



Printing Without Channels

RSCS, CUPS, And Every Other Printer
Known To Mankind

David Boyes/Jacob Welsh
Sine Nomine Associates



SINE NOMINE
ASSOCIATES

Agenda

- VM Printing 'Classic'
- VM Printing Today
- Configuring a RSCS LPR Link
- Configuring CUPS
- Using Your New Link from CMS
- Q & A

VM Printing 'Classic'

- In a classic VM system, CP handled all spooling and printing to line printers
 - Printers were assumed to be attached to System/390 channel hardware
 - CP controlled forms, skip destinations, and character set translations
 - Limited functionality: line printers
 - Control from operator commands
 - Remote printing handled with channel extenders

VM Printing Now

- In a modern VM system, channel attached printers are likely to be no longer available.
- Most printers are almost exclusively high-function devices that are far more than line printers
 - Still use the system spool as queuing device
 - Almost all use a complex formatting control language (HP PCL or PostScript) instead of simple text
 - Almost all are LAN-based and shared with other systems

VM Printing Now

- Driven by RSCS over LAN (lpr)
 - Remote impact printers
 - HP PCL
 - PostScript (*)
- RSCS converts output to the appropriate format and transmits to remote printer



SINE NOMINE
ASSOCIATES

VM Printing Now

- PostScript offers the most control over printer capabilities
 - Scaling
 - Page size
 - Vendor
 - But..... Requires printer to incorporate a interpreter for the PS language!

Configuring RSCS For LPR

- In RSCSTCP CONFIG (TCPMAINT 198):

```
LINKDEFINE <linkname> TYPE LPR FORM * AST
```

```
PARM <linkname> EXIT=LPRXPSE
```

```
HOSTN=print.devlab.sinenomine.net PRINTER=va2p2z
```



SINE NOMINE
ASSOCIATES

CUPS

- Common Unix Printing System
 - Adds direct managed API for print features on Unix/Linux
 - Accept Postscript and process using software interpreter and filters
 - PS to raster
 - Raster to printer data stream
 - Accept multiple input transports (lpr, IPP)

Configuring CUPS

- Use a Linux guest to host CUPS instance
 - Do transformation and print management as an appliance
 - Any printer can be a mainframe printer
 - User experience follows...



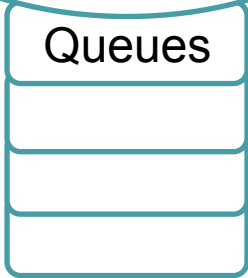
Configuration

Browsing Web Command Line

Output

Input

IPP
LPD



IPP
LPD
AppSocket/JetDirect
USB/Serial/Parallel
HPLIP
Others

IPP Internet Printing Protocol
LPD Line Printer Daemon
HPLIP HP Linux Imaging & Printing

CUPS Components

`cupsd`

- Listens on TCP port 631 (IPP)
- Receives jobs
- Manages print queues
- Invokes filters and backends
- Configuration: `/etc/cups/cupsd.conf`
 - Warning: can be modified by `cupsd`

`cups-lpd` (optional)

- Receives LPD jobs, resubmits to IPP

CUPS Components

PPD files

- PostScript Printer Description (Adobe standard)
- Extended by CUPS for non-PS printers
- Paper sizes, memory, color, PS level, fonts...
- Provided by manufacturers or others (OpenPrinting database)
- Installed to `/etc/cups/ppd/` on queue creation

CUPS Components

Filters

- External programs
- Standard interface for format conversions
- Provided by CUPS
- In `/usr/lib/cups/filters/`
- Configured via `mime.convs` and PPDs
(`*cupsFilter` lines)

