IPP FAX - Meeting Minutes

August 1, 2001
Toronto, ONT Canada

Authors: John Pulera, Marty Joel

Attendees:

Don Wright,	Lexmark
Peter Zehler,	Xerox
Shigeru Ueda,	Canon
Aisushi Uchino,	Epson
Patrick Pidduck,	PrinterOn
Mike Kuindersma,	PrinterOn
Marty Joel,	Netreon
Gail Songer,	Netreon
John Pulera,	Minolta
Ron Bergman,	Hitachi-Koki
Bill Wagner,	Netsilicon
Michael Wu,	Heidelberg Digital
Norbert Schade,	Oak Technology
Geoff Soord,	Software 2000
Jeff Christensen,	Novell
Lee Farrell,	Canon
Charles Kong,	Panasonic

Schedule Review

The schedule had to be revised, as there are still many outstanding issues.

Old Schedule:

June 2001	Specifications complete
September 2001	Bakeoff
January 2002	Revised specifications and possible implementers guide

New Schedule:

mid-October, 2001	Specifications complete (next IPPFax meeting)
April 2002	Bakeoff
July 2002	Revised specifications and possible implementers guide

UIF Issues Review

ISSUE 01: Should the capabilities discovery portion of this spec be removed and placed into a specification that deals solely with how IPPFAX uses capabilities discovery? Advantages: other applications interested in using UIF simply as a data format can do so (no prohibitive excess baggage).
The group thought it was a good idea to either create a new document or move the sections relating to capabilities discovery (e.g., UIF Section 4) to the IFX protocol spec.

ISSUE 02: Should we break UIF Profile C into two profiles—one to represent a baseline grayscale configuration and the other to represent a baseline color configuration? This way, a greater number of device capabilities configurations would be allowed without requiring an implementation of CONNEG. (The same could apply to UIF Profile L)

The group decided this is a good idea; however we felt it a good idea to check with Lloyd McIntyre about the direction in which TIFF-FX is heading and why TIFF-FX hasn't adopted a similarly tiered scheme for Profiles C and L. Also, we need to check with Lloyd about what the consequences should be to the tree diagram shown in section 3.1.

ISSUE 03: Should we add the CONNEG tag "profile" and tag values "uif-s", "uif-f", "uif-c", etc., to represent the incremental differences between minimum capabilities strings listed in sections 4.1.2.1 through 4.1.2.5? This would cut down on the length of the CONNEG strings, especially for the composite UIF profile M) and would make it immediately apparent from a human's perspective any OPTIONAL features that are advertised.

The group liked this idea. John Pulera will post proposed changes to the UIF spec to the mailing list.

ISSUE 04: (not in the document; see section 5.1.1) Should we change the MIME media registration to be simply a single value for the application parameter and add a new 'profile' parameter which is multi-valued to indicate the profiles that are in the document? For example:

Content type: image/tiff; application=uif; profile=uif-s,uif-c,uif-mcs

Yes this is a good idea, except the profile portion of the mime type (i.e., "profile=uif-s,uif-c,uif-mcs" in above example) should be OPTIONAL, and indicates those profiles that a Receiver can expect to (but not necessarily) find in the UIF data.

ISSUE 05: (not in the document; see table 15) Profile M only lists 200** and 300** for background and foreground layers and doesn't mention the binary mask layer at all. We need to add the binary mask layer back in. OK to add 400** as a REQUIRED value for all three uses?

MRC requires that the mask layer be an integral multiple (1, 2,3, ...) of the resolutions for the background and foreground layers. If we don't add 400** as a REQUIRED resolution, the mask layer would have to be the same as the background and foreground layers (200 or 300) unless an OPTIONAL resolution were used.
*Yes, "400**" should be added to both the X- and YResolution entries in table 15.*

Additional UIF-related issues raised at meeting:
ISSUE 06: (raised at meeting) In the current spec, the 'FaxProfile' tag introduced in RFC2301 is re-interpreted as the 'UIFProfile' tag for UIF Documents. Also, the meaning of each value for this tag is redefined to refer to inclusion of UIF profiles rather than IFAX ones. The group thought this is not a legitimate thing to do. Should we register a *new* TIFF tag to represent the UIF profiles present in a given IFD? Ask Lloyd about this…

IFX Issues Review

ISSUE 01 (TOC): We did a lot of name changing at the telecon. Are these attribute names ok now? Check the TOC to see all the names together.

Yes, the names are OK.

ISSUE 02 (TOC): I'm not completely happy with the organization of the document. Each attribute has its own section, so it appears in the TOC. Also I've tried to put the corresponding "xxx-supported" right next to the "xxx" attribute description, but in a separate section. However, several operation attributes appear more than once: "ippfax-semantics", "ippfax-profiles", and "document-format". Any suggestions or is this OK?
No problems with the new names.

ISSUE 03: OK that we are using the 'ipp:' scheme for both IPP and IPPFAX protocols?

