A Guide to the Lp Printer Spooler

Paul Glick
pg@plan9.att.com

ABSTRACT

Lp is a collection of programs used to provide an easy-to-use interface for printing a variety of document types on a variety of printers. *Lp* is the glue that connects various document language translators and printer communication programs together so that the users may have a consistent view of printers. Most of the glue is shell scripts, which can be easily modified. The user need not specify options to get sensible output in most cases. *Lp* is described here so that others may make additions and changes.

1. Introduction

Lp is used to format and print data on a variety of output devices. The need for *lp* was rooted in the inability of other printer spoolers to do simple tasks without a great deal of user specification of options. At the time *lp* was written, there were several printer languages, such as ImPress and PostScript, and an internally developed printer that would accept *troff* output. Now, all our printers take PostScript, but printers that use HPCL and HPGL abound and support for those printers may be added easily. A great deal of what underlies *lp* is taken from BSD's *lpr* and System V's *lp*. The important features of this system are that most of the programs are easily modified shell scripts and the user need not learn to use the large amount of underlying software developed by others. *Lp* runs under Plan 9 and several flavors of UNIX. This document deals with *lp* as it relates to Plan 9. *Lp* was developed using both Datakit and Ethernet to transport data between machines. Now only the Ethernet transport mechanism remains.

Text, graphics, and formatted text files are appropriately processed and placed into a spool directory from which they are taken to be printed by a daemon process. Additional functions include checking the status of a printer queue and removing jobs from the printer queue.

All the shell scripts (see *rc*(1)) associated with *lp* reside in the spool directory /sys/lib/lp except for the *lp* command itself, which resides in /rc/bin. Commands related to *lp* that are not shell scripts can most often be found in /$cputype/bin/aux. The directory where all the *lp* scripts reside is defined within *lp* by the shell variable LPLIB. In the remainder of this document, file names will be specified with this shell variable as their root.

2. Usage

Lp requires an output device to be specified before it will process input. This can be done in any of three ways described here.

1) The file $LPLIB/defdevice may contain the name of a default output device. This may not be practical for environments where there are many printers.

2) The user's environment variable LPDEST may be set to the name of the device to

be used. This is often a more practical solution when there are several printers available. This overrides the default device specification.

3) The -d *printer* option to the *lp* command specifies *printer* as the device to which output should be directed, overriding the previous two specifications.

If *printer* is ?, a list of printers and other information in the devices file is printed, as shown in Figure 1. Quote the question mark to prevent it from being interpreted by the shell language as a metacharacter.

```
% lp -d'?'
device     location    host              class
fn         2C-501      helix             post/2+600dpi+duplex
pcclone    -           -                 post+nohead
peacock    2C-501      cetus             post/2+300dpi+nohead+color
ps83       st8_fl3     rice              post+300dpi+reverse
psu        2C-501      cetus             post/2+1200dpi
      .
      .
      .
%
```

Figure 1. Sample listing of installed printers

Normally, *lp* uses the file command to figure out what type of input it is receiving. This is done within the generic process which is discussed later in this paper in the **Process directory** section. To select a specific input processor the -p *process* option is used where *process* is one of the shell scripts in the process directory.

Troff output can be printed, in this case, on printer *fn*, with

```
% troff -ms lp.ms | lp -dfn
```

A file can be converted to PostScript using the pseudo-printer stdout:

```
% troff -ms lp.ms | lp -dstdout > lp.ps
```

LaTeX (and analogously TeX) documents are printed in two steps:

```
% latex lp.tex
      .
      .
% lp lp.dvi
      .
      .
%
```

LaTeX produces a .dvi file and does not permit the use of a pipe connection to the standard input of *lp*. To look at the status and queue of a device, use -q:

```
% lp -dpsu -q
daemon status:
: 67.17% sent
printer status:
%%[ status: busy; source: lpd ]%%

queue on cetus:
job              user     try     size
rice29436.1      pg       0       17454
slocum17565.1    ches     1       49995
%
```

This command can print the status and queue of the local and remote hosts. Administrators should be advised that working in an environment where the *lp* spool directory is shared among the local and remote hosts, no spooling should be done on the local hosts. The format of the status and queue printout is up to the administrator. The job started above can be killed with -k:

```
$ lp -dpsu -k rice29436.1
rice29436.1   removed   from  psu  queue  on  cetus
```

3. Options

There are options available to modify the way in which a job is handled. It is the job of the *lp* programs to convert the option settings so they may be used by each of the different translation and interface programs. Not all options are applicable to all printer environments. Table 1 lists the standard *lp* options, the shell variable settings, and description of the options.

