The Printer Working Group

28 August 2017
Best Practices

Mapping CIP4 JDF to PWG Print Job Ticket v1.0 (JDFMAP)

Status: Approved

Abstract: For interoperability between job ticket-based printing systems, this Best Practices document defines a recommended mapping from XML objects and attributes in the CIP4 Job Definition Format [CIP4JDF] to XML elements in the PWG Print Job Ticket and Associated Capabilities [PWG5108.7].

This document is a PWG Best Practices. For a definition of a "PWG Best Practices", see:

http://ftp.pwg.org/pub/pwg/general/pwg-process30.pdf

This document is available electronically at:

http://ftp.pwg.org/pub/pwg/informational/bp-smjdfmap10-20170828.pdf
http://ftp.pwg.org/pub/pwg/ informational/bp-smjdfmap10-20170828.pdf

About the IEEE-ISTO

The IEEE-ISTO is a not-for-profit corporation offering industry groups an innovative and flexible operational forum and support services. The IEEE-ISTO provides a forum not only to develop standards, but also to facilitate activities that support the implementation and acceptance of standards in the marketplace. The organization is affiliated with the IEEE (http://www.ieee.org/) and the IEEE Standards Association (http://standards.ieee.org/).

For additional information regarding the IEEE-ISTO and its industry programs visit:

http://www.ieee-isto.org

About the IEEE-ISTO PWG

The Printer Working Group (or PWG) is a Program of the IEEE Industry Standards and Technology Organization (ISTO) with member organizations including printer manufacturers, print server developers, operating system providers, network operating systems providers, network connectivity vendors, and print management application developers. The group is chartered to make printers and the applications and operating systems supporting them work together better. All references to the PWG in this document implicitly mean “The Printer Working Group, a Program of the IEEE ISTO.” In order to meet this objective, the PWG will document the results of their work as open standards that define print related protocols, interfaces, procedures and conventions. Printer manufacturers and vendors of printer related software will benefit from the interoperability provided by voluntary conformance to these standards.

In general, a PWG Best Practices document is stable, well understood, and technically competent, and enjoys significant public support.

For additional information regarding the Printer Working Group visit:

http://www.pwg.org

Contact information:

The Printer Working Group
c/o The IEEE Industry Standards and Technology Organization
445 Hoes Lane
Piscataway, NJ 08854
USA

About the Semantic Model Work Group

The Semantic Model Work Group is responsible for the modeling of the services - Print, Copy, Scan, Fax, Resource, System Control, and Transform hosted on Multifunction Devices . Standardization of the Multifunction Device (MFD) model and semantics will support interoperability of devices and services in local and enterprise networks enabling improved job submission, job management, remote administration, and support. The goal of the project is to define a unified semantic model and set of abstract operations for the most common and essential service and device features of the Multifunction Device. For additional information regarding SM visit:

http://www.pwg.org/sm

Implementers of this Best Practices document are encouraged to join the SM mailing list in order to participate in any discussions of the document. Suggested additions, changes, or clarification to this document, should be sent to the SM mailing list for consideration.

Table of Contents

List of Tables

1. Introduction

For interoperability between job ticket-based printing systems, this Best Practices document defines a recommended mapping from XML objects and attributes defined in the CIP4 Job Definition Format [CIP4JDF] to XML elements defined in the PWG Print Job Ticket and Associated Capabilities [PWG5108.07].

2. Terminology

2.1 Conformance Terminology

Capitalized terms, such as RECOMMENDED, SHOULD, SHOULD NOT, MAY, and OPTIONAL, have special meaning relating to conformance as defined in Key words for use in RFCs to Indicate Requirement Levels [RFC2119].

2.2 Printing Terminology

Definitions and semantics of printing terms are imported from the Printer MIB v2 [RFC3805], Printer Finishings MIB [RFC3806], and Internet Printing Protocol/1.1: Model and Semantics [RFC8011].

This document also defines the following protocol roles in order to specify unambiguous conformance recommendations:

Client: Initiator of outgoing IPP session requests and sender of outgoing IPP operation requests.

Printer: Listener for incoming IPP session requests and receiver of incoming IPP operation requests that represents one or more Physical Devices or a single Logical Device.

Proxy: A Client that sends configuration and status information to and retrieves and manages Jobs and Documents from a shared Logical Device on behalf of one or more Output Devices.

2.3 Other Terminology

Document: An object created and managed by a Printer that contains the description, processing, and status information. A Document object may have attached data and is bound to a single Job.

Job: An object created and managed by a Printer that contains description, processing, and status information. The Job also contains zero or more Document objects.

Logical Device: a print server, software service, or gateway that processes jobs and either forwards or stores the processed job or uses one or more Physical Devices to render output.

Output Device: a single Logical or Physical Device.

Physical Device: a hardware implementation of an print device (that includes a marking engine).

2.4 Acronyms and Organizations

CIP4: International Cooperation for the Integration of Processes in Prepress, Press, and Postpress Organization, **http://www.cip4.org/**

IANA: Internet Assigned Numbers Authority, **http://www.iana.org/**

IEEE: Institute of Electrical and Electronics Engineers, **http://www.ieee.org/**

IETF: Internet Engineering Task Force, **http://www.ietf.org/**

JDF: CIP4 Job Definition Format, **http://www.cip4.org/**

ISO: International Organization for Standardization, **http://www.iso.org/**

NEA: IETF Network Endpoint Assessment WG, **http://datatracker.ietf.org/wg/nea/**

PJT: PWG Print Job Ticket, **http://www.pwg.org/**

PWG: IEEE-ISTO Printer Working Group, **http://www.pwg.org/**

3. Requirements

3.1 Rationale for Mapping of JDF to PJT

Before open print standards were developed, in order to print a document from a mainframe, desktop, or mobile computer, it was necessary to understand the characteristics, capabilities, and configured defaults of the target Printer: accessibility, supported document formats, color capability, duplex capability, etc. When submitting a Job, it was necessary to specify (or accept default) values for each of these Printer capabilities. When the Printer supported only unidirectional or simple print protocols, Printer capabilities were typically statically defined in device-specific drivers, Adobe PostScript Printer Description (PPD) files, and/or user-entered configuration information. The Printer capabilities to be used for a given Job were also embedded in document description formats such as Adobe Postscript and HP Printer Control Language (PCL). Later, as printers and print protocols evolved, capabilities and Job information were exchanged via separate commands using Adobe PostScript, HP Printer Job Language (PJL). Finally, as open print standards evolved, CIP4 JDF [CIPJDF4] and IETF/PWG Internet Printing Protocol (IPP) [RFC8011] [PWG5100.12] became widespread in production printers and digital network printers, respectively.

Creating, managing, storing, and accessing Printer-specific drivers and PPDs was always time-consuming, storage-intensive, and error-prone and had high implementation costs for Printer and operating system manufacturers.

It is not feasible to determine Printer capabilities when there are a large number of potential targets, when the target Printers are remote, and/or when there is no direct communication possible with each Printer.

To address the limitations of traditional print protocols, two types of data elements have evolved in CIP4 and PWG standards:

1) A set of elements with value ranges defining Printer capabilities (allowed processing choices for Jobs);

2) A set of element values specifying the capabilities values to be used for a specific Job (Print Job Ticket).

To enable interoperability, the CIP4 JDF [CIP4JDF] data elements should be consistently mapped to the corresponding data elements in the PWG Print Job Ticket and Associated Capabilities [PWG5108.07].

3.2 Use Cases for Mapping of JDF to PJT

This Best Practices document does not define any new elements, structure or protocol; it correlates different sets of elements defined elsewhere. Therefore, end-user scenarios are not applicable. Rather, users of this information will be print service designers and clients that interface with print services seeking to better provide and use the Printer Capabilities and submitted Print Job Ticket information.