CUPS Components

Backends

- ipp, lpd, socket, usb, parallel, hp...
- Configured on queue creation

CUPS Components



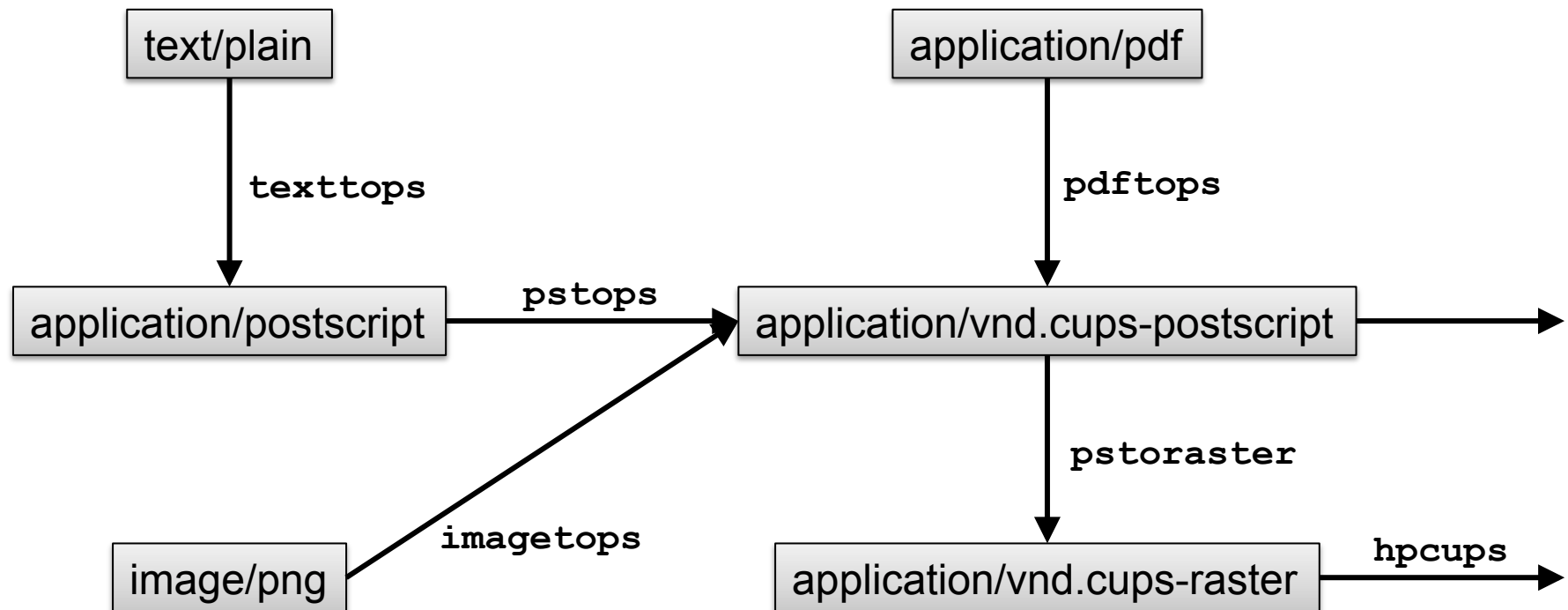
- “A high quality, high performance Postscript and PDF interpreter and rendering engine”
- Many CUPS filters are just shell script wrappers for Ghostscript



SINE NOMINE
ASSOCIATES

Filter Routing (Simplified)

Based on CUPS 1.4.2, HPLIP 3.13.3





The Problem With RSCS Postscript

DSC Conforming:

```
%!PS-Adobe-3.0
%%BoundingBox: 0 0 612 792
%cupsRotation: 0
%%Creator: texttops/CUPS v1.4.2
%%CreationDate: Tue 11 Jun 2013 04:14:35 PM EDT
%%Title: (title)
%%For: (root)
%%Pages: (atend)
%%DocumentNeededResources: font Courier-Bold
%%+ font Courier
%%DocumentSuppliedResources: procset texttops 1.1 0
%%+ font Courier-Bold
%%+ font Courier
%%EndComments
%%BeginProlog
%%Orientation: Portrait
%%BeginResource: font Courier-Bold

%%EndResource
%%BeginResource: font Courier

%%EndResource
...
```

From RSCS:

```
%!PS
0000 % serverloop password
/$brkpage where not {
  dup serverdict begin statusdict begin checkpassword
  { (Error Handler downloaded.\n) print flush exitserver }
  { pop (Bad Password on loading error handler.\n)
    print flush stop }
  ifelse
} {
  pop pop (Error Handler in place - not loaded again\n)
  print flush stop
} ifelse
/$brkpage 64 dict def
/=string where not { /=string 128 string def } { pop } ifelse
$brkpage begin
  /= { /cp 0 def typeprint nl } def
  /printpage {
    /prnt {
      dup type /stringtype ne { =string cvs } if dup length 6
      mul /tx exch def
      /ty 10 def currentpoint /toy exch def /tox exch def 1
      setgray newpath
    ...
```

“It doesn’t print!”

```
runEPS: Not DSC
Error Handler downloaded.
Error: /invalidaccess in /exitserver
Operand stack:
```

