Some Notes about Printer Drivers

Till Kamppeter
OpenPrinting

THE LINUX FOUNDATION

Make Printers Working on more than just Windows and Mac

- **IPP Everywhere**
  - This is the best and recommended way
  - Printer advertises itself by Bonjour
  - Detailed capabilities can be polled by IPP
  - Printer understands known languages:
    - PWG Raster (required)
    - PDF (optional)
    - PostScript (optional)
    - JPG (optional)
    - TIFF (optional)
  - → No model-specific software required
- **Auto-downloadable distribution-independent driver packages**
  - Easy setup, also if the distribution does not support the printer explicitly
  - With Internet access printer can get set up everywhere

Auto-Downloadable Printer Driver Packages

- **Problems**
  - IPP Everywhere printers not yet available
  - There will perhaps always be (host-based) cheapo printers
  - Distributions do not ship all available printer drivers
  - Free drivers from upstream need to be compiled by users -> driver installation too complicated for unexperienced users
  - Manufacturers make packages only for a few major distributions
  - Driver packages often difficult to find on manufacturer's web sites
  - Testing/packaging effort for manufacturers and driver developers too high to ship binary driver packages for all distributions
- **Existing Infrastructure**
  - **OpenPrinting database** (former linuxprinting.org), central database for printer/driver info
  - **LSB** provides tools and infrastructure to create **distribution-independent binary packages**

Auto-Downloadable Printer Driver Packages

Solution

Distribution-independent printer driver packages

- Based on **LSB 3.2** or any later version
- Using **CUPS**, **Ghostscript** (with IJS, CUPS Raster and OpenPrinting Vector interfaces), **Perl**, and **foomatic-rip** which is in any distribution (and is required since **LSB 3.2**)
- Installing everything in **`/opt/<supplier>/`** to avoid conflicts with distribution
- Linking PPDs to **`/usr/share/ppd/`**
- Discovering system directory/file locations at install time (maintainer scripts: pre/post (un)install) and symlinking system files appropriately
- Make packages part of OpenPrinting database, so that they can be easily found
- Infrastructure for automatic package lookup, download, and installation through the internet by printer setup tools

Advantages

Distribution-independent

One package for Linux, instead of one for Red Hat, one for SuSE, one for Ubuntu, ...

Binary packages

User does not need to compile, system is also suitable for closed-source drivers

Same installation method for all driver packages

A printer setup tool can easily install them automatically

One look-up location at the OpenPrinting site

Easy to find for both humans and printer setup tools

Advantages

Driver query API for printer setup tools

All needed info available: License, supplier, support contact, print quality indices. So the setup tool and the user can easily find the driver suiting best for him.

Distributions look up drivers at OpenPrinting

Distributions do not need to support all printer models

So drivers newer than the distro are available, for updates and for new printer models.

Problems

LSB does not have enough requirements for system software

- Complicated maintainer scripts needed in package to access system resources, scripts now provided by an **RPM macro set** for easier packaging
- **SANE** needed in LSB, for printer/scanner multi-function devices. For now added SANE to LSB DDK, SANE drivers compiled as LSB binaries work with the SANE of any distro. **From LSB 5.0 on SANE is part of LSB**.

Internet access needed to make use of the packages

- Distributions should ship some drivers for most common printers
- Manufacturers should ship them also on their CDs
- Most Linux environments have internet access

Available Driver Packages

For users (test the packages)

- Driver packages are marked on http://openprinting.org/driver_list.cgi
  - Epson inkjet drivers
  - PPD packages for PostScript printers
  - Gutenprint, SpliX, min12xxw, lm1100
- Easy setup on current distributions (on Ubuntu automatically done by system-config-printer and GNOME printer setup tool)
  - Download and install driver package
  - Set up printer

Web Query API

For developers of printer setup tools/distributions

- Web API for querying the OpenPrinting database and downloading driver packages
  - Queries by calling a CGI script via URLs like
    - http://openprinting.org/query.cgi?type=manufacturers
    - http://openprinting.org/query.cgi?type=printers
    - http://openprinting.org/query.cgi?type=drivers
  - Modifier for output format (Text or XML)
    - http://openprinting.org/query.cgi?type=printers&format=xml
  - Modifier to show complete entry and not only name
    - http://openprinting.org/query.cgi?type=printers&moreinfo=1
  - Filters (to be added as “&<name>=<value>” to the URL)
    - printer, make, model, driver, onlydownload, onlyppdfiles, onlydriverpackages, sourcedriverpackages, nobinarydriverpackages, onlynewestdriverpackages, architectures, noobsoletes, onlyfree, onlynonfree, onlymanufacturer