3.2.1 Print Service Designer

A print service designer is responsible for interfacing Printer devices from multiple vendors with several remote print services requiring Printer Capabilities and submitting Print Job Tickets using various standard but differing representations. All of the target Printers support IPP [RFC8011] [PWG5100.12], and as such can be considered to offer interfaces fully compliant with the PWG Print Job Ticket [PWG5108.07]. The print service designer needs a consistent guide in mapping the capabilities elements accessible from the Printers via IPP to the Printer Capabilities structures needed by JDF model [CIP4JDF]. Similarly, the print service designer needs a consistent guide in translating the job ticket information provided in JDF job tickets into IPP attributes.

3.2.2 Print Device Designer

A print device designer has a new, improved product that the marketing group wants to advertise as out-of-the box compatible with systems using various representations for Printer Capabilities and Print Job Tickets. This new Printer supports IPP [RFC8011] [PWG5100.12], and the print device designer wants to be able to support IPP and the data structures supplied in JDF Job Tickets [CIP4JDF] using a common internal mapping database.

3.3 Out-of-Scope for Mapping of JDF to PJT

The following are out of scope for thisBest Practices document:

1) Creation of any new element not in either the PWG model or the method to which it is being cross mapped.

2) Definition of a specific structure of how the Printer Characteristics and/or Job Ticket elements are to be presented.

3.4 Design Requirements for Mapping of JDF to PJT

This Best Practices document should:

1) Follow the naming conventions in PWG Print Job Ticket [PWG5108.07].

2) Conform to the specifications for each of the Printer Characteristic and each of the Print Job Ticket elements in [CIP4JDF] and [PWG5108.07].

3) Map elements from [CIP4JDF] to [PWG5108.07] in a one-to-one manner, whenever possible.

4) Whenever a one-to-one element mapping is not possible, define the best practice for complex mapping between elements and groups of elements.

5) Define conformance recommendations for implementations of Print Servers and Print Devices.

4. Mapping of CIP4 Job Definition Format (JDF)

4.1 Mapping JDF Attributes to PWG Print Job Ticket Elements

JDF attributes are encoded as XML attributes in XML schema instances – instead of as XML elements as in the PWG Print Job Ticket (PJT) – this complicates JDF/PJT mapping implementations. JDF attributes are defined in the CIP4 JDF Specification Release 1.5 [CIP4JDF] and profiled for various printing environments in a series of CIP4 ICS (Interoperability Conformance Specification) documents, in particular the CIP4 Integrated Digital Printing (IDP) ICS v1.3 [CIP4IDP]. The reader is also directed to the section "Mapping of parameters into JDF elements and attributes" on pages 167-173 of Adobe PDF Creation Settings [ADOBECREATE] for information on the extent of JDF support and mapping choices in Adobe Acrobat 9.0.

Notes:

1) JDF uses 0-based ranges. The PWG PJT and IPP use 1-based ranges. Implementations of mappings defined in Table 1 need to convert ranges accordingly.

2) JDF object/attribute and PWG element pairs listed in Table 1 in bold have corresponding complex mapping discussion sections below Table 1, based on the PWG element name.

3) In PWG Web Service and IPP protocol bindings, certain elements (attributes) are sent as operation parameters ("invisible" because not saved to the corresponding Job/Document objects). XxxSupplied elements in the PWG JobTicket and JobReceipt (and corresponding IPP objects) record these elements (attributes) for each Job/Document.

4) In the future, CIP4 may deprecate the usage of the LayoutPreparation process and replace it with the Stripping process which is more general.

5) In the future, PWG may define a Compression attribute that applies to individual Documents.

Table 1 – Mapping of CIP4 JDF to PWG PJT Elements

CIP4 JDF Object or Attribute	PWG Print Job Ticket Element
FileSpec/*@Compression*	**CompressionSupplied** (note 3) (note5)
[Output] ComponentLink/*@Amount*	Copies
<none>	CoverBack CoverType Media MediaCol

CIP4 JDF Object or Attribute	PWG Print Job Ticket Element
<none>	CoverFront CoverType Media MediaCol
FileSpec/@*Encoding*	**DocumentCharsetSupplied** (note 3)
FileSpec/*@CheckSum*	DocumentDigitalSignatureSupplied (note 3)
FileSpec/@*MimeType*	**DocumentFormat**
FileSpec /*@Application* /*@AppVersion* /*@AppOS* /@OSVersion /*@MimeType* /*@FileTargetDeviceModel* <none> /*@DocumentNaturalLang*	**DocumentFormatDetailsSupplied** (note 3) DocumentSourceApplicationName DocumentSourceApplicationVersion DocumentSourceOsName DocumentSourceOsVersion DocumentFormat DocumentFormatDeviceId DocumentFormatVersion DocumentNaturalLanguage
<none>	DocumentMessageSupplied (note 3)
FileSpec/*@UserFileName*	DocumentName
FileSpec/*@Password*	DocumentPassword
FileSpec/*@DocumentNaturalLang*	ElementsNaturalLanguage
[Input] ComponentLink/*@Orientation*	FeedOrientation
Finishings	**Finishings**
Finishings Wrapping/WrappingParams /*@WrappingKind* <none> **XxxBinding** (class) or **BindingIntent/** /XxxParams/HoleMakingParams/ *@HoleReferenceEdge* ***@BindingType*** **VarnishingParams** *@Side* (Front, Back, empty) Ink/*@SpecialInk* <none> Folding/Fold /*@To* /*@Travel* /*@From* LaminatingParams /*@Side* Media/*@FrontCoatings* HoleMaking/HoleMakingParams /*@Hole* /*@Center* ComponentLink/*@Orientation* StitchingParams	**FinishingsCol** FinishingTemplate Bailing BailingType BailingWhen **Binding** BindingReferenceEdge **BindingType** **Coating** CoatingSides **CoatingType** Covering CoveringName Folding FoldingDirection FoldingOffset FoldingReferenceEdge Laminating LaminatingSides LaminatingType Punching PunchingLocations PunchingOffset PunchingReferenceEdge Stitching

CIP4 JDF Object or Attribute	PWG Print Job Ticket Element
/*@StitchPositions* /*@Offset* ComponentLink/*@Orientation* **CuttingParams/Cut** or **PerforatingParams/Perforate** or **CreasingParams/Crease** /*@StartPosition* /*@WorkingPath* <implicit> <none>	StitchingLocations StitchingOffset StitchingReferenceEdge **Trimming** TrimmingOffset TrimmingReferenceEdge **TrimmingType** TrimmingWhen
FontPolicy/*@PreferredFont*	FontNameRequested
<none>	FontSizeRequested
LayoutPreparationParams/InsertSheet/ *@SheetUsage*=”FillForceFront” (and) *@SheetType*=”FillSheet”	ForceFrontSide (note 4)
LayoutPreparationParams/ **ExternalImpositionTemplate/FileSpec/** *@UserFileName*	**ImpositionTemplate** (note 4)
RunList/@Page (divided by) LayoutPreparationParams/NumberUp	Impressions
RunList/InsertSheet <complex mapping> <none> Layout/Media Layout/Media	**InsertSheet** ISheet InsertAfterPage InsertCount Media MediaCol
CustomerInfo/*@BillingCode*	JobAccountingID
InsertSheet *@SheetType*= ”AccountingSheet” <none> Layout/Media Layout/Media <none>	JobAccountingSheets JobAccountingSheetsType Media MediaCol JobAccountingOutputBin
CustomerInfo/*@CustomerID*	JobAccountingUserID
[Output] ComponentLink/*@Amount*	JobCopies
<none>	JobCoverBack CoverType Media MediaCol
<none>	JobCoverFront CoverType Media MediaCol
<none>	JobDelayOutputUntil
NodeInfo/@FirstStart	JobDelayOutputUntilTime