Because IPPFAX must use TLS, it would be best if there was a separate scheme. IETF won't allow a separate scheme for secure vs non-secure. Patrick Pidduck pointed out that it might be good to have separate schemes for allowing one for in-house and the other for the outside. By using that as our argument, we also gain TLS on a separate port. Bill Wagner raised the question of whether all data must be encrypted, or if TLS can negotiate down to no encryption.

TODO: Determine what IPPFAX's intentions are regarding authentication and encryption.

ISSUE 04: Can 'http' scheme be used in the "printer-uri" target attribute? Will 'http' be more likely to be configured to get through firewalls? What can a standards track RFC say about this since IPP/1.1 REQUIRES the use of the 'ipp' scheme?

Peter Zehler pointed out that you can use ipp:…:80.
We don't believe http scheme should be used with ipp.

ISSUE 05: OK that we are forced to use the same default port for IPPFAX as for IPP? So if a Receiver is configured to only receive IPPFAX Jobs from outside its firewall, but receive IPP Jobs from inside its firewall, one or the other will be forced to supply an explicit (different) port?

This issue is deferred pending on what's decided for ISSUE 03

ISSUE 06: If an IPPFAX Receiver is configured for IPP only, should it still accept an IPPFAX job, rather than rejecting it, but perform it with IPP semantics? That is what an IPP/1.1 Printer would do that doesn't know about the IPPFAX spec and the IPP Sender won't make this mistake, since it MUST query to determine if the Receiver is currently accepting IPPFAX requests.

Yes. Since the sender must query the printer, it knows if ippfax is supported.

ISSUE 07: OK to add the new 'client-error-missing-required-attribute' status code? The existing 'client-error-bad-request' status code isn't sufficient, since we want to return the missing attribute rather than indicate something wrong with what was submitted. Also the existing 'client-error-forbidden' is too mysterious, since it suggests an authorization and/or authentication problem. In the past, missing REQUIRED attributes are developer errors, so that the 'client-error bad-request' was sufficient. But this error can happen to a customer who has turned off IPP (or the implementation only supports IPPFAX semantics). This new status code can be used for other cases where 'client-error-bad-request' is used.

No. Gail suggested the missing attribute could be put in the unsupported attributes group. The only scenario for this is an ipp sender talking to an ippfax-only recipient, in which case existing ipp senders won't understand the new reply code.

ISSUE 08: How does coloring work when more than one UIF Profile is specified?

Gail suggested making ippfax-uif-profiles (1setof type2 keyword) instead be ippfax-uif-profile (type2 keyword), so it would return coloring for just that one profile. Agreed.

ISSUE 09: Should we REQUIRE the Receiver to color attributes with the "ippfax-uif-profiles" supplied by the Sender in a Get-Printer Attributes operation? If yes, should we REQUIRE the Sender to supply the "ippfax-uif-profiles" attribute in the Get-Printer-Attributes?

Peter Zehler pointed out that the "ippfax-uif-profile" attribute is only meaningful when the document format is UIF. If the Sender requests document format UIF but doesn't provide a UIF profile, it will use uif-s. 1st question: YES, if a document format is UIF, attributes must be colored by the Receiver. In fact, colored regardless of the document type, not just UIF. 2nd question, NO – colored to uif-s by default if document type is UIF . Agreed.

ISSUE 10: Should we REQUIRE the Receiver to color attributes with the "document-format" supplied by the Sender in a Get-Printer Attributes operation? If yes, should we REQUIRE the Sender to supply the "document-format" attribute in the Get-Printer-Attributes?

See ISSUE 09.

ISSUE 11: OK to REQUIRE the Sender to query the "ippfax-versions supported" Printer Description attribute, or is using Validate-Job sufficient if we change it from SHOULD to MUST? An IPP/1.1 Printer would return success, with the "ippfax-semantics" operation attribute in the Unsupported Group which the Sender could check for. What about an IPPFAX Receiver that is configured only for 'ipp'?

No – NOT REQUIRED. We agreed on adding a new operation attribute named "ippfax-version". " ippfax-semantics" is no longer going to be used as an operation attribute. " ippfax-semantics-supported" is now renamed to semantics-supported.

ISSUE 12: OK to REQUIRE the Sender to query the "ippfax semantics-supported" Printer Description attribute, or is using Validate Job sufficient if we change it from SHOULD to MUST? An IPP/1.1 Printer would return success, with the "ippfax-semantics" operation attribute in the Unsupported Group which the Sender could check for. What about an IPPFAX Receiver that is configured only for 'ipp'?

No – NOT REQUIRED, but specify that the Sender MUST determine if ippfax is supported (which can be via Get-Printer-Attributes or Validate-Job), and that the Sender may NOT send an ippfax job if it can get printed via ipp without the Sending User's okay.

ISSUE 13: Need to add some more UIF Profiles for color versus gray scale for C and L Profiles (same issue for UIF spec).

This is a UIF issue (see UIF ISSUE 02).

ISSUE 14: OK that we got rid of the new 'octetString32k' attribute syntax and use existing IPP/1.1 attribute syntaxes, so that existing IPP systems can be used as gateways?

Yes.

