Server-to-Device Protocol (SDP) Meeting Minutes
May 21, 1998
Washington, DC

The meeting started on May 21, 1998 at 1:00 AM led by Harry Lewis.

5   These minutes were written by Tom Hastings (with review by Harry). ==Action items are highlighted.== The attendees were:

Brian Batchelder - HP
Ron Bergman - Dataproducts
Roger deBry - IBM (by phone, since he was a presenter)
10   Lee Farrell - Canon
Tom Hastings - Xerox
Bob Herriott - SUN
Mark Hodges - HP
Henrik Holst - I-Data
15   Scott Isaacson - Novell
David Kellerman - Northlake
David Kuntz - HP
Greg LeClair - Epson
Harry Lewis - IBM Printing Systems
20   Jay Martin - Underscore
Sandra Matts - HP
Atsushi Uchino - Epson
Don Wright - Lexmark

1   Agenda

25   Scott Isaacson reviewed the agenda:

Thursday, 5/21:

1:00-5:00 SDP (Harry)

Charter

Requirements (David Kellerman, Bob Herriot email)

30   Proposals (Harry/Roger, Don)

1284.3 and 1284.4

2   Round the Room Thoughts on a standard SDP

Before we started the agenda, Don raised the important concern as to whether anyone would implement a standard SDP if we developed such a standard. He wanted to avoid the PWG
35   spending a lot of time on an SDP standard and then not have any vendors implement it. He didn't want to repeat the TIP/SI experience where a number of printer vendors participated in developing the TIP/SI standard, but did not implement TIP/SI. So we went around the room to get initial opinions on doing an SDP standard. These are unreviewed comments from my hurried notes. I apologize in advance if I mis-quoted anyone. Send me e-mail and I can fix the minutes before the
40   next meeting:

Scott - Novell is very leary of a new definition, does not think that something "new" (different) is useful to have. Either use IPP with extensions or TIP/SI with security and extensions; don't define something different.

45   Mark - thinks an SDP is a good idea (but the HP management (Armon Rankin?) didn't see a need five years ago when TIP/SI was being developed.

Lee - Good idea. Cannon not sure whether Cannon would implement an SDP.

David Kuntz - Wants right requirements that solves real problems first. has concern for a pre-determined outcome. Must meet market requirements. HP has sockets and PJL. SDP should work with NT 5.0 standard port monitor. Therefore, PWG needs to look at customer requirements.

50   David Kellerman - Would support SDP after SDP was widely deployed by printer vendors. Northlake would be a "follower".

Bob - Would like an SDP that is better than LP. Don't want to implement two different protocols. The semantic mapping between the client-to-server protocol and the server-to-device protocol should be as close to the identity mapping as possible. Want to use existing layers as much as
55   possible.

Jay - Supports many versions of UNIX. We need to say why TIP/SI is insufficient [if it is].

Don - Not interested in a new SDP. There is a need for an SDP and there is one: TIP/SI. What is wrong with TIP/SI (IEEE 1284.1)? Has a concern of the cost of implementing IPP as an SDP in devices. Need to look at requirements. Would implement an SDP. A standard SDP shouldn't be a
60   collection of protocols. A standard SDP needs to be a simple protocol.

Brian - From a standards person's point of view: an SDP is a good idea and would help OS vendors. A standard SDP would need to be robust and better for the user. Would be valuable. HP experience: need requirements, evaluate what alternatives exist. Can't commit that HP would implement, but would try [to convince HP to]. Need to consider the two alternatives: 80%
65   requirements in TIP/SI plus 20% versus starting with something different. Good thing to pursue, decide later when finished. Need Microsoft to participate in order for an SDP to succeed in the market place.

Tom - Xerox devices support many device protocols today. Having a standard SDP would help reduce the number and cost in the long run. Xerox wants to be able to add extensions to the device
70   and have them pass in both directions through the server to the client without requiring a new release of the server. Building an SDP with IPP semantics and encoding would help achieve that requirement. Also devices will have IPP, so having an SDP that is similar rather than very different will keep costs of devices down.

Ron - A standard SDP needs to be widely supported. A standard SDP would save drivers.

75   Harry - Good idea to pursue. I agree that TIPSI is a paper standard, at this point, it's history of limited acceptance does not warrant any more special consideration as a standard than any other potential solution.

Roger (by phone) - Good idea to pursue.

ACTION ITEM (all): Contact Microsoft to participate.

80 ## 3   SDP Requirements

3.1   David Kellerman's list of SDP requirements

We used David Kellerman's e-mail requirement summary as a starting point for developing a good set of requirements for a standard SDP. I've included his requirements list in quotes and interwoven the discussion after each point.