CIP4 JDF Object or Attribute	PWG Print Job Ticket Element

InsertSheet
@SheetType=”ErrorSheet”
  <none>
  <none>
  Layout/Media
  Layout/Media
	JobErrorSheet
	
	  JobErrorSheetType
	  JobErrorSheetWhen
	  Media
	  MediaCol

Finishings	**JobFinishings**
Finishings	**JobFinishingsCol**
<none>	JobHoldUntil
NodeInfo/*@FirstStart*	JobHoldUntilTime
@SettingsPolicy=”MustHonor” (generic JDF element attribute)	JobMandatoryElements
JDF/Comment[*@Name*=”OperatorText”]	JobMessageFromOperator
JDF/Comment[*@Name*=”Instruction”]	JobMessageToOperator
<none>	JobMoreInfo
CustomerInfo/*@CustomerJobName*	JobName
Contact/*@UserID*	JobOriginatingUserName
<none>	JobOriginatingUserUri
<none>	JobPassword
<none>	JobPasswordEncryption
CustomerInfo/Contact/ComChannel/*@Locator*	JobPhoneNumber
NodeInfo/@*Priority*	**JobPriority**
CustomerInfo/Contact/Person	**JobRecipientName**

<none>	JobSaveDisposition
	  SaveDisposition
	  SaveInfo
	  SaveDocumentFormat
	  SaveLocation
	  SaveName

<none>	JobSheetMessage
InsertSheet/*@SheetType*=”JobSheet”	JobSheets

InsertSheet
@SheetType=”JobSheet”
  @SheetUsage
  Layout/Media
  Layout/Media
	JobSheetsCol
	
	  JobSheets
	  Media
	  MediaCol

QueueEntryId	**JobUuid**
<none>	KOctets
Media	**Media**

CIP4 JDF Object or Attribute	PWG Print Job Ticket Element
Media	**MediaCol**
Media/*@BackCoatings*	MediaBackCoating
Media/@*MediaColorName*	MediaColor
Media/*@FrontCoatings*	MediaFrontCoating
Media/*@GrainDirectio*n	MediaGrain
Media/*@HoleTyp*e & Media/HoleList	MediaHoleCount
Media/Comment/*@Name*= "Description"	MediaInfo
Media/GeneralID	MediaKey
Media/*@MediaSetCoun*t	MediaOrderCount
Media/@*PrePrinted*	**MediaPreprinted**
Media/@*RecycledPercentage*	**MediaRecycled**
Media/@*Dimension*	**MediaSize**
(see Media/*@Dimension*)	XDimension
(see Media/*@Dimension*)	YDimension
<none>	**MediaSizeName**
Media/*@Thickness*	MediaThickness
Media/*@Texture*	MediaTooth
Media/*@MediaTypeDetails*	MediaType
Media/*@Weight*	MediaWeightMetric
DigitalPrintingParams/*@NonPrintableMarginBottom*	MediaBottomMargin
DigitalPrintingParams/*@NonPrintableMarginLeft*	MediaLeftMargin
DigitalPrintingParams/*@NonPrintableMarginRight*	MediaRightMargin
DigitalPrintingParams/*@NonPrintableMarginTop*	MediaTopMargin
Media/*@Location* (input tray name)	MediaSource
<none>	MediaInputTrayCheck
[Input] MediaLink/*@Amount*	MediaSheets
<none>	MultipleDocumentsHandling
LayoutPreparationParams/@*NumberUp*	**NumberUp**
LayoutPreparationParams/*@Rotate*	OrientationRequested
DigitalPrintingParams/*@OutputBin*	OutputBin
<none>	OutputDevice

CIP4 JDF Object or Attribute	PWG Print Job Ticket Element
RunListLink/Part or other **Link/Part** *@DocRunIndex* ***@DocIndex*** *@DocCopies* *<other JDF attribute>*	**Overrides** Override Pages PageRange Lowerbound Upperbound **DocumentNumbers** NumberRange Lowerbound Upperbound DocumentCopies PCopiesRange Lowerbound Upperbound **OverridingElements**
DigitalPrintingParams/*@PageDelivery*	PageDelivery
FileSpec/*@PageOrder*	PageOrderReceived
RunList/*@Pages* <none> (see Overrides) (see Overrides)	**PageRanges** PageRange Lowerbound Upperbound
<none>	PagesPerSubset Pages
<none>	PdlInitFiles PdlInitFile PdlInitFileLocation PdlInitFileName PdlInitFileEntry
LayoutPreparationParams/ *@PresentationDirection*	**PresentationDirectionNumberUp**
ColorantControl/*@ProcessColorModel*	**PrintColorMode**
<none>	PrintContentOptimize
ColorSpaceConversionParams/ ColorSpaceConversionOp/*@RenderingIntent*	**PrintRenderingIntent**
<none**>** DigitalPrintingParams/ *@DirectProofAmount* <none> <none>	**ProofPrint** Copies Media MediaCol
InterpretingParams/*@PrintQuality*	Quality
RenderingParams/ObjectResolution/*@Resolution*	Resolution
InsertSheet *@SheetType*=”SeparatorSheet” *@SheetUsage* Layout/Media Layout/Media	SeparatorSheets SeparatorSheetsType Media MediaCol
DigitalPrintingParams/*@Collate*	SheetCollate
DigitalPrintingParams/*@Sides*	Sides
<none>	TemplateCreatorUserName

CIP4 JDF Object or Attribute	PWG Print Job Ticket Element
<none>	TemplateId
JDF/*@TemplateVersion*	**TemplateInfo**
JDF/*@TemplateID*	**TemplateName**
<none>	TemplateType
LayoutPreparationParams/ImageShift/ *@PositionX*	**XImagePosition**
LayoutPreparationParams/ImageShift/ *@ShiftFront*	**XImageShift**
LayoutPreparationParams/ImageShift/ *@ShiftFront*	**XSide1ImageShift**
LayoutPreparationParams/ImageShift/ *@ShiftBack*	**XSide2ImageShift**
LayoutPreparationParams/ImageShift/ *@PositionY*	**YImagePosition**
LayoutPreparationParams/ImageShift/ *@ShiftFront*	**YImageShift**
LayoutPreparationParams/ImageShift/ *@ShiftFront*	**YSide1ImageShift**
LayoutPreparationParams/ImageShift/ *@ShiftBack*	**YSide2ImageShift**

4.1.1 Binding

PWG PJT Mapping: Binding

See detailed mappings for various JDF binding classes and attributes in section 4.2 below.

4.1.2 BindingType

PWG PJT Mapping: BindingType

See detailed mappings for various JDF binding classes and attributes in section 4.2 below.

4.1.3 CompressionSupplied

PWG PJT Mapping: CompressionSupplied (see note 3 and note 5 above)

JDF defines the FileSpec element in section 8.57 “FileSpec” of [CIP4JDF], which contains the *Compression* attribute, whose keyword values map directly to values of the PWG PJT CompressionSupplied element.

Note: The JDF *Compression* attribute only applies to the FileSpec element that it is attached to – but the PWG CompressionSupplied element applies to the entire Job.

4.1.4 DocumentCharsetSupplied

PWG PJT Mapping: DocumentCharsetSupplied (see note 3 above)

JDF defines the FileSpec element in section 8.57 “FileSpec” of [CIP4JDF], which contains the *Encoding* attribute. PWG and JDF both use the IANA Charset Registry [IANACHAR].