4. Devices file

The devices file is found in the spool directory. Each line in the file is composed of 12 fields, separated by tabs or spaces, that describe the attributes of the printer and how it should be serviced. Within the lp command, a shell variable is set for each attribute; the following list describes them:

LPDEST
: is the name of the device as given to *lp* with the -d option or as specified by the shell environment variable LPDEST or as specified by the file $LPLIB/defdevice . This name is used in creating directories and log files that are associated with the printers operation.

LOC
: just describes where the printer is physically located.

DEST_HOST
: is the host from which the files are printed. Files may be spooled on other machines before being transferred to the destination host.

OUT_DEV
: is the physical device name or network address needed by the printer daemon to connect to the printer. This field depends on the requirements of the daemon and may contain a - if not required.

SPEED
: is the baud rate setting for the port. This field depends on the requirements of the daemon and may contain a - if not required.

LPCLASS
: is used to encode minor printer differences. The keyword reverse is used by some of the preprocessors to reverse the order the pages are printed to accommodate different output trays (either face up or face down). The keyword nohead is used to suppress the header page. This is used for special and color printers. The keyword duplex is used to coax double sided output from duplex printers.

LPPROC
: is the command from the LPLIB/process directory to be used to convert input to a format that will be accepted by the device. The preprocessor is invoked by the spooler.

SPOOLER
: is the command from the LPLIB/spooler directory which will select files using the SCHED command and invoke the LPPROC command, putting its output into the remote spool directory. The output is sent directly to the spool directory on the destination machine to avoid conflicts when client and server machines share spool directories.

STAT
: is the command from the LPLIB/stat directory that prints the status of the device and the list of jobs waiting on the queue for the device. The status information depends on what is available from the printer and interface software. The queue information should be changed to show information

option	shell variable name	default	set	action
-D	DEBUG	N	1	turn on debugging mode.
-H	NOHEADER	N	1	suppress header page.
-L	LAND	N	1	make long page dimension horizontal.
-M *mach*	LPMACHID	N	*mach*	set the source machine name.
-Q	QONLY	N	1	do not execute daemon; for debugging.
-c *n*	COPIES	N	*n*	number of copies to be printed.
-d *printer*	LPDEST	U	*printer*	set job destination, override other settings.
-f *font.pt*	FONT POINT	N N	*font* *pt*	set font style and point size for printing.
-i *n*	IBIN	N	*n*	select input paper tray options. The argument given is dependent on the printer type. A number can be given to select a particular tray and/or simplex or duplex may be used to get single or double sided output, where applicable. Multiple options should be separated by commas.
-k	KILLFLAG	0	1	take non-option arguments as job numbers to be removed from queue.
-l *n*	LINES	N	*n*	for printed data, the number of lines per logical page.
-m *f*	MAG	N	*f*	magnify the image by a factor *f*. The factor should be a positive real number.
-n *n*	NPAG	N	*n*	put *n* logical pages on a single physical page. A simple algorithm is used to pack the pages.
-o *list*	OLIST	N	*list*	print only those pages specified in the list. The list may be a sequence of numbers or ranges separated by commas. A range is a pair of numbers separated by a hyphen.
-p *proc*	LPPROC	L	*proc*	use the preprocessor *proc* instead of the preprocessor given in the devices file for this printer.
-q	LPQ	N	1	print the status and queue.
-r	REVERSE	L	1	this toggles the REVERSE flag, changing whether or not page reversal should occur in preprocessing. Page reversal is needed if a printer delivers pages face up. The keyword reverse can be placed in the *lpclass* field of the devices file. If a document has already been processed this flag has no effect.
-u *user*	LPUSERID	U	*user*	change the userid that appears on the cover page.
-x *offset*	XOFF	N	*offset*	move the image *offset* inches to the right. A negative *offset* will move the image to the left. The *offset* may be any reasonable real number.
-y *offset*	YOFF	N	*offset*	same as for -x except a positive offset will move the image down.

default	setting definition
N	set to the null string ( ) initially in *lp*.
L	set from printer entry in devices file.
U	set from the users environment.

