New FileSpec and IPP Document Description attributes for version, etc.
From: Tom Hastings
Date: March 21,2003
File: New-FileSpec-and-IPP-attrs-030321.doc

NOTE: Do NOT accept revision marks in this document!! The revision marks must remain in order to show all of the changes from JDF/1.1a. If you choose to add some edits, please do so with revision marks turned on.

This document contains proposed additions and clarifications to the JDF FileSpec resource for JDF/1.2 and the corresponding IPP Document Description attribute proposed for the IPP Document object extensions. The reason for showing both in the same document is to try to align the semantics where possible. Some of us believe that there will be significant interworking between JDF and other systems, such as IPP. Having the same values of corresponding attributes will make gateways a lot simpler. Note: There is no need for the names of the attributes to be the same. The real gain is for the values.

These proposals build on the proposals from Martin Bailey, Bob Taylor, and Israel Viente in CIP4 and the proposals from Bob Taylor, Dave Hall, and Peter Zehler in the PWG for inclusion in the IPP Document Object, PWG Semantic Model and PWG Print Services Interface (PSI) which is a follow on to Bluetooth. The most recent version results from a telecon on March 21 with Craig Benson, Bob Taylor, Steve Hiebert, and Tom Hastings attempting to accommodate all of the comments received in email.

JDF issues are highlighted in yellow like this.

JDF additions and clarifications are highlighted like this.

IPP issues are highlighted in green like this and are duplicates of the ones in the Document Object specification that the PWG Semantic Model WG is reviewing.

General ISSUE 01: for JDF what to do about enumerations that have already been defined in JDF/1.1, but haven't been using IANA registered values. See http://www.iana.org for all their registries, such as Operating System Names, MIME Types, etc. Should these values be tweaked to make them be spelled like the IANA registered values for JDF/1.2? See below for specific cases. We suggest deprecating the values not in the IANA registry for JDF/1.2.

1 Summary of the additions to JDF and IPP

To align these two specs, add 54 attributes and 1 sub-element to the JDF/1.2 FileSpec resource spec:

- ~~*FileClass attribute*~~
- *FileSizeQuantum* attribute
- *FileTargetDeviceModel*~~*IEEE1284DeviceId*~~ attribute~~,~~
- *FileType attribute*
- *MimeOrFileTypeVersion*~~*FileSubClassFormatVersion*~~ attribute~~,~~
- *TransferEncoding* attribute
- FileTypeList~~FileSpecParts~~ *sub-element*

and clarified OSVersion.

To align these two specs, add 7 Document Description attributes (and their corresponding operation attributes) to the IPP Document object spec:

- document-creator-application-name (name(MAX)),
- document-creator-os-name (name(40)),
- document-creator-application-version (text(127),
- document-format-device-id (text(127)),
- document-format-version (text(127)),
- document-creator-os-version (text(40)),
- document-container-summary (1setOf collection).

Comments and questions on this comparison and the IPP specification are in order.

2 Proposed additions to JDF FileSpec resource for JDF/1.2

Here is the JDF/1.1 FileSpec specification with the proposed additions and clarifications indicated using revision marks:

7.2.55 FileSpec

Modified in JDF 1.2

[Added 5 attributes: *FileSizeQuantum*, *FileTargetDeviceModel*, *FileType*, *MimeOrFileTypeVersion*, and *TransferEncoding* attributes, added 1 sub-element: FileTypeList, modified 8 attributes: *Application*, *AppOS*, *AppVersion*, *DocumentNaturalLang*, *FileSize*, *MimeType*, *OSVersion*, *UserFileName*,16 ISSUES, 1 ISSUE (Ann), 3 ISSUE (Israel), SSUE (O&P), ISSUE (preflight), ISSUE (Rainer), ISSUE (Tom), 2 ACTION (Jim)]

Specification of a file or a set of files.

Resource Properties

Resource class: Parameter
Resource referenced by: Error! Reference source not found.~~**DBMergeParams**~~, Error! Reference source not found.~~**LayoutElement**~~, Error! Reference source not found.~~**PDLResourceAlias**~~, Error! Reference source not found.~~**ScanParams**~~
Example Partition: *Separation*
Input of processes: -
Output of processes: -

Resource Structure

Name	Data Type	Description
Application ? Modified in JDF 1.2	string	Creator application, such as Photoshop, Distiller, Quark, InDesign, or Microsoft Word. Should be the trademarked name. See *AppVersion* for the version of the application.
AppOS ? Modified in JDF 1.2	enumeration	Operating system of the application that created the file. The values are taken from the IANA Operating System Names [iana-os] which is up to 40 upper-case US-ASCII letters, "-", "_", and "/". See *OSVersion* for the version number of the operating system. ~~Possible~~ Example values are: *DG_UX* Deprecated in JDF 1.2 *HP_UX* Deprecated in JDF 1.2 *IRIX* Deprecated in JDF 1.2 LINUX *Linux* Deprecated in JDF 1.2 *Mac* Deprecated in JDF 1.2 MACOS OS/2 *Solaris* SUN-OS UNKNOWN - Default value *Unknown* – Default value Deprecated in JDF 1.2 UNIX WINDOWS *Windows* Deprecated in JDF 1.2

Name	Data Type	Description
AppVersion ? Modified in JDF 1.2	string	Version of ~~the value of~~ the *Application* in human readable form as would be displayed by the creator application to a human user ~~attribute~~. The intent of this attribute is for display to a human being, rather than being parsed by a JDF Consumer program for purpose of affecting the behavior of the programs and so may also include the name of the application, as well as build or service pack numbers. Examples: "Winzip® 8.1 (4331)" "Acrobat 5.0.5 10/26/2001" "Microsoft® Word 2000 (9.0.4119 SR-1)". ISSUE (preflight): Is it agreed that the purpose of AppVersion is for human consumption, rather than program consumption?
CheckSum ? New in JDF 1.1 Modified in JDF 1.1A	hexBinary	Checksum of the file being referenced using the RSA MD5 algorithm. In JDF 1.1a, the term RSA MD was completed to RSA MD5. The data type was modified to hexBinary to accommodate the 128 bit output of the MD5 algorithm.
Compression ? Modified in JDF 1.2	enumeration	Indicates how the file is compressed. Possible values are: *None* – The file is not compressed. Default value. *Deflate* – The file is compressed using ZIP public domain compression ~~(RFC 1951)~~[rfc1951] *Gzip* – GNU zip compression technology ~~(RFC 1952)~~[rfc1952] *Compress* – UNIX compression ~~(RFC 1977)~~[rfc1977]
Disposition ?	enumeration	Indicates what the device should do with the file when the process that uses this resource as an input resource completes. Possible values are: *Unlink* – The device should release the file. *Delete* – The device should attempt to delete the file. *Retain* – The device should do nothing with the file. Default value.
DocumentNaturalLang ? Modified in JDF 1.2	language	The natural language of the document this FileSpec refers to. If the document contains more than one language, the value is the first or primary language of the document. ISSUE: Or should we change "?" to "*" so that all of the languages can be identified in a single document that has more than one language?
FileFormat ?	string	A formatting string used with the *Template* attribute to define a sequence of filenames in a batch process. If neither *URL* nor *UID* are present, both *FileFormat* and *FileTemplate* must be present, unless the resource is a pipe. For more information, see the text following this table.
FileSize ? Modified in JDF 1.2	integer	Size of the file in Bytes in units specified by *FileSizeQuantum* when the size exceeds 2**31-1 bytes.
FileSizeQuantum ? New in JDF 1.2	integer	Unit of measure for *FileSize*. This attribute must not be used for files that are smaller than 2*31 bytes in order to provide backward compatibility and to provide a cannonical representation for file size. Default = 1 ISSUE: OK to add?