4.1.5 DocumentFormat

PWG PJT Mapping: DocumentFormat

JDF defines the FileSpec element in section 8.57 “FileSpec” of [CIP4JDF], which contains the *MimeType* attribute which maps directly to the PWG PJT DocumentFormat element. PWG and JDF both use the IANA Mime Media Types Registry [IANAMIME].

4.1.6 Document Format (w/ Charset)

PWG PJT Mapping: Document Format, DocumentCharsetSupplied (see note 3 above)

JDF defines the FileSpec element in section 8.57 “FileSpec” of [CIP4JDF], which contains the *MimeType* attribute which maps directly to the PWG PJT DocumentFormat element. PWG and JDF both use the IANA Mime Media Types Registry [IANAMIME].

Note: When a “charset” parameter is appended to the JDF MimeType attribute, the JDF MimeType can be mapped to the PWG PJT DocumentFormat and PWG PJT DocumentCharsetSupplied elements.

4.1.7 DocumentFormatDetailsSupplied

PWG PJT Mapping: DocumentFormatDetailsSupplied (see note 3 above)

JDF defines the FileSpec element in section 8.57 “FileSpec” of [CIP4JDF]. The PWG PJT DocumentFormatDetailsSupplied element has similar content. Therefore, each PWG PJT DcumentFormatDetailsSupplied member element SHOULD be mapped directly from the corresponding JDF FileSpec element – see details in Table 1 above.

4.1.8 DocumentNumbers

PWG PJT Mapping: DocumentNumbers

JDF defines the RunList resource in section 8.125 “RunList” of [CIP4JDF], which contains the *Docs* attribute.

Note: The JDF *Docs* attribute is a zero-based list of document indices in a multi-document file specified by the LayoutElement Resource – but the PWG PJT DocumentNumbers element is a single range of ‘1..MAX’. Therefore, only an approximate mapping is possible from the JDF *Docs* attribute to the PJT Document element. Also, every JDF *Docs* value SHOULD be incremented by one to map to the corresponding PJT DocumentNumbers value. All negative JDF Docs values SHOULD be calculated from the known actual range of Docs values for the given Job. For example, a JDF *Docs* value of ‘0 ~ -1’ specifies the first document through the last document in the RunList.

4.1.9 DocumentPages

PWG PJT Mapping: DocumentPages

JDF defines the RunList resource in section 8.125 “RunList” of [CIP4JDF], which contains the *Pages* attribute.

Note: The JDF *Pages* attribute is a zero-based list of indices in the documents in the LayoutElement resource – but the PWG PJT DocumentPages element has a range of ‘1..MAX’ and specifies the total number of pages in the Document. Therefore, only an approximate mapping is possible from the JDF *Pages* attribute to the PJT DocumentPages element. Also, every JDF *Pages* value SHOULD be incremented by one to map to the corresponding PJT DocumentPages value. All negative JDF Pages values SHOULD be calculated from the known actual range of DocumentPages values for the given Document.

4.1.10 DocumentPassword

PWG PJT Mapping: DocumentPassword

JDF defines the FileSpec element in section 8.57 “FileSpec” of [CIP4JDF], which contains the *Password* attribute.

Note: The JDF *Password* attribute contains either: (a) a cleartext password; or (b) the decryption key needed to read the document file contents – but the PWG PJT DocumentPassword *operation* parameter always contains a cleartext password supplied by the IPP Client over a secure Job submission channel.

4.1.11 Finishings and FinishingsCol

PWG PJT Mapping: Finishings, FinishingsCol

JDF does not define a single Finishings object – instead, section 6.4 “Postpress Processes” of [CIP4JDF] defines over 50 finishing processes including: ChannelBinding, CoilBinding, Cutting, Folding, HoleMaking, PlasticCombBinding, RingBinding, SaddleStitching, Stitching, StripBinding, Trimming, and WireCombBinding.

JDF section 7 “Intents” of [CIP4JDF] defines Intent Resource elements that closely correspond to the PWG Print Job Ticket including: section 7.5 “BindingIntent” (which includes stitching), section 7.9 “FoldingIntent”, section 7.10 “HoleMakingIntent”, and section 7.13 “LayoutIntent”.

JDF section 8 “Parameters” of [CIP4JDF] defines further Resource elements that closely correspond to the PWG Print Job Ticket including: section 8.17 “CoilBindingParams”, section 8.36 “CuttingParams”, section 8.58 “FoldingParams”, section 8.66 “HoleList”, section 8.67 “HoleMakingParams”, section 8.103 “PlasticCombBindingParams”, section 8.124“RingBindingParams”, section 8.126 “SaddleStitchingParams”, section 8.143

"StitchingParams", section 8.145 "StripBindingParams", section 8.157 "TrimmingParams", and section 8.162 "WireCombBindingParams".

4.1.12 ImpositionTemplate

PWG PJT Mapping: ImpositionTemplate

JDF defines the ExternalImpositionTemplate in section 8.55 of [CIP4JDF], which contains the FileSpec element, which contains the *UserFileName* attribute (user-friendly file name), which can be mapped to the PJT ImpositionTemplate element (keyword of 'None' or 'Signature' or site-specific name) defined in [PWG5100.3]. The PJT ImpositionTemplate element interacts in an implementation-specific manner with the PJT NumberUp element and PJT XImage and YImage layout elements.

4.1.13 InsertSheet

PWG PJT Mapping: InsertSheet, InsertAfterPage, InsertCount

JDF defines the RunList resource in section 8.125 "RunList" of [CIP4JDF], which contains the InsertSheet element defined in section 8.77 "InsertSheet" of [CIP4JDF]. The position of the InsertSheet can be inferred from the location of the InsertSheet element in the RunList for the mapping to the PWG PJT InsertPageAfter element. The chosen mapping for PWG PJT InsertCount is <none> because JDF normally has a separate InsertSheet for each inserted page.

4.1.14 JobFinishings and JobFinishingsCol

PWG PJT: JobFinishings and JobFinishingsCol

See section 4.1.11 Finishings and FinishingsCol above.

4.1.15 JobPriority

PWG PJT Mapping: JobPriority

JDF defines the NodeInfo element in section 8.91 "NodeInfo" of [CIP4JDF], which contains the *JobPriority* attribute.

Note: The JDF *JobPriority* attribute has a range of '0' (lowest) to '100' (highest) – but the PWG PJT JobPriority element has a range from '1' (lowest) to '100' (highest). Therefore, a JDF *JobPriority* value of '0' SHOULD be mapped to a PJT JobPriority value of '1' and all other JDF values SHOULD be mapped directly to the identical PJT values.

4.1.16 JobRecipientName

PWG PJT Mapping: JobRecipientName

JDF defines the CustomerInfo element in section 8.3 "CustomerInfo" of [CIP4JDF], which contains the Contact element defined in section 8.27 "Contact" of [CIP4JDF], which

contains the Person dlement defined in section 10.29 “Person” of [CIP4JDF], which contains the *FirstName, FamilyName, JobTitle, NamePrefix,* and *NameSuffix* attributes (all of which SHOULD be included in the value of PWG PJT JobRecipientName).

4.1.17 JobUuid

PWG PJT Mapping: JobUuid

JDF defines the JobPhase element in section 5.9.9.3 “JobPhase”, the QueueEntry element in section 5.14.2 “QueueEntry”, and various other elements of [CIP4JDF], all of which contain the *QueueEntryID* attribute, a string which could contain a UUID of a Job.

Note: PJT JobUuid contains a UUID in the form of a URN that conforms to [RFC4122]. JDF *QueueEntryID* is assigned by the PrintService at Job creation time. PJT JobUuid is assigned by the Print Service at Job creation time and therefore can only appear in a PWG JobReceipt and NOT in a PWG JobTicket (before Job submission). A JDF system that plans to forward a Job to a PWG Print Service SHOULD assign a *QueueEntryID* value in the form of a URN that conforms to [RFC4122].

