Study of the Print Channel Monitor Interface

Kentaro IDE

<ide.kentaro@exc.epson.co.jp>

OpenPrinting WG Japan/Asia

Seiko Epson Corp.

15-17 November 2004

Agenda

- Print Channel Monitor
    - Background
    - Requirement
    - APIs

Print Channel Manager

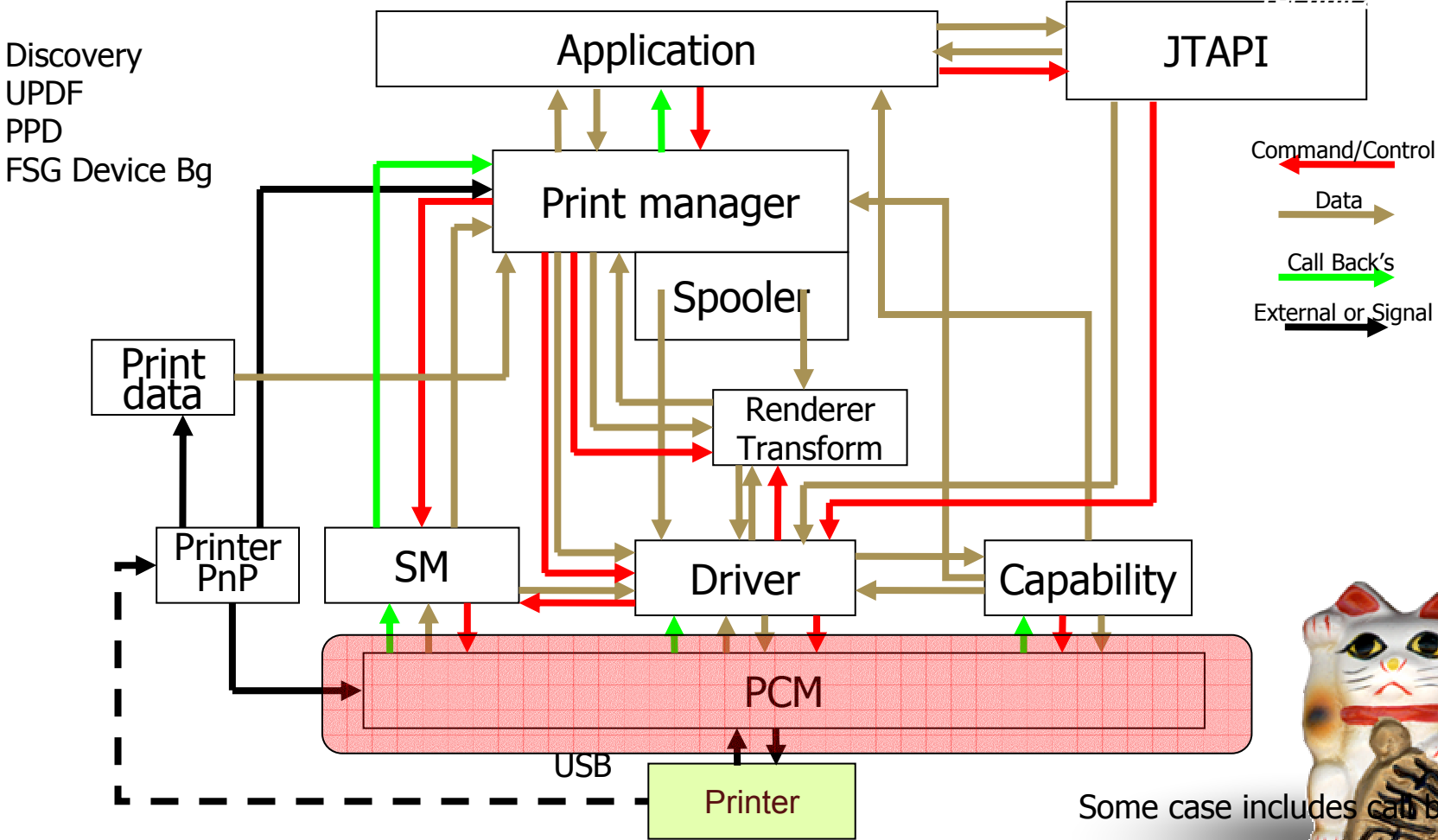# Background

- PCM will wrap the difference of physical layer and Vender depended protocol
  - Upper layer do not need to care about the difference of physical layer like USB, Parallel, Ethernet.
  - Wrap the difference of transaction protocol used by printer venders.
    - Port control between status port and data port have deep dependency with Vender specific protocol
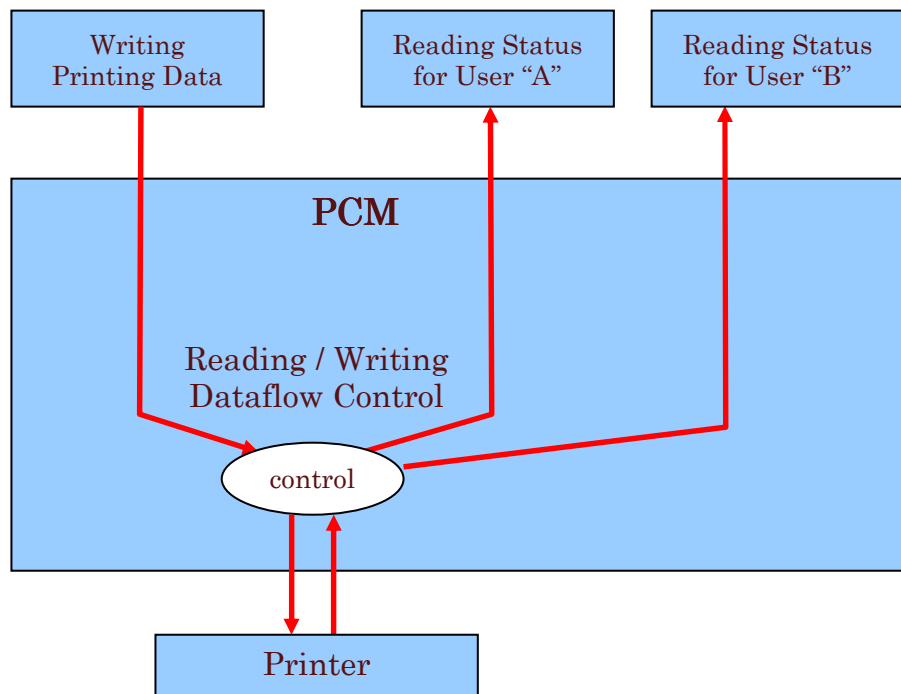
Background

Discovery
UPDF
PPD
FSG Device Bg

Command/Control
Data
Call Back's
External or Signal

Some case includes call back

Requirement

❀ PCM present port management functions

Writing Printing Data		Reading Status for User "A"		Reading Status for User "B"