Name	Data Type	Description
FileTargetDeviceModel ? New in JDF 1.2	string	ISSUE (O&P): OK, to add *FileTargetDeviceModel* to JDF/1.2? Identifies the model of the Device for which the document was formatted, including manufacturer name when the file is not device-independent. The value of this attribute must exactly match the IEEE 1284-2000 Device ID string, except the length field must not be specified. See the Microsoft Universal Plug and Play [upnp] section 2.2.6 DeviceId parameter for details. Here is an example showing only the required fields for a PostScript document formatted for a LaserBean 9: `MANUFACTURER:ACME Co.;COMMAND SET:PS;MODEL:LaserBean 9;` (See [IEEE1284] clause 7.6)
FileTemplate ?	string	A template, used with *FileFormat*, to define a sequence of filenames in a batch process. If neither *URL* nor *UID* is present, both *FileFormat* and *FileTemplate* must be present, unless the resource is a pipe.
FileType ? New in JDF 1.2	string	Type of file for file formats for which MIME types have not been registered with IANA [iana-mt]. If *FileType* is supplied, then *MimeType* must not be supplied. Example values are: "DCS" – Document Color Separation (DCS) [dcs2.0] "ICC Profile" – International Color Consortium (ICC) File Format for Color Profiles [icc.1] "TIFF/IT" – ISO 12639 - Tag image file format for image technology (TIFF/IT) [iso12639] "Type 1 Font" – Adobe Type 1 Font [type1font] ISSUE: Is "Type 1 Font" the correct title? See Appendix A.1 for a more complete list of examples.
FileVersion ? New in JDF 1.1	string	Version of the file referenced by this FileSpec. ISSUE (Rainer): Is *FileVersion* for file systems that have automatic version numbering, like Tenex and CVS?
MimeOrFileTypeVersion ? New in JDF 1.2	string	The level or version of the file format identified by *MimeType* or *FileType*. Some example values are the same as the Printer MIB [rfc1759] prtInterpreterLangLevel, such as: "1", "2", "3" for *MimeType* = "application/postscript" "3", "4", "5", "5e" for *MimeType* = "application/vnd.hp-PCL" Other example values are for those document formats that have well-defined version identification, either in the specification or as an actual field in the document content, such as: "5.0", "6.0", "2000", "XP" for *MimeType* = "application/msword" "1.3", "1.4", "PDF/X-1:2001", "PDF/X-1a:2001" for *MimeType* = "application/pdf" "2.0" for *FileType* = "DCS" "???" for *FileType* = "ICC Profile" ISSUE (Ann): What versions for ICC Profiles? "TIFF-IT/FP:1998", "TIFF-IT/CT:1998", "TIFF-IT/LW:1998" for *FileType* = "TIFF/IT" See Appendix A.1 "FileSpec Resource examples for *MimeType, FileType, MimeOrFileTypeVersion* attributes" for additional examples in common use by JDF applications.

Name	Data Type	Description
MimeType ? Modified in JDF 1.2	string	Mime type of the file. If *MimeType* is supplied, then *FileType* must not be supplied. See Appendix A.1 for combinations of *MimeType* or *FileType* in combination with *MimeOrFileTypeVersion*. Example *MimeType* values include: "application/msword" – Microsoft Word [iana-mt] "application/pdf" – Adobe Portable Document Format [iana-mt] "application/postscript" – Adobe PostScript™. See [rfc2045, rfc2046, iana-mt] "application/vnd.cip4-jdf+xml" – CIP4 Job Defintion Format [idfmime] "application/vnd.hp-PCL" – Hewlett Packard Printer Control Language (PCL™) [iana-mt] "application/vnd.Quark.QuarkXPress" – Quark Express "application/zip" – public domain ZIP compression [rfc1951]. See also FileTypeList element to summarize the contents. "multipart/related" – related files combined as a MIME multipart package [rfc2387]. See also FileTypeList element to summarize the contents. ISSUE (O&P): Which additional MIME type values from the IPP list should we add to JDF? See IPP List in 2nd half of this document. ISSUE (O&P): What about adding "application/octet-stream" which means auto sense?
OSVersion ? Modified in JDF 1.2	string	Version of the operating system identified by *AppOS*. Possible values include the version number part of the IANA Registry of Operating System Names [iana-os], not including the first HYPHEN (-) character that separates the name from the version. Some example values are: *"2.2."* for *AppOS* = "LINUX": "BSD", "V", "V.1", "V.2", "V.3", or "PC" for *AppOS* = "UNIX": "95", "98", "NT", "NT-5", "2000", or "XP" for *AppOS* = "Windows" See Appendix A.2 "FileSpec Resource examples for *AppOS and OSVersion* attributes" for additional examples in common use by JDF applications.
PageOrder ?	enumeration	Indicates whether the pages in the file are in reverse order. Possible values are: *Ascending* – The first page in the file is the lowest numbered page. *Descending* – The first page in the file is the highest numbered page.
ResourceUsage ?	NMTOKEN	If an element uses more than one FileSpec subelement, this attribute is used to refer from the parent element to a certain child element of this type, for example, see **Error! Reference source not found.**~~ColorSpaceConversionParams~~.

Name	Data Type	Description
TransferEncoding New in JDF 1.2	NMTOKEN	ISSUE (Israel): OR should *TransferEncoding* by a Pipe attribute or some attribute of the transfer mechanis, not the FileSpec resource, instead? Type of transfer encoding for purposes of transferring the files. *TransferEncoding* does not specify the character encoding of the files themselves. When receiving files, the receiver first decodes and applies the attributes in this order: (1) *TransferEncoding* (2) *MimeType* (3) if *MimeType* is a container type (e.g., multipart/related or application/zip), the *MimeType* value(s) in the FileTypeList subelement(s) Possible values include: *Base64* - A format for encoding arbitrary binary information for transmission by electronic mail. *BinHex* - Binhex encoding converts an 8-bit file into a 7-bit format, similar to uuencoding. *Binhex* format preserves file attributes, as well as Macintosh resource forks, and includes CRC (Cyclic Redundancy Check) error-checking. This encoding method works on any type of file, including formatted word processing and spreadsheet files, graphics files, and even executable files (i.e. programs or applications). Encoded files usually have a .HQX extension. Note: *BinHex* is not to be confused with *MacBinary* encoding, which is an 8-bit format. For more information see: http://www.natural-innovations.com/boo/binhex.html#info *MacBinary* -A format that combines the two forks of a Mac file, together with the file information (Name, Creator Application, File Type, etc) into a single binary data stream, suitable for storage or transferring through non-Mac systems. For more information see: http://astronomy.swin.edu.au/~pbourke/dataformats/macbinary/ *None* - Default *UUEncode* - A set of algorithms for converting files into a series of 7-bit ASCII characters that can be transmitted over the Internet. Originally, uuencode stood for Unix-to-Unix encode, but it has since become a universal protocol used to transfer files between different platforms such as Unix, Windows, and Macintosh. Uuencoding is especially popular for sending e-mail attachments. For more information see: http://www.webopedia.com/TERM/U/Uuencode.html
UID ? New in JDF 1.1	string	Unique internal ID of the referenced file. This attribute is dependent on the type of file that is referenced: PDF: Variable unique identifier in the ID field of the PDF file's trailer. ICC Profile: Profile ID in byte 84-99 of the ICC profile header. Others – Format specific.
URL ?	URL	Location of the file. If *URL* is not present, and neither *FileFormat* nor *FileTemplate* are present, the referencing resource must be a pipe.

Name	Data Type	Description
UserFileName ? Modified in JDF 1.2	string	A user-friendly name which may be used to identify the file, but is not guaranteed to be unique.
FileAlias *	element	Defines a set of mappings between file names that may occur in the document and URLs (which may refer to external files or parts of a MIME message).
FileTypeList * New in JDF 1.2 ~~ISSUE: Or should we just recursively refer to FileSpec?~~ ~~ISSUE: If we do, should we also rule out more than one level?~~	element	~~ISSUE: Or do we want a Manifest? Or both? Or be extensible to a Manifest?~~ When *MIMEType* is a container file format, such as "application/zip" or "multipart/related" [rfc2387], the FileTypeList subelement summarizes the distinct types of files in the container file. The purpose of the FileTypeList element is to allow a receiving Device to determine whether or not it supports all of the file formats and versions in the supplied instance of the container. However, the FileTypeList element does not provide means to associate each element instance with a particular file in the container file, so FileTypeList does not provide a "manifest" of the container. There must not be any duplicate FileTypeList elements values, that is, no elements with all the same attribute values; its a set, not a sequence. So 1 PCL file and 100 PostScript files with the same details in a .zip file would have *MimeType* = "application/zip" at the top level and 2 FileTypeList sub-elements: one with MimeType = "application/vnd.hp-PCL" and the other with MimeType = "application/postscript" If a file in a container file is itself a container file, the single FileTypeList element(s) SHOULD contain the flattened distinct collection values for all files at all nested levels. The FileTypeList element(s) is *not* recursively defined to contain further FileTypeList element(s).