4.1.18 Media and MediaCol

PWG PJT Mapping: Media, MediaCol

JDF defines the Media element in section 9.7 “Media” of [CIP4JDF]. Most of the PWG PJT MediaCol member elements SHOULD be mapped directly from the corresponding JDF Media attributes – see details in Table 1 above.

4.1.19 MediaPreprinted

PWG PJT Mapping: MediaPreprinted

JDF defines the Media element in section 9.7 “Media” of [CIP4JDF] which contains the *PrePrinted* attribute which is a boolean – but the PJT MediaPrePrinted element is a keyword (Blank, PrePrinted, LetterHead are currently defined). Therefore, only a partial mapping is possible from the JDF *PrePrinted* attribute to the PJT MediaPrePrinted element.

4.1.20 MediaRecycled

PWG PJT Mapping: MediaRecycled

JDF defines the Media element in section 9.7 “Media” of [CIP4JDF], which contains the *RecycledPercentage* attribute.

Note: The JDF *RecycledPercentage* attribute is an integer (0 to 100) and SHOULD be mapped to the PJT MediaRecycled element, by converting a JDF value of 51 percent to a PJT keyword of “Standard” for best fidelity.

4.1.21 MediaSize

PWG PJT Mapping: MediaSize

JDF defines the Media element in section 9.7 “Media” of [CIP4JDF], which contains the *Dimension* attribute.

Note: The JDF *Dimension* attribute is a pair of integers (X and Y dimensions in points) and has a direct mapping to the XDimension and YDimension members (in hundredths of millimeters) of the PJT MediaSize element [PWG5100.3]. When converting from a JDF *Dimension* attribute to a PJT MediaSize element, an approximate mapping with a tolerance of 1-2 percent SHOULD be used, in order to select the correct media actually intended by the Job Owner.

4.1.22 MediaSizeName

PWG PJT Mapping: MediaSizeName

JDF defines the Media element in section 9.7 “Media” of [CIP4JDF], which contains the *Dimension* attribute.

Note: The JDF *Dimension* attribute is a pair of integers (X and Y dimensions in points) and has no direct mapping to a PJT MediaSizeName element [PWG5101.1], since the class (“na”, “iso”, etc.) cannot be reliably inferred.

4.1.23 Number Up

PWG PJT Mapping: NumberUp

JDF defines the LayoutPreparationParams element in section 8.86 “LayoutPreparationParams” of [CIP4JDF], which contains the *NumberUp* attribute.

Note: The JDF *NumberUp* attribute is a pair of integers (the number of columns and number of rows) – but the PWG PJT NumberUp element is a single integer (1, 4, 6, 9, etc.) that does NOT constrain the column/row layout implementation choices. Therefore, only an approximate mapping is possible from the JDF *NumberUp* attribute to the PJT NumberUp element.

4.1.24 Overrides

PWG PJT Mapping: Overrides

JDF defines the RunList resource in section 8.125 “RunList” of [CIP4JDF] (a Process resource). See section 4.3 JDF RunList Resources and section 4.5 JDF Part Elements below for more details.

4.1.25 OverridingElements

PWG PJT Mapping: OverridingElements

JDF defines Partitioned resources in section 3.11.5 “Description of Partitioned Resources” of [CIP4JDF]. See section 4.4 JDF Partitioned Resources and section 4.5 JDF Part Elements below for more details.

4.1.26 PageRanges

PWG PJT Mapping: InsertAfterPage, PageRanges

JDF defines the RunList resource in section 8.125 “RunList” of [CIP4JDF], which contains the *Pages* attribute.

Note: The JDF *Pages* attribute is a list of zero-based integer ranges – but the PWG PJT PageRanges element is a single range of integer (1..MAX). Therefore, only an approximate mapping is possible from the JDF *Pages* attribute to the PJT PageRanges element. Also, every JDF *Pages* value SHOULD be incremented by one to map to the corresponding PJT PageRanges value. All negative JDF Pages values SHOULD be calculated from the known actual range of PJT PageRanges values for the given Document.

4.1.27 PresentationDirectionNumberUp

PWG PJT Mapping: PresentationDirectionNumberUp

JDF defines the LayoutPreparationParams element in section 8.86 “LayoutPreparationParams” of [CIP4JDF], which contains the *PresentationDirection* attribute.

Note: The JDF *PresentationDirection* attribute is a keyword (either ‘FoldCatalog’ or permutations of ‘XYZ’, each letter in uppercase or lowercase to specify the order in which finished pages are flowed along each axis with respect to the coordinate system of the front side of the Sheet – but the PJT PresentationDirectionNumberUp element is a keyword (e.g., TorightTobottom, TobottomToright, ToleftTotop). Therefore, only an approximate mapping is possible from the JDF *PresentationDirection* attribute to the PJT PresentationDirectionNumberUp element.

4.1.28 PrintColorMode

PWG PJT Mapping: PrintColorMode

JDF defines the ColorantControl element in section 8.21 “ColorantControl” of [CIP4JDF], which contains the *ProcessColorModel* attribute.

Note: The JDF *ProcessColorModel* attribute is a keyword (DeviceCMY, DeviceCMYK, DeviceGray, DeviceN, DeviceRGB, or None) – but the PJT PrintColorMode attribute is a keyword (Auto, BiLevel, Color, Highlight, Monochrome, ProcessBiLevel, ProcessMonochrome). Therefore, only an approximate mapping is possible from the JDF *ProcessColorModel* attribute to the PJT PrintColorMode element.

4.1.29 PrintRenderingIntent

PWG PJT Mapping: PrintRenderingIntent

JDF defines the ColorSpaceConversionParams element in section 8.25 of [CIP4JDF], which contains the ColorSpaceConversionOp element defined in section 10.7 of [CIP4JDF], which contains the *RenderingIntent* attribute.

Note: The JDF *RenderingIntent* attribute is an enumeration (AbsoluteColorimetric, Perceptual, RelativeColorimetric, Saturation, or ColorSpaceDependent (i.e., implementation-defined)) – but the PJT PrintRenderingIntent element is a keyword (Absolute, Auto, Perceptual, Relative, RelativeBPC, or Saturation).

Notes:

1) JDF RelativeColorimetric and equivalent PWG PJT Relative specify “white point compensation” – defined in section 5.2.4 of [PWG5100.13] as “Clip out-of-gamut colors to preserve in-gamut accuracy, adjusting the white point as necessary.”

2) PWG PJT RelativeBPC specifies “black point compensation” – defined in section 5.2.4 of [PWG5100.13] as “Clip out-of-gamut colors to preserve in-gamut accuracy, adjusting both the white and black points as necessary.”

Table 2 – Mapping of CIP4 JDF to PWG PJT for Rendering Intent

JDF RenderingIntent	PJT PrintRenderingIntent
AbsoluteColorimetric	Absolute
ColorSpaceDependent	Auto
Perceptual	Perceptual
RelativeColorimetric	Relative
<none>	RelativeBPC (note 2)
Saturation	Saturation

4.1.30 ProofPrint

PWG PJT Mapping: ProofPrint

There is no JDF support for specifying specific media for ProofPrint. Use standard Media or MediaCol from PrintJob for ProofPrint.

4.1.31 TemplateInfo

PWG PJT Mapping: TemplateInfo

JDF defines the JDF Node in section 3.2 of [CIP4JDF], which contains the *TemplateVersion* attribute.

4.1.32 TemplateName

PWG PJT Mapping: TemplateName

JDF defines the JDF Node in section 3.2 of [CIP4JDF], which contains the *TemplateID* attribute.

