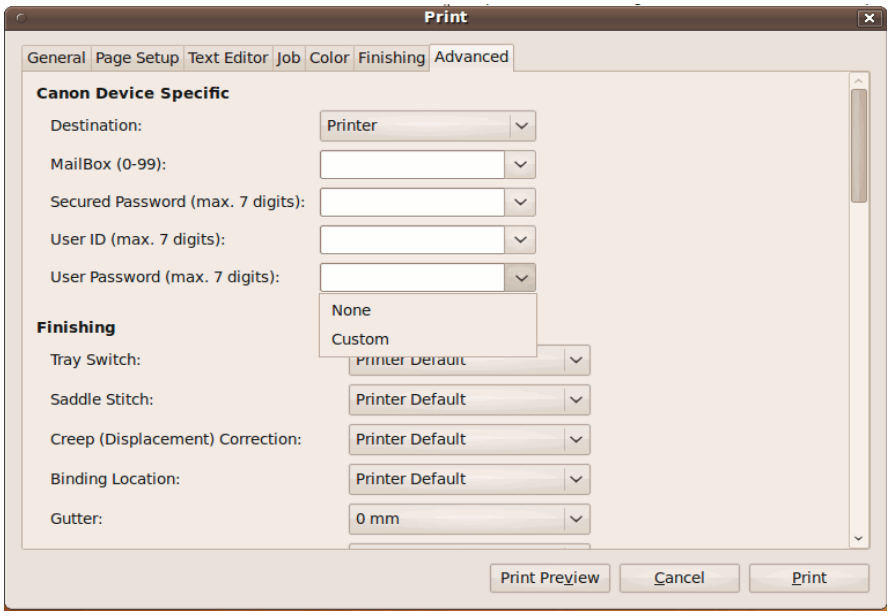


Figure 39 Advanced Options: GTK+ user interface

they would still be active.

This may be a limitation of the current GTK+ implementation. Please note that ***gedit*** maintains a hidden file ($HOME/.gnome2/gedit/gedit-print-settings) into which all printer settings are stored and thus retrieved when restoring a previous session.

Other applications, which also use the GTK+ print interface, do not have this behaviour.

If you want to reset an advanced option, i.e. de-activate it, please click on the right drop down menu of the option you want to reset, and there select “NONE”. If you were to select “CUSTOM,” then you must specify a value (password or ID).

3.4.9.1 How CQue Manages File Boxes, User IDs and Passwords for Different Linux Users

As we have seen the Canon device knows about features like *file boxes, user ID-s* and *passwords*. The Linux, operating system knows about *user names* or *login names*. The *root* user plays a special role, both on the Linux level and for CQue.
Of course, the root is the only user with enough rights to modify the global behaviour of these advanced options of the printer queue. But for CQue the root user plays another role too, as will be clear from what follows.

Within CQue you may create a list of users or logins and specify Canon device features for each user. CQue will associate Linux logins to Canon device features in the following way:

	root	***normal user***
Filebox	Each job of a normal user not in the list will go to the file box.	Each job of this user will go to the file box.
Secured printing	Each job of a normal user not in the list will use the specified password.	Each job of this user will use the specified password.
Accounting printing	Each job of a normal user not in the list will use the specified user ID and password.	Each job of this user will use the specified user ID and password.

Perhaps not all possibilities will be of interest to all applications, but this approach is very powerful as some examples will show further below in this section.

All advanced entries also come with a so-called “force” flag: If this flag is set, then the associated value cannot be changed from the printer setting menu of any application (OpenOffice etc.). If the flag is not set, then the user has the freedom to modify a value, if the user interface properly allows for such modifications.

The force flag does not have very much meaning for the root user (except when root himself would be printing): once a root value is set, it overrides any selection made by a user who is not in the list. This is not the case for a user who is in the list.

If the force flag for a certain user is set (by the system administrator) then that user will not be able to modify the associated value from any application.
Otherwise values for this user may be changed.

3.4.9.2 File Box Specification for UNIX Users

To specify a file box, click on the *box button* on the Advanced options menu. The following window will display:

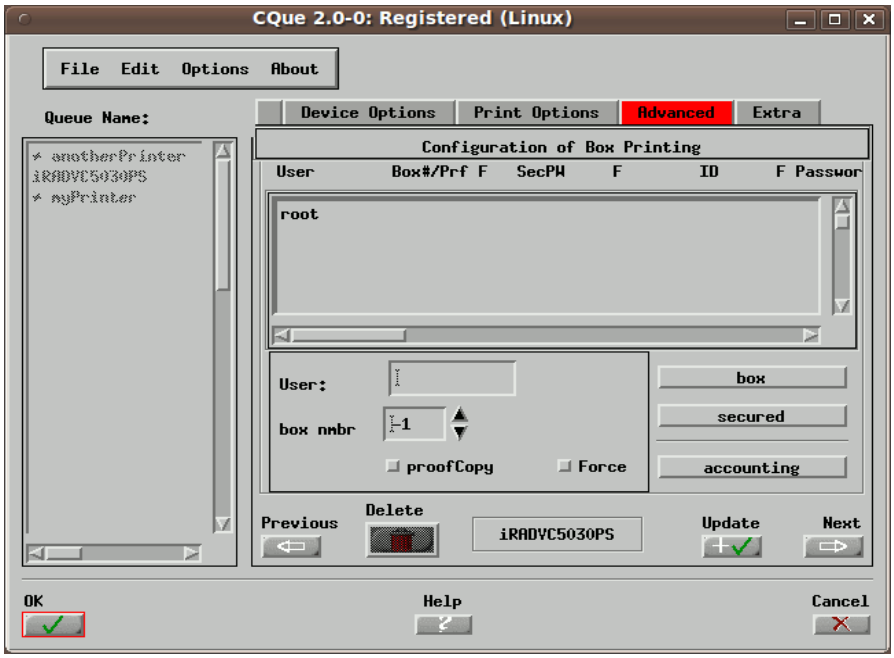


Figure 40 Advanced Options: Filebox

From this window you must specify the:

1. user name.
2. file box number.
3. force flag.
4. proofcopy flag.

The *user name* can be any legal UNIX login.
If you specify *root*, please note that the values for that user will hold for any other user not in the list.

The root user has a special meaning:
If root is present, then the jobs of all users will be sent to the file box of 'root', except if a user itself is specified in the list (with a different file box number). In that case, and only for that user, his own file box number would be used.

The *file box number* can be '0' up to a maximum specified by the copier, normally '99'. A value of '-1' indicates no file box.

The *force flag* will determine whether a user has the right to modify the setting from within any application like OpenOffice: If set, then the value may not be changed, otherwise the user may, if he wishes, still modify the file box number.
Please note: if a file box is specified here, automatically the destination of the document will not be the printer, but the file box. This is independent of the setting for the destination in any application printer menu.

The *proofcopy* flag indicates whether a proof copy should be printed when a job is submitted to the Canon device. If this is set, then each time a job is sent to the Canon device, a copy will be printed and the job will then be stored in the file box. This is useful if you need to print further copies of the document after inspecting the proof copy.

You can add and/or update the values for a particular user using the *Update button*.

Examples:

- If you would like to create a queue that will print to file box 3 for all users, then you would specify *root* as user and *3* as file box number.

- If you would like all jobs for the user *sic* to go to file box number 7, then figure 41. shows the parameters.

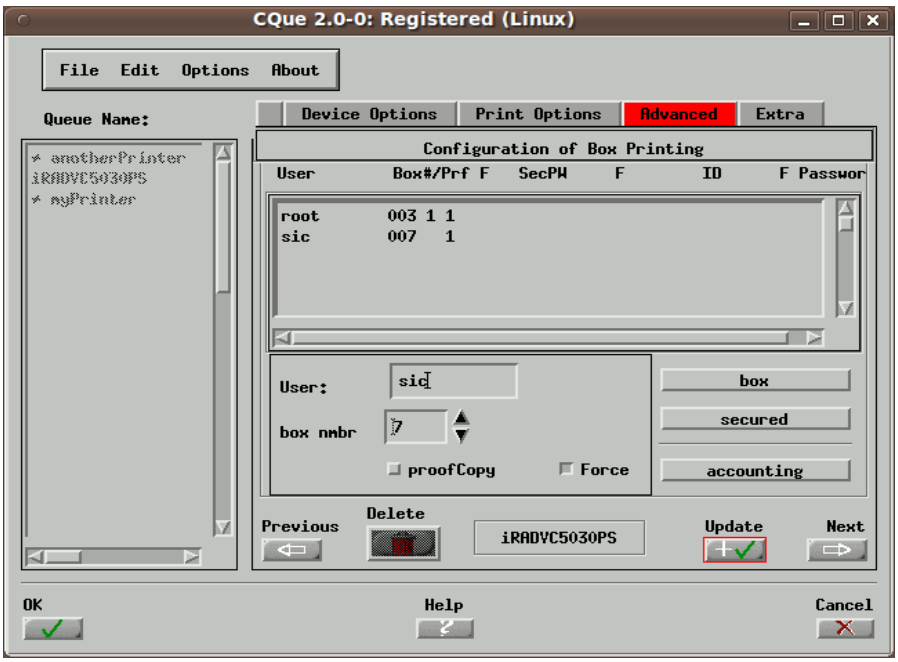


Figure 41 File Box for the user 'sic' and 'root' (all other users)

3.4.9.3 Accounting Printing Specification for UNIX Users

To specify Accounting Printing click on the *accounting button* on the Advanced options menu. The following window will display:

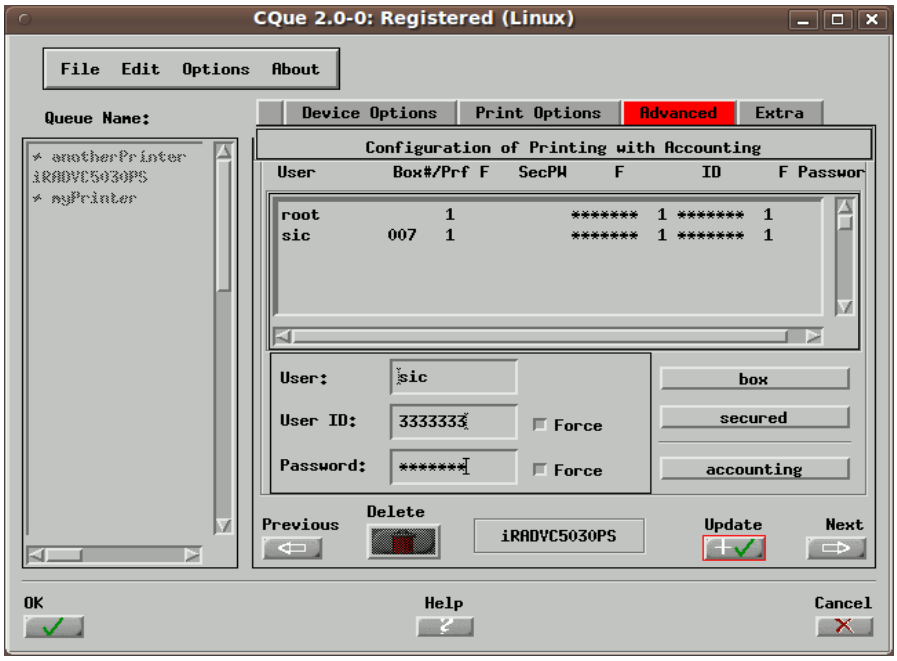


Figure 42 Advanced Options: Accounting

From this window you must specify the:

1. user name.
2. user ID.
3. user password.
4. force flags.

The user name can be any legal UNIX login.
If you specify *root*, please note that the values for that user will hold for any other user not in the list. The root user has a special meaning: If root is present, then the jobs of all users will be processed using the user id and password of 'root', except if a user itself is specified in the list. In that case, and only for that user, his own user id and password would be used.

The user ID and password are numerical values of up to seven digits.
You *must* specify a user ID, but the password is optional.

The current version of CQue does support department accounting, but is not able to give any feedback. Therefore, if printing quota are used up, further printing will be refused silently by the Canon device.

3.4.9.4 Force a User ID / Password

Next to the text field for the *user ID* and the *password* there is a toggle button *force*. If this button is selected, then the corresponding value will be forced.

If a value is *forced*, it will be taken into account automatically by CQue. Thus each job sent to a queue with forced values for the user ID and password will print making use of this user ID and password, *without any possibility for the user to modify these values*.