PCM

Reading / Writing Dataflow Control

control

Printer

- Read/Write dataflow control
- Port arbitration with plural port
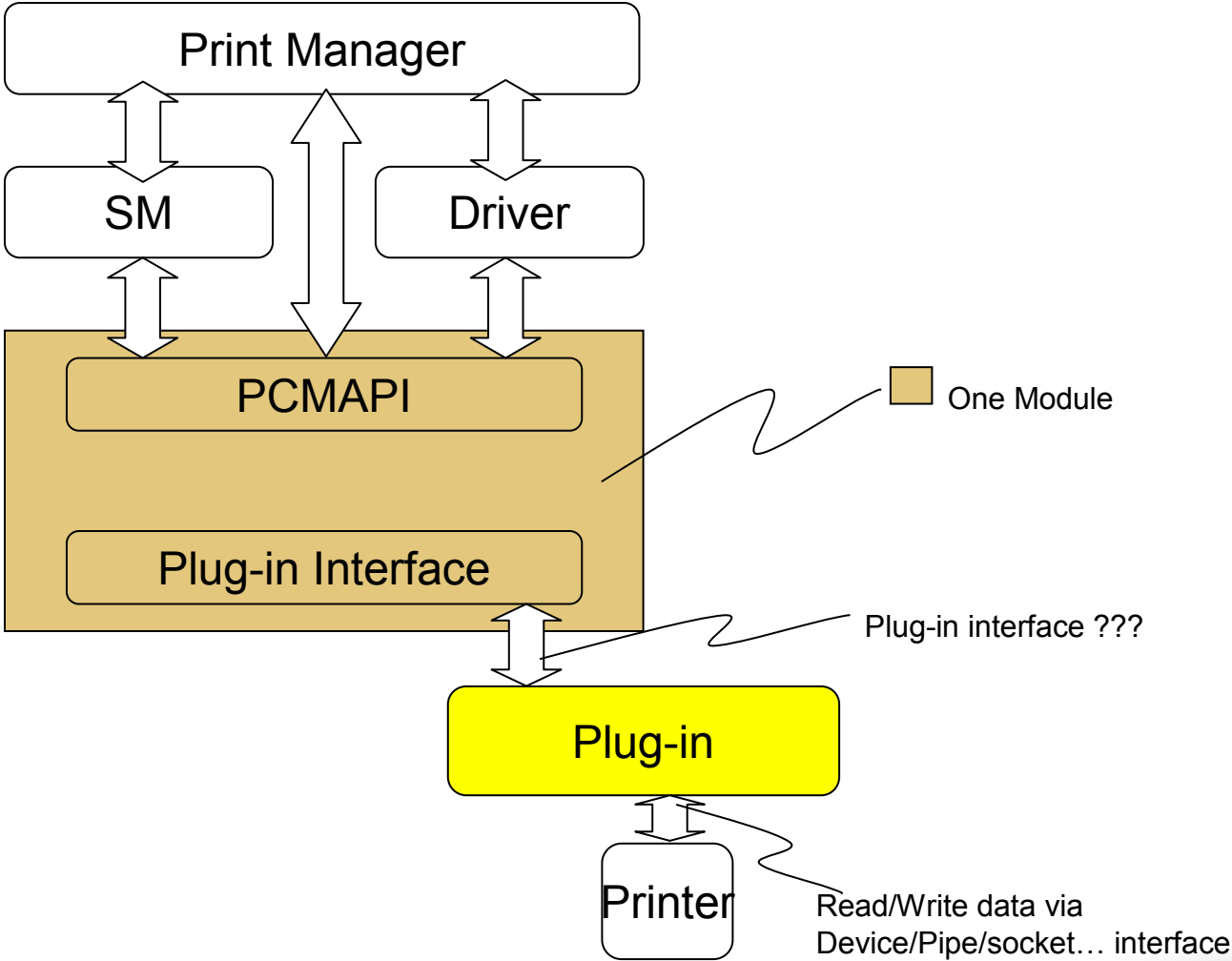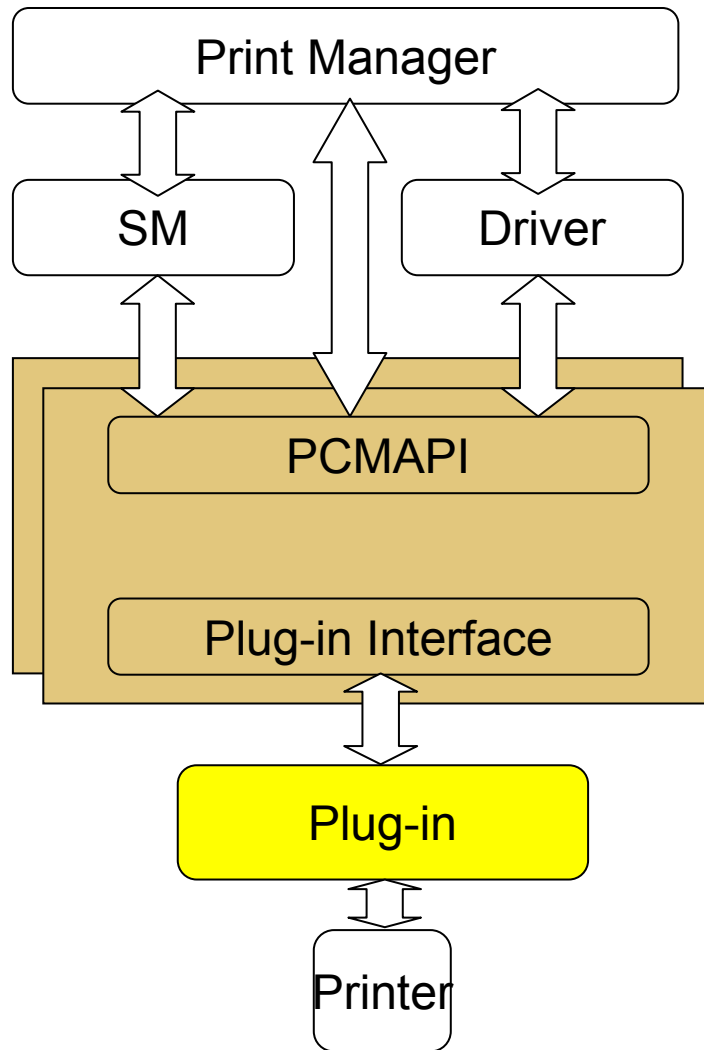- Device arbitration
  with plural device

Requirement

✿ PCM present port management functions

- ✿ Printer selection
- ✿ Port open/close by SM, Driver and extra
- ✿ Data read/write by Driver
- ✿ Status read by SM
- ✿ Job control from upper module
- ✿ Extra control port

Requirement - PCM working flow

Requirement - PCM working flow

Print Manager

SM   Driver

PCMAPI

Plug-in Interface

Plug-in

Printer

1. Print Manager select Printer
2. PCM fork itself as
   "Reading / Writing Control Process"
   with linking a Plug-in
3. SM or Driver Read/Write via PCMAPI

APIs

Object control
• New
  • Create object for port management
  • One object will connect one device (in many case, printer)
  • PCM fork itself and link Plug-in

• Destroy
  • Destroy object

Port control
• Open
  • Open port for status, data or extra…
  • A kind of port specified by port ID

• Close
  • Close port specified by port ID

APIs

Job control
- StartJob
    - Declare the beginning of the printing job

- EndJob
    - Declare the end of the printing job

- CancelJob
    - Cancel the job specified by job ID

Data R/W
- Read
    - Read the specified byte length of a device
    - A kind of data specified by port ID

- Write
    - Write the specified byte length of a device
    - A kind of data specified by port ID

Open questions

❀ PCM Interface must be…

    ❀ IPP?

    ❀ Socket?

    ❀ Named PIPE?

    ❀ Or, anything else?

PCM is specification for OpenPrinting, but want to grow as standard that retrieve by other framework, isn't it?