85 ```
"Recent SDP discussions have "drilled down" pretty deeply. To try to gain some
perspective, here's a general observation, and my list of key technical
requirements that are still at issue for a driver-printer protocol.   The
general observation is that everyone has already "solved" the driver-printer
communications (where "internet printing" was relatively uncharted territory),
```
90 ```
so a new driver-printer protocol has to have compelling advantages over
existing solutions in order to get implemented.  In fact, most vendors have
heavy investments in their existing solutions."
```

```
"In contrast to IPP as an "internet printing" solution, I doubt that vendors
would give serious thought to replacing their driver-printer connection with
```
95 ```
IPP -- they'd lose too much functionality. The question we ought to keep
revisiting is, do we have a driver-printer protocol that is compelling enough
to get implemented?"
```

```
"Here's my list of technical requirements that are specific to the driver-
printer interface. Most are lifted from prior discussion."
```
100 ```
"1. Non-HTTP transport -- HTTP seems a poor choice of transport for a driver-
printer protocol, for a variety of reasons: it's "heavy," it constrains the
model of driver-printer interaction, it potentially requires the driver to be
integrated with the host HTTP services."
```

SDP must be light weight. HTTP has a difficult model for interaction. The abstract IPP model is a
105 better match for an SDP.

```
"2. Transport independence -- does the driver-printer protocol need to be
designed to run over different (possibly non-network) transports; what is the
practical (not just hypothetical) scope of the protocol."
```

Transport independent could mean several things: physical link layer independent, transport layer
110 independent. Need to map an SDP to different "transports": TCP, UDP, USB, 1394, parallel, ,
serial connect (questioned), … We need to make a list that we are going to cover and those that we
are not. This list should go in the SDP requirements document.

ISSUE: Some favored keeping layers distinct and only adding functionality for a deficient layer in
an implementation that uses that configuration. Others favored have a single SDP that made up for
115 the deficiencies in lower layers directly in the SDP. The latter approach is simpler to understand,
though it does mean that there is redundancy in certain configurations. No consensus was reached
on this issue. We agreed that the SDP should not be tied to a particular transport.

Part of this issue is whether the SDP should have an acknowledgement message at the application
level or should depend on lower layers to detect problems. A related issue is whether there should
120 be an ACK for every write or at the end of an entire operation?

We need some assumptions about reliability of the environment over which the SDP is deployed.

ACTION ITEM (Brian): Make a table of which physical connections had sockets functionality.

```
"3. Handling the data stream -- it isn't just an issue of "chunking" the data
stream; there is a body of experience that suggests separating data transfer
from the control channel is a desirable feature of a driver-printer protocol."
```

125

RFC 63 FTP experience is that a separate data from control is a good idea. Same for TIP/SI and CPAP v2.

Separating data from control is a solution to some requirement. What is that requirement?.

Possible requirements:

130   No framing of data at the application layer in order to improve efficiency of data transmission and receipt.

        No blocking of control information during data transfer.

        Need flow control at a lower layer

```
"4. Notification, printer status and exception monitoring -- yes, this has been
beaten around a lot; a driver-printer protocol needs a good answer, and IPP
doesn't yet have one; sockets have issues with scalability and persistence of
connections."
```

135

Notification should be done with the same protocol, in-band asynchronous exception handling. In-band means same data packaging.

140 Should be back over the same channel for ease of implementation.

HP experience: two channels asynchronous up one channel.

Question: Should server be assumed to be multi-threaded or not? In other words, is there a requirement that the SDP be implementable in a server that is not multi-threaded?

TIP/SI: every data sent - ACK by recipient always or on a different asynch channel.

145 If a multiplexor communications layer, then assume 1283.3 and 1283.4.

```
"5. Security -- it's shortsighted not to address security even in a driver-
printer protocol (or at least plan for it); it needs to handle coordinated
authentication of multiple connections if the protocol allows that (for data
stream, notification, etc.), as is desirable for a driver-printer protocol."
```

150 Enough said.

```
"6. Device management - it was out-of-scope for IPP 1.0, but a driver talking
to a printer typically has to deal with device management issues; IPP plus
"some other protocol for management" gets ugly quickly (configuration,
coordination, security), and you at least need to spec the alignment between
the multiple protocols."
```

155

We agreed that we need further detailed requirements. [We need to have the same understanding about what "device management"? Query device configuration? Query device capabilities? Set device defaults. Down-line load resources? Delete resources? Set policy? what else? Then we need specific requirements in each category -TH].

160
```
"7. Resource management - it is valuable for a driver-printer protocol to allow
the printer to request resources (fonts, forms, etc) as it needs them; it can
simplify drivers, make printer configuration and maintenance easier."
```

ISSUE of whether the server pushes the resources to the device, perhaps in response to a
notification and/or by analyzing the, or whether the device initiates pulling the resources from a
165   resource server as needed, as in CPAP.

```
"8. Localization - at my last reading, IPP placed a significant localization
load on the printer; the model for a driver-printer protocol should be machine-
to-machine, place localization at the driver, and avoid localized data on the
printer."
```

170   Do localization in server.

```
"9. Who's in control - IPP assumes the client controls all interaction (push
the job across, poll for information); a driver-printer protocol can benefit
from a more balanced division of control (printer gating of job processing,
printer delivery of event notifications, printer request of data and resources,
for instance)."
```
175

IEEE 1394 is a memory mapped approach which allows push or pull.

```
"10. Client contention - when multiple clients share a non-spooling printer, do
they just poll until they gain access; alternatively, a notification-style
approach ("I want to send a job," "I'm ready to accept your job") might be more
efficient and "fair.""
```
180

Agree this is a requirement.

```
"11. Low-level discovery - to allow driver (or directory service) to be
configured automatically with printer identity, features"
```

Agree this is a requirement.