If a value is *not forced*, it will be taken into account by CQue, but the user still has the possibility to modify the values from the application he is printing from (if the printer setup menu of that application properly supports modification of these values).
The example of figure 41. shows the case for our user 'sic' with user ID 3333333 and a password which is hidden from display.

3.4.9.5 Secured Printing Specification for UNIX Users

To specify Secured Printing click on the *secure button* on the Advanced options menu. The following window will display:

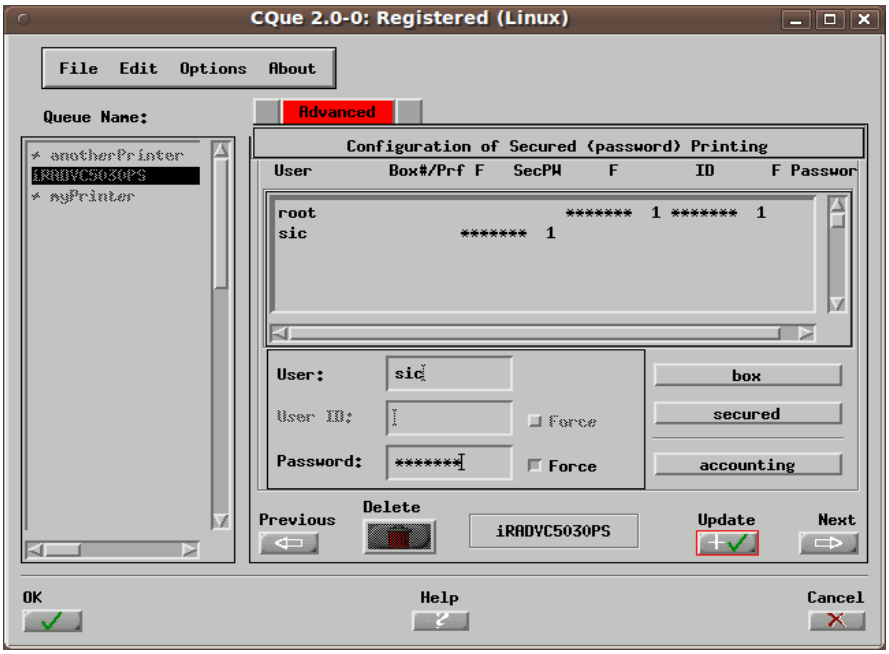


Figure 43 Advanced Options: Secured Printing

From this window you must specify the:

1. user name.
2. password.
3. force flag.

The user name can be any legal UNIX login.
If you specify *root*, please note that the values for that user will hold for any other user not in the list. The root user has a special meaning:

If ‘root’ is present, then the jobs of all users will be sent making use of the secured password of ‘root’, except if a user itself is specified in the list. In that case, and only for that user, his own secure password would be used.

The user password consists of a numerical value of up to seven digits.

3.4.9.6 Force a Password

Next to the text field for the *password* there is a toggle button *force*. If this button is selected, then the corresponding value will be forced.

If a value is *forced*, it will be taken into account automatically by CQue. Thus each job sent to a queue with forced values for the secured password will print making use of this user password, *without any possibility for the user to modify this value*.

If a value is *not forced*, it will be taken into account by CQue, but the user still has the possibility to modify the values from the application he is printing from (if the printer setup menu of that application properly supports modification of these values).

The example of figure 43. shows the case for our user ‘sic’ with a password which is hidden from display.

3.4.9.7 Deleting an Entry of the Advanced Options Menu

If you were to delete an entry from the list of users, double click on the entry to select it. The waste basket (*Delete* button) will become accessible.

When you hit the *Delete* button the entry will not be deleted immediately. The following verification window will display:

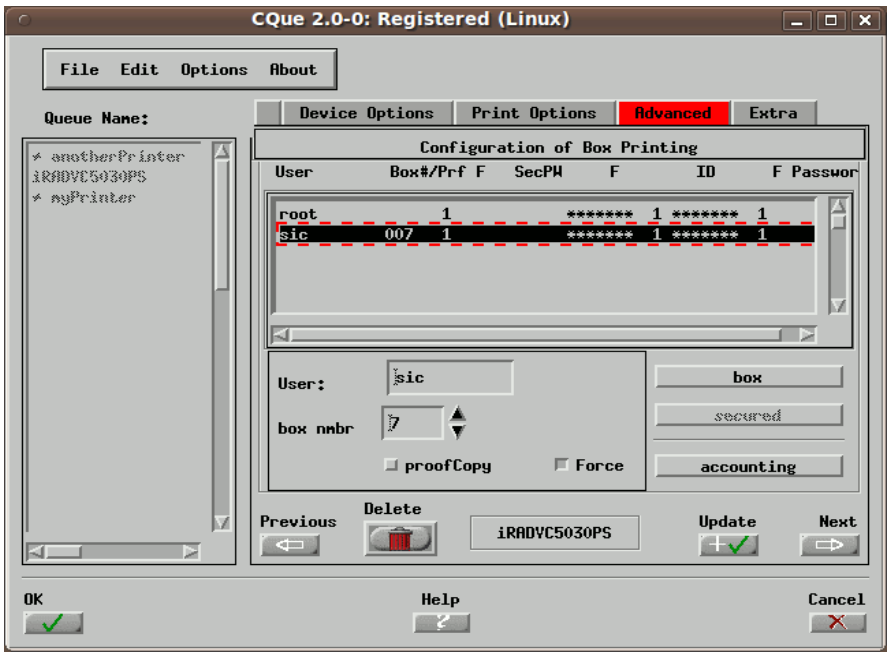


Figure 44 Advanced Options: Selecting an entry to delete

The selected user name, in our example 'sic', will display and if you are really sure, you may click the *OK* button to activate the deletion.

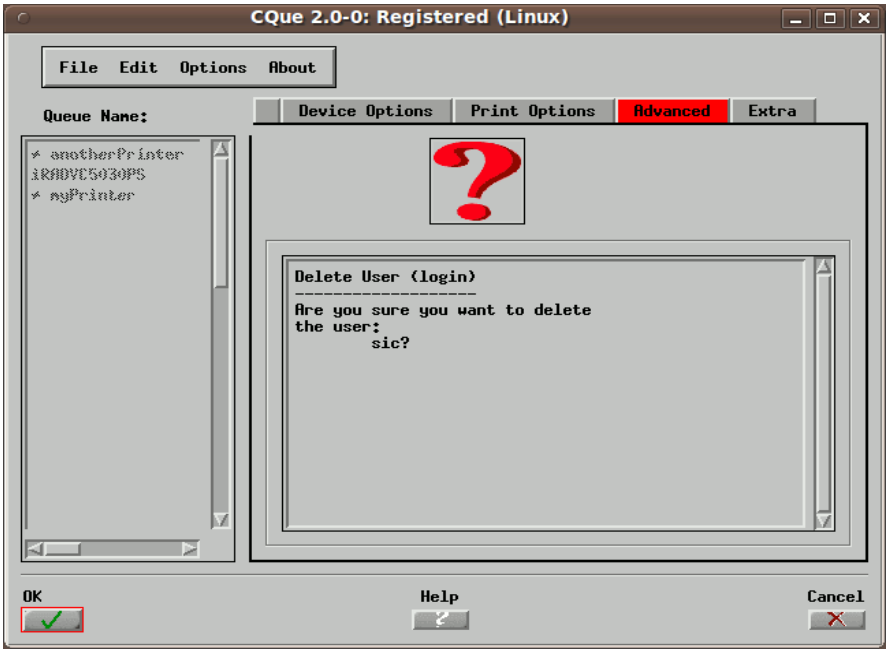


Figure 45 Advanced Options: Verification before deleting the entry

3.4.10 Extra & Banner Options

The *Extra tab* contains first two entries for the option of a *banner page*: A banner page before and/or after the job are possible.

The possible types of banners, before and or after the document, supported and generated by CUPS are displayed in fig. 47.
Please note that these banner pages are generated by CUPS and for the printer manager may just be another job not necessarily with the same finishing options as specified for the main document.

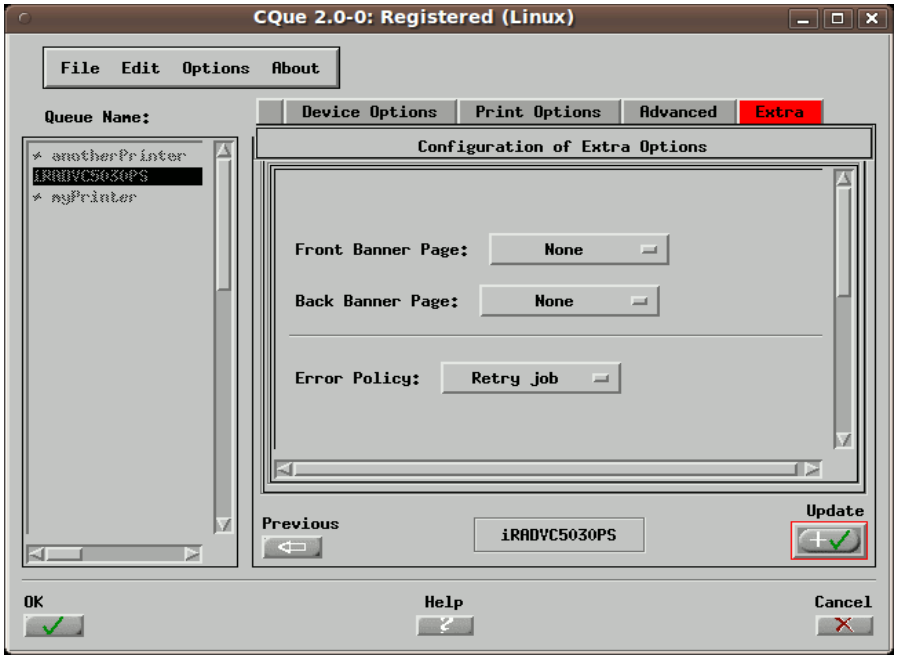


Figure 46 Extra and Banner Options

Also the *Error Policy* of the queue can be modified here. The possible error policies are displayed in fig. 48.

Note that if *"Stop the queue"* is selected, then in case of a problem, like for example a paper jam, the queue will indeed be disabled (stopped) and needs to be enabled manually after the original problem has been solved.

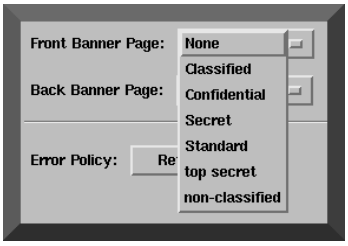


Figure 47 Banner Type Options

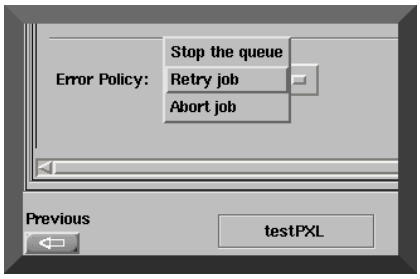


Figure 48 Error Policy Options

If *"Retry job"* is selected then the job will continue to retry, possibly with some time out between different tries.
If *"Abort job"* is selected the job will simply be aborted in case of a problem.

3.4.11 Printing Multiple Copies of a Document with Stapling

When printing multiple copies (say N copies) of a document with stapling most CUPS implementations will concatenate the output N times and then staple the resulting concatenated file. To prevent this CQue 2.0-2 PPDs support a feature: ***Repeat Job*** within the ***Canon Specific Options*** section. This allows you to specify from 1 to 99 copies for applications with the GTK+ print interface and 1 to 25 otherwise, with proper stapling of each copy. This feature is supported for PostScript, PCL5 and PCL6 Canon devices allowing PJL headers, i.e. some pure PostScript EFI devices are excluded.
Note: N copies of a document may generate N separate entries in the log file of the Canon device.

4 Non-standard Installations of CQue 2.0

4.1 Upgrade Installation of CQue 1.0 to CQue 2.0

If a version of CQue is already installed on your Linux workstation the setup program will detect that. If you would like to upgrade any existing CQue printer queues to the new version CQue 2.0 automatically, you may chose any of the three installation possibilities outlined in section 3.2. By default setup will upgrade any CQue printer queue with the new PPD files. The setup program will never touch any non CQue printer queues.

If you would like to upgrade CQue, but not upgrade any existing CQue printer queues, that is possible only by using the .tgz package installation, as outlined in section 3.2.2.