Structure of FileAlias Subelement

Modified in JDF 1.2

Name	Data Type	Description
Alias	string	The filename which is expected to occur in the file.
Disposition Deprecated in JDF 1.2	enumeration	Indicates what the device should do with the file referenced by this alias when the process that uses this resource as an input resource completes. Possible values are: *Unlink* – The device should release the file. *Delete* – The device should attempt to delete the file. *Retain* – The device should do nothing with the file. In JDF/1.2, use *Disposition* in FileSpec subelement.
MimeType ? Deprecated in JDF 1.2	string	Mime type of the file. In JDF/1.2, use *MIMEType* in FileSpec subelement.
URL Deprecated in JDF 1.2	URL	The URL which identifies the file the alias refers to. In JDF/1.2, use *URL* in FileSpec subelement.
FileSpec	refelement ISSUE: element or refelement?	List of file formats in the container file. If some of the container files themselves are container files, this list of file formats may include all levels of the container file, or just the next level, depending on implementation. Thus a Device must look at the entire structure if nested, to determine whether or not

Structure of FileTypeList Subelement

ISSUE: Or should we just recursively refer to FileSpec or use the FileAlias subelement which now includes FileSpec and not define this *FileTypeList* subelement?

Name	Data Type	Description
Application ?	string	See FileSpec::*Application*.
AppOS ?	enumeration	See FileSpec::*AppOS*.
AppVersion ?	string	See FileSpec::*AppVersion*.
DocumentNaturalLang ?	language	See FileSpec::*DocumentNaturalLang*.
FileType ?	string	See FileSpec::*FileType*
FileTargetDeviceModel ?	string	See FileSpec::*FileTargetDeviceModel*
MimeOrFileTypeVersion ?	string	See FileSpec::*MimeOrFileTypeVersion* .
MimeType ?	string	See FileSpec::*MimeType*
OSVersion ?	string	See FileSpec::*OSVersion*

Usage of Format and Template

The function defined when using the attributes *FileFormat* and *FileTemplate* is drawn from the same root as the standard C print function and, therefore, overtly resembles the model of that function. *FileFormat* is the first argument and *FileTemplate* is a comma-separated list of the additional arguments. *FileTemplate* may contain the following operators : +,-,*,/,%,(,) which are evaluated using standard C-operator precedence and the variables defined in the following table:

Table 27-186 Predefined variables used in FileTemplate

Name	Description
element	Integer iterator over all elements in a given page. Restarts at 0 for each page.
i	Integer iterator over all files produced by this process. 0-based numbering.
page	Integer iterator over the page number of a document. This is equivalent to r for the case that each run contains exactly one page.
r	Integer iterator over all RunList partitions with a partition key of "*Run*" in an input **RunList**.
ri	Integer iterator over all indices in an input Run of a **RunList**. This index is equivalent to looping over a RunIndex.
sep	Separation as defined in the separation PartIDKey of a partitioned resource.
surf	Surface string, "Front" or "Back"
SheetName	SheetName string of a partitioned resource.
SignatureName	SignatureName string of a partitioned resource.
TileX	X coordinate of a Tile
TileY	Y coordinate of a Tile
PartVersion	PartVersion string of a partitioned resource.
jobPartID	JobPartID string
jobID	Job ID string
jobName	*DescriptiveName* of the Node that is being processed.
Time	Current *Time* in ISO 8601 format.
Date	Current *Date* in ISO 8601 format.
CustomerID	CustomerID

Example:

```
<FileSpec FileFormat = "file://here/next/%s/%4.i/m%4.i.pdf" FileTemplate =
"JobID,i/100,i%100"/>
```

with JobID = "j001" and a **RunList** defining 2023 created files will iterate all created files and place them into:

```
"file://here/next/j001/0000/m0000.pdf"
```

…
"file://here/next/j001/0020/m0023.pdf"

A.1 FileSpec Resource examples for *MimeType*, *FileType*, *MimeOrFileTypeVersion* attributes

New in JDF 1.2

ACTION (Jim): Add this as a new Appendix in JDF/1.2.

This appendix lists examples values for the following attributes of the **FileSpec** resource: *MimeType*, *FileType*, and *MimeOrFileTypeVersion*. The listing is intended to be exhaustive for the most likely document formats that are routinely used in JDF applications. However, other document formats and other combinations of the listed document formats may be used as well. When these format standards are revised with new version numbers, they may be used and should follow the patterns established in this table.

MimeType	*FileType*	*MimeOrFileTypeVersion*	Description
application/msword	must omit	5.0	Microsoft Word [iana-mt]
application/msword	must omit	6.0	Microsoft Word [iana-mt]
application/msword	must omit	2000	Microsoft Word [iana-mt]
application/msword	must omit	XP	Microsoft Word [iana-mt]
application/pdf	must omit	1.3	Adobe Portable Document Format [iana-mt]
application/pdf	must omit	1.4 ISSUE: Any other PDF versions?	Adobe Portable Document Format [iana-mt]
application/pdf	must omit	PDF/X-1:2001	Portable Document Format (PDF) [cgats.12/1] ISSUE: What are the titles?
application/pdf	must omit	PDF/X-1a:2001	Portable Document Format (PDF) [cgats.12/1] ISSUE: What are the titles?
application/pdf	must omit	PDF/X-2:2001	Portable Document Format (PDF) [pdf/x-2] ISSUE: What are the titles?
application/pdf	must omit	PDF/X-3:2001	Portable Document Format (PDF) [pdf/x-3] ISSUE: What are the titles?
application/postscript	must omit	1	Adobe PostScript™ See [rfc2045, rfc2046, iana-mt]
application/postscript	must omit	2	Adobe PostScript™ See [rfc2045, rfc2046, iana-mt]
application/postscript	must omit	3	Adobe PostScript™ See [rfc2045, rfc2046, iana-mt]
application/vnd.cip4-jdf+xml	must omit	1.0	CIP4 Job Defintion Format (JDF) version 1.0, April 2001
application/vnd.cip4-jdf+xml	must omit	1.1	CIP4 Job Defintion Format (JDF) version 1.1, May 2002 ISSUE: Do we need both 1.1 and 1.1a? If not, which one should it be?
application/vnd.cip4-jdf+xml	must omit	1.1a	CIP4 Job Defintion Format (JDF) version 1.1a, August 2002
application/vnd.cip4-jdf+xml	must omit	1.2	CIP4 Job Defintion Format (JDF) version 1.2, ??? 2003
application/vnd.hp-PCL	must omit	3	Hewlett Packard Printer Control Language (PCL™) [iana-mt]
application/vnd.hp-PCL	must omit	4	Hewlett Packard Printer Control Language (PCL™) [iana-mt]
application/vnd.hp-PCL	must omit	5	Hewlett Packard Printer Control Language (PCL™) [iana-mt]
application/vnd.hp-PCL	must omit	5e	Hewlett Packard Printer Control Language (PCL™) [iana-mt]
application/vnd.hp-PCL	must omit	6	Hewlett Packard Printer Control Language (PCL™) [iana-mt]
application/vnd.hp-PCL	must omit	X	Hewlett Packard Printer Control Language (PCL™) [iana-mt]
application/vnd.Quark.QuarkXPress	must omit	??	Quark Express