3.2   *Roger and Harry's list of SDP requirements*
185

Then we looked at the requirements in the Appendix of Harry's and Roger's SDP proposal:

ftp://ftp.pwg.org/pub/pwg/sdp/sdp-proposal.txt

ftp://ftp.pwg.org/pub/pwg/sdp/sdp-proposal.pdf

The appendix items are in Courier and in quotes. The discussion follows each requirement and is
190   in Times Roman. We did not look at the body of Harry's and Roger's SDP proposal (that is for next
meeting).

```
               Appendix:  Detailed Requirements Statement
```

```
195    The following requirements come from discussions held at past PWG
meetings.
```

```
 "1. The SDP protocol must provide a means to synchronize
communications between the Server and Device, regardless of the
underlying transport (i.e. start of packet indicator)."
```

200   Synchronization between server and device (start packet at application layer).

Detect out of synchronization at application layer

ISSUE: delivery a message as a chunk. Maybe lower layers should do chunking?

```
  "2. The SDP protocol must provide for packatized data flow with
  the ability to segment data for efficient use of underlying
  services with a means to indicate final segment. ("Chunking")"
```
205

Chunking so can send data stream.

Define to be a byte stream serving the lower layers (vs. message stream service)

```
  "3. Must provide the ability for either Server or Device to
  mandate synchronization (ACK) to their message flow, if
  appropriate.
```
210

```
     - Server -- Any message flow as Server sees appropriate based
  on data integrity needs, perceived communications reliability etc.

   · - Device -- Limited message flow, mainly for the purpose of
  determining when a server may have lost connection. (**I'm not
  sure about this one**), or during the reverse transmission of FAX
  or SCAN data"
```
215

Sender asks for an ACK (or send to server so server can synchronize with job in printer.

Requirement assumed by lower layers

Server asks for events to come back from device.

```
  "4. The SDP protocol must asynchronous notifications including
  support for registration, de-registration and event type
  filtering."
```
220

Register and unregister for event notification.

Need event type filter

```
  "5. The SDP protocol must support distinction between
  communications with Print Job Objects and Printer Objects, as
  defined in the IETF IPP Model standard."
```
225

distinguish between printers and jobs

```
  "6. The SDP protocol must accommodate the IPP encoding as defined
  in the IETF IPP Protocol standard."
```
230

IPP encoding (not transport)

```
  "(The following are from Portland and may be a bit redundant or
  need restating, rewording, or we may choose not to include some of
  them in our document):"
```

```
  "7. The SDP protocol must be completely Transport independent."
```
235

need more details. See discussion above in David Kellerman's requirements paper, item number 2.

```
  "8. Need way to send FAX or SCAN data from device to server (for
  MFPs only)"
```

Rephrase：Don't disallow FAX or scan.

240   `"9. Control channel can't be blocked by data. Server can query and control with quick response."`

`"10. Need configuration and status info like what's provided in the printer MIB."`

Printer MIB information

245   `"11. Ability to recover job accounting information"`

Need job accounting `"12. Device can retrieve resources like fonts and forms"`

`"13. Server-to-Device protocol must not significantly limit the function of any major existing client to server protocols and must`
250   `accommodate IPP without any loss."`

Accommodate IPP without loss.

`"14. Must Cover the case of spooling both in the server and the device or other multiple levels. The user should get the same functionality (CANCEL, MODIFY etc.)."`

255   Server manages the device whether the device spools or not.

Server maintains control of the job.

`"15. Submit, Cancel, and list jobs (end-user and administrator)"`

Security for cancel and list jobs

`"16. Provide a means of client contention resolution (lunch`
260   `counter ticket vs. underlying protocol)."`

`"17. Allow printer to throttle data from the server."`

4   SDP Charter

We reviewed the SDP charter draft:

ftp://ftp.pwg.org/pub/pwg/sdp/CharterSDP.pdf

265   1. Some devices are not IPP at all.

2. All protocol to server

3. Job submission and Printer Management

4. Handle any PDL

Trade-offs:

270   1. Time to develop standard

2. Features

3. OS support

4. Faster route to market

5. Need "some better" to a "lot better" than current vendor server-to-device protocols

275   ISSUE: One protocol versus a suite of protocols (including SNMP).

5   ACTION ITEMS and NEXT STEPS

ACTION ITEM (Harry): Inform Alan that the Finisher MIB will meet Tuesday night a the July PWG meeting in Monterey.

ACTION ITEM (Harry): Update the Charter and Requirements, or make two documents.

280   Need SDP officers.

Presentations of proposal at the next SDP meeting:

       Harry

       Randy

       Don

285   ACTION ITEM (Tom): Warn Randy.