Web Query API

- Printers can be selected by supplying the **IEEE-1284 device ID**
  - to find drivers for an **auto-detected printer**
- Printers can also be searched by a **model name not exactly matching the database** (fuzzy matching) or by **name fragments**
  - to find an auto-detected printer without the device ID being in the database
- The output can be restricted to only entries with
  - downloadable drivers and/or PPDs
  - selected architectures
  - only manufacturer-supplied drivers
  - only free or non-free drivers
  - only current version
  - ...
- The output contains for each driver
  - Driver name
  - Driver package/PPD URL(s)
  - License, free/non-free?
  - Supplier, manufacturer/third-party?
  - Support contact, commercial/voluntary?
  - Output/quality/performance ratings for different printing tasks

Web Query API

Typical steps for a printer setup tool

- Auto-detect printer, get IEEE-1284 device ID
- Check for locally installed drivers and versions
- Query OpenPrinting database for available drivers for system's architecture
- Check whether driver download is needed, either due to no available local driver or downloadable driver being newer
- For each suitable downloadable driver show an info panel to the user, with supplier, license, support contact/level, quality ratings
- Let user choose driver/confirm download
- If user agrees, download and install driver
- Set up print queue

Possible extra functions

- Let user configure preferences: Only free drivers, only manufacturer-supplied, ...
- Let distribution's build server download all available source RPMs and rebuild them to distribution-specific packages

The LSB DDK

For developers of printer drivers, especially manufacturers

- The **LSB Driver Development Kit** (DDK) provides all tools and resources to develop distribution-independent printer driver packages
- The **LSB DDK** contains:
  - The RPM macro set with macros
    - to facilitate installing everything in **`/opt`**
    - to generate the maintainer scripts (pre/post (un)install) to link the system files to the correct system directories
    - to put absolute paths for print filters installed in /opt into the PPDs
    - to rename and re-order the PPDs according to the planned LSB 3.2 requirements.
  - The printer driver development HOWTO:

http://www.linux-foundation.org/en/OpenPrinting/WritingAndPackagingPrinterDrivers

Developing Drivers

- Compiling drivers into Ghostscript is obsolete, use the following renderer interfaces for driver plug-ins:
  - CUPS Raster
  - OpenPrinting Vector
  - IJS
- Take care that your driver can be installed in **`/opt`**
- CUPS backends, extra daemons, auxiliary programs (nozzle cleaning, ...) are supported
- SANE was added to the LSB DDK, which allows building LSB binaries of scanner drivers. They work with the SANE packages of all distros. SANE searches drivers always in **`/usr/lib/sane`**.

Packaging drivers

- Install the build environment
  - LSB Build Environment chroot
  - Add LSB DDK packages CUPS, Ghostscript, foomatic-rip
  - Install RPM macro set of LSB DDK
- Build RPM packages inside the build environment chroot
- Use the macro set:
  - Let everything go into **`/opt/<supplier name>`**:

```
%define supplier <supplier name>
%define drivername <driver name>
%install_into_opt
```

  - Put absolute paths to filter calls in the PPDs and arrange the PPDs according to LSB 3.2:

```
%adjust_ppds
```

  - (In post-install script) Link the PPD directory into **`/usr/share/ppd`**:

```
%set_ppd_links
```

The LSB DDK

Statically link libcups and libcupsimage

`%uses_libcups_and_libcupsimage`

(post-install script) Symlink all CUPS files (backends, filters, mime rules) into appropriate directories

`%set_cups_links`

Set executable and man paths so that the executables and man pages in **`/opt`** get found

`%has_bin_executables`
`%has_sbin_executables`
`%has_manpages`
`%set_opt_paths`

There are also macros for

- Restarting CUPS
- Setting up and starting services
- Building PPDs from Foomatic XML data
- Linking PAM modules
- ...

Testing drivers

- Test the LSB compliance of the executables with
  `lsbappchk <name of the executable with full path>`
  Link libraries statically if they are not covered by the LSB
- Check Adobe-compliance of PPD files with
  `cupstestppd <name of the PPD file with full path>`
- Install the driver package. Should work like described in the user instructions
- Try to set up print queues and to print

More Info

General info

http://www.openprinting.org/

For developers

http://www.linux-foundation.org/en/OpenPrinting/Development

http://www.linux-foundation.org/en/Developers

For driver developers

http://www.linux-foundation.org/en/OpenPrinting/WritingAndPackagingPrinterDrivers

For developers of printer setup tools

http://www.linux-foundation.org/en/OpenPrinting/Database/Query

Available driver packages

http://www.openprinting.org/driver_list.cgi

How to install driver packages

http://www.linux-foundation.org/en/OpenPrinting/Database/DriverPackages