When upgrading from CQue 1.0 to CQue 2.0 the setup program makes a backup of various PPD files:

- All system PPD files are saved in to a file /var/tmp/CQuePPDFiles2.0-0.tar.
- Each PPD file corresponding to a printer queue (in /etc/cups/ppd) is saved into a file in the directory /etc/cups with the extension .rpmsave.

A review of the upgrade installation will be available in the file */var/log/CQue2.0_UpdateLog*.

4.2 Upgrade Installation of SQue 9.X to CQue 2.0

If you would like to upgrade SQue 9.X to CQue 2.0 this has to be done in two steps:

1. First upgrade SQue 9.X to CQue 1.0. This process will upgrade any SQue printer queue to a CQue printer queue. Section 4.3 tells you how to do this. Please note that section 4.3 is taken from the CQue 1.0 manual, and thus makes reference to the setup of CQue 1.0!
2. Then upgrade CQue 1.X to CQue 2.0. This process will upgrade any CQue printer queue to the new CQue 2.0 printer queues. This is outlined in section 3.2 of this manual.

4.3 Upgrade Installation of SQue 9.X to CQue 1.0

For an upgrade from SQue to CQue the setup program may convert the SQue type of printer queues to CQue type of queues.
It may convert the queues depending on the printer language of the queue as follows:

SQue queue type	Recommended CQue queue type	Comment
PostScript	PostScript	Behaviour of SQue and CQue rather similar for PostScript processing.
PCL	PCL	CQue does not support overlay printing. the PCL interface of SQue is not supported.
CPCA	PostScript if supported by the Canon device, otherwise PCL.	CQue currently does not support CPCA. Transformation of PDLs is supported better by CQue, therefore the choice primarily depends on the PDL of the Canon device.

The following window will display:

After you made your choice, select OK.
This will upgrade CQue and eventually the same windows as in Figs. 6 and 7 of the ***CQue 1.0 Rerefence Manual*** will display.

Depending on the possibilities of the printer as known to CQue you may for each SQue queue:

- ✓ convert to a PostScript queue
- ✓ convert to a PCL queue
- ✓ convert to a PXL queue
- ✓ skip this queue
- ✓ delete this queue

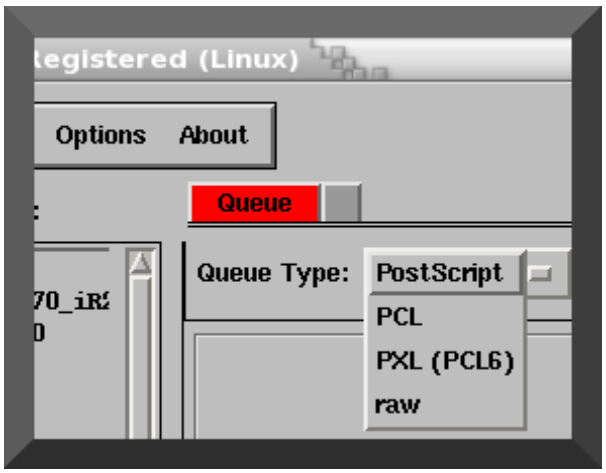


Figure 49 Keep or Destroy SQue Queues

If you decided to "Keep queues", then for each SQue queue for which an equivalent CQue queue can be created, the following window will display during the upgrade procedure:

If CQue would not support one or more of these choices, the corresponding items will be greyed, i.e. not selectable.

SQue does not know of PXL queues, so that choice would always be new. If you select PXL, please be sure the Canon device has the necessary PCL board.

For each converted queue the setup program will try to make device and print options of the old queue to correspond with the new queue. In some cases checking of the result is recommended.

If you select to skip an SQue queue, it will remain as an existing queue managed by SQue.

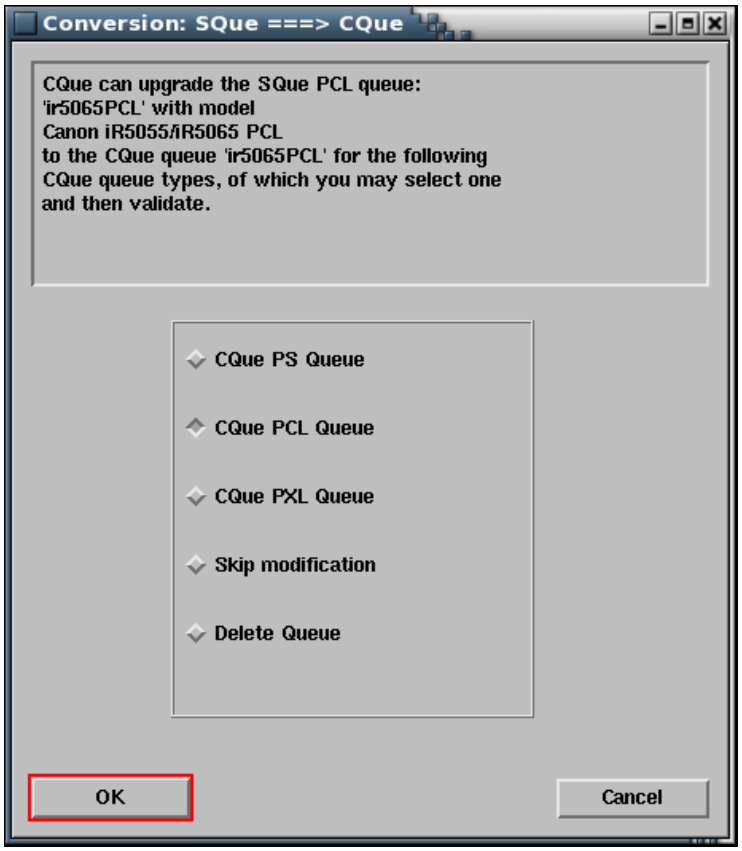


Figure 50 Specify SQue to CQue conversion

If you select to delete the SQue queue, then the queue will be deleted and not converted to a CQue queue.
A review of the upgrade installation will be available in the file */var/log/CQue1.0_UpdateLog*. In the case of conversion of queues a detailed report of each queue that has been converted will be found in this file.

Please note: the setup program will not delete SQue itself. It will only possibly convert SQue queues to CQue equivalents.
If the ***setup*** program encounters both an SQue installation and a CQue installation, it will only upgrade the CQue installation. In other words: The transformation from SQue to CQue can be done only once.

4.4 De-Installation of CQue 2.0

If you want to de-install an existing version of CQue 2.0 from your Linux workstation, you may do so with a system tool if you installed CQue as an rpm or deb package. Such a de-installation, however, will not remove any CQue printer queues. This means that these printer queues most probably will not function correctly any more. A CQue printer queues relies on the existence of the filter ***sicgsfilter***. If that program is removed because of a de-installation, the queues will not work properly, and in the best case continue to print in A4 only.

Alternatively you may runs ***setup*** from a terminal window with the options -d:

```
/opt/cel/bin/setup  -d
```

The program will ask for confirmation and then delete any CQue queues, as well as (most of) the CQue programs and tools. As you run ***setup***, it is probable that the Linux system will prevent to delete the program setup itself! This means that eventually you would have to remove the program /opt/cel/bin/setp and possibly the directory /opt/cel/bin.

A review of the actions taken can be found in */var/log/CQue2.0_UpdateLog*.

5 Miscellaneous Topics of CQue 2.0

5.1 Blob or Balloon Help of CQue 2.0

The program CQue 2.0 has help mechanisms at various levels. The main Help, which consists of selecting the "Help" button, and then selecting the object about which help is required, is already explained in the Introduction.

There is also a so-called "blob help", sometimes called "balloon help" too: When you point the mouse cursor to a button or other element of the CQue graphic interface, a small window pops up with a short description of that element. The window remains for a short time and disappears also when you move the cursor away from the element.

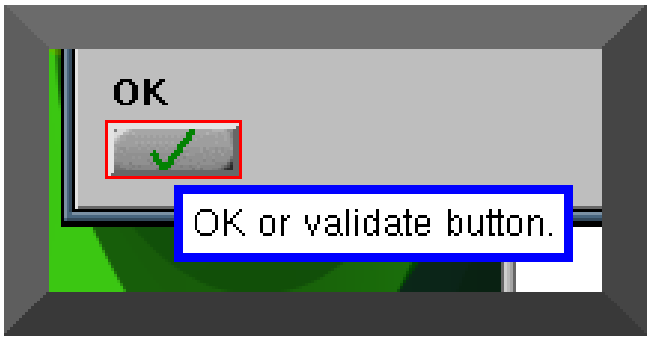


Figure 51 Blob Help

The figure 51 shows an example of the blob help for the *OK* button.

Sometimes this may have a side affect: When you would have different windows of CQue open at the same time and these windows would overlap, then the blob help mechanism tends to systematically put one of the windows in front of all the others.

In such cases you may wish to turn off the blob help mechanism. On the main menu of CQue select the *"Options"* button and there you may toggle on or off the blob help.

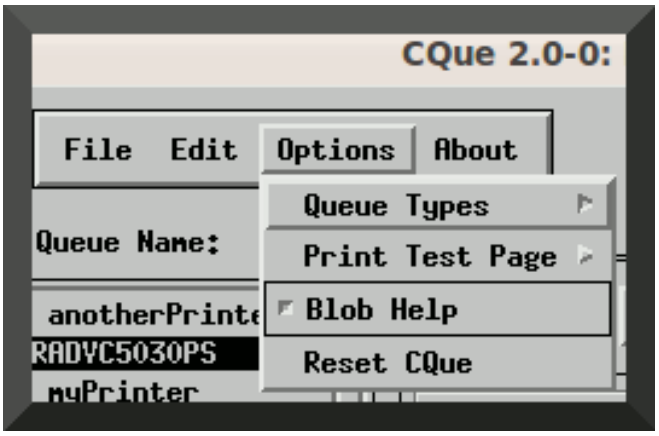


Figure 52 Turning on or off Blob Help

5.2 Filtering the list of Queues of CQue 2.0

On the most of the windows of CQue 2.0 a list of queues is visible with, as explained in the Introduction, two type of queues: with an asterisks for system or non-CQue type of queues and without for CQue type of queues.

If this list is very long you may wish to show only one type of queue. To select which type of queue will be displayed, you select *"Options"* on the main window of CQue. Then select the sub-menu *"Queue Types"* and there make your choice:

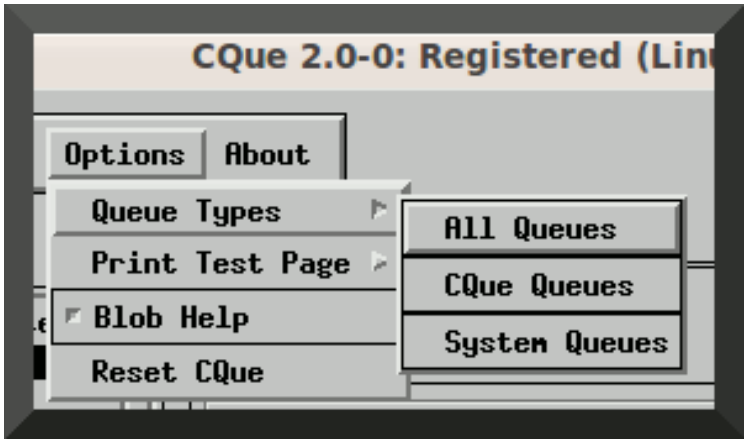


Figure 53 Selecting which Type of Queues will display in the List

All Queues	Displays all queues found on the workstation, with the asteriks for non-CQue queues.
CQue Queues	Displays only CQue queues.
System Queues	Displays only system or non-CQue type of queues.

Be careful: Currently CQue only shows which filter is active (or not) under the "Options" menu of the main window. So if you wouldn't find a queue in the list you were expecting to be there, please verify which filter is set.

5.3 Printing a Test Page with CQue 2.0

Once you have created a queue with CQue 2.0 you might want to test it. CQue has the following possibilities to print one or more test pages.

