

CUPS

The Mac OS X Printing System

About...well...me

- John C. Welch
 - Mac Admin for Kansas City Life Insurance
 - Technical Strategist for Provar
 - Writer behind bynkii.com
 - Columnist for MacTech
 - Contributor to MacJournals and Your Mac Life

About this session

- We're here to talk about CUPS
 - While we will talk about how Mac OS X uses CUPS and how to get the most out of CUPS in Mac OS X, this is NOT a complete Mac OS X printing session
 - That topic involves huge parts of Quartz and a lot more time

What is CUPS?

- CUPS
 - The Common Unix Printing System
 - Designed to give Unix, and particularly Linux, the same ease of use that Mac users have gotten from PAP, only over TCP/IP
 - It is the plumbing of the Mac OS X printing subsystem, that allows all that gooey Quartz goodness talk to printers since Mac OS X 10.2
 - 10.1 and earlier used a different system, but those were bad times, and we do not speak of them

What is CUPS?

- CUPS is a Client/Server system that is based around IPP as its network protocol
 - IPP was designed to avoid having to open yet more ports. It uses either HTTP, (port 80), or the IPP port, (port 631 tcp & udp)
 - IPP is an extension to HTTP to provide remote printing support
 - Since it's an independent standard, it's not an *Apple* thing

What is CUPS?

- IPP is *widely* supported
 - Windows 2000 and later
 - Linux
 - Solaris
 - Etc
 - Yadda

What is CUPS?

- One advantage here is that unlike Mac OS 9, you can support multiple printing architectures and protocols without needing totally separate printing systems
 - Along with IPP, CUPS supports PAP, LP(R), SMB, HP JetDirect, and can support any print protocol
 - It supports logging, quotas, SSL, sharing, etc.

What is CUPS?

- CUPS is a modular, extensible system
 - So as new printers, protocols, etc, are needed, they can simply be added to the system as plugins
 - Like most Unix - based printing systems, this assumes everything is postscript.
 - Usually, only Windows has a problem with this
 - You can easily use non-postscript printers though
 - This is why older printer support is easier under CUPS

What is CUPS?

- CUPS ends at the UI level
 - Quartz Imaging, the Printer Setup Utility, etc. are not part of CUPS
- CUPS is what connects the user levels to the printer(s)
- CUPS runs background printing and is why any Mac OS X system running 10.2 or later can be an effective print server

What is CUPS?

- In fact, as of Panther, you can build a better print server from CUPS *without* Mac OS X Server



CUPS

CUPS Concepts

CUPS Concepts

- While CUPS is “standard”, there are still some concepts unique to CUPS that will make using it much easier

CUPS Concepts

- Job: Anything you submit to be printed
 - Jobs have server-unique job numbers, and is assigned a destination, either a printer, or another server
- Class: a group of identical printers or servers that appear as one object
 - This allows for load balancing for printers, servers, or both

CUPS Concepts

- Filters: Filters convert data from what Quartz hands to CUPS (pdf) to what the printer needs.
- Backends: Backends connect CUPS to the various ports, such as USB/IPP/PAP/LPD, etc.
 - Obviously “ports” are not strictly physical in this case

CUPS Concepts

- **Printer Drivers**
 - These are analogous to filters, but are printer - specific
 - Provided by Apple and printer vendors
 - CUPS has some basic drivers that will kill trees, but not optimally
- **Client:** Anything submitting a job to a server

