CUPS Plenary

Michael Sweet, Apple Inc.
April 26, 2016
Boise, ID (HP)
Topics

- Introduction
- We've Moved!
- Developer "Cheats"
- CUPS 2.1 Release History
- CUPS 2.2 Preview
- ippsample Project
- CUPS Future
- Q&A
Introduction

- CUPS is the standards-based, open source printing system developed by Apple Inc. for OS X and other UNIX®-like operating systems.

- CUPS 2.1.x is the current stable branch
  - 2.1.4 coming out soon
  - One more 2.1.x release planned

- CUPS 2.2.x is the current development branch
  - Beta testing will start June/July 2016
  - Probable 2.2.0 release October 2016
We've Moved!

- CUPS is now hosted on Github
  - Repository: https://github.com/apple/cups
  - Web Site: http://www.cups.org

- Mailing lists still hosted by Apple:
  - https://lists.cups.org/mailman/listinfo

- Same license, same contribution policies, etc.

- Security bug reports should be submitted to "security@cups.org" instead of Github
  - Routed through Apple Product Security
CUPS Developer “Cheats”

- `#define _CUPS_NO_DEPRECATED 1`
  - Turns off compatibility defines/typedefs for enums
  - Marks deprecated functions and types as unavailable so you get a compile error instead of a warning

- `#define _IPP_PRIVATE_STRUCTURES 1`
  - Makes ipp_t structure public for existing source code
  - Not a long-term solution - use public API instead

- `#define _PPD_DEPRECATED ""`
  - Turns off PPD warnings
CUPS 2.1 Release History

- CUPS 2.1.0 released August 31, 2015
- CUPS 2.1.1 released November 30, 2015
  - but withdrawn due to a bad tarball/tag
- CUPS 2.1.2 released December 2, 2015
  - USB and IPP printing fixes
  - Security hardening changes
- CUPS 2.1.3 released February 5, 2016
  - General bug fixes
CUPS 2.2 Preview

- Local Print Queues
- Performance Improvements
Local Print Queues

- The scheduler now supports a new CUPS-Creat-Local-Printer user operation
  - Local printers only accept local print jobs and live only as long as needed to print jobs to IPP printers

- The CUPS printing APIs now create local print queues automatically when printing to a discovered printer

- Any uncompleted jobs are lost if the system is rebooted

- Local print queues can be "upgraded" to regular queues by an administrator
Local Print Queues (con't)

- CUPS-Create-Local-Printer Request:
  - Operation attributes group:
    - attributes-charset (charset)
    - attributes-natural-language (naturalLanguage)
  - Printer attributes group:
    - printer-name (name(127))
    - device-uri (uri)
    - OPTIONAL printer-device-id (text(1023))
    - OPTIONAL printer-geo-location (uri)
    - OPTIONAL printer-info (text(127))
    - OPTIONAL printer-location (text(127))
Local Print Queues (con't)

- CUPS-Create-Local-Printer Response:
  - Operation attributes group:
    - attributes-charset (charset)
    - attributes-natural-language (naturalLanguage)
  - Printer attributes group:
    - printer-id (integer(0:65535))
    - printer-is-accepting-jobs (boolean)
    - printer-state (type1 enum)
    - printer-state-reasons (1setOf type2 keyword)
    - printer-uri-supported (1setOf uri)
Performance Improvements

- Restart time worse than cold start time
  - Memory deallocations (free calls) for printer state/attributes dominate restart times
  - Still investigating solutions (always do cold start is one way)

- Get-Jobs with first-job-id can be slow
  - Search time for "job-id" in job list
  - Switching to "first-index" attribute for the web interface (faster and standard IPP attribute defined in PWG 5100.13)
ippsample Project

- New Github project:
  - https://github.com/istopwg/ippsample
- Sample implementations of IPP Client, Printer (server), and Proxy
  - Experimental code
- Based on CUPS code base with same license (LGPL2)
- Printer and Proxy implementations support transforms from PDF and JPEG to PWG Raster and HP PCL
ippsample Programs

- ippfind - general purpose "find" program for printers (as found in CUPS)
- ippproxy - implementation of IPP Proxy for generic HP PCL and IPP Everywhere printers
- ippserver - implementation of IPP Printer/Infrastructure Printer
- ipptool - general purpose program for sending requests and doing tests
- ipptransform - tool for converting PDF and JPEG files into PWG Raster and HP PCL
ippserver

- Enhanced version of the sample code included with CUPS

- Supports previous "single queue" mode like the original sample code, plus a new configuration directory mode that allows for the configuration of multiple queues (IPP Printers) and other settings

- Supports notifications

- Supports transforms (via ipprtransform tool)

- Supports both regular ("direct printing") and Infrastructure Printer ("Cloud printing") modes
ipptransform

- Uses CoreGraphics (OS X) or MuPDF (all) to rasterize files
  - Configurable memory limits (banded output)
- Supports sRGB, sGray, and Black color spaces
- Supports "copies", "media", "media-col", "page-ranges", "print-color-mode", "print-quality", "print-scaling", "printer-resolution", and "sides" Job Template attributes
ippsample Demo
CUPS Future
CUPS Future

- Continue development/investigation of ippsample code
  - launchd/systemd integration
  - Additional auth mechanisms (MutualAuth, OAuth, etc.)
  - System Service implementation, local queues?
  - Release printing proxy
  - 3D extensions support
  - User commands (lp, lpr, etc.)?
CUPS Future

- Additional discovery/directory service support
  - Bring back LDAP support, this time using the standard schema
  - DNS-SD/mDNS enhancements being discussed in the IETF
  - Configuration profiles
Resources

- CUPS Web Site
  - http://www.cups.org/
- CUPS Repository
  - https://github.com/apple/cups
- IPP Sample Code Repository
  - https://github.com/istopwg/ippsample
Q&A