MimeType	*FileType*	*MimeOrFileTypeVersion*	Description
application/zip	must omit	??	public domain ZIP compression [rfc1951]. See also FileTypeList element to summarize the contents. [iana-mt]
image/jpeg	must omit	??	JPEG [rfc2045, rfc2046, iana-mt]
image/tiff	must omit	??	Tag Image File Format [rfc2302]
multipart/related	must omit	??	related files combined as a MIME multipart package [rfc2387]. See also FileTypeList element to summarize the contents.
must omit	DCS	2.0	Document Color Separation (DCS), version 2.0. [dcs2.0]
must omit	ICC Profile	ISSUE: What version strings for ICC Profiles?	International Color Consortium (ICC) File Format for Color Profiles [icc.1]
must omit	ICC Profile	ISSUE: What version strings for ICC Profiles?	International Color Consortium (ICC) File Format for Color Profiles [icc.1]
must omit[1]	TIFF/IT	TIFF/IT-FP:1998	TIFF/IT [iso12639] - Full Page - baseline
must omit	TIFF/IT	TIFF/IT-CT:1998	TIFF/IT [iso12639] - Continuous Tone picture data - baseline
must omit	TIFF/IT	TIFF/IT-LW:1998	TIFF/IT [iso12639] - Color Line art data - baseline
must omit	TIFF/IT	TIFF/IT-HC:1998	TIFF/IT [iso12639] - High-resolution Continuous tone image data - baseline
must omit	TIFF/IT	TIFF/IT-MP:1998	TIFF/IT [iso12639] - monochrome picture image data - baseline
must omit	TIFF/IT	TIFF/IT-BP:1998	TIFF/IT [iso12639] - Binary Picture image data - baseline
must omit	TIFF/IT	TIFF/IT-BL:1998	TIFF/IT [iso12639] - Binary Line art image data - baseline
must omit	TIFF/IT	TIFF/IT-FP/P1:1998	TIFF/IT [iso12639] - Full Page - profile 1
must omit	TIFF/IT	TIFF/IT-CT/P1:1998	TIFF/IT [iso12639] - Continuous Tone picture data - profile 1
must omit	TIFF/IT	TIFF/IT-LW/P1:1998	TIFF/IT [iso12639] - Color Line art data - profile 1
must omit	TIFF/IT	TIFF/IT-HC/P1:1998	TIFF/IT [iso12639] - High-resolution Continuous tone image data - profile 1
must omit	TIFF/IT	TIFF/IT-MP/P1:1998	TIFF/IT [iso12639] - monochrome picture image data - profile 1
must omit	TIFF/IT	TIFF/IT-BP/P1:1998	TIFF/IT [iso12639] - Binary Picture image data - profile 1
must omit	TIFF/IT	TIFF/IT-BL/P1:1998	TIFF/IT [iso12639] - Binary Line art image data - profile 1
must omit	TIFF/IT	TIFF/IT-FP:2003[2]	TIFF/IT [iso12639] - Full Page - baseline
must omit	TIFF/IT	TIFF/IT-CT:2003	TIFF/IT [iso12639] - Continuous Tone picture data - baseline
must omit	TIFF/IT	TIFF/IT-LW:2003	TIFF/IT [iso12639] - Color Line art data - baseline
must omit	TIFF/IT	TIFF/IT-HC:2003	TIFF/IT [iso12639] - High-resolution Continuous tone image data - baseline
must omit	TIFF/IT	TIFF/IT-MP:2003	TIFF/IT [iso12639] - monochrome picture image data - baseline
must omit	TIFF/IT	TIFF/IT-BP:2003	TIFF/IT [iso12639] - Binary Picture image data - baseline

[1] The file format TIFF/IT must not use the "application/tiff" *MimeType*. The "application/tiff" *MimeType* conforms to baseline TIFF 6.0 [rfc3302 obsoletes rfc2302], where as TIFF/IT does not conform to TIFF 6.0. Consequently, the widely-deployed TIFF 6.0 readers are not able to read TIFF/IT. [rfc3302] requires that an RFC be published in order to extend image/tiff with a parameter that would be needed in order to distinguish TIFF/IT from TIFF. There is no plan by the ISO committee that oversees TIFF/IT to register TIFF/IT with either a parameter to image/tiff or as new separate MIME type. Therefore, TIFF/IT will use the *FileType* attribute instead of the *MimeType* attribute.

[2] The revision of ISO 12639 TIFF/IT is being balloted as a Draft International Standard (DIS) and is expected to be published in the latter half of 2003.

MimeType	*FileType*	*MimeOrFileTypeVersion*	Description
must omit	TIFF/IT	TIFF/IT-BL:2003	TIFF/IT [iso12639] - Binary Line art image data - baseline
must omit	TIFF/IT	TIFF/IT-SD:2003	TIFF/IT [iso12639]
must omit	TIFF/IT	TIFF/IT-FP/P1:2003	TIFF/IT [iso12639] - Full Page - profile 1
must omit	TIFF/IT	TIFF/IT-CT/P1:2003	TIFF/IT [iso12639] - Continuous Tone picture data - profile 1
must omit	TIFF/IT	TIFF/IT-LW/P1:2003	TIFF/IT [iso12639] - Color Line art data - profile 1
must omit	TIFF/IT	TIFF/IT-HC/P1:2003	TIFF/IT [iso12639] - High-resolution Continuous tone image data - profile 1
must omit	TIFF/IT	TIFF/IT-MP/P1:2003	TIFF/IT [iso12639] - monochrome picture image data - profile 1
must omit	TIFF/IT	TIFF/IT-BP/P1:2003	TIFF/IT [iso12639] - Binary Picture image data - profile 1
must omit	TIFF/IT	TIFF/IT-BL/P1:2003	TIFF/IT [iso12639] - Binary Line art image data - profile 1[3]
must omit	TIFF/IT	TIFF/IT-FP/P2:2003	TIFF/IT [iso12639] - Full Page - profile 2
must omit	TIFF/IT	TIFF/IT-CT/P2:2003	TIFF/IT [iso12639] - Continuous Tone picture data - profile 2
must omit	TIFF/IT	TIFF/IT-LW/P2:2003	TIFF/IT [iso12639] - Color Line art data - profile 2
must omit	TIFF/IT	TIFF/IT-HC/P2:2003	TIFF/IT [iso12639] - High-resolution Continuous tone image data - profile 2
must omit	TIFF/IT	TIFF/IT-MP/P2:2003	TIFF/IT [iso12639] - monochrome picture image data - profile 2
must omit	TIFF/IT	TIFF/IT-BP/P2:2003	TIFF/IT [iso12639] - Binary Picture image data - profile 2
must omit	TIFF/IT	TIFF/IT-BL/P2:2003	TIFF/IT [iso12639] - Binary Line art image data - profile 2
must omit	TIFF/IT	TIFF/IT-SD/P2:2003	TIFF/IT [iso12639]
must omit	Type 1 Font	?? ISSUE: What versions for Type 1 Fonts?	Type 1 Font [type1font]
must omit	TrueType Font	?? ISSUE: What versions for True Type Fonts?	TrueType Font [truetypefont]
must omit	OpenType Font	?? ISSUE: What versions for Open Type Fonts?	OpenType Font [opentypefont]

A.2 FileSpec Resource examples for *AppOS and OSVersion* attributes

New in JDF 1.2

ACTION (Jim): Add this as a new Appendix in JDF/1.2.

This appendix lists examples values for the following attributes of the **FileSpec** resource: *AppOS* and *OSVersion*. The listing is intended to be exhaustive for the most likely Operating Systems that are routinely used in JDF applications. However, other Operating Systems and combinations may be used as well. When operating systems have new versions, they may be used and should follow the patterns established in this table.

AppOS	*OSVersion*	Description
LINUX	*2.2*	
MACOS	*X*	

[3] Note: There is no TIFF/IT P1 conformance level of SD in ISO 12639:2003.

AppOS	*OSVersion*	Description
OS/2	?? ISSUE: What versions are there for OS/2?	ISSUE: Should we even list OS/2 as an operation system in common use by JDF applications?
SUN-OS	4.0	
UNIX	BSD	
UNIX	V	
UNIX	V.1	
UNIX	V.2	
UNIX	V.3	
UNIX	PC	
WINDOWS	95	
WINDOWS	98	
WINDOWS	NT	
WINDOWS	NT-5	
WINDOWS	2000	
WINDOWS	XP	

3 Additional references to add to both JDF/1.2 and IPP Document Object specs

Add the following References to JDF/1.2 References section and use these symbolic tags to refer to them throughout the spec:

[cgats.12/1] - CGATS.12/1 Graphic technology—Prepress digital data exchange—Use of PDF for composite data—Part 1: Complete exchange (PDF/X-1).