ISSUE 15: are these additional capabilities restricted to the OPTIONAL capabilities in the UIF Profile according to the UIF spec ([14]), or MAY they include other capabilities as well?

Yes.

ISSUE 16: Should the UIF specification [14] add registered UIF Profile tags so that the entire minimum string becomes a single named token. Lloyd McIntyre thought this would be a good idea in order to shorten the strings and make the processing easier by the Sender.

This is a UIF issue (see UIF ISSUE 03).

ISSUE 17: Should we add the new "ippfax-uif-profile" operation attribute to the Get-Printer-Attributes operation and then REQUIRE the Receiver to perform attribute coloring for the "ippfax-uif-profile" operation attribute? Then the Sender could determine the resolutions supported for a particular UIF Profile without having to do the CONNEG stuff?

See IFX ISSUE 09.

ISSUE 18: What restrictions on the vCard content do we need to make? vCard can have image, logos, sound!

Leave it up to the implementation to place limits as it sees fit. There is an error code to indicate that an attribute is "too long". Maybe selectively discard fields (such as the images, sound…) in an implementation specific way. Required fields? Change the first SHOULD in Section 6.1 to MAY.

ISSUE 19: Denial of service problem: a Sender could bog down a Receiver Job with a huge amount of data which the Receiver is supposed to copy to the Job object.

True, but per Peter Zehler's suggestion, allow the implementation to limit the size, and return client-error-request-value-too-long.

ISSUE 20: What restrictions on the vCard content do we need to make? vCard can have image, logos, sound!

See ISSUE 18.

ISSUE 21: Denial of service problem: a Sender could bog down a Receiver Job with a huge amount of data which the Receiver is supposed to copy to the Job object.

See ISSUE 19.

ISSUE 22: Did we agree to delete the ippfax-sender-uri (uri) operation/Job Description attribute in favor of depending on TLS authentication?
No, we will not delete (leave as a MUST). Make clear that the ippfax-sender-uri is only a comment (e.g., for logging purposes) – has nothing to do with authentication. That way, it's more akin to a 'reply-to' field.

ISSUE 23: SHOULD be using a client URL by preference and NOT a MAC address (generally totally unknown to an IPP client application). In any case the IEEE and IETF don't approve the use of MAC address for identifiers anymore except in EUI-64 format (an IEEE standard), which is the basis for canonical IPv6 self-configured global addresses. Ira will look up the RFC references later, if you want EUI-64

No, the MAC address should NOT be used. Yes, a unique, syntactically correct URI that is traceable to the IPPFAX Sender MUST be used for "ippfax-sending-user-identity". Remove Section 6.3 paragraph 3 sentence 2 ("A value derived from a MAC address…").

ISSUE 24: Or should the spec be changed to REQUIRE the Sender to use Validate-Job? Currently the spec only RECOMMENDS using Validate-Job and REQUIRES that the Sender query a number of Printer Description attributes in order to submit a job the Receiver will accept.

NO. It can be validated by Get-Printer-Attributes if the client wants.

ISSUE 25: Need to add the multi-valued profile parameter with 'uif-x' values to the image/tiff MIME Media Type registration and only have a single 'uif' value for the 'application' parameter (instead of 'uif-s', 'uif-c', 'uif-l', etc.).

See UIF issue 4.

ISSUE 26 : OK to REQUIRE the Sender to supply the "ippfax-uif profiles" of the document being sent? What if the Sender didn't create the document?

No. If "ippfax-uif-profiles" value differs with actual UIF data, the UIF data takes precedence. This field is useful as a hint or for rejecting a job if the given profile is not supported.

ISSUE 27: Need to fill in the TBD entries to indicate the IPPFAX semantics for the Job Template attributes.

(Postponed)

ISSUE 28 (8.1): Are the entries in the Operations Conformance Table 6 correct?

(Postponed)

ISSUE 29: What does the 'tel' scheme do for IPPFAX?

Patrick Pidduck would like to table the whole offramp discussion, as detail needs to be given about the whole route. The group agreed this whole issue needs to be readdressed perhaps during a later revision of the specification.

10.1.1 needs to be renamed (maybe ippfax-final-destination-uri), and the description needs to make it clear that routing can happen transparently to ipp. We agreed we need to keep the "final-destination-uri" attribute. We agreed to discard the schemes supported as well as the retry attributes.

Also, the final destination could be the URN of the recipient, even if the receiver is not acting as an off-ramp.

ISSUE 30: OK that we got rid of the new 'octetString32k' attribute syntax and use existing IPP/1.1 attribute syntaxes, so that existing IPP systems can be used as gateways?

Yes. (Same as ISSUE 14).

ISSUE 31: Is the description of this new 'client-error-missing-required-attribute' (0x0419) status code sufficient?

We are not going to use this status code (see ISSUE 07).

ISSUES 32 – 41:
(Postponed)

Next IPPFAX Meetings

Telecon:	Thursday August 9, 2001, 8am – 10am PDT Agenda: discuss remaining IFX and IPPGET issues
Face-to-Face:	Tuesday, October 23 in Seattle (which day is still tentative)