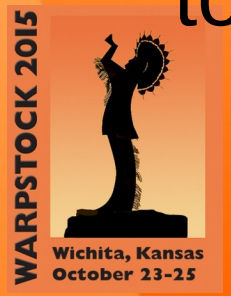
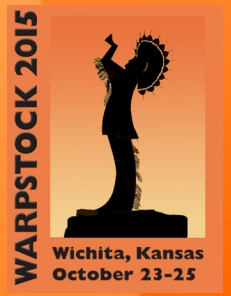
What is CUPS?

- CUPS - Common Unix Printing System
- Originally developed by Easy Software Products, was adopted as the default print system for MacOS in 2002; was purchased by Apple in 2007.
- CUPS allows postscript (or PDF) files to be sent to its daemon, which then translates the data into a printer specific format and sends it to the printer



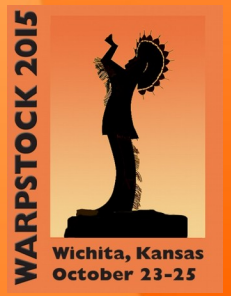
How does it work?

- CUPS provides a mechanism that allows print jobs to be sent to printers in a standard fashion.
- The print-data goes to a scheduler which sends jobs to a filter system that converts the print job into a format the printer will understand.



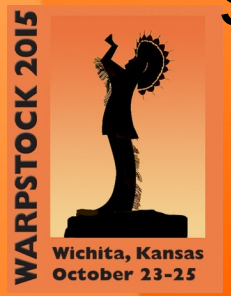
How does it work (cont)?

- The filter system then passes the data on to a backend—a special filter that sends print data to a device or network connection.
- The system makes extensive use of PostScript and rasterization of data to convert the data into a format suitable for the destination printer.



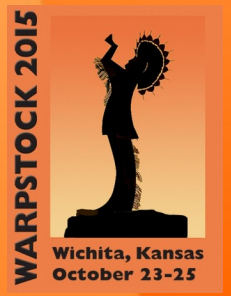
History of CUPS for OS/2 & eComStation

- Idea or porting CUPS to OS/2 originally suggested to me by Bart van Leeuwen and Adrian Gschwend on #netlabs IRC channel
- Concept was that as OS/2 could generate postscript; and CUPS could process postscript – then why couldn't CUPS be used as the print engine for OS/2?
- Whilst it is all working now, the concept took some time to be understood by me.



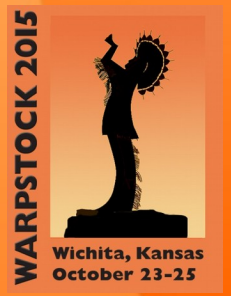
History of CUPS on OS/2 & eComStation

- First builds of CUPS for OS/2 generated in late 2006
- Libc fixes were necessary to get the scheduler stable(ish)
- Other components also required include:
  - Ghostscript (a version that supports CUPS)
  - A printer driver (eg hplip, gutenprint, splix, etc)



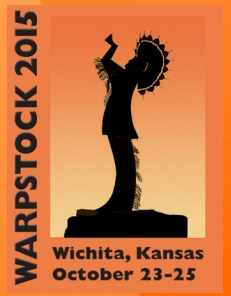
CUPS print job processing (example for CUPS <=1.4.8)

- Postscript file sent to CUPS
- Pstops filter run to 'clean postscript'
- Pstoraster filter run to convert postscript into CUPS raster format (using Ghostscript)
- Printer specific backend (eg rastertogutenprint) run to convert CUPS raster format into printer specific format
- Backend filter (eg usb.exe or socket.exe) run to send data to the printer



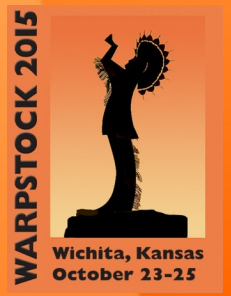
Current Status:

- CUPS on OS/2 currently at v2.04 level (latest is v2.1)
- Also requires:
  - cups-filters package – currently at 1.0.71 level
  - Ghostscript – currently at v9.15
  - Printer driver (eg hplip or gutenprint)
  - Some patience



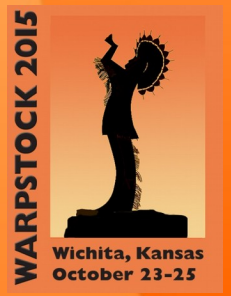
Available printer drivers - HPLIP

- HPLIP (current build v3.14.3) – supports HP devices. To see which models are potentially supported, check http://hplipopensource.com/hplip-web/supported_devices/index.html
- Whilst most models listed can be supported, any that reference a 'Drive plugin' CANNOT be supported.



Available printer drivers - HPLIP

- Example statement for a model that can't be supported due to the non-open license of the source
- This printer REQUIRES a downloadable driver plug-in, which is required to enable print, fax or scan support. Use hp-setup to install the printer, and to download and install the plug-in. Driver plug-ins are released under a proprietary (non-open) license and are not part of the HPLIP tarball release.