Execution stack:

```
%interp_exit .runexec2 --nostringval-- --nostringval-- --nostringval-- 2 %stopped_push --nostringval--
--nostringval-- --nostringval-- false 1 %stopped_push 1862 1 3 %oparray_pop 1861 1 3
%oparray_pop 1845 1 3 %oparray_pop 1739 1 3 %oparray_pop --nostringval-- %errorexec_pop .runexec2
--nostringval-- --nostringval-- --nostringval-- 2 %stopped_push --nostringval--
```

Dictionary stack:

```
--dict:1151/1684(ro)(G)-- --dict:0/20(G)-- --dict:70/200(L)-- --dict:6/10(L)-- --dict:87/89(L)--
```

Current allocation mode is local

Current file position is 344

GPL Ghostscript 8.70: Unrecoverable error, exit code 1

From the Ghostscript docs...

-dJOBSEVER

Define \004 (^D) to start a new encapsulated job used for compatibility with Adobe PS Interpreters that ordinarily run under a job server. The -dNOOUTERSAVE switch is ignored if -dJOBSEVER is specified since job servers always execute the input PostScript under a save level, although the exitserver operator can be used to escape from the encapsulated job and execute as if the -dNOOUTERSAVE was specified.

This also requires that the input be from stdin, otherwise an error will result (Error: /invalidrestore in --restore--).

[Credit: Leland Lucius found that this made it work]



“It still doesn’t print!”

```
D [12/Jun/2013:13:12:19 -0400] [Job 36] prnt/hpcups/HPCupsFilter.cpp 266: ERROR: Unsupported resolution
D [12/Jun/2013:13:12:19 -0400] [Job 36] prnt/backend/hp.c 839: ERROR: null print job total=0
D [12/Jun/2013:13:12:19 -0400] PID 12257 (/usr/lib/cups/backend/hp) exited with no errors.
D [12/Jun/2013:13:12:19 -0400] PID 12256 (/usr/lib/cups/filter/hpcups) stopped with status 1!
D [12/Jun/2013:13:12:19 -0400] [Job 36] cups_print_chunked: xflip = 0, yflip = 0, height = 1074
D [12/Jun/2013:13:12:19 -0400] PID 12255 (/usr/lib/cups/filter/pstoraster) did not catch or ignore signal 13.
D [12/Jun/2013:13:12:19 -0400] Discarding unused job-state-changed event...
E [12/Jun/2013:13:12:19 -0400] [Job 36] Job stopped due to filter errors; please consult the error_log file for details.
```

A closer look into the pipeline...

Temporarily modified `pstoraster` script to capture final stage before rasterization:

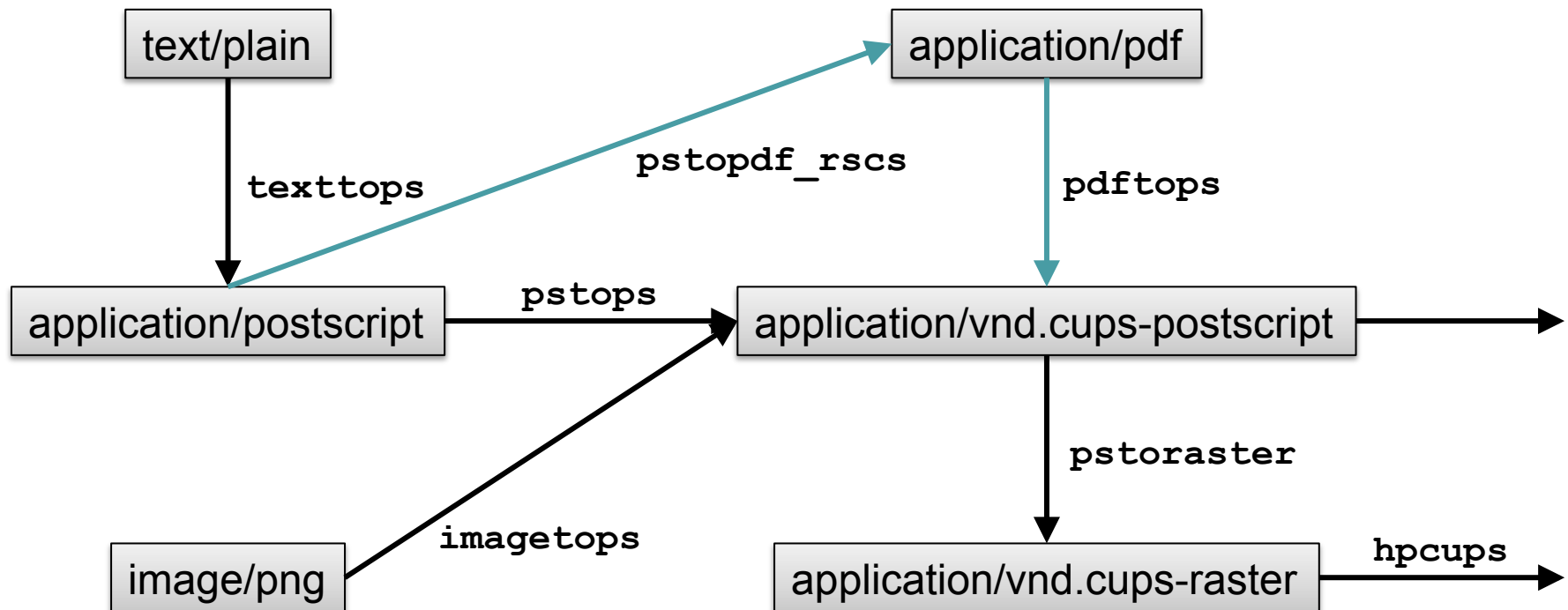
```
cat "$6" > /var/log/cups/test.ps  
exit 1
```

