# Minutes of IPP Working Group Meeting

December 15-16, 1999

## 1. Meeting Attendees

| | |
|---|---|
| Lee Farrell | Canon Information Systems |
| Bill Wagner | DPI/Net Silicon |
| Greg LeClair | Epson |
| Atsushi Uchino | Epson |
| Trevor Wells | Hewlett Packard |
| Ron Bergman | Hitachi-Koki |
| Harry Lewis | IBM |
| Henrik Holst | i-data |
| Vinnie Finn | Kodak |
| Jim Vitkus | Kodak |
| Michael Wu | Kodak |
| Stuart Rowley | Kyocera |
| Don Wright | Lexmark |
| Weihai Chen | Microsoft |
| Paul Moore | Peerless |
| Eric Random | Peerless |
| Gail Songer | Peerless |
| Nick Webb | Peerless |
| Grant Gilmour | Qmaster |
| Fumio Nagasaka | Seiko Epson |
| Tom Hastings | Xerox |
| Bob Herriot | Xerox |
| Carl-Uno Manros (Chairman) | Xerox |
| Peter Zehler | Xerox |

### 1.1 Internet Printing Project (IPP) – Day 1

Carl Uno-Manros opened the IPP meeting and provided the suggested agenda topics:

- Short presentation on the IETF46 meeting and on where we stand with our documents
- Set 2 and Set 3 Operations
- IPP Notifications
- IPP/1.1: Notifications over SNMP
- IPP/1.1: The ‘ipp-get’ Notification Delivery Method
- IPP/1.1: The ‘ipp-ntfy’ Notification Delivery Method and Protocol
- IPP/1.1: Job Progress Attributes
- IPP/1.1: attribute syntax for Collection
- IPP/1.1: Comparison of Notification Delivery Method alternatives
- IPP/1.1: Comparison of IPP Notification Delivery Methods
- Look at questions and answers from the IPP DL and determine which ones should be documented in the IPP Implementer's Guide
- Further testing/bakeoff, trigger criteria, form/venue for testing
- Presentation on initial ideas for document/page level attributes
- QualDocs

### 1.1.1 IETF Meeting

Carl-Uno gave a brief review of the slides that he had presented at the IPP Extensions "Birds of a Feather" (BOF) session of the November IETF Plenary:

ftp://ftp.pwg.org/pub/pwg/ipp/IETF-Presentations/ietf46-IPP.pdf
ftp://ftp.pwg.org/pub/pwg/ipp/IETF-Presentations/IPP-Status-9912.pdf

He mentioned that the QualDocs (Quality Document Transfer) BOF was also held at the Plenary, and included many of the same participants.

### 1.1.2 QualDocs

Paul Moore provided some information on QualDocs. He explained that QualDocs came out of the Internet Fax (IFax) effort within the IETF. All IFax implementations (and the related IETF specifications) are currently based on store-and-forward methods. However, it has been noted that several features of "classic facsimile" (e.g., format negotiation) are not provided by the existing IFax protocol.

IPP has been identified as a good basis for providing these additional features. Paul has written a document that describes a protocol built on IPP to achieve the desired capability for transferring high quality documents over the Internet. The TIFF-FX image format (RFC 2301) has been adopted as the base document format requirement supported by QualDocs, and the protocol allows the client to query the target device to discover the exact features supported.

Paul noted that the QualDocs protocol is not limited to the transfer of documents only—it is suitable for a variety of image transfer applications.

Paul plans to submit his document to the IETF as an Internet-Draft in the next few weeks.

### 1.1.3 Set 2 and Set 3 Operations

Tom Hastings led a review of the two documents on new IPP operations:

ftp://ftp.pwg.org/pub/pwg/ipp/new_OPS/ipp-ops-set2-991208.pdf
ftp://ftp.pwg.org/pub/pwg/ipp/new_OPS/ipp-ops-set3-991208.pdf

He first listed the Set 2 Operations, which include both Printer object and Job object operations:

<u>Printer Operations</u>
Set-Printer-Attributes
Enable-Printer
Disable-Printer
Pause-Printer-After-Current-Job (IPP/1.1 Pause-Printer clarified)
Pause-Printer-After-All-Current-Jobs
Deactivate-Printer
Activate-Printer
Restart-Printer
Shutdown-Printer
Startup-Printer

Job Operations
Set-Job-Attributes
Reprocess-Job
Cancel-Current-Job (though the target is the Printer object)
Suspend-Current-Job (though the target is the Printer object)
Resume-Job
Promote-Job

The Set3 operations are Device operations that operators/administrators may perform and that directly affect the output device:

Disable-Device
Pause-Device-Now
Pause-Device-After-Current-Job
Deactivate-Device
Purge-Device
Power-Off-Device

Enable-Device
Pause-Device-After-Current-Copy
Resume-Device
Activate-Device
Reset-Device

Tom reviewed the terminology, requirements and use cases in the Set3 document. During the review, someone suggested that a possible additional requirement would be to pause the device after a *specific* job.