From the main menu of CQue select the *"Options"* menu. There select the *"Print Test Page"* sub-menu, which shows the following possibilities:

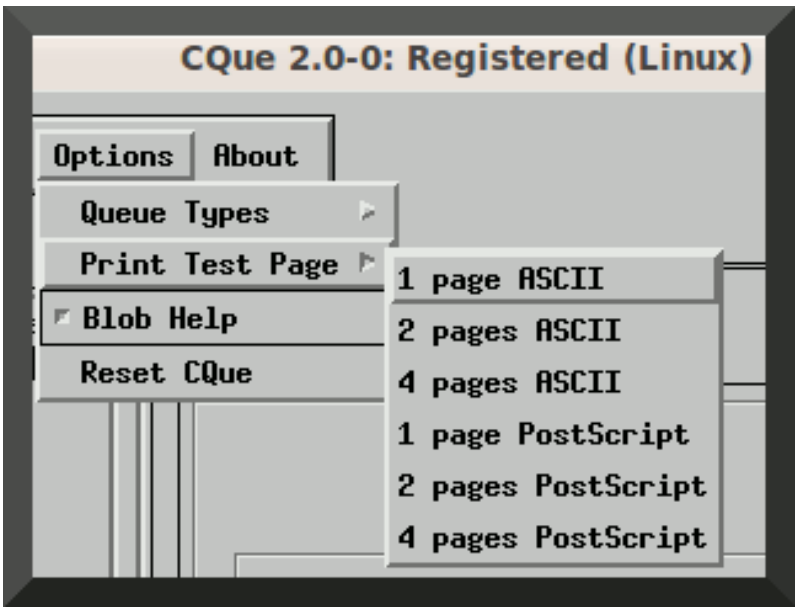


Figure 54 Printing a Test Page

1 page ASCII	Prints a simple test page with only ASCII text.
2 pages ASCII	Prints twice the same simple test page with only ASCII text.
4 pages ASCII	Prints four times the simple test page with only ASCII text.
1 page PostScript	Prints a PostScript test page, possibly in colour, with information about the imageable area and the resolution of the printer.
2 pages PostScript	Prints twice the same PostScript test page.
4 pages PostScript	Prints four times the same PostScript test page.

You may send a test page to any printer, either of CQue type or not. Once the job is sent, CQue displays a message asking whether the page(s) printed correctly.

The option to print more pages at once is included to test duplex and/or stapling of the Canon device.

5.5 Reset of CQue 2.0

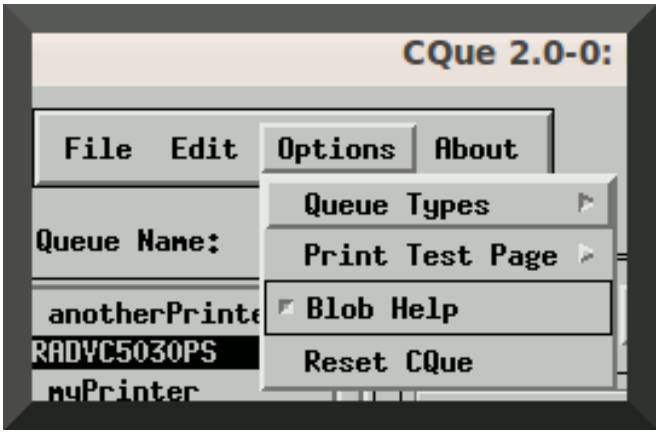


Figure 55 Reset of CQue

If, for some reason or another, you want to reset all options at once, select the *"Options"* menu from the main window of CQue. There select *"Reset CQue"*. This may be useful if you are creating various printer queues in one session of CQue. CQue will retain some of the data, for example the printer options. If you change the model from one queue to another, then this may generate conflicting settings because of differing constraints of the PPD file. In such cases it may be helpful to reset CQue completely.

On the other hand, if you were to create various printer queues for the same model (same PPD) then it may be advisable to not reset CQue each time you create a new queue.

If creating both PostScript and PCL printer queues for the same printer, then we do advise to reset CQue between the creation of the different queues.

5.6 Printing to a File with CQue 2.0

If you were to print to a file the CQue Port interface does provide the necessary options to do that, as described in section 3.4.4.6.
The CUPS printer manager, however, may not agree with such a strategy. By default CUPS does not allow to print to data files on the file system.
To modify this you must edit the file: ***/etc/cups/cupsd.conf*** (we recommend to make a backup of the file first).

Add a line with:

```
FileDevice Yes
```

and then restart thee CUPS printer manager so that the modified configuration file will be taken into account.
Restarting CUPS can be done by:

```
service cups  restart
```

or:

```
/etc/init.d/cups restart
```

or

```
/etc/init.d/cupsys restart
```

depending on the type of Linux system you are using.

5.7 About CQue

When you click on the ***About*** button of the main window (Fig. 17) the following window will open, showing some general information on the origin of CQue.
For support you are invited to contact the technical division of your local Canon company.

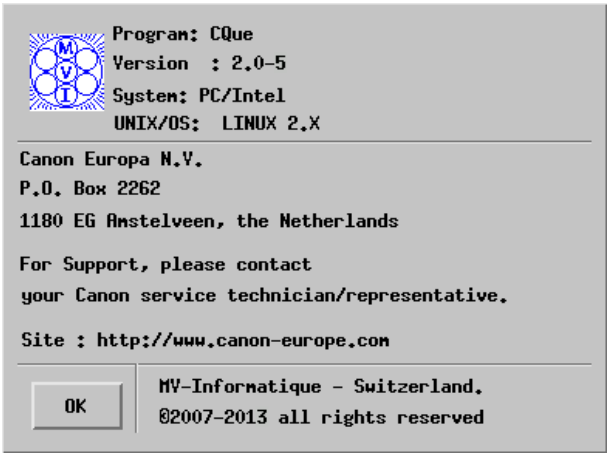


Figure 56 About window of CQue

6 What to do with Problems of CQue 2.0

6.1 Introduction

If you installed a version of CQue 2.0 on your Linux workstation, and things are not functioning as you expected, this chapter tries to outline some debugging strategies.
Section 6.2 discusses problems related to the CQue interface itself.
Section 6.3 discusses problems related to printer queues in a first approach.
Section 6.4 discusses debugging in a more detailed manner.

6.2 After Installation of CQue 2.0 the Program Does Not Start

If you installed CQue with an RPM or DEB package the program /opt/cel/bin/setup will be executed automatically.
Once the setup program run, the final CQue program is /opt/cel/bin/cque.
And a link to that location should be made in /usr/bin.
If you installed CQue on your workstation and CQue would not start, please first check the file

/var/log/CQue2.0_UpdateLog

This file contains a trace of the complete installation and normally will give you an indication of possible problems.

You may also try to start CQue from a terminal. First open a terminal session. Then go to the directory /opt/cel/bin.
From there try the command:

./cque

To run CQue from a terminal, though, the value of the environment variable DISPLAY should be set! To verify, type

```
echo  $DISPLAY
```

If there is no answer, or the answer is empty, try setting the default value:

```
DISPLAY=:0.0
export DISPLAY
```

(We suppose you are not running in a csh, for which you would have to use the *setenv* command: setenv DISPLAY=:0.0).



And retry the ***./cque*** command.

If you opened a terminal window as root in a non-root desktop, you may have to execute from yet another terminal winod, i.e. as the owner of the dekstop:

```
xhost  +
```

This will allow others users than the owner to open a window on the desktop.

If the CQue interface would display in outline, but text fields are invisible, this is probably due to a lack of colours of the X11 server.
Running ***cque*** in a terminal would display an error message complaining about the allocation of colours.
To solve this problem try running ***cque*** with the option -a2. This will essentially run CQue in black/white mode. If this solves the problem, you may edit the ***cque*** script and include the -a2 option in the last line, which starts ***CQue.exe***. (see section 9.6 for a description of the script ***cque***).

To activate ***cque*** with full debug options you may run:

```
/opt/cel/bin/cque  -v  -Z2
```




6.3 A Created Queue Does Not Print

If you created a queue and for one reason or the other it does not print, perhaps the following suggestions may help. Some topics seem rather obvious, though experience learned that sometimes they are enough to solve the problem...

1. Check that the Canon device (and possibly the printer controller) are both powered up and connected with the appropriate cables. Of course there should be paper or media in to the paper trays.
2. To check the connection with the Canon device do a *ping* on the system. I.e. from a Linux terminal window type:

   ```
   ping  pName
   ```

   We suppose here that the Canon device is known by the network as *pName*.
   The system should respond with something like:

   ```
   pName is alive
   ```

   or it could return an infinite loop of sequences reported to be send and received. (You may stop this by *Ctrl-C* or the *break* key of the keyboard).
   If the system does not respond like that, there is a network connectivity problem, which must be resolved first. *There is no point in doing any other tests, before the connection with the printer is established.*
   Check the cables, check also the IP address of the Canon device to be the same as entered by the installation. The file */etc/hosts* contains the IP address entry on the Linux system, in the most simple case. If your system uses DNS or NIS check the main databases. To get the IP address of the Canon device you might print the configuration page.

3. Check that the printer manager is running. On Linux systems the command

   ```
   lpstat  -r
   ```

   should respond with

   ```
   scheduler is running.
   ```

   If it is not running for one reason or the other you may restart it by the command

   ```
   /etc/init.d/cups  start          (or possibly:  /etc/init.d/cupsys)
   ```

4. Send a test print to the printer by hand. This could be done with CQue itself (see section 5.3) but it could also be done by hand. For example the command

```
lp  -dqName  /etc/hosts
```

should print a short text file on the printer.
For CUPS you may check the file

/var/log/cups/error_log

The first file is created if you print as *root*, the second if you print as user *sic*.
If no errors occurred these files will be empty, but one or both of them will exist as soon as a print cycle through the queue qName passed. If the log-file does not exist, this means that there is a problem with the qName logical printer queue. If the log file is not empty please verify its contents, which may give a clue to the problem.

5. You may also verify the details of the created printer queue in the CUPS printer data base: /etc/cups/printers.conf.
For the printer queue qName you would see something like:

```
<Printer qName>
Info CQue 2.0 LPD Queue
DeviceURI lpd://192.168.1.152/lp
State Idle
StateTime 1199811929
Accepting Yes
Shared Yes
JobSheets none none
QuotaPeriod 0
PageLimit 0
KLimit 0
OpPolicy default
ErrorPolicy retry-job
</Printer>
```

Finally the general command:

```
lpstat –t
```

should give an answer like:

```
scheduler is running
device for qName: lpd://192.168.1.152/lp
printer qName is idle. enabled since Mon Oct 30 22:21:21 WET 2010. available.
printer remPrinter is idle. enabled since Mon Oct 30 22:21:22 WET 2010. available.
```




7. If you activated file or mail box support check that the document has the correct destination. Sometimes it happens that printing to a certain mailbox is applied, and thus no output to the copier will appear. You will see the communication indicator of the copier flicker, but no printing will occur. In such cases verify that no mailbox is activated.
8. You may also use the print manager of the Linux operating system. This tool, however, may be used to delete printers or to (re-)create *raw* printers, but it should *not* be used to (re-)create the *qName* CQue printer because the Linux print manager is not capable of handling this kind of printer queues. You may modify printer options though making use of the Linux print manager for a CQue printer queue.
9. It may also help to have a look at possible mail that might be send by the printer managers if a problem arises.

The next section describes a more detailed approach activating some of the debugging features of CQue.

6.4 Introduction to Debugging an Installation of CQue

Chapter 7 details a number of command line options of most of the programs of CQue. If you have problems with ***cque*** (or CQue.exe) itself, the program ***cque*** supports the option -Zx to set debugging.
The value if *x* has the following meaning:

x	Description	Remarks
0	Normal usage. No particular debug mode.	This is the default value.
1	Will display messages of system commands etc. but will not execute them.	This is useful if you do not want to create a queue, but simply want to see what CQue intends to do.
2	Will display messages of system commands etc. but will execute them too.	This is the most appropriate method for debugging: CQue displays what it wants to do and executes too.

To catch the messages issued by ***cque*** you would have to run ***cque*** from a terminal.

If a queue does not print as expected and the suggestions of the previous sections did not help, you may activate debug flags too.
The most important program is the ***sicgsfilter*** which will be started in a script run by the CUPS printer scheduler. To see how this program is run by CUPS you need to activate the debug mode of CUPS. This can be done either by the HTML interface at localhost:631 (port 631) and then under the administrator window activate debug mode. Or you may edit the file ***/etc/cups/cupsd.conf***.
Search for the line:

LogLevel info

and modify it to

LogLevel Debug

After editing the file ***cupsd.conf*** you must restart the CUPS printer scheduler:

services cups restart
or ***/etc/init.d/cups restart***
or ***/etc/init.d/cupsys restart***

The useful place to look for log files is the directory /var/log/cups.
There are three types of files:

access_log: reporting all access events of all CUPS printers.
page_log: reporting all pages printed by CUPS printers.
error_log: reporting a full history of each print process of a CUPS printer. This file is by far the most important one.