Results:

```
%!PS-Adobe-3.0  
%%Creator: texttops/CUPS v1.4.2  
%%LanguageLevel: 3  
%%DocumentSuppliedResources: (attend)  
%%DocumentMedia: plain 612 792 0 () ()  
%%For: (root)  
%%Title: (test.ps)  
%RBI NumCopies: 1  
%%Pages: (attend)  
%%BoundingBox: (attend)  
%%EndComments  
%%BeginDefaults  
%%PageMedia: plain  
%%EndDefaults  
...
```



Filter Routing, Modified



pstopdf_rscs Modifications

Add the necessary option:

```
gsopts="-dQUIET -dDEBUG -dPARANOIDSAFER -dNOPAUSE -dBATCH -dJOBSERVER "  
gsopts="$gsopts -dNOMEDIAATTRS -sDEVICE=pdfwrite -sstdout=%stderr"
```

Remember, -dJOBSERVER mandates standard input, but CUPS may supply filters with a named file. Handle this case with a redirect:

```
if test -z "$6"; then  
    $bindir/gs $gsopts -sOUTPUTFILE="%stdout" $profile -  
else  
    $bindir/gs $gsopts -sOUTPUTFILE="%stdout" $profile - < "$6"  
fi
```

And remove the original gs command:

```
# $bindir/gs $gsopts -sOUTPUTFILE="%stdout" $profile $ifile
```



SINE NOMINE
ASSOCIATES

mime.convs Additions

```
application/postscript application/pdf          20      pstopdf_rscs
application/pdf         application/vnd.cups-postscript 20      pdftops
```


Appendix: CUPS quick reference (RHEL 6 flavor)

Installing, enabling, starting:

```
yum install cups foomatic ghostscript
chkconfig cups on
service cups start
```

For LPD support:

```
yum install cups-lpd
chkconfig xinetd on
chkconfig cups-lpd on
service xinetd (re)start
```

For HP printers: Install hplipfull RPM from <http://hplipopensource.com/>

Important files

/etc/cups/cupsd.conf: Main server configuration (rewritten by cupsd on web or CLI admin)
/etc/cups/printers.conf: Queue definitions (periodically rewritten by cupsd)
/etc/cups/ppd/<queue>.ppd: Written by cupsd but can be edited
/usr/lib/cups/filters/*: Filter executables
/usr/share/cups/mime/mime.convs, mime.types: Filter routing config
/etc/cups/mime.convs, mime.types: Local filter config (not overwritten by upgrades)
/var/spool/cups/: Queued job data
/var/cache/cups/: Various caches, such as remote printers discovered by browsing
/var/log/cups/error_log: Useful for problem solving
/etc/pam.d/cups: Password authentication config

Some commands

```
lp -d <queue> <file>
lpstat
cancel
lpadmin
cupsctl
```

Web interface (user & admin)

<http://your-cups-server:631/> (Default config only listens on localhost)

Useful docs

```
man cupsd.conf
```

Using Your New Link

CP SPOOL PRT TO RSCS

CP TAG DEV PRT <local> <linkname>

PRINT fn ft fm

PPS fn ft fm (options



Q&A

- dboyes@sinenomine.net
- jwelsh@sinenomine.net