Printer Setup Tool: How it works currently

- **Printer setup tools**
  - CUPS command line tools: `lpadmin`, `lpinfo`, `lpstat`
  - `system-config-printer` – GUI
  - GNOME Control Center – Print module – GUI
  - `cups-browsed` – daemon – Automation of setup

- Tools **control CUPS**, the running `cupsd`
  - List available printers and drivers and create print queues
  - List queues and jobs
  - Modify queues
  - Server settings: Owner/everyone can cancel jobs, debug mode, ...
Printer management in the New Architecture

• We assume any form of the New Architecture
  - The CUPS Snap - OR -
  - CUPS 3.x or newer
• All Printers are driverless IPP printers, native or Printer Applications
• CUPS auto-creates virtual queue for each IPP printer → No manual queue creation required
• CUPS fully automatic → Admin action moves to the IPP printers
• Tasks
  - List IPP services
    • Buttons to web admin interfaces, IPP System Service, ...
  - Discover non-driverless printers
    • Find Printer Applications, local and in Snap Store
Printer Setup Tool: GUI Design

• **Similarities** between old and new
  
  - **Main Window**
    • Old: List CUPS queues, buttons/pop-up to modify
    • New: List IPP devices, buttons to web IF/IPP System Service
  
  - **Add-Printer Window**
    • Old: List printer devices and drivers, create CUPS queue
    • New: List non-driverless printers, install Printer Application, open Printer Application’s web interface
Printer Setup Tool: GNOME Control Center

- **Support for classic CUPS AND New Architecture**
  - No hard dependency between GNOME and CUPS versions
  - Current CUPS already supports IPP services, Printer Applications, ...

- **Main view**
  - CUPS queues with “Set options”, “Change driver”, “Remove queue”, ...
  - IPP service with “Open web admin interface”
  - IPP: Group entries of same hardware device/Printer Application

- **“Add Printer” dialog**
  - Discover non-driverless printers
  - Search for both classic drivers and Printer Applications
Main discussion with drawings by Mohit Verma and Elio Qoshi
https://github.com/vermamohit13/GSOC_2022_Summary/issues/1

Drawings by Mohit Verma
- Main View
  https://drive.google.com/file/d/1hPpdW0icnZIE1njhw6jRFj9YfmsO70pB/view?usp=sharing
- Add Printer
  https://drive.google.com/file/d/1wg1xxPqM2C2K0H__MTAvzRiWDj8fIJgo/view?usp=sharing

Screencasts by Mohit Verma
- Main View
  https://drive.google.com/file/d/1LYw0T150sV3o4vJkqScmAFz-ThMIsgb/view?usp=sharing
- Add Printer
  https://drive.google.com/file/d/1eSJinN_NxyimeTPr_ZQDc0omeI-0lZwH/view?usp=sharing
New non-driverless IPP...

User clicks on an entry

Setup Successful

Web Interface
G-C-C “Printers” – Main View – Mohit
Select Printer Application UX

12 Jun 2023 - Ubuntu Desktop

Select an already installed printer application (default selection)

Select printer application from the internet (results dynamically populated)

- Specify printer name more semantically in the title bar
- Radio button selection for selecting preferred method to add printer
- Dropdown to select from a list of installed printer application
- Set up printer automatically based on selection
- Configure printer via web interface of the printer application
- List of available printer applications online are automatically populated here as soon as the 2nd radio button is selected
- Set up actions disabled as there are no applications selected yet
Select printer application from the internet (application selected)

Once list is populated, a printer application can be selected

Set up printer automatically based on selection

Configure printer via web interface of the printer application

Select a classic driver

Driver GUI will be initiated

No web-interface configuration possible for classic drivers
GTK library – Separate Printing API?

- Matthias Clasen posted issue to suggest **moving printing API into separate library** or even **separate project**:
  
  https://gitlab.gnome.org/GNOME/gtk/-/issues/5816

- Matthias started work in this merge request **moving printing API code into separate directory** (already **merged**):
  
  https://gitlab.gnome.org/GNOME/gtk/-/merge_requests/6067

- Matthias suggests to **keep a minimum print API**:
  
  https://gitlab.gnome.org/GNOME/gtk/-/issues/5562
GTK library – Separate Printing API?

• Why separating?
  - **Size** of printing code compared to total size of GTK library?
  - **Sophisticated print API rarely needed?**
    • Few apps which print?
    • Minimum API good enough?
  - **Maintainership?** Should be done by separate maintainers?

• Dependencies
  - Dependencies on **print technologies** (libcups, cpdb-libs) are print backends.
    • Are the print backends needed for both (included) simple API and (separate) sophisticated API? Do we need new, separate CPDB implementation for the simple API?
  - **Further dependencies:** Rendering? Filtering?