4.1.33 Trimming

PWG PJT Mapping: Trimming

JDF defines the CuttingParams element in section 8.36 “CuttingParams” of [CIP4JDF], which contains the Cut element defined in section 10.12 “Cut” of [CIP4JDF], which contains details of the requested cutting.

JDF defines the PerforatingParams element in section 8.101 “PerforatingParams” of [CIP4JDF], which contains the Perforate element defined in section 10.28 “Perforate” of [CIP4JDF], which contains the details of the requested perforating.

JDF defines the CreasingParams element in section 8.32 “CreasingParams” of [CIP4JDF], which contains the Crease element defined in section 10.11 “Crease” of [CIP4JDF], which contains the details of the requested creasing (i.e., scoring).

JDF defines the Trimming element in section 6.4.49 “Trimming” of [CIP4JDF], which contains the TrimmingParams element defined in section 8.1.57 “TrimmingParams” of [CIP4JDF], which contains the *TrimmingType* attribute (deprecated in JDF 1.2) with values of ‘Detailed’ (use *Height*, *Width*, and *TrimmingOffset* values) and ‘SystemSpecified’ (specified by System, but not exposed).

Note: Since the deprecated JDF *TrimmingType* attribute is not useful for determining the equivalent value of PJT TrimmingType, the JDF Trimming element is not used in this mapping specification.

4.1.34 TrimmingType

PWG PJT Mapping: TrimmingType

The value of the PJT TrimmingType element [PWG5100.1] can be inferred from the use of the JDF Cutting, Perforating, or Creasing element and the details in the respective Cut, Perforate, or Crease elements. The PJT TrimmingType element contains a keyword value (Full, Partial, Perforate, Score, or Tab). The JDF Cut element maps to PJT TrimmingType value of Full. The JDF Perforate element maps to PJT TrimmingType value of Perforate. The JDF Crease element maps to PJT TrimmingType value of Score. There is no reasonable mapping to PJT TrimmingType values of Partial or Tab.

4.1.35 XImage Layout

PWG PJT Mapping: XImagePosition, XImageShift, XSide1ImageShift, XSide2ImageShift

JDF defines the LayoutPreparationParams resource (from JDF/1.1) in section 8.86 of [CIP4JDF], which contains the ImageShift element, which contains the *PositionX*, *ShiftFront*, and *ShiftBack* attributes with simple mappings to the equivalent PJT XImage layout elements – in this version of this JDFMAP, this is the recommended mapping for XImage layout elements. If only JDF ShiftFront is used without a ShiftBack, then JDF ShiftBack is calculated from ShiftFront so that the content remains aligned. To explicitly do JDF ShiftBack of zero, you have to specify JDF ShiftBack.

Note: In the future, CIP4 may deprecate LayoutPreparationParams, in favor of the newer and more general StrippingParams resource (from JDF/1.2) defined in section 8.146 of [CIP4JDF], which contains the Position and BinderySignature elements.

Note: PWG image shift units are Integer 1/2540th inch (1/100 of a millimetre). JDF image shift units are DoubleFloat 1/72 inch (points). Also JDF ShiftFront and ShiftBack are pair of both X and Y values.

The JDF *PositionX* attribute is a keyword (None, Center, Left, Right, or Spline), while the equivalent PJT XImagePosition element is a keyword (None, Center, Left, or Right), so there is no mapping for the JDF ‘Spline’ value.

The JDF LayoutPreparationParams resource (from JDF/1.1) in section 8.86 of [CIP4JDF] also contains the ExternalImpositionTemplate element, which contains the FileSpec element, which contains the *UserFileName* attribute (user-friendly file name), which can be mapped to the PJT ImpositionTemplate element (keyword of ‘None’ or ‘Signature’ or site-specific name) defined in [PWG5100.3]. The PJT ImpositionTemplate element interacts in an implementation-specific manner with the PJT NumberUp element and PJT XImage and YImage layout elements.

The JDF Position element contains the *MarginBottom*, *MarginLeft*, *MarginRight*, and *MarginTop* attributes (margins in points outside of the BinderySignature) and the *AbsoluteBox* attribute (absolute position in points of the BinderySignature display area).

JDF defines the BinderySignature element (a conceptual folding dummy) in section 8.7 of [CIP4JDF], which contains the *TrimBottom*, *TrimLeft*, *TrimRight*, and *TrimTop* attributes (cutoff margin widths around the sides of the BinderySignature – the remainder contains the Strips).

The JDF BinderySignature element also contains the *NumberUp* attribute (pair of integers, columns and rows) that interacts with the PJT NumberUp element (single integer product of columns and rows). See the discussion of JDF and PJT page layout in the section 4.1.23 NumberUp above.

4.1.36 YImage Layout

PWG PJT Mapping: YImagePosition, YImageShift, YSide1ImageShift, YSide2ImageShift

JDF defines the LayoutPreparationParams resource (from JDF/1.1) in section 8.86 of [CIP4JDF], which contains the ImageShift element, which contains the *PositionY*, *ShiftFront*, and *ShiftBack* attributes with simple mappings to the equivalent PJT YImage layout elements – in this version of this JDFMAP, this is the recommended mapping for YImage layout elements. If only JDF ShiftFront is used without a ShiftBack, then JDF ShiftBack is calculated from ShiftFront so that the content remains aligned. To explicitly do JDF ShiftBack of zero, you have to specify JDF ShiftBack.

Note: In the future, CIP4 may deprecate LayoutPreparationParams, in favor of the newer and more general StrippingParams resource (from JDF/1.2) defined in section 8.146 of [CIP4JDF], which contains the Position and BinderySignature elements. See the discussion of the StrippingParams resource and Position and BinderySignature elements in section 4.1.35 XImage Layout above.

Note: PWG image shift units are Integer 1/2540th inch (1/100 of a millimetre). JDF image shift units are DoubleFloat 1/72 inch (points). Also JDF ShiftFront and ShiftBack are pair of both X and Y values.

The JDF *PositionY* attribute is a keyword (None, Center, Left, Right, or Spline), while the equivalent PJT YImagePosition element is a keyword (None, Center, Left, or Right), so there is no mapping for the JDF ‘Spline’ value.

4.2 Mappings of JDF to PJT Binding Types

JDF defines various binding classes in section 6.4 Postpress Binding and of [CIP4JDF] and also defines the binding resource in section 7.5 “BindingIntent” of [CIP4JDF].

Notes for Table 3 and Table 4:

1) JDF Postpress WireCombBinding and JDF BindingIntent/@*BindingType* value of WireComb can be mapped to the *new* PWG PJT value of WireComb, while JDF Postpress PlasticCombBinding and and JDF BindingIntent/@*BindingType* value of can be mapped to PWG PJT value of Comb, because the PWG PJT value Comb is defined as specifically plastic comb binding.

2) JDF BindingIntent/@*BindingType* value of Adhesive is deprecated in JDF 1.1 – use JDF SoftCover or HardCover instead – JDF BindingIntent/@*BindingType* value of SoftCover includes Perfect binding – JDF PWG PJT mapping is still Adhesive???

3) JDF BindingIntent/@*BindingType* values of CornerStitch, SaddleStitch, Sewn (deprecated in JDF/1.4), SideSewn (deprecated in JDF/1.4), and SideStitch SHOULD be mapped to PWG PJT Stitch.

4) JDF BindingIntent/@*BindingType* value of ThreadSealing (deprecated in JDF/1.4) of sewing signatures at the spine and also well as Sewn (deprecated in JDF/1.4), SideSewn (deprecated in JDF/1.4), and SideStitch SHOULD be mapped to PWG PJT Stitching.

