



Contribution ID: 258

Type: **not specified**

Printer/Scanner Applications - The new format for printer and scanner drivers

Tuesday, 10 September 2019 12:00 (30 minutes)

The upstream author of CUPS has deprecated the classic way to implement printer drivers, describing the printer's capabilities in PPD (PostScript Printer Description) files and providing filters to convert standard PDLs (Page Description Languages) into the printer's own, often proprietary data format. With the background of PostScript not being the standard PDL any more, most modern (even the cheapest) printers being IPP driverless printers (using standard PDLs and printer's capabilities can get polled from the printer via IPP), and modern systems using sandboxed application packaging (Snappy, Flatpak, e.t.c.) the new Printer Application concept got introduced.

A Printer Application is a (simple) daemon emulating a driverless IPP printer (can be in the local network but also simply on localhost). Like a physical printer this daemon advertises itself via DNS-SD, takes get-printer-attributes IPP requests and answers with printer capability info so that the client can create a local print queue pointing to it, takes print jobs, converts them to the physical printer's data format and sends them off to the printer.

This way the client "sees" a driverless IPP printer and the Printer Application is the printer driver (printer-model-specific software to make the printer work). So with the driver being connected to the system's printing stack only via IP and no consisting of files spread into directories of the printing stack, both the printing stack and the driver can be in separate, sandboxed applications, provided as sandboxed packages in the app stores of the appropriate packaging systems (Snappy, Flatpak, e.t.c.). And this allows the driver not depending on a specific operating system distribution any more. A printer manufacturer only needs to make a driver "for Snappy", not for Ubuntu Desktop/Server, Ubuntu Core, Red Hat, SUSE, e.t.c. making development and testing much easier and cheaper.

And one can even go further: As the Printer Working Group (PWG) also has created an IPP driverless scanning standard, we can create Scanner Applications emulating a driverless IPP scanner and internally using scanner drivers, like SANE, to communicate with the scanner, allowing the same form of OS-distribution-independent sandboxed driver packages for ANY scanner, especially also stand-alone scanners without printing engine.

For multi-function printers one could also have a combined Printer/Scanner application. Any such Printer and/or Scanner Application can even provide an IPP System Service interface to allow configuring the driver without need of specialized GUI applications on the client.

We have a Google Summer of Code student working on a framework for Printer Applications, to convert classic printer drivers into Printer Applications to kick off the new standard.

In this session we will present the new format, its integration into real life systems, problems we got into during the work with our student, and how to present it to hardware manufacturers as the new way to go.

I agree to abide by the anti-harassment policy

Presenter: KAMPPETER, Till

Session Classification: Open Printing MC