[dcs2.0] - Document Color Separation (DCS) ISSUE: Need a reference.

[iana-mt] - IANA Registry of MIME Media Types. Available: http://www.iana.org/assignments/media-types

[iana-os] - IANA Registry of Operating System Names. Available: http://www.iana.org/assignments/operating-system-names

[icc.1] - International Color Consortium (ICC) File Format for Color Profiles [icc.1], 2001, Produced by: International Color Consortium (ICC), http://www.color.org, Available at: .http://www.color.org/ICC_Minor_Revision_for_Web.pdf

[ieee1284] - ISSUE (Tom): Get reference

[iso12639] - ISO 12639:1998 Graphic technology -- Prepress digital data exchange -- Tag image file format for image technology (TIFF/IT).

[iso15930] - ISO 15930-1:2001 Graphic technology -- Prepress digital data exchange -- Use of PDF -- Part 1: Complete exchange using CMYK data (PDF/X-1 and PDF/X-1a).

[jdfmime] - "The MIME application/vnd.cip4-jdf+xml Content-Type", Hastings, T., and I McDonald, 25 January 2003, <draft-mcdonald-cip4-jdf-mime-00.txt>, work in progress.

[opentypefont] - Open Type font ISSUE: Is this the correct title? ISSUE: Need a reference.

[pdf/x-2] - Portable Document Format/X-2 ISSUE: Is this the correct title? ISSUE: Need a reference.

[pdf/x-3] - Portable Document Format/X-3 ISSUE: Is this the correct title? ISSUE: Need a reference.

All IETF (Internet Engineering Task Force) RFCs (Request for Comments) are available at RFC Database search: http://www.rfc-editor.org/rfcsearch.html

[rfc1759] - Smith, R., Wright, F., Hastings, T., Zilles, S., and Gyllenskog, J., "Printer MIB", RFC 1759, March 1995.

[rfc1951] - P. Deutsch, “DEFLATE Compressed Data Format Specification version 1.3”, May 1996.

[rfc1952] - P. Deutsch, “GZIP file format specification version 4.3”, May 1996.

[rfc1977] - V. Schryver, “PPP BSD Compression Protocol”, August 1996.

[rfc2045] - N. Freed, N. Borenstein, “Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies”. November 1996. (Updated by RFC2184, RFC2231)

[rfc2046] - N. Freed, N. Borenstein, “Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types”, November 1996. (Updated by RFC2646)

[rfc2302] - G. Parsons, J. Rafferty, S. Zilles, “Tag Image File Format (TIFF) - image/tiff MIME Sub-type Registration”, March 1998. (Obsoleted by RFC3302)

[rfc2387] - E. Levinson, “The MIME Multipart/Related Content-type”, RFC 2387, August 1998.

[rfc3302] - G. Parsons, J. Rafferty, “Tag Image File Format (TIFF) - image tiff MIME Sub-type Registration”, September 2002. (Obsoletes RFC2302)

[truetypefont] - True Type font ISSUE: Is this the correct title? ISSUE: Need a reference.

[type1font] - Adobe Type 1 Font [type1font] ISSUE: Is this the correct title? ISSUE: Need a reference.

4 Comparison of JDF/1.1 FileSpec and IPP/1.1 Document Description attributes

This section is a comparison of the JDF FileSpec resource including additions for JDF/1.2 and the proposed IPP Document object attributes. The IPP semantics are taken from [RFC2911] and the IPP Document object specification, version 0.7, 3/14/03. See ftp://ftp.pwg.org/pub/pwg/ipp/new_DOC/wd-ippdoc10-20030314-doc.zip
The Document Description attributes are proposed to be added to IPP to meet the requirements of a number of print protocols to have additional information about a Document Format, than just its MIME type. So this attribute has additional fields (member attributes) for version, natural language, platform (OS on which the document was generated), and device ID. It also caters to MIME Media Types that are containers, such as application/zip and multipart/related, where contain additional document of possibly differing MIME media types.

It would be good to add the same attributes to JDF, presumably to the FileSpec resource. Also JDF has a number of attributes which would be good to add to the IPP Document object.

The ISSUES for JDF are a repeat of the ones above and so are not numbered.

Here is a comparison of JDF/1.1 FileSpec and IPP/1.1 and IPP "document-format-detail" attribute:

JDF FileSpec attributes and elements	IPP Document Description attributes	Comments
Application (string) Creator application, such as Photoshop, Distiller, Quark, InDesign, or Microsoft Word. Should be the trademarked name. See *AppVersion* for the version of the application.	document-creator-application-name (name(MAX)) This OPTIONAL Document Description member attribute identifies the application that created the document. The version number MUST NOT be included (see the "document-creator-application-version" attribute). Examples: "Photoshop", "Microsoft Word".	Same semantics
AppOS (enumerations) Modified in JDF 1.2 Operating system of the application that created the file. The values are taken from the IANA Operating System Names [iana-os] which is up to 40 upper-case US-ASCII letters, "-", "_", and "/". See *OSVersion* for the version number of the operating system. ~~Possible~~ Example values are:	document-creator-os-name (name(40)) This OPTIONAL Document Description member attribute identifies the name of the operating system on which the document was generated. Valid values are the operating system names defined in the IANA document [os-names] with the version number portion removed (see the "document-creator-os-version" attribute)IANA Operating System Names are consist of up to 40 uppercase US-ASCII letters, hyphen ("-"), period ("."), and slash ("/") characters. The zero length string represents unknown (rather than the UNKNOWN value in the IANA OS Registry, since clients are not expected to localize names). Example IANA OS Registry values:	Same semantics~~, but JDF doesn't use the IANA OS Name Registry, so JDF has some values that represent operating systems that aren't registered with IANA.~~ ~~Clarify JDF *AppOS*~~
DG_UX Deprecated in JDF 1.2	'AIX, 'DOS', 'LINUX', 'MACOS', 'MSDOS', 'MVS', 'NETWARE', 'OS/2', 'SUN', 'SUN-OS', 'UNIX', 'VMS', 'WINDOWS'.	~~ISSUE 11: JDF uses its own enumerations, instead of the~~

JDF FileSpec attributes and elements	IPP Document Description attributes	Comments
HP_UX Deprecated in JDF 1.2 *IRIX* Deprecated in JDF 1.2 LINUX *Linux* Deprecated in JDF 1.2 *Mac* Deprecated in JDF 1.2 MACOS OS/2 *Solaris* SUN-OS UNKNOWN - Default value *Unknown* – Default value Deprecated in JDF 1.2 UNIX WINDOWS *Windows* Deprecated in JDF 1.2		~~IANA OS name registry. Also need to sort JDF values alphabetically.~~
AppVersion *(*string) Version of ~~the value of~~ the ***Application*** in human readable form as would be displayed by the creator application to a human user ~~attribute~~. The intent of this attribute is for display to a human being, rather than being parsed by a JDF Consumer program for purpose of affecting the behavior of the programs and so may also include the name of the application, as well as build or service pack numbers. Examples: ISSUE (preflight): Is it agreed that the purpose of AppVersion is for human consumption, rather than program consumption?	document-creator-application-version (text(127)) ~~The~~ This OPTIONAL Document Description member attribute identifies the version number of the application that created the document. The intent of this attribute is for display to a human being, rather than being parsed by the Printer for purpose of affecting the interpreting by the Printer and so may also include the name of the application, as well as build or service pack numbers. ~~The version number MUST NOT include the application name. See "document-creator-application-name" attribute.~~ Examples: ISSUE: OK that the purpose is human consumption, instead of program consumpation?	Same semantics and values
"Winzip® 8.1 (4331)" "Acrobat 5.0.5 10/26/2001" "Microsoft® Word 2000 (9.0.4119 SR-1)".	~~'V3.0.', 'V6.0'~~"Winzip® 8.1 (4331)" "Acrobat 5.0.5 10/26/2001" "Microsoft® Word 2000 (9.0.4119 SR-1)"	
CheckSum *(*hexBinary) New in JDF 1.1 Modified in JDF 1.1A	-	Handled in IPP by the TLS lower layer security.