The group agreed that additional clarification should be added to explain the full impact of a Shutdown-Printer command.

It was suggested that the Note in the paragraph describing the requirement for Startup-Printer ("This operation is unlikely to be supported for the embedded Printer configuration.") should be removed.

Some people suggested that there should be an equivalent Startup operation for a device—Startup-Device. This led to a discussion about what is actually meant by "turning power off." Can a "turned off" device still listen for commands—or have its power turned back on remotely?

While reviewing the configuration diagrams shown in the Set3 document, there was much discussion about whether device operations can be supported for fan-out configurations. At one of the IPP teleconferences, it was determined that this would be too complicated. As a result, device operations can only occur on fan-out configurations if the fanned-out device is hosted by a subordinate IPP Printer object.

Because there was much confusion about the meaning of a "subordinate Printer object", Tom will modify figure 5 in the Set3 document.

Two Issues in the Set3 document were addressed:

ISSUE 01 – Ok that every Device operation REQUIRES the IPP Printer to perform the corresponding Printer operation, if implemented?
The group agreed that it will *not* be ok. The paragraph following Table 1 in the document will be removed or modified.

ISSUE 02 – Which corresponding Printer operations MUST an implementation support, if it supports a particular Device operation?
Deferred.

Should the Cancel-Current-Job operation (in Set2 document) be forwarded? What is the behavior on a fan-out configuration to multiple devices? Which is the “current job”? Should it be left as “implementation dependent”?

While reviewing the “Set Printer Attributes” operation (in Set2 document), it was suggested that a more general mechanism should be defined for handling and supporting side effects—such as modifying values of any related attributes.

The group agreed that all the “Set” operations should be moved to a separate document.

Should the attribute names be modified (e.g., with a unique suffix) to identify them as “factory defaults”? Some individuals thought that this would make some of the factory default handling easier to specify/implement. Also, some of the attributes could be prefixed/suffixed to indicate that they are [only] applicable to specific document formats. Although preliminary discussion appears encouraging, the topic was deferred for more detailed examination.

There was general agreement that the protocol should support the ability to “unset” values.

## 1.2 Internet Printing Project (IPP) – Day 2

### 1.2.1 Set 2 and Set 3 Operations – cont’d

Tom continued the review by addressing the Issues in the Set2 document.

ISSUE 01 – Do we want to define whether the response to the client for Job operations can happen before the non-leaf Printer gets the response from its subordinate Printer or MUST the non-leaf Printer wait until its gets the response from its subordinate Printer?
Yes.

ISSUE 02 – The "requesting-user-name" operation attribute is the parent Printer's host, not the original requesting user, correct?
Correct.

ISSUE 03 – Presumably the "job-originating-user-name" remains as the authenticated original user, not the parent Printer's authenticated host, correct?
Correct.

ISSUE 04 – What new 'moving-to-xxx' and 'xxx' values do we need to support the new operations defined in this document?
Deferred.

ISSUE 05 – Should Pause-Printer-After-Current-Job be a new operation with a new operation-id code or be a clarification of the existing IPP/1.1 Pause-Printer operation and use its operation-id? Or should the Pause-Device-Now operation be a new operation-id code or be the clarification of the existing IPP/1.1 Pause-Printer operation and use its operation-id? Or should both Pause-Printer-After-Current-Job and Pause-Device-Now be new operation-id codes and leave the IPP/1.1 Pause-Printer with its current ambiguous (implementer free-for-all) semantics?

The new operations for Pause Printer and Pause Device will be defined. The IPP/1.1 Pause operation will not change.

ISSUE 06 – Should the Printer's "printer-state" attribute be independent of the Pause Printer operations so that the Pause Printer operations don't set the "printer-state" to 'stopped', i.e., the "printer-state" tries to reflect 'idle', 'processing', or 'stopped' of the output device(s) as best it can independent of whether the IPP Printer object is paused or not?

No (no change is required.)

ISSUE 07 – Would a better name for Pause-Printer-After-All-Current-Jobs be Hold-Future-Jobs? Unfortunately, unlike Pause-Printer-After-All-Current-Jobs which gets to 'paused', the state transition would just be to 'idle' when all of the current jobs have completed? But what operation would undo this condition? Do-Not-Hold-Future-Jobs, Release-All-Jobs? Or how about having a single Schedule-Jobs operation that has a parameter that says whether to hold all future jobs or not?