CUPS Concepts

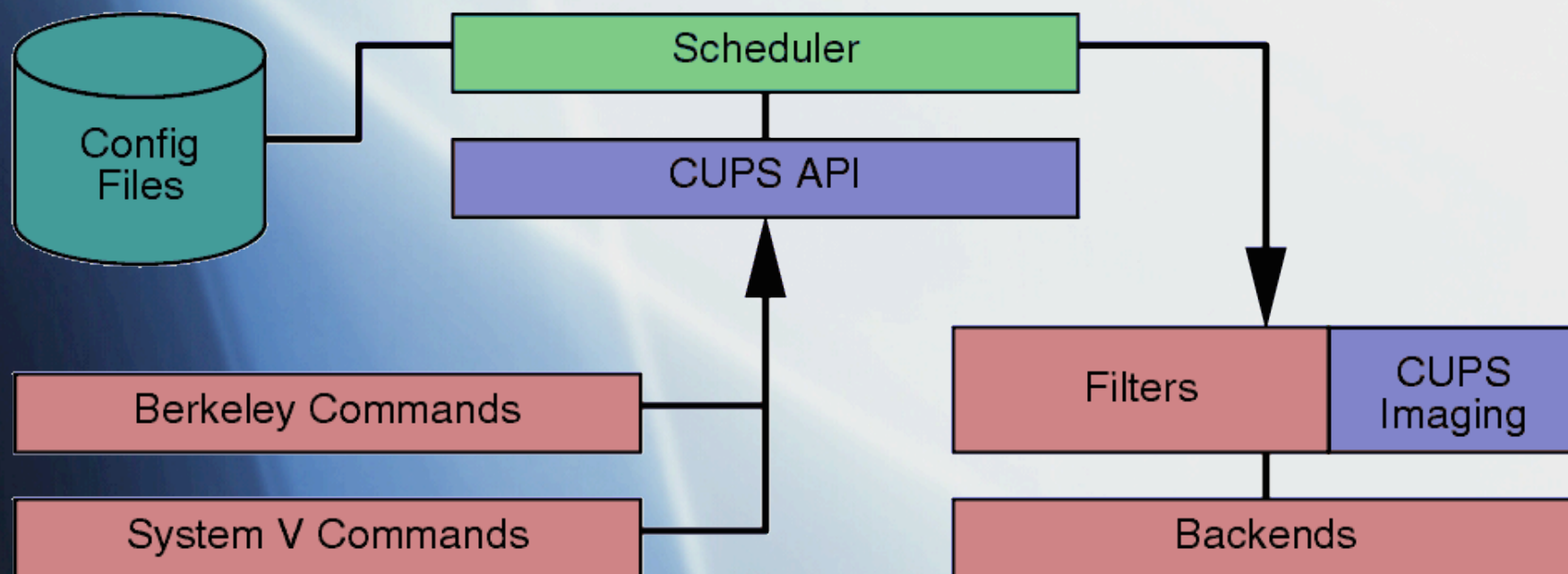
- Server: Anything accepting print jobs
- CUPS is a client-server architecture
 - Even if you only ever print to local USB printers, you're a local client printing to a local server
 - Forgetting this can make CUPS very odd

CUPS Architecture

- CUPS is a modular system
 - Each part of CUPS has a specific set of duties
 - This makes things simpler in many ways, and helps with reliability too
 - It makes updating CUPS a LOT easier
 - This way, we avoid things like the QuickDraw GX debacle