Of course the file *error_log* will be of most interest.

7 Program Description of CQue 2.0

7.1 Introduction

This chapter describes some of the programs or modules of CQue 2.0. For the occasional reader this chapter will be far too technical, but for a Linux system administrator it may contain valuable information. It will mainly explain the command line options of some of the modules of CQue. Most of these modules will be of little use if used as stand alone programs. But for debugging purposes, for example, it may be useful to know the meaning of the various command line options.

7.2 The Program CQue.exe

The program ***CQue.exe*** is the main interface of the package. It should never be run without the correct environment variables associated with it. Section 8.2 gives details on the script ***cque***, which is normally used to run ***CQue.exe***.
The command line options known by the program ***CQue.exe*** include the following:

-ax : *colour* mode:
- 0: use system default colours;
- 1: limited number of colours, with grey background; [*default*]
- 2: use black and white exclusively;
- 3: use all colours in true colour mode;

If on your display only the outlines of CQue would be visible but no text fields, try using the option -a2, which switches to the most basic black/white version of the CQue colour scheme.

-h : print a *help* message.

-rx : define how the *previous session* will be *restored*:
- 0: no restoring of the previous session.
- 1: restore the previous session at start up, [*default*]
- 2: re-read the PPD/PDD file without restoring;

-sx : define the name of the *main configuration file*. By default the name of this file is: /opt/cel/cqueInfo2.0. This name may also be specified with the environment variable CQUEINFO.

-ux : define the *working directory*. By default : /opt/cel. Normally this directory may also be redefined by the configuration file cqueInfo2.0.



-v : define *verbose* mode [*false*];

-xx : set the width of the window of CQue [*640*];

-yy : set the height of the window of CQue [*480*];

-Hx : define users *home directory* [*$HOME*];

-Mxy : defines the *mailbox mode*.
x = 0: no mailbox support. [default]
x = C: mailboxes of type Canon, also EFI ;
y = the number of available mailboxes (default : 100).

-Ux : define the user name. By default CQue will detect itself the name of the user who started CQue. Normally CQue would be run by the root only.

-Wx : define the *working directory*. By default : /opt/cel. Normally this directory may also be redefined by the configuration file cqueInfo2.0.

-Zx : *debug mode*:
0: normal mode. [*default*]
1: debug mode: Don't execute commands, but echo them to the console only.
2: debug mode: Echo the commands and execute them at the same time.

Normally the script ***cque*** will take care of the most important options through environment variables.
The environment variables of ***CQue.exe*** include:

CQUEINFO :	The main configuration file (by default: /opt/cel/cqueInfo2.0).
PREPATH:	The working directory of CQue.exe.
PREFIX:	Identical to PREPATH.
PPDDIR:	The directory of the PPD files, i.e. the printer model files.
CQUECUPSDIR:	The directory root of the CUPS distribution.
CQUECUPSBINDIR:	The directory 'bin' of the CUPS distribution.
XKEYSYMDB:	If XKEYSYMDB is present it may overrule the default keyboard mapping. On some UNIX systems, particularly Linux, the standard XKeysymDB file is located in another than the standard location: /usr/lib/X11.
XENVIRONMENT :	If XENVIRONEMNT is present it may overrule some of the X11 features of CQue.exe. Default all graphic interface widgets and components will have reasonable values.

7.3 The Program sicgsfilter

The program or printer driver *sicgsfilter* - the main working horse during printing next to foomatic - supports a number of options, which are simply listed without much technical explication. The following options are supported:

-ax : set booklet mode to x.
-bx : set back cover mode to x.
-dx : set duplex to x; (0=simplex,1=duplexLong,2=duplexShort) [0].
-ex : set line refine mode to x.
-fx : set front cover mode to x.
-lx : set binding location to x.
-mx : set media type to x.
-px : set paper size to x.
-qx : set staple mode to x.
-rx : set long/short offset registration [1].
-r xfl,yfl[,xbl,ybl][,xfs,yfs[,xbs,ybs]] : set fixed PCL offsets (see below).
-sx : set sorter mode to x.
-tx : set input tray to x.
-ux : set user to x.
-Ax : set destination to x (0=prt,1=mailbx,2=mailb+proof,3=Secure,4=Form) [0].
-Bxy: set mailbox to y,x='B':iR, x='N':EFI, [-1].
-Cx : set back cover source to x [0].
-Dx : set debug flag to x [0].
0: standard mode; [*default*]
1: debug mode without actions;
2: debug mode with actions;
-Fx : set front cover source to x [0].
-Gx : set punch mode to x [0].
-Hx : set halftone mode to x [0].
-Kx : set colour mode to x [0].
-Lx : set slip sheet mode to x [0].
-Mx : set work mode to x (MPS=PS,MPCL=PCL5,MPXL=PXL) [PS].
-Nx : set flag to keep PaperSize (x=P), or paper tray (x=T).
-Ox : set output bin x [0].
-Px : set PCL level to x [5].
-Qx : set print density to x.
-Rx : set resolution to x [600].
-SPW=x : set secured print password to "x".
-Tx : set toner reduction to x [0].
-UID=x : set account user id to "x".
-UPW=x : set account print password to "x".
-Wx : set wideA4 mode to x [0].

The option –M (for the printer description language) and –P (for the name of the printer queue) are the most important ones and are almost required options.

The -r option allows for fixed PCL offsets. The unit of dimension is decipoint, which means 1/720". The meaning of the variables is:
xfl,yfl = (x,y) offset for front page (simplex or long edge duplex).
xbl,ybl = (x,y) offset for back page (long edge duplex).
xfs,yfs = (x,y) offset for front page (simplex or short edge duplex).
xbs,ybs = (x,y) offset for back page (short edge duplex).
Appendix C gives some more details on applying these options.

The program ***sicgsfilter*** supports the following environment variables:

SICPSBINDIR :	The directory where binary files related to PostScript processing would be located. For example: CQue 2.0 will try to transform PDF files with the program *pdf2ps*. It will search for *pdf2ps* in a number of default directories, but you may force the location by SICPSBINDIR. (this option is not used anymore)
CUPS_SERVER:	The CUPS server on which the document is printed.
DEVICE_URI:	The URI of the Canon device or output device of the queue.
PPD:	The name of the PPD file used for the printing of the document.
PRINTER:	The name of the printer queue itself.
SERVER_ADMIN:	The CUPS administrator.
TITLE:	The title of the document to print.
CQUEDATADIR:	The directory for the data of CQue [/opt/cel].
USER	Name of the user of the print job.

7.4 The Program setup

The program *setup* is the *installation program* for ***CQue.exe***. It supports the following options:

-fx : Allow installation from any directory. (x=0: no, 1:yes). Default: yes. It makes a slight difference for setup up to know whether it is launched from the default directory /opt/cel/bin or not. The rpm and deb package installations launch setup with the option -f0 to force installation from /opt/cel/bin.

-d : Run setup in de-installation mode. In this mode, if an installation of CQue is detected, setup will ask for permission to de-install CQue (including any CQue printer queues). If confirmed it will proceed with the de-installation.

-qx : Query the steps of setup:
- x = 0 : No, installation proceeds without questions [default].
- 1 : Ask permission to proceed and to upgrade each CQue printer queue if found.

-Sx : Set SELinux mode (x=0:ignore SELinux, 1:process SELinux) [1].

-Dx : Set debug mode:
- x = 0 : Normal processing without debug information
- 1 : Send debug information to *stderr*, but do not execute commands.
- 2 : Send debug information to *stderr*, and do execute commands normally.
- >2: May even output more debug information, while executing normally.

This program must be run as ***root*** (or super-user).
A log file with a review of the installation or update is written in:
/var/log/CQue2.0_UpdateLog.

7.5 CQue and SELinux

The program *setup,* the installation program for ***CQue.exe,*** will detect whether the Linux distribution is running SELinux in enforced mode. If this is the case, setup will install a CQue SELinux policy module, ***cque.pp***. This will prevent any SELinux warning messages during setup process.

The ***cque.pp*** module, however, is not necessary for the correct functioning of CQue.exe and one of it components. If, for whatever reason, you might prefer to de-activate the CQue SELinux policy, you may do so by the following command:

```
semodule  -r  cque
```

7.6 The program sicnc

The program *sicnc* is a home made version of the popular *netcat* or *nc* program shipped with most Linux distributions. It is used for various TCP/IP work internal to CQue. This version has the property that, when listening to a TCP/IP port, it stops doing so when an end of file is encountered. Some versions of nc would continue listening indefinitely.



8 Configuration Files and Scripts of CQue 2.0

8.1 Introduction

This chapter describes some of the configuration file of CQue 2.0. For the occasional reader this chapter will be far too technical, but for a Linux system administrator if may contain valuable information.
All of the configuration files of CQue are in readable ASCII format.

In this chapter the term *advanced options* of CQue will refer to either file box printing, printing with department accounting or secured printing, or any combination thereof.

8.2 Directory Layout of CQue 2.0

The program CQue is installed in a directory, which is fixed during setup of CQue.
The so-called working directory is: ***/opt/cel***.
For each printer that is created by CQue, and which is of CQue type, a subdirectory of ***/opt/cel*** will be created, with the same name as that of the printer queue. Within that directory there is always another subdirectory called *root*.
For a particular printer we, of which the configuration files will be located in a subdirectory of /opt/cel, we will denote that directory by $PRINTERDIR.

During setup there will be the following directories created as sub-directory of /opt/cel:

- ***bin***: directory with binary programs, and executable scripts, basically: ***CQue.exe, cque, sicgsfilter, setup, sicnc***.
- ***doc***: directory with documentation.
- ***etc***: directory for the CQue configuration files.
- ***ppd***: directory for the printer model files or PPD files for PostScript, PCL and PXL.

8.3 File Box and Accounting Configuration File

If advanced printing options are applied then CQue will have created a file called ***CQueMailBoxFile*** into the $PRINTERDIR. This file associates user names or login names on the Linux level with mail or file boxes, user IDs or passwords on the copier level.

The syntax of this file is as follows:

1. Each line corresponds with the data of one user or login.
2. The first entry of a line corresponds with the login or user name.
3. The login name is followed by any of the following sequences or a combination thereof:

Start sequence	Followed by	Followed by	Meaning
m=	S followed by X=number between -1 and 99	P	Stores into file box X. If P is present, then prints too (proofcopy). If S is present value is 'saved'
u=	S	X=number of max. 7 digits	User ID for department accounting. If S is present value is 'saved'
s=	S	X=number of max. 7 digits	User password in combination with user ID for department accounting. If S is present value is 'saved'
p=	S	X=number of max. 7 digits	User password for secured printing. If S is present value is 'saved'

4. The character S is optional. The character P for the m-sequence is optional too.
5. No comment lines are allowed.

An example:

```
mv      m=S2P      u=S2800    s=S2800
gz      m=3        u=2802     s=2804
sic     m=S49
```

The first line indicates that for the user or login 'mv' each job will be stored into the file box number 2 and a proofcopy will be printed too. (m=S2P). This user will be automatically accounted for with user ID = 2800 (u=S2800) and password = 2800 (s=S2800). Note that the values of the mailbox, user ID and password are 'saved' (all have the character 'S' before the digits). This means that the mailbox, user ID and password will be taken into

account automatically and the user does not have to enter them when submitting a job.

The second line indicates that for the user or login 'gz' each job will be stored into the file box number 3, but no proofcopy will be printed. (m=3). This user will be accounted for with user ID = 2802 (u=2802) and password = 2804 (s=S2804). Note that the values of the user ID and password are not 'saved' in this case. This means that the user ID and password may be modified by means of a "Properties" window before submitting a job. It the user does not enter the correct values, the job will not be printed. If no values are specified, then the values specified here will be taken into account automatically.

The third line indicates that for the user sic each job will be stored into the file box number 49. There will be no proof copy nor will there be department accounting for this user.

Please note that you may edit this file by hand. It will possibly be re-edited by CQue as soon as you run CQue, but no information will be lost.

For the sake of completeness we repeat here the tables displaying the possible combinations allowed from section 3.4.9:

Combination	**Supported**
Mailbox + Accounting	Yes
Mailbox + Secured Printing	No
Accounting + Secured Printing	Yes

If, for example, you were to include all sequences, (m, p, u and s) this is *not* allowed and the result will be undefined.