Deferred.

ISSUE 08 – Any better name than 'moving-to-paused-all' Printer state reason to distinguish Pause-Printer-After-All-Current-Jobs from Pause-Printer-After-Current-Job which uses 'moving-to-paused'?

Deferred.

ISSUE 09 – Should the Printer's "printer-state" attribute be independent of the Pause Printer operations so that the Pause Printer operations don't set the "printer-state" to 'stopped', i.e., the "printer-state" tries to reflect 'idle', 'processing', or 'stopped' of the output device(s) as best it can independent of whether the IPP Printer object is paused or not?

No (no change is required.)

ISSUE 10 – Should Cancel-Current-Job really be moved to Set3 as a Device operation, i.e., Cancel-Current-Device-Job or do we need both a Printer and a Device operation that cancels the current job?

This should be moved to Set3.

Tom asked if anyone would like to form a small sub-group to work out more details on the Set2/Set3 Operations documents. Peter Zehler, Harry Lewis, Michael Wu, and Ron Bergman all volunteered. Tom will publish an updated version of the documents and will distribute them for review.

A separate sub-group (which will probably be much larger) will be established to review the new “Set Operations” document when it is generated.

[Tom requested that the Minutes should reflect that the group should “wait for the Op codes before sending out the document for WG last call.”]

### 1.2.2 IPP Notifications

Tom led a review of the IPP Notification specification document:
ftp://ftp.pwg.org/pub/pwg/ipp/new_NOT/ipp-not-spec-991014.pdf

During the review, Bob Herriot suggested that the size of subscriber-user-data (currently octetString(63)) should be changed to 255.

ISSUE 1 – Once a number of delivery solutions have been developed and evaluated, we may want to make one or several of them REQUIRED for implementation to ensure a minimum set of interoperability. Which one or ones should be REQUIRED?
Deferred.

ISSUE 2 – Should we change the Notification Model to allow notification delivery methods that are request and response (in addition to the current model which has only one-directional notification delivery using the 'application/ipp' operation response format?
“Make it look like a request whether the response comes back from recipient or not, except when the notification is a response to a request.”

ISSUE 3 – If the answer to Issue 2 is yes, should we change the format of the notification content using 'application/ipp' to always be a (new) Send-Notification operation request, whether the scheme returns a response or not?
Yes.

ISSUE 4 – Ok that "job-uri" isn't defined for use with the Create-Job-Subscription operation?
Initially it was suggested that we should be consistent with IPP/1.1, and that “job-uri” should be defined. However, after more discussion, this Issue was deferred.

ISSUE 5 – Ok that we aren't passing the operation attributes that are copied to the Subscription object in the new Subscription object attributes group? Some of the "notify-xxx" attributes aren't Subscription object attributes.
It would be good to create a separate tag for subscriptions.

### 1.2.3 The 'ipp-ntfy' Notification Delivery Method and Protocol

Tom led a review of the “IPP-ntfy” document:
ftp://ftp.pwg.org/pub/pwg/ipp/new_NOT/ipp-not-ntfy-delivery-991209.pdf

ISSUE 01 – What should the name of this delivery method and protocol be that we use in the title of this document?
Evidently, the IESG does not approve of abbreviated names. Deferred.

ISSUE 02 – What should the scheme name be? Consider 'ipp-ntfy' a working title, until we see several schemes. The 'ipp-get' delivery method is another example. Should the scheme name somehow include "notification", i.e., 'ntfy'? How about 'ipp-ntfy-send' or 'ipp-ntfy-push' and 'ipp-ntfy-get' or 'ipp-ntfy-pull' to go with the Send-Notifications and Get-Notifications operations, respectively?
Deferred.

ISSUE 03 – Should the scheme name be used in the title?
Deferred.

ISSUE 04 – Ok to add a "Generic Attributes" group tag to [ipp-pro], instead of adding a special tag for each new object and/or operation that needs a different set of attributes than Job or Printer? The same issue for the Subscription object in [ipp-ntfy]. Either we define separate tags for both or use a single generic tag for both and future objects and attribute groups.
A tag for Notifications should be defined. A separate tag for Subscriptions will also be defined.

ISSUE 05 – Ok to extend Notification Model to allow a single notification to have both Human Consumable form and Machine Consumable form when the client asks for Human Consumable form by supplying the "notify-text-format" attribute rather than the Human Consumable being sent instead or in addition to the Machine Consumable using MIME multi-part-related?
Yes, each method can behave differently. In some cases, the Human Consumable form could be added to the Machine Consumable. In other cases, only one will be sent.