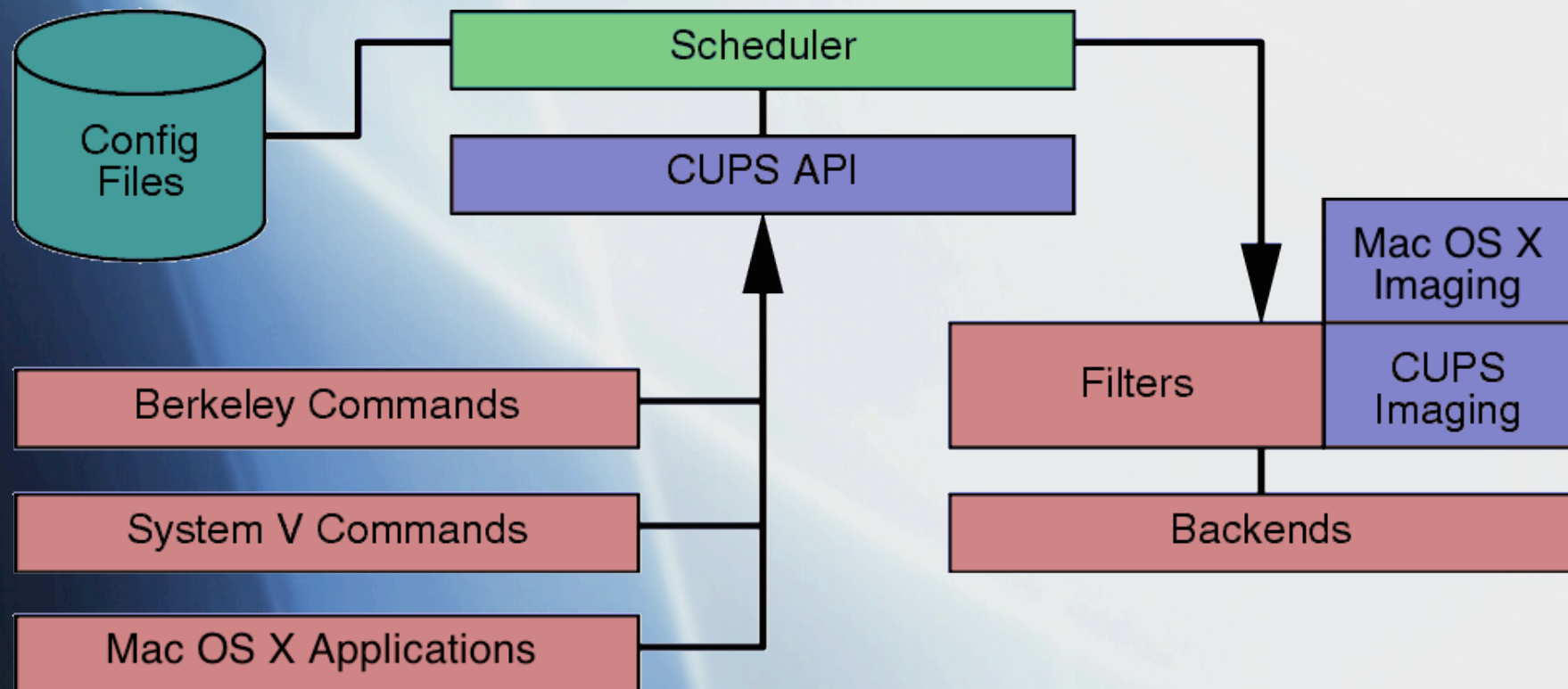
CUPS Architecture

- Basic CUPS diagram



CUPS Architecture

- The Mac OS X version



CUPS Architecture

- Scheduler

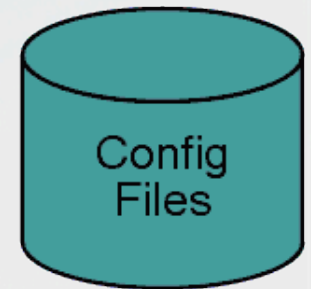


Scheduler

- The center of CUPS
- Basically, it's a specialized HTTP server
 - This comes in handy, as it uses Apache - Style directives in its config files
- This is what you connect to when you go to <http://127.0.0.1:631> on your Mac
- Manages all CUPS operations

CUPS Architecture

- Config Files
 - Like all Unix config files, these control how CUPS works
 - There's a bunch throughout Mac OS X
 - /etc
 - /etc/cups
 - Most of them resemble Apache config files
 - This was deliberate



CUPS Architecture

- CUPS API

CUPS API

- This is the programmatic interface used by applications and the OS to talk to CUPS

CUPS Architecture

- BSD & System V commands

Berkeley Commands

System V Commands

- These are the shell commands that let you control CUPS for printing, printer creation, printer control, etc.
- Two primary locations:
 - /usr/bin/
 - /usr/sbin/

CUPS Architecture

- BSD & System V Commands

Berkeley Commands

System V Commands

- For example you can create printer entries in Print Setup Utility via lpadmin and (optionally) gunzip. You can either do this on large amounts of remote computers via ARD or other tools, or do it on one, then copy the PPD and config files out across the network

CUPS Architecture

■ Filters

Filters

- Used by CUPS to convert files to the correct output format
 - On Mac OS X, the input's always PDF*, so that's easy
 - *this assumes printing from Aqua. If you're doing command line printing, you don't need me to tell you anything
 - PostScript filter for Postscript, PDF, HPGL, etc.
 - A PostScript Raster filter is used for non-PostScript Printers, or the vendor can write their own

CUPS Architecture

- CUPS Imaging
 - Library that provides imaging functionality:
 - Color Management
 - Image Scaling
 - etc.
 - Used by CUPS for filters, the PostScript RIP, and the raster drivers

CUPS
Imaging

CUPS Architecture

- Backends

Backends

- Backends are filters that connect CUPS to the printer port
- As of the CUPS version in Panther, you get Parallel, Serial, USB, LPD, IPP, and JetDirect, (HP Printing)
- SMB printing backends are provided by Samba 2.0.6 and higher

CUPS Architecture

- Log and Spool Files
 - These all live in `/var/` or `/private/var/` for the pedantic
 - Spool files are in `/var/spool/cups/`
 - You need root access to even browse that directory
 - However, if you have one of those zombie spool files that just won't go away, and keeps trying to print, here's where it is
 - Also a good place to look if you're getting tight on disk space, sometimes that directory doesn't get cleaned up correctly

CUPS Architecture

- Log Files
 - These live in `/var/log/cups/`
 - World - readable, but root - writeable, so you are able to use them without needing root access
 - Three main log files used by CUPS
 - `access_log`
 - `error_log`
 - `page_log`

CUPS Architecture

- Log Files
 - Panther Server's print server keeps its log files in a different location:
 - /Library/Logs/PrintService
 - The format of the logs are different, and not as nice to use for things like reports
 - Panther Server also ignores most of CUPS. This makes CUPS logs useless when Panther Server's print service is used, because all jobs are then printed by root

Conclusion

- Hopefully, you have a better idea of what CUPS is and is not
 - For more details, go to <http://127.0.0.1:631/documentation.html> on your Mac OS X box, and check out the various files there, since that's the complete documentation for CUPS
 - There are also mailing lists, etc.
- Q&A?