5) JDF Postpress EndSheetGluing applies to hard cover binding and differs from PWG PJT value Adhesive (which only applies the glue but does not apply the hard cover).

6) JDF BindingIntent/@*BindingType* value of None can be mapped to PWG PJT by explicitly omitting the PJT BindingType element

7) JDF PostPress RingBinding and JDF BindingIntent/@*BindingType* value of Ring both define the punching of holes for a 2/3-ring binder and the application of the actual 2/3-ring binder – use PJT Punching to punch the appropriate holes, but PJT does not support applying the actual 2/3-ring binder.

Table 3 – Mapping of CIP4 JDF Postpress Binding Class to PWG PJT BindingType

CIP4 JDF Postpress Binding Class	PWG PJT BindingType
AdhesiveBinding	Adhesive (note 5)
ChannelBinding	<none>
CoilBinding	Spiral
EndSheetGluing	<none> (note 5)
LongitudinalRibbonOperations	<none>
PlasticCombBinding	Comb (note 1)
RingBinding	<none> (note 7)
SpineTaping	Tape
StripBinding	Velo
WireCombBinding	WireComb (note 1)

Table 4 – Mapping of CIP4 JDF BindingIntent/*@BindingType* to PWG PJT BindingType

CIP4 JDF BindingIntent/*@BindingType*	PWG PJT BindingType
AdhesiveBinding	Adhesive (note 2)
ChannelBinding	<none>
CoilBinding	Spiral
CornerStitch	<none> (note 3)
EdgeGluing	Padding
HardCover	<none> (note 2)
None	<no BindingType element> (note 6)
PlasticComb	Comb
Ring	<none> (note 7)
SaddleStitch	<none> (note 3)
Sewn	<none> (note 3)
SideSewn	<none> (note 3)
SideStitch	<none> (note 3)
SoftCover	Perfect (note 2)
StripBind	Velo
Tape	Tape
ThreadSealing	<none> (note 4)
WireComb	WireComb (note 1)

4.3 JDF RunList Resources

JDF defines the RunList resource in section 8.125 “RunList” of [CIP4JDF] (a Process resource).

RunList resources describe an ordered set of LayoutElement or ByteMap elements. Ordering and structure are defined using the generic Partitioning mechanisms as described in JDF section 3.11.5 “Description of Partitioned Resources” [CIP4JDF].

RunList resources are used whenever an ordered set of page descriptions elements are specified. Depending on the Process usage of a RunList, only certain types of LayoutElement could be valid.

In essence, a RunList is a virtual document or set of documents. It allows a document to either be physically spread over multiple files, or multiple documents to be contained within a single file (e.g., PPML, PDF/VT). It retains the same properties as the original documents (e.g., the pages of a document that is described by a RunList are ordered).

4.4 JDF Partitioned Resources

JDF defines Partitioned resources in section 3.11.5 “Description of Partitioned Resources” of [CIP4JDF]. JDF defines the Part element in section 3.11.6.2 “Part” of [CIP4JDF].

Printing workflows contain a number of Processes that are repeated over a potentially large number of individual files, Sheets, surfaces or separations. In order to define a Partitioned Resource in a concise manner without having to create a large number of individual Nodes and Resources, a set of Resources might be Partitioned by factoring them by one or more attributes. The common attributes and defaults are placed in the parent element while Partition-specific attributes and overrides are placed in the child elements. This avoids redundancy. Also, by providing a single parent ID for each Resource, it allows easy access to the entire Resource or iteration over each Part.

To reference part of a Resource, a ResourceLink references the parent Resource and supplies a Part element that contains an actual value for a Partition. The result is all the child Elements with matching Partition values, including common values and defaults from the parent Resource. If *@PartUsage = "Implicit"*, the parent Attributes are returned if there is no matching Partition.

4.5 JDF Part Elements

JDF defines the Part element in section 3.11.6.2 “Part” of [CIP4JDF], which contains the *DocRunIndex, DocIndex,* and *DocCopies* attributes.

The JDF *DocRunIndex* attribute (pair of signed integers that represent a range of pages) maps to the PWG PJT Overrides/Override/Pages/PageRange in the sub-elements LowerBound and UpperBound.

The JDF *DocIndex* attribute (pair of signed integers that represent a range of documents) maps to the PWG PJT Overrides/Override/DocumentNumbers/NumberRange in the sub-elements LowerBound and UpperBound.

The JDF *DocCopies* attribute (pair of signed integers that represent a range of document copies) maps to the PWG PJT Overrides/Override/DocumentCopies/PCopies in the sub-elements LowerBound and UpperBound.

4.6 Mapping JDF Job State Model to PWG Job State Model

JDF defines the NodeInfo element in section 8.91 “NodeInfo” of [CIP4JDF], which contains the *NodeStatus* attribute. PWG Semantic Model [REF] defines the JobState element.

This mapping is actually for the JMF Job states (in progress jobs) and the corresponding mapping of PWG Job states is actually for the PWG JobReceipt (results, not Job submission states). The PWG Client does not directly control Job state via the submitted PWG JobTicket – instead the Spooler/Printer does so.

Notes:

1) JDF NodeStatus of Aborted may map to either PJT Aborted or Canceled (the distinction is between system Abort and operator Cancel is not preserved in JDF).

2) JDF NodeStatus of Cleanup is ephemeral and SHOULD be mapped to PJT Processing (during cleanup) or Completed (at completion of cleanup).

3) JDF NodeStatus of Setup (holding for Operator setup/load/preparation) SHOULD be mapped to PJT PendingHeld.

4) JDF NodeStatus of Stopped and Suspended SHOULD be mapped to PJT ProcessingStopped.

5) JDF NodeStatus of Waiting SHOULD be mapped to PJT Pending.

6) PJT JobStateReason of ProcessingToStopPoint can be ephemeral during various JobState transitions.

Table 5 – Mapping of CIP4 JDF Node Status to PWG SM Job States

JDF NodeStatus	PWG JobState	PWG JobStateReasons
Aborted	Aborted or Canceled (note 1) (note 6)	AbortedBySystem, UnsupportedCompression, CompressionError, UnsupportedDocumentFormat, JobCanceledByUser, JobCanceledByOperator, (ephemeral) ProcessingToStopPoint
Cleanup	Processing (note 6) Completed (note 2)	(ephemeral) ProcessingToStopPoint <none>
Completed	Completed	JobCompletedSuccessfully, JobCompletedWithWarnings, JobCompletedWithErrors, DocumentAccessError, QueuedInDevice
InProgress	Processing	JobQueued, JobInterpreting, JobTransforming, JobQueuedForMarker, JobPrinting
Ready	Pending	
Setup	PendingHeld (note 3)	JobHoldUntilSpecified, ResourcesAreNotReady, JobQueuedForMarker, ServiceOffline
Stopped	ProcessingStopped	PrinterStopped, PrinterStoppedPartly
Suspended	ProcessingStopped (note 4) (note 6)	(ephemeral) ProcessingToStopPoint
Waiting	Pending (note 5)	JobDataInsufficient, JobIncoming

5. Conformance Recommendations

5.1 Print Server Conformance Recommendations

Conforming Print Servers that implement this Best Practices document SHOULD:

1) Conform to the all of the element mappings defined in section 4.1 Mapping JDF Attributes to PWG Print Job Ticket Elements;

2) Conform to all of the state mappings defined in section 4.2 Mapping JDF State Model to PWG Printer State Model;

3) Conform to section 6 Internationalization Considerations;

4) Conform to section 7 Security Considerations.

