OpenPrinting®

cups-filters, Driverless Scanning, CUPS Snap, ippusbxd, ipp-usb, Avahi

Till Kamppeter – OpenPrinting

5 May 2020

Cups-filters: Introduction

- **cups-filters takes up everything from CUPS which Mac OS X does not need** (CUPS 1.6.x)
  - Started end of 2011 by OpenPrinting, **overtaking most of CUPS' filters**
  - Switched filters over from PostScript-centric to **PDF-centric workflow**
  - **cups-browsed** introduced end of 2012, to introduce **browsing of DNS-SD-advertised remote CUPS queues**, as CUPS dropped its own broadcasting/browsing
  - 9 years of **further development** added things like driverless printing support, clustering, support for common PDF renderers, IPP standards, ...

cups-filters Development: cups-browsed

- **New features:**
  - **Clustering of different printers:** Remote CUPS, network, destination selected by job and options (Deepak Patankar, GSoC 2018)
  - **No download of remote CUPS printer PPDs** (PPD handling in locally created queues will go away when cups-browsed turns Printer Application)
  - Generally **IPP printers** and **remote CUPS** are handled **the same**
  - Local queues are **created/modified in small chunks**, to allow quick reaction to events, like jobs
  - Cups-browsed needs only to know **domain socket** to attach to CUPS, not port

cups-filters Development: cups-browsed

- **Bug fixes**
  - **Keep track of all interfaces** through which a remote printer is advertised and remove it only if it disappears on all interfaces (Issue #136)
  - Many fixes for (potential) **crashers** and **memory leaks**

cups-filters Development: Filters

- **New features:**
  - Support **zero-page input files**, passing on zero-page output, and making nothing be printed, without error output (Issue #117)
  - **pdftoraster**: Switched to **stable Poppler APIs** (Tanmay Anand, GSoC 2019), no use of undocumented Poppler APIs in cups-filters any more
  - Added "**print-scaling**" option with settings "auto", "auto-fit", "fill", "fit", "none" (Issue #108).
  - **mupdftoraster, pdftops**: Improved and simplified MuPDF-based filters
  - **Removed deprecated pdftoijs, pdftoopvp**

cups-filters Development: Filters

- **Bug fixes** (many are fixed by GSoC 2020 student candidates):
  - Improved **handling of margins and media sizes**, more precise selection, use of smaller margins if in doubt
  - Improved handling of **color spaces and modes**
  - Transition between **PPD option names** and **IPP attribute names**
  - **CJK Text** Printing
  - Continuous adaptation to **changes in CUPS and Ghostscript**

cups-filters Development: libcupsfilters

- **New features:**
  - Provide **get-printer-attributes IPP request function** to both cups-browsed and driverless, using "all" and "media-col-database" like CUPS and falling back to IPP 1.1 and "all"-only, to work with the maximum amount of printer models with various firmware bugs (Issue #22).

cups-filters Development: Next Steps

- **License change** from [many, many different licenses] to **Apache 2.0 + (L)GPL2 exception**, the same license as CUPS and PAPPL, most contributors have agreed, no one rejected.
- Make all **filters working without PPD**, with IPP options. Exceptions are filters especially made for PPDs, as foomatic-rip.
- **Move more functionality into the libcupsfilters library**, so in Printer Applications filter chains could be implemented as library function calls instead of external executable calls
- **Add build options** for no-PPD-supporting CUPS, raster-only Printer Applications, no cups-browsed. Snaps will often have their own cups-filters and then a very limited part of it.

cups-filters Development: Next Steps

- Move all the **PPD-related functions of the CUPS library** (libcups) into a **new libppd**, to allow converting legacy drivers into Printer Applications with minimum need of creating new code
- Turn **cups-browsed** into a **Printer Application**

We have agreed on **not to rename cups-filters**. It will only get a **new generation number (versions 2.x.y)**. cups-browsed will perhaps get spun out into its own project.

Driverless Scanning

- **3 Standards**
  - **IPP Scan**, open PWG standard
  - **eSCL**, proprietary, from HP, used by Apple AirScan
  - **WSD**, from Microsoft and W3C
- All are mainly intended for **multi-function printers**
- **eSCL** and **WSD** one **already available** in devices
- **2 SANE drivers** for eSCL: “escl” from Thierry Hucahrd and “airscan” from Alexander Pevzner, I have made both joining forces and work together on one project
- Alexander has added **WSD support** and will add **IPP Scan**
- **eSCL** also works via **IPP-over-USB** (ippusbxd, ipp-usb)

Sandboxed Scanner Drivers

- **Current situation: SANE**
  - Scanner driver (**SANE backend**) is shared library
  - Scanning app (**SANE frontend**) links backends dynamically
  - **To add a driver it needs to be dropped in backend dir** => not good for sandboxed packaging
- **New scanning environment: IPP Scan**
  - Scanner drivers in **Scanner Applications, emulating IPP scanner**
  - Scanning app is IPP Scan client
  - Legacy: App uses IPP Scan SANE backend, SANE drivers enclosed in legacy Scanner Application
  - Alexander Pevzner will mentor student on Scanner Applic.

IPP-over-USB: ippusbxd and ipp-usb

- **ippusbxd simple TCP ↔ USB packet relay**, leaves packets in USB buffer on client closing connection, packets re-appear in next connection messing it up → **Creation of alternative ipp-usb** by Alexander Pevzner
- Ipp-usb written in **Go**, as Go has **sophisticated HTTP library** to read out buffer on closed connection
- Ipp-usb **works perfectly**, esp. web admin interface
- **Chrome OS not accepting software in Go** due to high memory footprint
- Thierry Hucahrd improved ippusbxd to **poll device for getting complete DNS-SD records**
- **eSCL scanning** and **IPP Fax Out** works with both daemons

CUPS in a Snap

- Development of a **Snap containing CUPS**, cups-filters, cups-browsed, Ghostscript, QPDF
- **Complete CUPS printing stack** in a Snap
- **No support for classic drivers**, as filters and PPDs cannot get dropped into Snap’s file system
- Sorting out all the problems with Canonical’s Snap gurus on the snapcraft.io forum
- **First production release** probably before Ubuntu 20.10, but needs also **conversion of all legacy printer drivers** to Printer Applications
- Feature request to Snap Store: **Search Snaps by hardware signature**, to easily install driver Snaps

Printer Applications

- With Michael Sweet introducing PAPPL **Dheeraj Yadav's GSoC 2019 project printer-application-framework discontinued**
  - Thanks anyway to Dheeraj, but PAPPL provides a complete base for Printer Applications.
- In this year's GSoC **continuing Printer/Scanner Application work based on PAPPL**:
  - Gutenprint as sample driver, legacy driver conversion, snapping Printer Applications, work on cups-filters to use code of the existing filters in Printer Applications, ...
- If we get accepted into this year's GSoD, we will let a technical writer write a **tutorial on designing and packaging printer and scanner drivers** for us.

Avahi

- Patch for **DNS-SD advertising local services** (on localhost, loopback interface) on **local machine only**
- Needed for **Printer Applications** and **IPP-over-USB**
- Finally! 3 years after I have created the patch it **got adopted into the upstream code** and version 0.8.0 with it released!
- **Thanks to Rithvik Patibandla** for contributing to the patch.
- Also **thanks to Michael Sweet** for helping here.

Questions / Comments