Table 1. Lp Option List

useful in tracking down problems. The SCHED command is used to show the jobs in the order in which they will be printed.

KILL is the command from the LPLIB/kill that removes jobs from the queue. The jobs to be removed are given as arguments to the *lp* command. When possible, it should also abort the currently running job if it has to be killed.

DAEMON is the command from the LPLIB/daemon that is meant to run asynchronously to remove jobs from the queue. Jobs may either be passed on to another host or sent to the printing device. *Lp* always tries to start a daemon process when one is specified.

SCHED is the command from the LPLIB/sched that is used to present the job names to the daemon and stat programs in some order, e.g., first-in-first-out, smallest first.

5. Support programs

The following sections describe the basic functions of the programs that are found in the subdirectories of $LPLIB. The programs in a specific directory vary with the type of output device or networks that have to be used.

5.1. Process directory

The generic preprocessor is the default preprocessor for most printers. It uses the *file* (1) command to determine the format of the input file. The appropriate preprocessor is then selected to transform the file to a format suitable for the printer.

Here is a list of the preprocessors and a description of their function.

dvipost Converts TeX or LaTeX output (dvi files) to PostScript

ppost Converts UTF text to PostScript. The default font is Courier with Lucida fonts filling in the remainder of the (available) Unicode character space.

tr2post Converts (device independent) troff output for the device type utf. See /sys/lib/troff/font/devutf directory for troff font width table descriptions. See also the /sys/lib/postscript/troff directory for mappings of troff UTF character space to PostScript font space.

p9bitpost Converts Plan 9 bitmaps (see *bitfile* (9.6)) to PostScript.

g3post Converts fax (CCITT-G31 format) to PostScript.

hpost Does header page processing and page reversal processing, if necessary. Page reversal is done here so the header page always comes out at the beginning of the job. Header page processing is very location-dependent.

5.2. Spool directory

The generic spooler is responsible for executing the preprocessor and directing its output to a file in the printer's queue. An additional file is created containing information such as the system name, userid, job number, and number of times this job was attempted.

Certain printer handling programs do not require separate preprocessing and spooling. For such circumstances a nospool spooler is available that just executes the preprocessing program. The processing and spooling functions are assumed by this program and the output is sent to OUT_DEV or standard output if OUT_DEV is - .

The pcclone spooler is used to send print jobs directly to a printer connected to a 386 compatible printer port (see *lpt*(3)).

5.3. Stat directory

The function of the shell scripts in the stat directory is to present status information about the printer and its queue. When necessary, the stat scripts may be designed to return information about the local queue as well as the remote queue. This is not done on Plan 9 because many systems share the same queue directory. The scheduler is used to print the queue in the order in which the jobs will be executed.

5.4. Kill directory

The kill scripts receive command line arguments passed to them by *lp* and remove the job and id files which match the arguments for the particular queue. When a job is killed, the generic kill procedure:

1) kills the daemon for this queue if the job being killed is first in the queue,

2) removes the files associated with the job from the queue,

3) attempts to restart the daemon.

5.5. Daemon directory

The daemon shell scripts are the last to be invoked by *lp* if the -Q option has not been given. The daemon process is executed asynchronously with its standard output and standard error appended to the printer log file. The log file is described in a subsequent section. Because the daemon runs asynchronously, it must catch signals that could cause it to terminate abnormally. The daemon first checks to see that it is the only one running by using the LOCK program found in the /$cputype/bin/aux directory. The LOCK command creates a LOCK file in the printer's queue directory. The daemon then executes the scheduler to obtain the name of the next job on the queue.