We also remind that the root has a special meaning:

	root	***normal user***
Filebox	Each job of a normal user not in the list will go to the file box.	Each job of this user will go to the file box.
Secured printing	Each job of a normal user not in the list will use the specified password.	Each job of this user will use the specified password.
Accounting printing	Each job of a normal user not in the list will use the specified user ID and password.	Each job of this user will use the specified user ID and password.

An example with the root entry:

```
root        m=S2          u=S2800     s=S2800
gz          m=S3           u=2802      s=2804
sic         m=S49
```

In this case all users would print, rather store, into file box number 2, *except for the users* 'gz' and 'sic', who would store into file box numbers 3 and 49 respectively.
This is a convenient way of constructing a queue which print or stores to a file box for all users.

8.4 Adding a Printer Model to CQue

When running CQue the Menu for the selection of the Printer Model shows a list of available Canon devices, depending slightly on the printer description language (PDL). It may occur that from time to time new models become available and/or old models become obsolete. To add a printer model to the data base of CQue consists simply of adding one or two files.

The name of a CQue PPD file is as follows:

cel-<Model>-<PDL>-<COUNTRY>.ppd

where:

<Model>	= the Canon model name, for example iR4580
<PDL>	= the printer description language: ps = PostScript, pcl = PCL or pxl=PXL.
<COUNTRY>	= the language code: en=English, fr=French, de=Geman, it=Italian, es=Spanish.

For example: ***cel-iR4580-ps-en.ppd*** is a PPD for an iR4580 PostScript Canon device in the English language.
And ***cel-mf7170i-pxl-fr.ppd*** is a PPD for a MF7170i PXL Canon device in the French language.
This naming convention is imperative for CQue. In other words otherwise valid PPD files not adhering to this naming convention will not be recognised by CQue!

For PostScript Canon devices you could use the PPD files from MS Windows. We do not, however, recommend using these files because the advanced features (mailbox, department account, secured printing) are not supported by these files.

Thus we recommend using CQue compatible PPD files. CQue will be upgraded regularly to include any new Canon devices when available.
You might also check the web site of M.V.I. www.mv-informatique.com, which will be updated regularly to make the PPD files for the latest Canon devices available in different languages.

To add support for a new Canon device, first get the PPD files for that device.
Then copy these files to the appropriate PPD directory. This sounds easier than it may be for certain Linux distributions. Despite the standard of LSB (Linux Standard Base) which is proposed, only very few current Linux distributions do adhere to this standard.
At installation time setup figures out where the proper system PPD directory is located. This is reported in the CQue log file ***/var/log/CQue2.O_UpdateLog***.
The table on the next page shows some known locations where you may look for.

Please note: The directory ***/etc/cups/ppd*** is ***not*** the right place to put the PPD file! This is a system directory where PPD files are copied after the queue has been created.
The correct PPD directory should already contain a series of files with the above described naming convention.

The following table shows some potential directories where PPD files could be located:

PPD directory	Comment / Linux system
/usr/share/foomatic/db/source/PPD	Fedora, Mandriva
/usr/share/ppd	Recommended LSB
/usr/share/cups/model	CUPS standard directory
/opt/share/ppd	Debian
/usr/share/cups/model/custom	UBUNTU
/usr/share/cups/model/foomatic-db-ppds	Several Linux

If no directory could be found, you might use /usr/share/cups/model as a starting point and find any subdirectories or symbolic links from there. For most current Linux distributions the /usr/share/cups/model directory is the basis for PPD files. On more recent Linux distributions we would expect /usr/share/ppd to be the principal location to look for, as this is the LSB recommended directory.
Finally you may have a look at the script ***cque*** and try to locate the environment variable PPDDIR therein. This is the directory ***CQue.exe*** will take as a starting point.

8.5 The Script cque

The general script ***cque*** which runs ***CQue.exe*** would look like:

```
#!/bin/sh
#
# Start up script for CQue 2.0. with environment variables.
# 1999-2010 MV-Informatique.
# all rights reserved.
# For system colours use option -ax:
#       x=0 use system colours
#       x=1 use grey colours (default).
#       x=2 use black and white only (basic version).
#       x=3 use full colours.
# Creation date: Tue Jan  8 16:26:44 2010.
# This script is multi-platform
#
prepath=/opt/cel
machine=`uname -s`
CQUEINFO=$prepath/cqueInfo2.0
CQUETMPDIR=/tmp/
PREPATH=$prepath
PREFIX=$prepath/bin
CQUECUPSDIR=/
CQUECUPSBINDIR=/usr
PPDDIR=/usr/share/foomatic/db/source/PPD/Canon/cel
XKEYSYMDB=/usr/share/X11/XKeysymDB
XENVIRONMENT=$prepath/CQue2_0
export CQUEINFO
export CQUETMPDIR
export XKEYSYMDB
export XENVIRONMENT
export PREPATH
export PREFIX
export PPDDIR
export CQUECUPSDIR
export CQUECUPSBINDIR
$PREFIX/CQue.exe -MC100 -W$prepath $*
```

Please note that this script contains some important environment variables:

CQUEINFO : default the file /opt/cel/cqueInfo2.0. This file is the heart of CQue. We outline its syntax further below. If this file does not exist, CQue will try to create one itself or it will fail to start.

PREPATH and PREFIX are path variables to directories of CQue. These variables are exported because other modules of CQue may use them.

XENVIRONMENT : This is a standard X11 environment variable and describes the X11 resources which may possibly override those of CQue itself.

XKEYSYMDB : This is a standard X11 environment variable and describes the X11 resources for keyboard mappings.

PPDDIR : This environment variable describes where the PPD files are to be found. It is not very important.

CQUECUPSDIR : This environment variable describes where the CUPS installation is to be found. The CUPS configuration files will be found in the /etc subdirectory relative to this one.

CQUECUPSBINDIR : This environment variable describes where the CUPS shared or common data are to be found. The CUPS model directory will be found in the /share subdirectory relative to this one.

We recommend to always starting ***CQue.exe*** by means of this type of scripts in order to assure that the correct environment variables are defined.
Note, please, that the program ***CQue.exe*** is run with $*, so any arguments given to the script ***cque*** will be transferred to ***CQue.exe*** itself. It also has the standard argument -MC100 allowing for 100 file boxes support.
If you were to run CQue in black and white mode, i.e. without colour widgets, then you would add the option -a2 to the last line of the script (before the '$*'-entry).



8.6 The file cqueInfo2.0

The file ***cqueInfo2.0*** will be regularly updated and consulted by ***CQue.exe.*** It looks like:

```
#
# CQue 2.0 Information File
#
%%SICBLOCK_GENERAL_BEGIN
USRDIR          /opt/cel
HOMEDIR          $HOME
TMPDIR          /tmp
%%SICBLOCK_GENERAL_END
#
#
%%SICBLOCK_PRINTERS_BEGIN
Default
qName
%%SICBLOCK_PRINTERS_END
#
#
%%SICBLOCK_USERS_BEGIN
root
%%SICBLOCK_USERS_END
#
#
#
%%SICBLOCK_PRNT_BEGIN Default
root
%%SICBLOCK_PRNT_END Default
#
#
%%SICBLOCK_PRNT_BEGIN qName
root
%%SICBLOCK_PRNT_END qName
#
%%SICEND
```

It starts with a general block of some environment variables.
Then follows the block between %%SICBLOCK_PRINTERS_BEGIN and %%SICBLOCK_PRINTERS_END. This block contains a list of currently known printers of CQue. The entry ***Default*** is just a place holder, which will not show up in the list of printers on the X11 CQue window.
For each printer follows an entry with %%SICBLOCK_PRNT_BEGIN and %%SICBLOCK_PRNT_END. In the example above the printer queue ***qName*** is registered.



9 Using XPP for Passing of Printer Options

9.1 Introduction

CQue supports department accounting and secured printing. Some main applications, like OpenOffice etc., however, do not support a mechanism to pass strings (i.e. user id or password strings) to the printer. Therefore the PPD files of CQue supply some dummy values, like 11111, 22222, 33333. These values are, most probably, never to be used.

To bypass this problem you may want to make use of XPP, an ***X11 Printer Panel***.

This chapter will detail some issues related to XPP

9.2 Installing XPP

To make use of XPP you would first install the XPP package if it is not installed yet.

If the official package repository of your Linux distribution does not have a ready-to-install package for XPP you may have a look at: http://cups.sourceforge.net/xpp. The binary will normally be installed as ***/usr/bin/xpp***. (Note: XPP depends on the library ***fltk***, which, if not available by default, will have to be installed too).

9.2 Creating an OpenOffice Printer for XPP

Once ***xpp*** is installed, you may add a printer to OpenOffice by using the ***spadmin*** program of OpenOffice itself. The OpenOffice suite is normally installed in ***/usr/lib/openoffice*** (or a name resembling this, perhaps with the version added). Within the OpenOffice directory there is a subdirectory ***program*** and there you will find the program ***spadmin***.
When you run ***spadmin*** (as super-user or root), the following window will display:

Click on the ***New Printer*** button to arrive at the window on the next page:

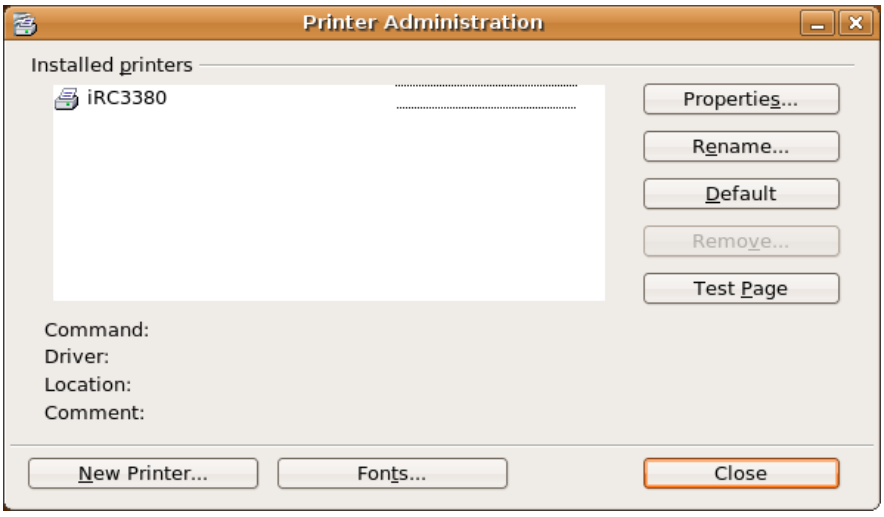


Figure 57 Start of OpenOffice spadmin

Here we are asked for the device type. Please choose a ***PDF converter***. It is used here only as a dummy. Click ***Next*** to arrive at the next window:

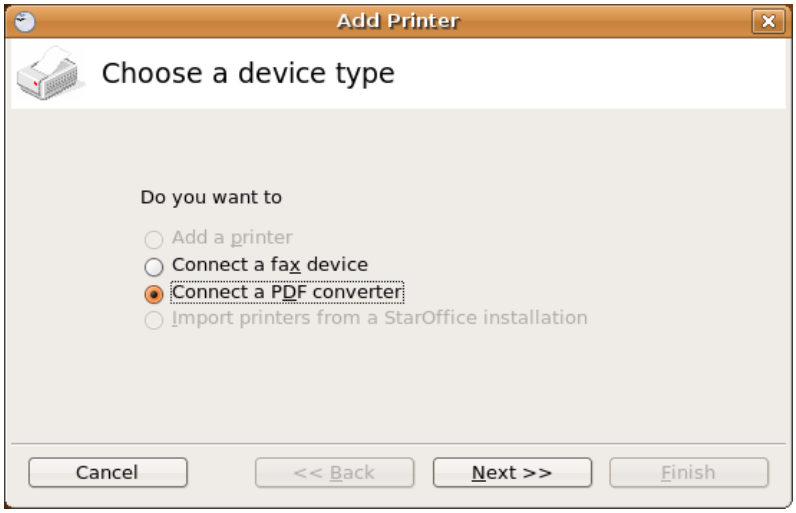


Figure 58 Choose PDF converter device type

We choose the default PDF driver (it is a dummy anyway). Click ***Next*** again to arrive at the following window:

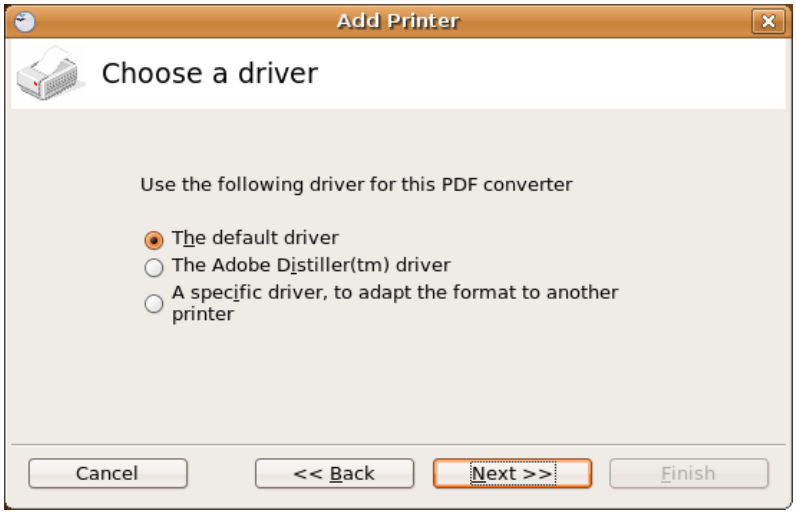


Figure 59 Choose PDF default driver

Specify the xpp execution path (by default: ***/usr/bin/xpp***). You do not need to specify any additional arguments. Click ***Next*** to arrive at the next window:

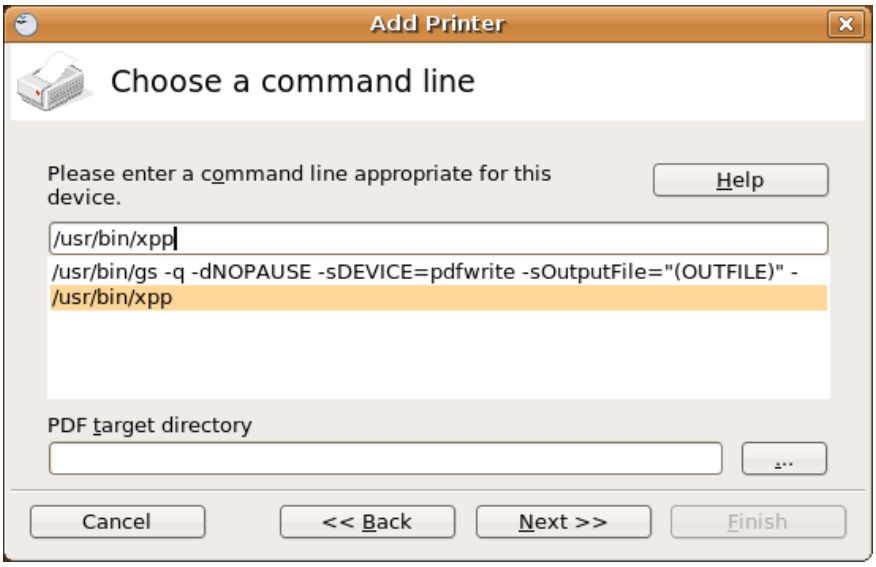


Figure 60 Specify xpp command line execution path

Choose a name for the printer, in our example: ***XPP Print***. This is the name, which will show up in the printer dialog of OpenOffice. Click ***Finish*** to terminate the installation.

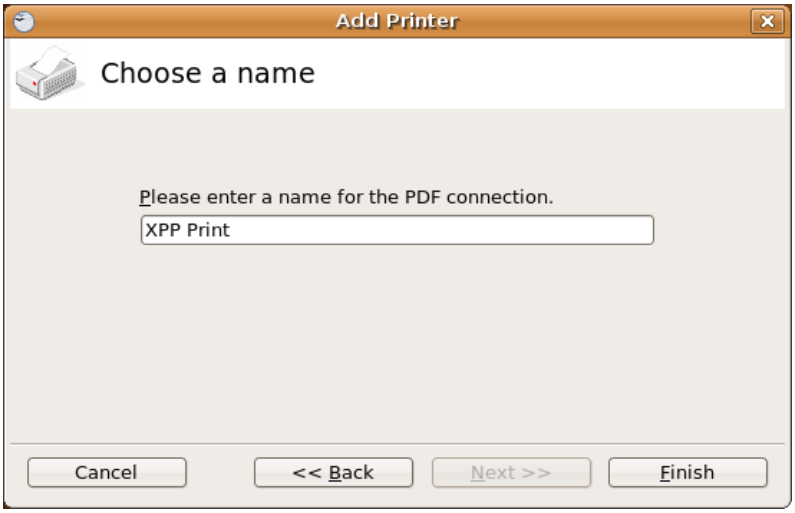


Figure 61 Choose a name for the printer

After termination the printer will show up in the list of printers known to OpenOffice.

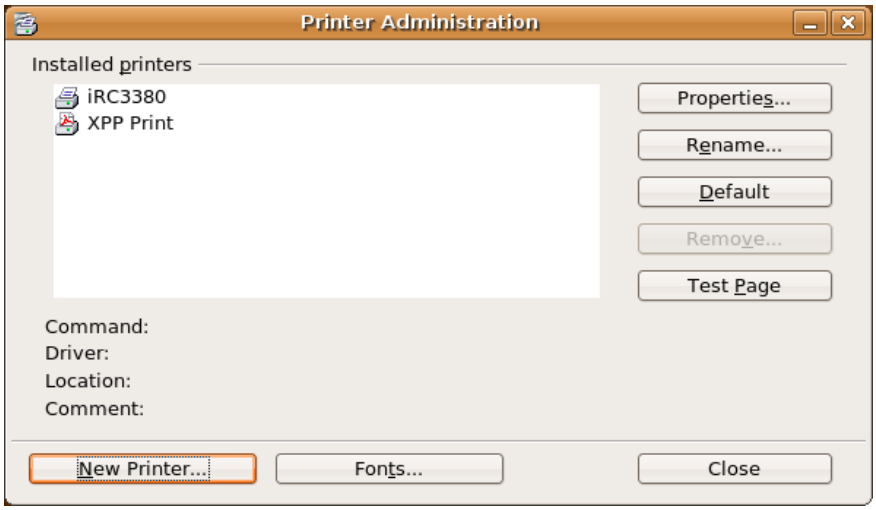


Figure 62 List of printers supported by OpenOffice

9.3 Printing from OpenOffice with XPP

Once you have edited a document in OpenOffice you can print it selecting the printer ***XPP Print***. Do not specify any properties here. The print properties will be specified after XPP opens up, as shown on the next window:

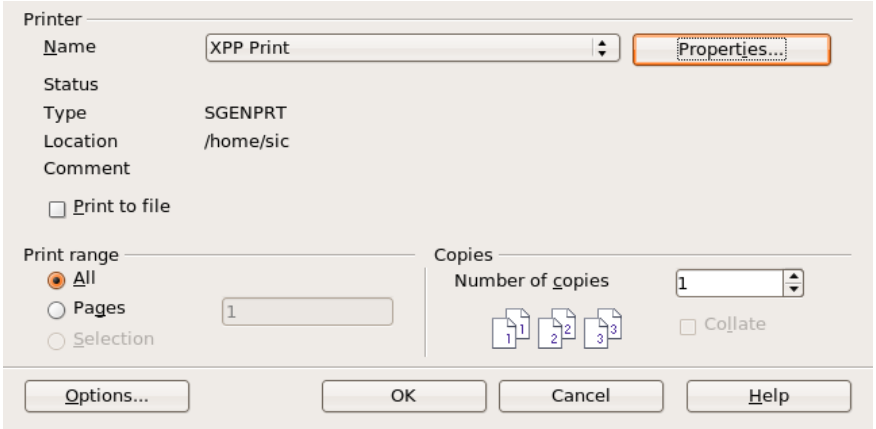


Figure 63 Printing from within OpenOffice

Now from the XPP window select the printer you want to use and click on the ***Options*** button. This will open up the last window:

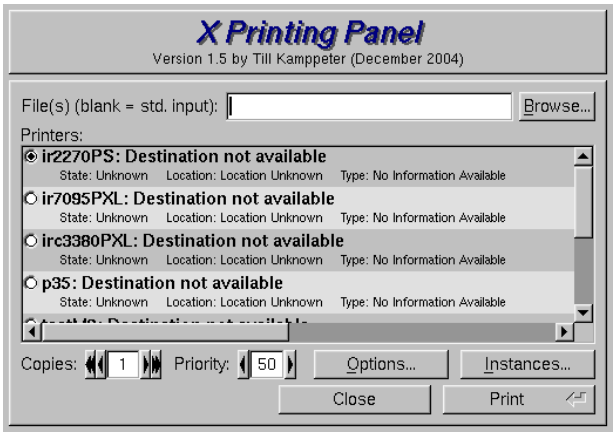


Figure 64 The XPP Window

Select the ***Canon Device Specific*** tab to possibly specify userID and or password strings. On the other tabs you can specify all other finishing options. With the ***Save settings*** button you can store your parameters as default setting. Thus, next time when you print, you would only select the ***XPP Print*** printer and then click on the ***Print*** button to send the job to the Canon device.

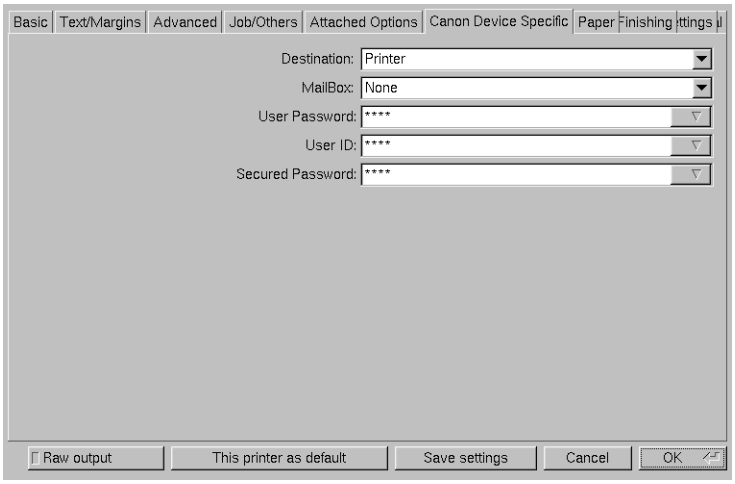


Figure 65 The XPP Printer Options Window

This procedure is only necessary when the passing of strings like user ID and/or passwords is an issue. Otherwise all relevant finishing options will be available directly through the OpenOffice print dialog itself.

The procedure is known to work for all versions of ***Mandriva***, ***Suse*** and ***Ubuntu***. It is known to fail for ***Fedora Core*** (the installation of the printer with OpenOffice works, but it will not show up in the print dialog). You may consult the ***sicforum*** site for additional information.

9.4 An XPP Printer Queue

In the case that XPP might not work for OpenOffice, or if you want to print from other applications like ***acroread***, or ***gimp*** making use of all the options of CQue, you may create a genuine xpp printer queue too.

There is two or three conditions to such a queue:

- The queue has to be created locally on the Linux workstation you are working on.
- The user ***lp*** should be allowed to open a window on your desktop.
- The printing to a file should be permitted by CUPS. This is recommended but not required.

The second condition could be arrived at by running:

```
xhost +lp
```

or possibly

```
xhost +
```

To create such a queue you may make use of the script MAKEXPP, found in the directory ***/opt/cel/bin***.

The third condition is described in Section 5.6.

The script, when run, will require two arguments:

- The name of the queue to create.
- The URL of the Canon device. If printing to a file is allowed, we recommend this argument to be ***/dev/null***. Otherwise you could specify the URL of the Canon device itself.