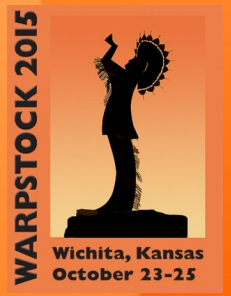


Available printer drivers

- Gutenprint – current build is v5.2.10
  - Supports primarily Canon & Epson printers – device list at http://gimp-print.sourceforge.net/p_Supported_Printers.php
- Splix – current build is 2.0.0 – supports Samsung SPL printers. Device list is at http://splix.ap2c.org/

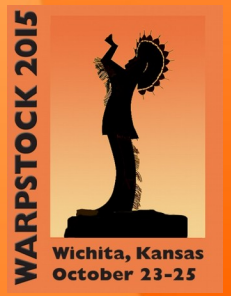
Available device backends

- Usb – utilising libusb10.dll & usbcalls
- Socket – for printing to devices that utilise HP Jetdirect
- Lpd
- Ipp – Internet Printing Protocol
- Bjnp – 3rd party backend for certain Canon printers
- Parallel not supported by CUPS on OS/2



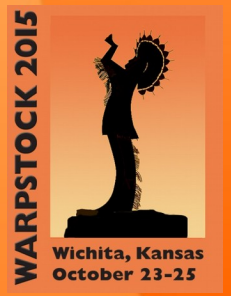
Installation help

- Warpin packages – provided by Alex Taylor are available: http://trac.netlabs.org/ecups/wiki/CupsPort at the time of writing, these are based on CUPS 1.4.8 and Ghostscript 8.71
- ECS 2.2 beta has the option to install as part of the system



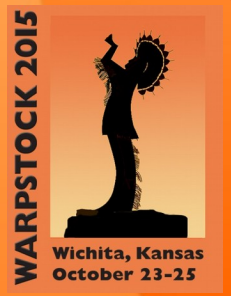
Integrating CUPS with OS/2

- Whilst Qt apps can print natively to a CUPS scheduler, OS/2 native applications cannot.
- A postscript driver is required to be installed on the system, and the output device set to CUPS using the CUPS port driver.
- Cupswiz can assist with this process - http://www.altsan.org/os2/printing/#cupswiz



Printer specific options

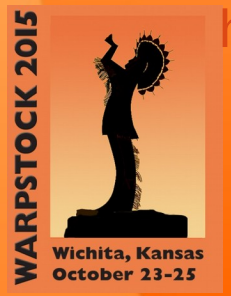
- Printer specific options are defined in a printers PPD (Postscript Printer Definition) file.
- Pin.exe (part of the IBM postscript driver) allows these files to be imported into the Postscript driver so that printer specific features are available in the print settings dialog
- Unfortunately, modern PPD files often crash pin.exe or generate data that is invalid. More work is required to enhance/rewrite pin



Updated pin/postscript driver

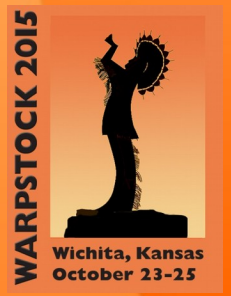
- Through testing, some commands/syntax in modern PPD files that can cause issues with pin have been identified and resolved.
- Recommend use of Alex Taylor's psprint driver which includes fixes from myself and Alex to try and workaround issues. Cupswiz also helps automate this process.
- Psprint is available from:

http://www.altsan.org/os2/printing/#psprint



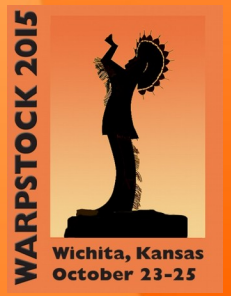
Current issues

- Lack of testing:
  - Lions share of testing has been done by David Mckenna and Pete Brown – most progress in resolving issues has been as a result of their testing and clear feedback on issues
- Lack of documentation:
  - Wiki at http://trac.netlabs.org/ecups has a HowTo and FAQ, but these don't reflect latest available versions.



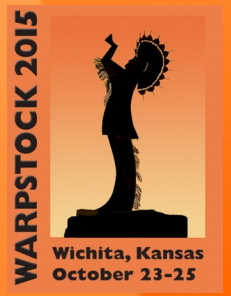
Available Resources

- Ecups mailing list:
  http://dir.gmane.org/gmane.org.netlabs.ecups.devel
- OS2World threads:
  http://www.os2world.com/forum/index.php?topic=631.0
- Bug smedles on #netlabs on freenode
- http://trac.netlabs.org/ecups
- Alex's presentation from WSE2011 -
  http://www.warpstock.eu/uploads/tx_wseevents/ECUPS.PDF



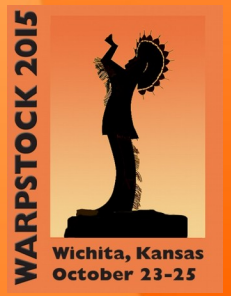
Future Plans

- Current port driver manually executes lpr.exe from CUPS with parameters to print a file, and requires manual configuration
- A future build could poll the network for available printers and automatically populate the server and device name fields. It could also transfer the file to the server using CUPS API's



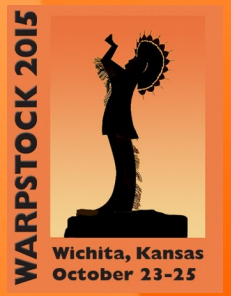
Recommended printers for OS/2

- Device with native postscript support – most likely avoids the need for CUPS, although CUPS can still be of use if the printer doesn't support a protocol supported by OS/2 – eg it needs ipp printing.
- A HP device (with the earlier caveats on devices that require a closed source component)



Conclusion

- CUPS provides a mechanism to support modern printers on OS/2, without direct support from OEMs (although more open source drivers in the style of hplip would help).
- More help is required to test CUPS and help improve the documentation that is available.



Questions

- ???