Technical Goals

Open Printing
(Free Standards Group)

Doc #: FSG102-2b

Version   0.1

Document Information

This page provides a sequential record of changes for a multi-page document. All pages shall carry the same revision letter as shown on this page.

Title:	**Open Printing Technical Goals Proposal**
Written By:	Shawn Pratt
Doc #:	FSG102-2b

REVISION HISTORY

Rev	Revision Description	Date	Approved by:	MF
0.10	Draft	5/25/02		

File Name:	**OP TG (FSG102-2b)**
File Format:	**PDF – viewing, DOC – archival & development**
Document Status:	**NOT EXITED**

This document defines the proposed Standards Working group position and plan for a Linux Standard on printing.

The information in this document is subject to change without notice.

If updates and changes need to be made to this document, please contact the Free Standards Open Printing group chairperson. Currently that is:

Shawn Pratt
Hewlett-Packard
11311 Chinden Blvd., MS 235
Boise, ID 83714
Phone: 208-396-4628
Fax: 208-396-5161
Email: Shawn_Pratt@hp.com

Table of Contents

Overview

This document provides the general technical overview of the problem and proposed solutions. This document will cover the following areas:

- ❖ General description of problem,
- ❖ Brief abstract of the proposed solution,
- ❖ Current existing solutions,
- ❖ Current existing projects/working groups related to printing,
- ❖ Companies and organizations who would benefit from a print standard,

System Overview / Components Affected

Customer Models

The Linux environment printing comes under several customer models:

Network of One (direct connect)
Home/Small Office Network
Campus/Metro/Corporate Network
Internet
Mobile Devices

User Needs

Each model has similar and different needs:

Network of One:
- ➢ Resources are local
- ➢ Direct access to device

Home/Small Office Network
- ➢ Printers may be shared
- ➢ One physical network
- ➢ Homogeneous
- ➢ Replicated configuration
- ➢ Replicated state
- ➢ Broadcasting used for discovery
- ➢ Legacy protocols limit functionality

Campus/Metro/Corporate Network
- ➢ Heterogeneous
- ➢ Multi-protocol
- ➢ Translation loses functionality
- ➢ Multiple Administrators
- ➢ Multiple locales

Internet
- ➢ No managing authorty
- ➢ Systems may be firewalled
- ➢ Information is spotty

Mobile Devices
- ➢ Systems may access multiple networks
- ➢ Need to discover resources dynamically
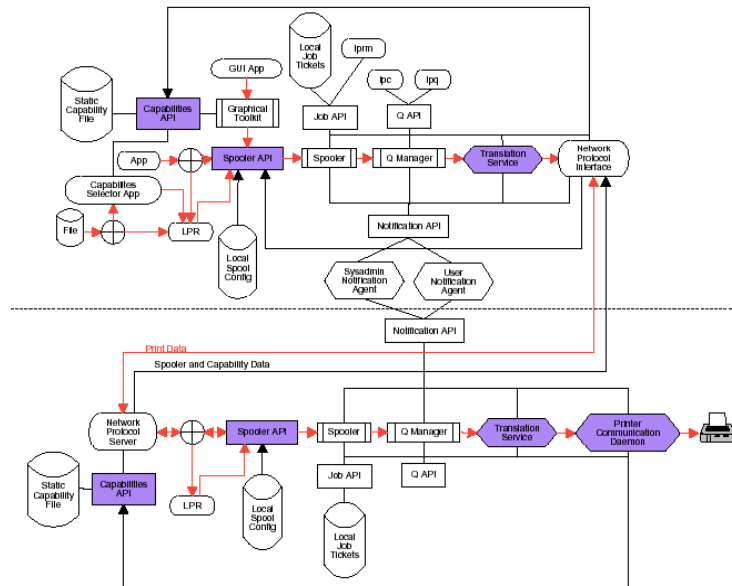- ➢ Need to use multiple protocols

Overall Architectures

Within the Linux operating system, the architectures to support these models fall into two architectures:
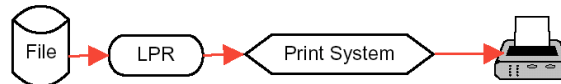Basic Print System architecture
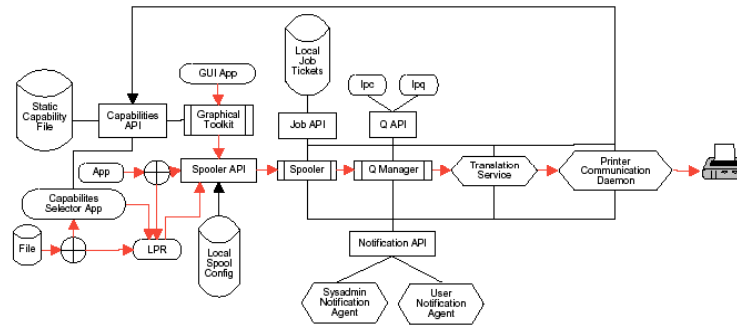
Network Print System architecture

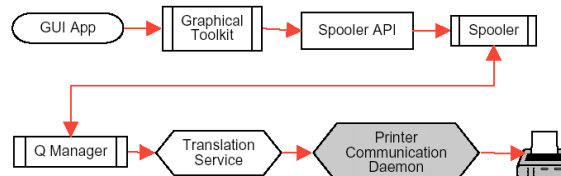Given these architectures, the identified scenarios for printing within the Linux print system are:

1. Command Line LPR

2. Submitting Job via API

3. GUI Printing

Standard Definitions (Goals)

Standards need to be generated around the following areas:

Desktop
- ❖ *Device Discover* – Ability to identify the device and establish a connection for printing to said device.
- ❖ *Spooler* – Ability for applications to enumerate the device and submit jobs.
- ❖ *Capabilities* – Ability to understand what the capabilities of the device are.
- ❖ *Queue* – Ability to fetch print queue information.
- ❖ *Print Job* – Ability to fetch job information or to act on a job already in the process of being printed.
- ❖ *Notification* – Ability to provide information back to the system about the state of a job.
- ❖ *Basic Print Support* – Ability to provide a consistent set of print support regardless of the connection type or the print device. Consistent refers to a consistent page format given paper size, resolution, paper type, paper handling, and graphics output.
- ❖ *Content Rendering* – Ability to send document data (in the form of raster or PS) with a job description file (job ticket) to a print server. This print server represents a particular printer. It establishes the printing process with the printer, and collects and reports status back to the initiator.

Network
- ❖ *Network Queue* – Ability to provide print queue information to a non-Linux client (Samba interface to a Windows client).
- ❖ *Dynamic Discovery* – Ability to discover a device and its capabilities "on the fly".