### 1.2.4 Document/Page Attributes – Exception Attributes

Bob Herriot presented his proposal on Exception Attributes. He explained that certain pages or even whole documents within a job might need to be handled differently than the rest of a job. Different finishing requirements or page media types (e.g., letterhead) were given as examples.

Bob said that he will post his slides and the associated proposal document.

During the presentation, Don Wright suggested that the “last page” should be referenceable—even when its exact page number is not known.

Don also suggested that different copies of the same document should be able to have different attributes assigned. For example, it should be possible to send a document only once, but request one copy printed duplex, one copy on transparencies, and multiple copies printed 2-up format.

NOTE: This document will be a PWG specification. Don noted that it should be published with PWG copyright statements.

### 1.2.5 Registering Private Operations

It was noted that the support for “private” operations needs to be established—especially to ensure collision avoidance. A publicly accessible PWG registry was suggested. Gail Songer volunteered to work on developing the registry and its associated processes.

### 1.2.6 The 'ipp-get' Notification Delivery Method

Tom led a review of the “IPP-get” document:

ftp://ftp.pwg.org/pub/pwg/ipp/new_NOT/ipp-not-get-delivery-991207.pdf

ISSUE 01 – Is there a limit to the number of outstanding Get-Notifications requests that an IPP Printer supports? What is this number? How does it relate to the maximum number of Subscriptions? Can the client determine the number?
Deferred.

ISSUE 02 – Should an implementation be able to queue event Notifications, so that a client can get event Notifications that had occurred prior to the Get-Notifications? If so, how long does the IPP Printer keep the event Notifications before discarding them (for this delivery method only)? The lease time of the Subscription object? If this is possible, should the subscriber get to say whether to queue or not, or is it just baked into the implementation. If the former, does the subscriber indicate via a parameter in the notification method URL? If the latter, how does a client discover whether event Notifications are queued or not? Should we have two different notification methods, one the queues and one that doesn't?
It was suggested that any "notification queuing service" should be the responsibility of the notification recipient, not the Printer. However, the Issue was not completely resolved.

ISSUE 03 – What "printer-state-reasons" might cause an error return, if any? 'paused', 'shutdown', 'deactivated'?
None of the printer-state-reasons should cause an error return.

ISSUE 04 – Is this correct for MIME multi-part-related responses? This needs prototyping.
Deferred.

There was mixed opinion on whether the "ipp-get" scheme needs to be registered with IANA.

### 1.2.7 IPP Interoperability Testing

Pete Zehler led a discussion on future IPP testing. When will the next "Bake-off" be held? It depends on when we achieve a critical "trigger" threshold of additional content in sufficient implementations. Other "trigger" items suggested include:

- IPP/1.1 is published as an RFC
- [Some of the] Set Operations
- Notifications

Once the trigger items are achieved, a lead time of at least 3 months notice is probably reasonable. Initial estimate for the event: 3$^{rd}$ Quarter 2000.

Pete thinks that we should use a more rigorous test plan, and says that a formal test suite would be very beneficial.

Other suggestions for the next bake-off include:

- more client instances
- more security testing

Someone needs to organize and/or host the event. If anyone is interested in being the Interoperability Bake-off Host, please contact Pete Zehler. Cost issues and "entry fee" considerations still need to be resolved. The cost will likely be affected by the location facilities.

Could IEEE-ISTO host the event? Stardust Labs, Quality Logic, or a University are other possibilities to consider.

Pete recommends that pair-wise testing is beneficial, and should be done before the Bake-off. However, he does not think that it would provide sufficient testing to replace a Bake-off.

### 1.2.8 Basic IPP Notifications using HTTP

Harry Lewis presented his proposal document:

<URL?>

His proposal includes server-directed polling and asynchronous notification. It is intended to reduce the number of required connections, and the length of time that they remain open.

### 1.2.9 What's in a Name?

Carl-Uno wanted to generate new names for the notification method documents under development. The following list was identified:

- ipp-ntfy:
- ipp-get:
- ipp-snmp:
- mailto:
- ipp-notify-xxx:
- inpoll:
- inmix:
- insnmp:
- inget:
- insend:

It was agreed that the documents should be submitted as individual Internet-Drafts and will be "named" by the authors.

The document that will contain the Set operations will be named "Set Attributes Operations."

Henrik Holst volunteered to write the e-mail Internet-Draft submission.

### 1.2.10 How About a Date?

The group proposed a few delivery estimates for the documents:

| | |
|---|---|
| Set Attributes Operations | March 2000 WG Last Call |
| Set2 Operations | June 2000 WG Last Call |
| Set3 Operations | June 2000 WG Last Call |
| Notifications | September 2000 WG Last Call |

There was no time to address the remaining agenda topics.

IPP Meeting adjourned.