JDF FileSpec attributes and elements	IPP Document Description attributes	Comments
Checksum of the file being referenced using the RSA MD5 algorithm. In JDF 1.1a, the term RSA MD was completed to RSA MD5. The data type was modified to hexBinary to accommodate the 128 bit output of the MD5 algorithm.		
Compression (enumeration) Indicates how the file is compressed. Possible values are:	"compression" (type3 keyword) This REQUIRED Document Description attribute ~~i~~identifies the set of supported compression algorithms for document data. Compression only applies to the document data; compression does not apply to the encoding of the IPP operation itself. Standard keyword values are:	Same semantics and values (except for case).
None – The file is not compressed. Default value. *Deflate* – The file is compressed using ZIP public domain compression ~~(RFC 1951)~~[rfc1951] *Gzip* – GNU zip compression technology ~~(RFC 1952)~~[rfc1952] *Compress* – UNIX compression ~~(RFC 1977)~~[rfc1977]	'none': no compression is used. 'deflate': ZIP public domain inflate/deflate) compression technology in RFC 1951 [RFC1951] 'gzip' GNU zip compression technology described in RFC 1952 [RFC1952]. 'compress': UNIX compression technology in RFC 1977 [RFC1977]	
*Disposition (*enumeration) Indicates what the device should do with the file when the process that uses this resource as an input resource completes. Possible values are:	-	ISSUE12: I think LPR has this, right? So should we add "document-disposition" to IPP?
Unlink – The device should release the file. *Delete* – The device should attempt to delete the file. *Retain* – The device should do nothing with the file. Default value.		
*DocumentNaturalLang (*language) The natural language of the document this FileSpec refers to. If the document contains more than one language, the value is the first or primary language of the document. ISSUE: Or should we change "?" to "*" so that all of the languages can be identified in a single document that has more than one language?	"document-natural-language" (naturalLanguage) This OPTIONAL Document Description attribute specified the natural language of the document (see [rfc2911] §3.2.1.1 and [pwg5100.4] §5.1.7). If the document contains more than one language, the value is the first or primary language of the document. The Printer sets this Document Description attribute from the corresponding operation attribute supplied by the client in the Document Creation operation (see section 3.1). The Printer MAY use this value to select fonts or other Globalization processing. Examples include: ~~ISSUE: The definition in [rfc2911] §3.2.1.1 and [pwg5100.4]~~	Same semantics and values

JDF FileSpec attributes and elements	IPP Document Description attributes	Comments
	§5.1.7 is single-valued. OK that this document attribute isn't 1setOf? ISSUE: Or should we extend "document-natural-language" to 1setOf naturalLanguage) and keep the same name? ISSUE: Or change the name to "document-natural-languages"?	
In language data type in Appendix A.1: <Example Language="de"/> - German <Example Language="de-CH"/> - Swiss German <Example Language="en"/> - English <Example Language="en-GB"/> - British English	'en': for English 'en-us': for US English 'fr': for French 'de': for German	
FileFormat (string) A formatting string used with the *Template* attribute to define a sequence of filenames in a batch process. If neither *URL* nor *UID* are present, both *FileFormat* and *FileTemplate* must be present, unless the resource is a pipe. For more information, see the text following this table.	-	Not sure I understand. Don't put in IPP.
FileSize (integer) Size of the file in Bytes in units specified by *FileSizeQuantum* when the size exceeds 2**31-1 bytes.	"k-octets" (integer(0:MAX)) This OPTIONAL "k-octets" Document Description attribute has the same semantics as the corresponding "job-k-octets" Job Description attribute (see [rfc2911] §4.3.17.1) applied to the Document object. The Printer sets this Document Description attribute from the corresponding operation attribute supplied by the client in the Document Creation operation (see section 3.1). "job-k-octets" (integer(0:MAX)) This attribute specifies the total size of the document(s) in K octets, i.e., in units of 1024 octets requested to be processed in the job. The value MUST be rounded up, so that a job between 1 and 1024 octets MUST be indicated as being 1, 1025 to 2048 MUST be 2, etc.	Same semantics, different units of measure.
FileSizeQuantum ? (integer) Unit of measure for *FileSize*. This attribute must not be used for files that are smaller than 2*31 bytes in order to provide backward compatibility and to provide a cannonical representation for file size. Default = 1 ISSUE: OK to add?	-	Not needed in IPP, since the quantum is fixed at 1024 octets.
ISSUE (O&P): OK, to add *FileTargetDeviceModel* to JDF/1.2?	document-format-device-id (text(127))	Same semantics and values

JDF FileSpec attributes and elements	IPP Document Description attributes	Comments
Identifies the model of the Device for which the document was formatted, including manufacturer name when the file is not device-independent. The value of this attribute must exactly match the IEEE 1284-2000 Device ID string, except the length field must not be specified. See the Microsoft Universal Plug and Play [upnp] section 2.2.6 DeviceId parameter for details. Here is an example showing only the required fields for a PostScript document formatted for a LaserBean 9:	This OPTIONAL Document Description member attribute identifies the type of device for which the document was formatted, including manufacturer and model. This attribute is intended to identify document formats that are not portable, e.g., PDLs that are device dependent. The value of this variable MUST exactly match the IEEE 1284-2000 Device ID string (see [IEEE1284] clause 6), except the length field MUST NOT be specified. See the Microsoft Universal Plug and Play [upnp] section 2.2.6 DeviceId parameter for details and examples. Here is an example showing only the required fields for a PostScript document:	
`MANUFACTURER:ACME Co.;COMMAND SET:PS;MODEL:LaserBeam 9;` *(See IEEE 1284-2000 clause 7.6)*	MANUFACTURER:ACME Co.;COMMAND SET:PS;MODEL:LaserBeam 9;	
FileTemplate (string) A template, used with ***FileFormat***, to define a sequence of filenames in a batch process. If neither ***URL*** nor ***UID*** is present, both ***FileFormat*** and ***FileTemplate*** must be present, unless the resource is a pipe.	-	~~Not sure I understand.~~ Don't put in IPP.
FileType (string) Type of file for file formats for which MIME types have not been registered with IANA [iana-mt]. If ***FileType*** is supplied, then ***MimeType*** must not be supplied. Example values are:	-	ISSUE: Don't add to IPP at this time, until there is a request to be able to supply files for printing that don't have mime types registered. It is still the WG 3-year old plan to register all Printer MIB MIME types under application/vnd.pwg-xxx, unless already registered or the vendor doesn't want the PWG to register it at all.
"DCS" – Document Color Separation (DCS) [dcs2.0] "ICC Profile" – International Color Consortium (ICC) File Format for Color Profiles [icc.1] "TIFF/IT" – ISO 12639 - Tag image file format for image technology (TIFF/IT) [iso12639] "Type 1 Font" – Adobe Type 1 Font [type1font] ISSUE: Is		