For example:

```
/opt/cel/bin/MAKEXPP  xpp  /dev/null
```

would create the printer queue ***xpp***.
Otherwise:

```
/opt/cel/bin/MAKEXPP  foo  socket://192.168.1.1:9100
```

would create the printer queue ***foo*** to the TCP/IP 9100 port of the Canon device.
For completeness we show here the contents of the relevant files, once the queue is

created. For the example we suppose the printer queue itself is named xpp. Then there will be a file ***/opt/cel/bin/xpp.sh***:

```
#!/bin/sh
DISPLAY=:0.0
export DISPLAY
/bin/cat $1 | /usr/local/bin/xpp
```

And a file ***/opt/cel/etc/xpp.model***:

```
#!/bin/sh
NAME=xpp
TYPE=Dumb
WDIR=/opt/cel/bin
DISPLAY=":0.0"
export DISPLAY
# Get proper user name
getTrueUserName()
{
  echo $1 | grep @ >/dev/null
  ret=$?
  if [ $ret -eq 0 ] ; then
    theName=`echo $1 | tr "@" "\\012" | head -1` >/dev/null
  else
    echo $1 | grep ! >/dev/null
    ret=$?
    if [ $ret -eq 0 ] ; then
      theName=`echo $1 | tr "!" "\\012" | tail -1` >/dev/null
    else
      theName=$1
    fi
  fi
  echo $theName
}
#
printer=`basename $0`                  # the printer's name
seqid=$1                               # job sequence id number
username=`getTrueUserName $2`
title=`echo "$3" | tr "\040" "\055"`           # job title
copies=$4                              # number of copies
options="$5"                           # -o options
shift 5                                # discard args up to filenames
files="$*"                             # input file(s)
LOGFILE=/tmp/${printer}_${username}.log

for file in $files
do
  $WDIR/xpp.sh  "$file" 2>$LOGFILE
  stat=$?
done
exit $stat
```

The MAKEXPP will create a genuine CUPS queue, which should be accessible from any



application.

If the MAKEXPP script cannot find the location of xpp on your system, then it will abandon. If xpp is installed, but on a non-standard location, you may run MAKEXPP with the full path of xpp as the third argument.



Appendix A: Accessing CQue from the Desktop

Once the program is installed you may want to access it in a rather convenient way. One way to do this is to include access to CQue in the startup menu.

With a file manager go to the installation directory of CQue, i.e. */opt/cel*. Locate there the item "CQue 2.0 Printer Driver" with the MVI logo (not the CQue2.0.png file, which contains the same logo image) and drag it to a position on the workspace bar of the desktop. You may now click on MVI icon in the toolbar to start *cque*. This would work for Gnome as well as KDE.

Another way to do this is to include access to CQue in the startup menu. To this end the setup procedure creates a file *CQue2.0.desktop* in the working directory of CQue (by default: */opt/cel*).
We describe here the procedure for RedHat which is rather general and applies mutatis mutandis to most current Linux versions. You might consult the FAQ section of the website: http://www.mv-informatique.com/forum which will contain updates on this topic.

You should copy the file *CQue2.0.desktop* to the directory:

```
/usr/share/applications
```

After that execute the following commands:

```
cd  /var/lib/menu/kde/Applications/System/MoreSystem
ln  -s  /usr/share/applications/CQue2.0.desktop  CQue2.0.desktop
```

This creates a link to the file CQue91.desktop in the /usr/share/applications directory.
You may now access CQue from the startup menu. It will be in the sub-section *"Extra System Utilities"* of the *"System tools"* section.

Finally, if you want to have a shortcut of CQue on the desktop, you may also do the following: Locate with a browser the program *cque* in the */opt/cel* directory. Then copy-paste it onto the desktop.

You might consult the FAQ section of the website: http://www.mv-informatique.com which will contain updates on this topic.
There is also a MVIForum at: http://www.mv-informatique.com/forum. Here you may find much more information about CQue and related UNIX printing topics.

Appendix B: Command Line Options

CQue relies on the CUPS printer manager for the processing of printjobs. The CUPS lp command supports a number of command line options, frequently used in classical UNIX printing.

The syntax for command line options is:

lp -d*printer* -o *option1* -o *option2* ....

Here ***option1*** has for example the syntax: *duplex=DuplexNoTumble*, or: *PageSize=a4*, etc.

Some main applications, for example ***acroread***, allow for the specification of such kind of options in the printer interface menu or windows.

To get a full listing of all options supported by the printer you may use the CUPS specific command:

lpoptions -l -d*printer*

Appendix C: A Note on PCL Printing

CQue supports PCL printers through the foomatic package, which uses the ghostscript program to transform various PDLs to PCL. Due to the intrinsics of this mechanism sometimes the output page needs an offset registration in both horizontal (x) and vertical (y) directions in order to fit completely on the paper. The CQue driver program **sicgsfilter** does apply known offsets for each Canon device.
The offsets may be different for long-edge and short-edge duplex printing. If, for whatever reason, the offsets applied by CQue do not give satisfactory output, you may modify the behaviour of CQue by applying the -r option of the sicgsfilter program.

The -r option allows for fixed PCL offsets.
The syntax is:

-r xfl,yfl[,xbl,ybl][,xfs,yfs[,xbs,ybs]]

The unit of dimension is decipoint, which means 1/720th of an inch.
The meaning of the variables is:

xfl,yfl = (x,y) offset for front page (simplex or long edge duplex).
xbl,ybl = (x,y) offset for back page (long edge duplex).
xfs,yfs = (x,y) offset for front page (simplex or short edge duplex).
xbs,ybs = (x,y) offset for back page (short edge duplex).

A negative value implies a shift to the left, or top. A positive value implies a shift to the right or bottom. This is on the front page. For the back page definitions differ.

To apply these options please follow the following steps.

1. Locate the PPD file for the printer queue. If the queue has the name 'foo,' the PPD file corresponding to that queue will be located at **/etc/cups/ppd/foo.ppd**.

2. Edit the file **/etc/cups/ppd/foo.ppd** and search the line containing the **sicgsfilter** program. An example:
   *FoomaticRIPCommandLine: "gs -q -dBATCH -dPARANOIDSAFER -dPDFFitPage -sDEVICE=ljet4 -dNOPAUSE%B%A%C -sOutputFile=- - %D%E | sicgsfilter -MPCL -NP %G%H%I -u&user; -V"&title;" -n&copies; "

   If we were to apply our own shifts, for long edge duplex only, an example might be:

   *FoomaticRIPCommandLine: "gs -q -dBATCH -dPARANOIDSAFER -dPDFFitPage -sDEVICE=ljet4 -dNOPAUSE%B%A%C -sOutputFile=- - %D%E | sicgsfilter -MPCL -NP %G%H%I **-r155,10,-155,20** -u&user; -V"&title;" -n&copies; "

If you specify values for short-edge duplex, you must first specify values for long-edge duplex.

3. Save the file and restart the queue (or cups):

   Restarting the queue is done by:

   ```
   disable  foo
   cupsenable foo
   ```

   Restarting CUPS is done by:

   ```
   service  cups restart
   ```

   or

   ```
   /etc/init.d/cups  restart
   ```

   (possibly: /etc/init.d/cupsys restart, depending on the type of Linux used).



Appendix D: Release Notes

This Appendix describes some of the known limitations and bugs on specific releases of Linux operating systems. You may always have a look at the ***sicforum*** for more up-to-date information.

1. *First release version 1.0: January 2008*

   This is the first official release of CQue 2.0.

2. *CUPS 1.3.2 (distributed by UBUNTU 7.10)*

   The CUPS 1.3.2 release as distributed by UBUNTU 7.10 has some limitations for legacy SystemV printing. For example the /etc/cups/interface directory is non-existent and the /usr/share/cups/model directory, though existing, is empty.

3. *Foomatic-rip and duplex printing*

   Some foomatic-rip distributions do not correctly process duplex printing. We strongly recommend using the latest version of foomatic-rip, which may be found at: http://www.openprinting.org/foomatic-rip.

4. *Printing multiple copies from OpenOffice*

   When printing multiple copies of the same document from OpenOffice, please note the behaviour for the various types of PDL as described in the following table.

Number of Copies	PS	PCL	PXL
1	prints 1 copy	prints 1 copy	prints 1 copy
N > 1 with collate	prints N copies	prints N copies	prints N copies
N > 1 without collate	prints N copies	prints 1 copy	prints 1 copy

   Thus, if possible, we recommend using the collate option of OpenOffice.

5. *OpenSuse*

   For OpenSuse the RPM version should be used with “rpm --nodeps”.
   OpenSuse checks for “foomatic-filters,” whereas CQue requires “foomatic.”

6. *LBP Support*

   For the LBP5970/5975 advanced features (mailbox, secured printing, department accounting) are not supported. Stapling is supported in PostScript only.

7. *PCL Offsets*

CQue allows for very precise positioning of the print data on the page. As not all Canon devices use the same PCL interpreter, details may differ between different Canon devices. The program sicgsfilter supports an option (-r) to manually specify PCL offsets in the PPD file. See Appendix C for more details.

8. *SELinux*

Several Linux distributions, including Debian and Fedora, come with the SELinux layer by default. If SELinux enforced then some CQue log messages would generate a SELinux warning. Therefore the setup program will now install a SELinux module if it detects the presence of enforced SELinux protection. (See Section 7.5)

9. *CQue 1.0-8 PPD revision*

Starting with CQue 1.0-8 all PPD files have been revised to be more stable and compatible with the latest foomatic specifications, as well as the Adobe specifications.

10. *Linux 64 bit (x86_64) support*

As of version 1.0-9 CQue comes with a separate package for Linux 64 bits version. The kernel version should be 2.6-27 or newer.

11. *CQue 1.0-10*

CQue 1.0-10 adds new devices and corrects some bugs for IPP and USB printing.

12. *CQue 1.0-11*

CQue 1.0-11 adds new devices and corrects some bugs in PPD files.

13. *CQue 2.0-0*

With respect to the version ***CQue 1.0*** some important changes of ***CQue 2.0*** include:

- All PPD files have been revised to allow for support of secured printing and printing with department accounting through the GTK+ interface. Unfortunately *OpenOffice* and *acroread* for Linux do not yet use this type of interface. It is expected, however, that future releases of these programs will support the CUPS specifications of PPD file syntax, as used by GTK+.
- CQue now ships in *rpm*, *deb* and *tgz* packages.
- The installation of an *rpm* or *deb* package includes the execution of the *setup*

program and installs the PPD files in an appropriate system directory. Any existing CQue 1.0 printer queues will be automatically upgraded with the new PPD files.
- For special purposes the *tgz* package allows for in installation without upgrading existing CQue 1.0 printer queues.
- Upgrading from SQue 9.X for Linux to CQue 2.0 for Linux is not supported. If you want to upgrade SQue 9.X to CQue, you would fist have to install CQue 1.0 (which transforms SQue printers queues to CQue printer queues) and then CQue 2.0.

Please note: Due to a bug #3631 in CUPS versions up to and including 1.4 it is not possible to set default values for the advanced options (mailbox, secured printing and printing with department accounting) making use of the user interface of CQue. This bug will be fixed in release 1.5 of CUPS.
When printing from applications making use of GTK+ advanced options are supported as described in this manual, and they do work correctly.

13. *CQue 2.0-1*

CQue 2.0-1 adds some new devices especially for professional printing.

14. *CQue 2.0-2*

CQue 2.0-2 adds some new devices especially entry level devices.

15. *Printing multiple copies of a document with stapling*

When printing multiple copies (say N copies) of a document with stapling most CUPS implementations will concatenate the output N times and then staple the resulting concatenated file. To prevent this CQue 2.0-2 PPDs support a feature: ***Repeat Job*** within the ***Canon Specific Options*** section. This allows you to specify from 1 to 99 copies for applications with the GTK+ print interface and 1 to 25 otherwise, with proper stapling of each copy. This feature is supported for PostScript, PCL5 and PCL6 Canon devices allowing PJL headers, i.e. some pure PostScript EFI devices are excluded.
Note: N copies of a document may generate N separate entries in the log file of the Canon device.

16. *CQue 2.0-3*

CQue 2.0-3 adds some new iR-ADV B/W and LBP devices.

17. *CQue 2.0-4*

CQue 2.0-4 adds some new iR-ADV B/W and Colour devices.

18. *CQue 2.0-5*

CQue 2.0-5 adds some new iR-ADV B/W and Colour devices and fixes some minor bugs.

19. *CQue 2.0-6*

CQue 2.0-6 adds some new iR-ADV B/W and Colour and B/W LBP devices.
CQue 2.0-6 revises PPD files for compatibility with CUPS versions 1.6 and later.
If you use CUPS 1.6 or later, please use CQue 2.0-6 or later.
Manual paper selection bug fixed for PCL drivers.
Limit for password of printing with department accounting fixed and some minor bugs were fixed too.

20. *CQue 2.0-7*

CQue 2.0-7 adds some new iR-ADV B/W and Colour and B/W LBP devices.
CUPS 1.7 integrates the foomatic package into CUPS itself. CQue 2.0-7 is the first version to be compatible with this new approach of CUPS.

21. *CQue 2.0-8*

CQue 2.0-8 adds some new iR-ADV B/W and Colour devices.

22. *CQue 2.0-9*

CQue 2.0-9 adds some new iR-ADV B/W and Colour devices. Also some minor bugs have been solved.