The processing of jobs may entail transfer to another host or transmission to a printer. The details of this are specific to the individual daemons. If a job is processed without error, it is removed from the queue. If a job does not succeed, the associated files may be moved to a printer specific directory in $LPLIB/prob. In either case, the daemon can make an entry in the printer's log file. Before exiting, the daemon should clean up lock files by calling UNLOCK.

Several non-standard daemon programs have been designed to suit various requirements and whims. One such program announces job completion and empty paper trays by causing icons to appear in peoples seemail window. Another, using a voice synthesizer, makes verbal announcements. Other daemons may be designed to taste.

5.6. Sched directory

The scheduler must decide which job files should be executed and in what order. The most commonly used scheduler program is FIFO, which looks like this:

```
ls -tr $* | sed -n -e 's/.*    *//' \
   -e '/^[0-9][0-9]*.[1-9][0-9]*$/p'
```

This lists all the job files in this printer's queue in modification time order. Jobs entering the queue have a dot (.) prefixed to their name to keep the scheduler from selecting them before they are complete.

6. When Thing Go Wrong

There are four directories where *lp* writes files. On the Plan 9 release these directories may be found in the directory /n/bootes/other/lp. The four directories are log, prob, queue, and tmp. *Lp* binds (see *bind*(1)) the first three into the directory /sys/lib/lp for its processes and their children. The tmp directory is bound to the

/tmp directory so that the lp daemons, which run as user none , may write into this directory.

6.1. Log directory

The log files for a particular *printer* appear in a subdirectory of the spool directory log /*printer* . There are currently two types of log files. One is for the daemon to log errors and successful completions of jobs. These are named *printer.day* where *day* is the three letter abbreviation for the day of the week. These are overwritten once a week to avoid the need for regular cleanup. The other type of log file contains the status of the printer and is written by the program that communicates with the printer itself. These are named *printer* .st . These are overwritten with each new job and are saved in the $LPLIB/prob directory along with the job under circumstances described below. When a printer does not appear to be functioning these files are the place to look first.

6.2. Prob directory

When a job fails to produce output, the log files should be checked for any obvious problems. If none can be found, a directory with full read and write permissions should be created with the name of the printer in the $LPLIB/prob directory. Subsequent failure of a job will cause the daemon to leave a copy of the job and the printer communication log in $LPLIB/prob/ *printer* directory. It is common for a printer to enter states from which it cannot be rescued except by manually cycling the power on the printer. After this is done the print daemon should recover by itself (give it a minute). If it does not recover, remove the LOCK file from the printer's spool directory to kill the daemon. The daemon will have to be restarted by sending another job to the printer. For PostScript printers just use:

```
echo '%!PS' | lp
```

6.3. Repairing Stuck Daemons

There are conditions that occur which are not handled by the daemons. One such problem can only be described as the printer entering a comatose state. The printer does not respond to any messages sent to it. The daemon should recover from the reset and an error message will appear in the log files. If all else fails, one can kill the first job in the queue or remove the LOCK file from the queue directory. This will kill the daemon, which will have to be restarted.

7. Interprocessor Communication

A Plan 9 CPU server can be set up as a printer's spooling host. That is, the machine where jobs are spooled and from which those jobs are sent directly to the printer. To do this, the CPU must listen on TCP port 515 which is the well known port for the BSD line printer daemon. The file /rc/bin/service/tcp515 is executed when a call comes in on that port. The Plan 9 lpdaemon will accept jobs sent from BSD systems. The /$cputype/bin/aux/lpdaemon command is executed from the service call and it accepts print jobs, requests for status, and requests to kill jobs. The command /$cputype/bin/aux/lpsend is used to send jobs to other Plan 9 machines and is usually called from within a spooler or daemon script.

8. Acknowledgements

Special thanks to Rich Drechsler for supplying and maintaining most of the PostScript translation and interface programs, without which *lp* would be an empty shell. Tomas Rokicki provided the TeX to PostScript translation program.

9. References

[Camp86] Ralph Campbell, 4.3BSD Line PrinterSpooler Manual , UNIX System Manager s Manual, May, 1986, Berkeley, CA
[RFC1179] Request for Comments: 1179, Line Printer Daemon Protocol, Aug 1990
[Sys5] System V manual, date unknown