JDF FileSpec attributes and elements	IPP Document Description attributes	Comments
"Type 1 Font" the correct title? See Appendix A.1 for a more complete list of examples.		
FileVersion (string) New in JDF 1.1 Version of the file referenced by this FileSpec. ISSUE (Rainer): Is *FileVersion* for file systems that have automatic version numbering, like Tenex and CVS?	-	Not sure I understand. Don't put in IPP.
MimeOrFileTypeVersion ? (string) The level or version of the file format identified by *MimeType or, FileType*. Some example values are the same as the Printer MIB [rfc1759] prtInterpreterLangLevel, such as: Other example values are for those document formats that have well-defined version identification, either in the specification or as an actual field in the document content, such as:	document-format-version (text(127)) This REQUIRED Document Description member attribute contains the level or version of the document format identified by the "document-format" member attribute. Possible values are the same as the Printer MIB [rfc1759] prtInterpreterLangLevel (not prtInterpreterLangVersion), such as: Other example values are for those document formats that have well-defined version identification, either in the specification or as an actual field in the document content, such as:~~For those document formats that are defined in standards, the "document-format-version" may contain the official designation of that standard.~~ ~~ISSUE 10: OK that "document-format-version" is REQUIRED for a Printer to support?~~ ~~ISSUE: 11: The problem with separating "document-format" and "document-format-version" is how can a Printer describe what versions are supported, since the versions have to be associated with the document format.~~ Standard text values are:	Same semantics and values
"1", "2", "3" for *MimeType* = "application/postscript" "3", "4", "5", "5e" for *MimeType* = "application/vnd.hp-PCL" "5.0", "6.0", "2000", "XP" for *MimeType* = "application/msword" "1.3", "1.4", "PDF/X-1:2001", "PDF/X-1a:2001" for *MimeType* = "application/pdf" "2.0" for *FileType* = "DCS" ~~"??" for *FileType* = "ICC Profile"~~ ISSUE (Ann): What	'2000': For MS-WORD 2000. '1', '2', '3': For Postscript level 1, 2, 3, respectivelly [rfc1759]. '3', '4', '5', '5e': For PCL 3, 4, 5, 5e, respectively [rfc1759]. '1.3', '1.4': For PDF version 1.3 and 1.4, respectively. '**PDF**/**X**-1:2001', 'PDF/X-1a:2001': For PDF/X-1 and PDF/1-1a, respectively, specified by [iso15930]~~the ISO standard that specifies PDF/X [iso15930]~~. ~~'ISO 12639-1:1996': For ISO 12639:1996 standard that specifies TIFF/IT Profile 1 [ISO 12639]~~	

JDF FileSpec attributes and elements	IPP Document Description attributes	Comments
versions for ICC Profiles? "TIFF/IT-FP:1998", "TIFF/IT-CT:1998", "TIFF/IT-LW:1998" for *FileType* = "TIFF/IT" See Appendix A.1 "FileSpec Resource examples for *MimeType*, *FileType, MimeOrFileTypeVersion* attributes" for additional examples in common use by JDF applications.	'TIFF/IT-FP:1998": For TIFF/IT [iso12639] - Full Page - baseline 'TIFF/IT-CT:1998': For TIFF/IT [iso12639] - Continuous Tone picture data - baseline 'TIFF/IT-FP/P1:1998': For TIFF/IT [iso12639] - Full Page - profile 1 'TIFF/IT-CT/P1:1998': For TIFF/IT [iso12639] - Continuous Tone picture data - profile 1 ~~ISSUE 12: Or should the official ISO standard number, part number and date, be used instead, e.g., "ISO 15930-1:2001"?~~ ~~ISSUE: If so, how do you specify PDF/X-1a which is also specified in Part 1?~~	
MimeType (string) Mime type of the file. If *MimeType* is supplied, then *FileType* must not be supplied. See Appendix A.1 for combinations of *MimeType* or *FileType* in combination with *MimeOrFileTypeVersion*. Example *MimeType* values include:	document-format (mimeMediaType) This REQUIRED Document Description attribute specifies the document format (see [rfc2911] §3.2.1.1) for the Document object. The standard values for this attribute are Internet Media types (sometimes called MIME types). For further details see the description of the 'mimeMediaType' attribute syntax in [rfc2911] section 4.1.9. ~~If it is a MIME Media Type, such as 'multipart/related' or 'application/zip', that is a container format that contains document parts, t~~The "document-format-details~~container-summary~~" attribute summarizes the content (see section 8.2.8) whether the "document-format" is a single document format or a container document format, such as "multipart/related' or 'application/zip'. The Printer sets this Document Description attribute from the corresponding operation attribute supplied by the client in the Document Creation operation (see section 3.1). Example values:	Same semantics and values
"application/msword" – Microsoft Word [iana-mt] "application/pdf" – Adobe Portable Document Format [iana-mt] "application/postscript" – Adobe PostScript™ See [rfc2045, rfc2046, iana-mt] "application/vnd.cip4-jdf+xml" – CIP4 Job Defintion Format [jdfmime] "application/vnd.hp-PCL" – Hewlett Packard Printer Control Language (PCL™) [iana-mt]	'text/html': An HTML document 'text/plain': A plain text document in US-ASCII (RFC 2046 indicates that in the absence of the charset parameter MUST mean US-ASCII rather than simply unspecified) [RFC2046]. 'text/plain; charset=US-ASCII': A plain text document in US-ASCII [RFC2046]. 'text/plain; charset=ISO-8859-1': A plain text document in ISO 8859-1 (Latin 1) [ISO8859-1]. 'text/plain; charset=utf-8': A plain text document in ISO 10646	

JDF FileSpec attributes and elements	IPP Document Description attributes	Comments
"application/vnd.Quark.QuarkXPress" – Quark Express "application/zip" – public domain ZIP compression [rfc1951]. See also FileTypeList element to summarize the contents. "multipart/related" – related files combined as a MIME multipart package [rfc2387]. See also FileTypeList element to summarize the contents. ISSUE (O&P): Which additional MIME type values from the IPP list should we add to JDF? See IPP List in 2nd half of this document. ISSUE (O&P): What about adding "application/octet-stream" which means auto sense?	represented as UTF-8 [RFC2279] 'application/postscript': A PostScript document [RFC2046] 'application/vnd.hp-PCL': A PCL document [IANA-MT] (charset escape sequence embedded in the document data) 'application/pdf': Portable Document Format - see IANA MIME Media Type registry 'application/octet-stream': Auto-sense - see [rfc2911] section 4.1.9.1 'application/zip': ZIP container file package [rfc1951]. See also "document-container-summary" (1setOf collection) which summarizes both the top level file and all levels of the contents. 'multipart/related': related files combined as a MIME multipart package [rfc2387]. See also "document-container-summary" (1setOf collection) which summarizes both the top level file and all levels of the contents.	
OSVersion (string) Version of the operating system identified by *AppOS*. Possible values include the version number part of the IANA Registry of Operating System Names [iana-os], not including the first HYPHEN (-) character that separates the name from the version. Examples of *AppOS* and *OSVersion* values. Some example values are:	document-creator-os-version (text(40)) This OPTIONAL Document Description attribute identifies the version of the operating system on which the document was generated. Valid values include the version portion of any of the operating system names defined in the IANA Registry [os-names]. The value MUST NOT include the name portion of the registered OS name (see "document-creator-os-name" attribute). The zero length string represents unknown (rather than the UNKNOWN value in the IANA OS Registry, since clients are not expected to localize names). Example values for the indicated "document-creator-os-name" value:	Same semantics and values ~~ISSUE: Clarify JDF *OSVersion*.~~ ~~Note: while JDF doesn't reference the IANA Registry, the version numbers will tend to be the same or a superset of the IANA registry.~~
"2.2." for *AppOS* = "LINUX". "BSD", "V", "V.1", "V.2", "V.3", or "PC" for *AppOS* = "UNIX". "95", "98", "NT", "NT-5", "2000", or "XP" for *AppOS* = "Windows" See Appendix A.2 "FileSpec Resource examples for *AppOS and OSVersion* attributes" for additional examples in common use by JDF applications.	For 'AIX': '370', 'PS/2' For 'LINUX': '1.0', '1.2', '2,0', '2.2', '2.4' For 'MVS': 'SP' For 'NETWARE': '3', '3.11', '4.0', '4.1', '5.0' For 'SUN-OS': '3.5', '4.0' For 'UNIX': 'BSD', 'V', 'V.1', 'V.2', 'V.3', 'PC' For 'WINDOWS': '95', '98', 'CE', 'NT', 'NT-2', 'NT-3', 'NT-3.5', 'NT-3.51', 'NT-4', 'NT-5', '2000', 'XP' [not registered yet]	

