



Printing and the Linux Standards Base (LSB)

Till Kamppeter

OpenPrinting Manager, The Linux Foundation



Printing Requirements in the LSB: Goals

- **For printer manufacturers:**
 - Easy way to make distribution-independent driver packages
 - Less effort for packaging and testing
 - Printers simply work with “Linux” easy to install for every user
- **For application developers:**
 - Easy implementation of printing functionality
 - Common interfaces in all distributions
 - Access to full functionality of all printers
- **For distribution vendors:**
 - One central, shared printer driver repository
 - No necessity of shipping all drivers
 - Less effort and expenses on maintaining driver packages



- **Additions to the LSB 3.2**
 - **For application developers:** Interfaces to give full functionality printer access
 - **For printer driver developers/printer manufacturers:** Interfaces and facilities to build distribution-independent printer drivers
- **Additions to OpenPrinting web services**
 - API for printer setup tools to be able to find, choose, and download driver packages (**DONE**)
 - First set of driver packages for testing (for LSB 3.1 + CUPS + Ghostscript + foomatic-rip, **DONE**)
- **LSB Driver Development Kit**
 - Tools: LSB packages of CUPS, Ghostscript, foomatic-rip (**DONE**)
 - RPM macro set (**DONE**)
 - Driver development and packaging HOWTO (**DONE**)
 - Planned: LSB packages of CUPS DDK and SANE



- **Testing and Certification program**

- To be integrated with the LSB and the OpenPrinting database
- Testing program to be developed with manufacturers, distribution vendors and certification organizations
- Infrastructure for driver packages reached a working state -> Make testing and certification main subject of the Montreal Printing Summit



LSB 3.2: Application-Level Standards

- **Interfaces from the CUPS convenience API**
 - This gives access to all printers, their options, and to authentication functionality
 - Functions: cupsAddOption, cupsEncryption, cupsFreeDests, cupsFreeOptions, cupsGetDest, cupsGetDests, cupsGetPPD, cupsLangDefault, cupsLangEncoding, cupsLastError, cupsMarkOptions, cupsPrintFile, cupsServer, cupsSetDests, cupsSetPasswordCB, cupsSetUser, cupsUser
 - See <http://www.linux-foundation.org/en/CUPS>
- **GUI Interfaces**
 - From principal desktop toolkits Qt and GTK
 - LSB 3.2 will only require **Qt 4.0**
 - GTK 2.10.x not in all current distros -> LSB 4.0
 - Common printing dialog to interface via Portland and to be designed by OpenPrinting and OpenUsability -> LSB 4.0+



LSB 3.2: Driver-Level Standards

- **Renderer/driver interfaces**

- IJS (version 0.35)
- CUPS Raster API (CUPS 1.2.x)
- OpenPrinting Vector (version 0.2 + UPDF support required for drivers and no text/font support allowed in drivers)

- **Ghostscript**

- Required output devices: “ijs”, “cups”, “opvp”, “pxlmono”, “pxlcolor”

- **foomatic-rip**

- Universal filter for driver integration

- **Directory for PPD files**

- The printing system must search **/usr/share/ppd** and subdirectories for the available PPD files
- Packages should symlink their PPD directories into this directory



Problems with OpenPrinting Vector

- **Current distributions ship version 0.2 with these problems (fixed in 1.0):**
 - OPVP 0.2 does not require UPDF properties (current drivers do)
 - OPVP 0.2 supports text/fonts partially, 1.0 not (therefore text/font support of 0.2 excluded in LSB 3.2)
 - Name space change between 0.2 and 1.0
- **Text/font support will come in LSB 4.0+**
 - As soon as it gets re-introduced in OPVP 1.x



- **CUPS**

- API Documentation/Specification (Mike Sweet)
- Test Coverage

- **CUPS Raster Format**

- Documentation/Specification
- Test Coverage

- **IJS API**

- specification exists but need tests (Glen Petrie/Epson will contribute)

- **OPVP API**

- Need to subset OPVP 0.2 specification (OpenPrinting Japan)
 - Can we simply include OPVP 1.0 by reference since OPVP 0.2 and OPVP 1.0 are compatible?
- Test Coverage (multiple test suites exist, OpenPrinting Japan will contribute)



- **Ghostscript (The Linux Foundation)**
 - May need help with functional tests (i.e. input provided to driver produces expected output)
 - George Liu (Ricoh) will contribute pxlmono and pxlcolor specifications/tests
- **foomatic-rip (Till Kamppeter)**
 - Specification
 - Test Coverage



- **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

-
-
-