JDF FileSpec attributes and elements	IPP Document Description attributes	Comments
*PageOrder (*enumeration) Indicates whether the pages in the file are in reverse order. Possible values are:	"page-order-received" (type2 keyword) This attribute specifies the page order of the print-stream pages defined in the document data. The "page-order-received" attribute does not provide any direct processing instructions, it only provides information about the page order so that the client can specify ordinal page numbers with respect to the original source document, rather than having to take into account whether the print stream pages are being sent "one to N" or "N to one". For example, consider such Job Template attributes as "insert-sheet" ([pwg5100.3] section 3.5) and "page-overrides" (see [ipp-override]). See [pwg5100.3] section 2.5 for a complete discussion of print-stream page order. Standard keyword values are:	Same semantics, mappable values
Ascending – The first page in the file is the lowest numbered page. *Descending* – The first page in the file is the highest numbered page.	'1-to-n-order' 'n-to-1-order'	Mappable values
ResourceUsage (NMTOKEN) If an element uses more than one FileSpec subelement, this attribute is used to refer from the parent element to a certain child element of this type, for example, see **Error! Reference source not found.**~~ColorSpaceConversionParams~~.	-	Don't put in IPP.
*TransferEncoding (*NMTOKEN*)* ISSUE (Israel): OR should *TransferEncoding* by a Pipe attribute or some attribute of the transfer mechanis, not the FileSpec resource, instead? Type of transfer encoding for purposes of transferring the files. *TransferEncoding* does not specify the character encoding of the files themselves. When receiving files, the receiver first decodes and applies the attributes in this order: (1) *TransferEncoding* (2) *MimeType* (3) if *MimeType* is a container type (e.g., multipart/related or application/zip), the *MimeType* value(s) in the FileTypeList subelement(s) Possible values include:	-	Don't add to IPP. It's a function of the transport. ~~now, but add "document transfer encoding" to IPP in the future when needed.~~

JDF FileSpec attributes and elements	IPP Document Description attributes	Comments
Base64 - A format for encoding arbitrary binary information for transmission by electronic mail. *BinHex* - Binhex encoding converts an 8-bit file into a 7-bit format, similar to uuencoding. *Binhex* format preserves file attributes, as well as Macintosh resource forks, and includes CRC (Cyclic Redundancy Check) error-checking. This encoding method works on any type of file, including formatted word processing and spreadsheet files, graphics files, and even executable files (i.e. programs or applications). Encoded files usually have a .HQX extension. Note: *BinHex* is not to be confused with *MacBinary* encoding, which is an 8-bit format. For more information see: http://www.natural-innovations.com/boo/binhex.html#info *MacBinary* -A format that combines the two forks of a Mac file, together with the file information (Name, Creator Application, File Type, etc) into a single binary data stream, suitable for storage or transferring through non-Mac systems. For more information see: http://astronomy.swin.edu.au/~pbourke/dataformats/macbinary/ *None* - Default *UUEncode* - A set of algorithms for converting files into a series of 7-bit ASCII characters that can be transmitted over the Internet. Originally, uuencode stood for Unix-to-Unix encode, but it has since become a universal protocol used to transfer files between different platforms such as Unix, Windows, and Macintosh. Uuencoding is especially popular for sending e-mail attachments. For more information see: http://www.webopedia.com/TERM/U/Uuencode.html		
UID (string) Unique internal ID of the referenced file. This attribute is dependent on the type of file that is referenced: PDF: Variable unique identifier in the ID field of the PDF file's	-	Don't put in IPP.

JDF FileSpec attributes and elements	IPP Document Description attributes	Comments
trailer. ICC Profile: Profile ID in byte 84-99 of the ICC profile header. Others – Format specific.		
*URL (*URL*)* Location of the file. If *URL* is not present, and neither *FileFormat* nor *FileTemplate* are present, the referencing resource must be a pipe.	"document-uri" (uri) The URI of the document to be printed by reference. The Printer pulls the file at its convenience.	Same semantics and values.
UserFileName (string) A user-friendly name which may be used to identify the file, but is not guaranteed to be unique.	-	~~Clarify JDF *UserFileName*~~ Don't add "user-file-name" to IPP until requested.
FileAlias *(element) Defines a set of mappings between file names that may occur in the document and URLs (which may refer to external files or parts of a MIME message).	-	ISSUE: Add to IPP too? ~~Not sure I understand.~~ ~~Don't put in IPP.~~
FileTypeList * (element) ~~ISSUE: Or do we want a Manifest? Or both? Or be extensible to a Manifest?~~ When *MIMEType* is a container file format, such as "application/zip' or "multipart/related" [rfc2387], the FileTypeList subelement summarizes the distinct types of files in the container file. The purpose of the FileTypeList element is to allow a receiving Device to determine whether or not it supports all of the file formats and versions in the supplied instance of the container. However, the FileTypeList element does not provide means to associate each element instance with a particular file in the container file, so FileTypeList does not provide a "manifest" of the container. There must not be any duplicate FileTypeList elements values, that is, no elements with all the same attribute values; its a set, not a sequence. So 1 PCL file and 100 PostScript files with the same details in a .zip file would have *MimeType* = "application/zip" at the top level and 2 FileTypeList sub-elements: one with MimeType = "application/vnd.hp-PCL" and the other with MimeType = "application/postscript"	document-format-details ~~container-summary~~ (1setOf collection) This OPTIONAL Document Description attribute summarizes the details of the document content (see the member attributes in **Error! Reference source not found.**). ~~document format content of the body parts, if the document's document format is a container type, such as 'multipart/related' or 'application/zip'.~~ If a Printer supports ~~such~~ a container MIME type, such as 'multipart/related' or 'application/zip', the Printer MUST support this "document-format-details~~container-summary~~" Document Description attribute ~~and all the member attributes in Table 10 that the Printer supports as top level Document Description attributes~~. The purpose of the "document-format-details~~container-summary~~" attribute is to allow a receiving Printer to determine whether or not it supports ~~all of~~ the exact document format or formats in the supplied instance of the ~~container~~Document object. ~~However,~~ For a container format, the "document-format-details~~container-summary~~" attribute does not provide means to associate each collection value with particular document in the archive file, so it does not provide a "manifest" of the container. The first collection value provides the details for the container format itself and the remaining collection values provides the summary of the details of the contained files.	Same semantics, except that IPP doesn't have the equivalent of *FileType* attribute.

JDF FileSpec attributes and elements	IPP Document Description attributes	Comments
If a file in a container file is itself a container file, the single FileTypeList element(s) SHOULD contain the flattened distinct collection values for all files at all nested levels. The FileTypeList element(s) is *not* recursively defined to contain further FileTypeList element(s). ISSUE: Or should we just recursively refer to FileSpec or use the FileAlias subelement which now includes FileSpec and not define this *FileTypeList* subelement?	~~The member attributes defined for this collection are listed in Table 10 and are the same as those defined for the corresponding Document Description attributes themselves, i.e., a recursive definition. But t~~There MUST NOT be any duplicate collection values, that is, no collection values with all the same member attribute values; its a set, not a sequence. So 100 PostScript files with the same details in a .zip file would have 'application/zip' as the MIME type for the top level "document-format" Document Description attribute for the Document object and a "document-format-details~~container-summary~~" Document Description collection attribute with only one collection value containing a "document-format" member attribute with the 'application/postscript' MIME type value. If a file in a container file is itself a container file, the single "document-format-details~~container-summary~~" (1setOf collection) Document Description attribute SHOULD contain the flattened distinct collection values for all files at all nested levels. Therefore, "document-format-details~~container-summary~~" attribute is not recursively defined to contain "document-format-details". ~~ISSUE 06: OK that "document-container-summary" is only one level deep?ISSUE 07: Is the description of "document-container-summary" attribute OK?~~	
Attributes defined for in FileTypeList: *Application ?* *AppOS ?* *AppVersion ?* *DocumentNaturalLang ?* *FileType* *FileTargetDeviceModel ?* *MimeOrFileTypeVersion ?* *MimeType* *OSVersion ?*	Member attributes of the "document-format-details" are: document-creator-application-name (name(MAX)) document-creator-application-version (text(127)) document-creator-os-name (name(40)) document-creator-os-version (text(40)) document-format (mimeMediaType) document-format-device-id (text(127)) document-format-version (text(127)) document-natural-language (naturalLanguage)	