5.2 Print Device Conformance Recommendations

Conforming Print Devices that implement this Best Practices document SHOULD:

1) Conform to the all of the element mappings defined in section 4.1 Mapping JDF Attributes to PWG Print Job Ticket Elements;

2) Conform to all of the state mappings defined in section 4.2 Mapping JDF State Model to PWG Printer State Model;

3) Conform to section 6 Internationalization Considerations;

4) Conform to section 7 Security Considerations.

6. Internationalization Considerations

For interoperability and basic support for multiple languages, conforming implementations of this Best Practices document SHOULD support the Universal Character Set (UCS) Transformation Format -- 8 bit (UTF-8) [RFC3629] encoding of Unicode [UNICODE] [ISO10646] and the Unicode Format for Network Interchange [RFC5198].

When processing human names that are being concatenated from multiple CIP4 JDF attributes (e.g., JDF “CustomerInfo/Contact/Person” object attributes such as *FirstName, FamilyName, JobTitle, NamePrefix,* and *NameSuffix*) into a PWG PJT single human-readable string attribute (e.g., JobRecipientName), the correct layout order for these components of human names is locale-dependent.

For more information on locale-dependent processing, see ICU User Guide/Software Internationalization [ICU-SWI18N].

Implementations of this Best Practices document SHOULD conform to the following standards on processing of human-readable Unicode text strings:

- Unicode Bidirectional Algorithm [UAX9] – left-to-right, right-to-left, and vertical
- Unicode Line Breaking Algorithm [UAX14] – character classes and wrapping
- Unicode Normalization Forms [UAX15] – especially NFC for [RFC 5198]
- Unicode Text Segmentation [UAX29] – grapheme clusters, words, sentences
- Unicode Identifier and Pattern Syntax [UAX31] – identifier use and normalization
- Unicode Collation Algorithm [UTS10] – sorting
- Unicode Locale Data Markup Language [UTS35] – locale databases

Implementations of this Best Practices document SHOULD also follow the recommendations of the documents below:

- Unicode Character Encoding Model [UTR17] – multi-layer character model
- Unicode in XML and other Markup Languages [UTR20] – XML usage
- Unicode Character Property Model [UTR23] – character properties
- Unicode Conformance Model [UTR33] – Unicode conformance basis

7. Security Considerations

Implementations of this Best Practices document SHOULD conform to security requirements specified in section 8 Security Considerations of IPP/1.1 Model and Semantics [RFC8011] and in section 10 Security Considerations of IPP/2.0 Second Edition [PWG5100.12].

Implementations of this Best Practices document SHOULD conform to the following standard on processing of human-readable Unicode text strings:

- Unicode Security Mechanisms [UTS39] – detecting and avoiding security attacks

Implementations of this Best Practices document SHOULD also follow the recommendations of the document below:

- Unicode Security FAQ [UNISECFAQ] – common Unicode security issues

8. IANA and PWG Considerations

There are no IANA or PWG registration considerations for this document.

9. References

[CIP4JDF] CIP4, "JDF Specification Release 1.5, December 2013, http://www.cip4.org

[CIP4IDP] CIP4, "Integrated Digital Printing (IDP) ICS Version 1.3 Errata Revision A", February 2009, http://www.cip4.org

[ISO10646] ISO, "Information technology -- Universal Coded Character Set (UCS)", ISO/IEC 10646:2014, 2014, http://standards.iso.org/ittf/PubliclyAvailableStandards/index.html

[PWG5100.1] M. Sweet, "IPP Finishings 2.1 (FIN)", PWG 5100.1, December 2014, http://ftp.pwg.org/pub/pwg/candidates/cs-ippfinishings21-20170217-5100.1.pdf

[PWG5100.3] T. Hastings, K. Ocke, "Internet Printing Protocol (IPP): Production Printing Attributes – Set1", PWG 5100.3, February 2001, http://ftp.pwg.org/pub/pwg/candidates/cs-ippprodprint10-20010212-5100.3.pdf

[PWG5100.12] R. Bergman, H. Lewis, I. McDonald, M. Sweet, " IPP Version 2.0, 2.1, and 2.2 ", PWG Standard 5100.12, October, 2015, http://ftp.pwg.org/pub/pwg/standards/std-ipp20-20151030-5100.12.pdf

[PWG5100.13] I. McDonald, M. Sweet, P. Zehler, "IPP Job and Printer Extensions – Set 3 (JPS3)", July 2012, http://ftp.pwg.org/pub/pwg/candidates/cs-ippjobprinterext3v10-20120727-5100.13.pdf

[PWG5100.14] J. Hutchings, I. McDonald, A. Mitchell, M. Sweet, "IPP Everywhere", PWG 5100.14, January 2013, http://ftp.pwg.org/pub/pwg/candidates/cs-ippeve10-20130128-5100.14.pdf

[PWG5101.1] R. Bergman, T. Hastings, M. Sweet, "PWG Media Standardized Names 2.0 (MSN2)", PWG 5101.1, March 2013, http://ftp.pwg.org/pub/pwg/candidates/cs-pwgmsn20-20130328-5101.1.pdf

[PWG5108.07] P. Zehler, "PWG Print Job Ticket and Associated Capabilities v1.0 (PJT)", PWG 5108.7, http://ftp.pwg.org/pub/pwg/candidates/cs-sm20-pjt10-20120801-5108.07.pdf

[RFC2119] S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119/BCP 14, March 1997, http://www.ietf.org/rfc/rfc2119.txt

[RFC3805] R. Bergman, H. Lewis, I. McDonald, "Printer MIB v2", RFC 3805, June 2004, http://www.ietf.org/rfc/rfc3805.txt

[RFC3806] R. Bergman, H. Lewis, I. McDonald, "Printer Finishing MIB", RFC 3806, June 2004, http://www.ietf.org/rfc/rfc3806.txt

[RFC4122] P. Leach, M. Mealling, R. Salz, “A Universally Unique IDentifier (UUID) URN Namespace”, RFC 4122, July 2005, http://www.ietf.org/rfc/rfc4122.txt

[RFC5198] J. Klensin, M. Padlipsky, "Unicode Format for Network Interchange", RFC 5198, March 2008, http://www.ietf.org/rfc/rfc5198.txt

[RFC5646] A. Phillips, M. Davis, “Tags for Identifying Languages”, RFC 5646 / BCP 47, September 2009, http://www.ietf.org/rfc/rfc5646.txt

[RFC8011] M. Sweet, I. McDonald, "Internet Printing Protocol/1.1: Model and Semantics", RFC 8011, January 2017, http://www.ietf.org/rfc/rfc8011.txt

[RFC8126] M. Cotton, B. Leiba, T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", RFC 8126 / BCP 26, June 2017, http://www.ietf.org/rfc/rfc8126.txt

[STD63] F. Yergeau, "UTF-8, a transformation format of ISO 10646", RFC 3629/STD 63, November 2003, http://www.ietf.org/rfc/rfc3629.txt

[UAX9] Unicode Consortium, “Unicode Bidirectional Algorithm”, UAX#9, June 2014, http://www.unicode.org/reports/tr9/tr9-31.html

[UAX14] Unicode Consortium, “Unicode Line Breaking Algorithm”, UAX#14, June 2014, http://www.unicode.org/reports/tr14/tr14-33.html

[UAX15] Unicode Consortium, “Normalization Forms”, UAX#15, June 2014, http://www.unicode.org/reports/tr15/tr15-41.html

[UAX29] Unicode Consortium, “Unicode Text Segmentation”, UAX#29, June 2014, http://www.unicode.org/reports/tr29/tr29-25.html

[UAX31] Unicode Consortium, “Unicode Identifier and Pattern Syntax”, UAX#31, June 2014, http://www.unicode.org/reports/tr31/tr31-21.html

[UNICODE] Unicode Consortium, "Unicode Standard", Version 10.0.0, June 2017, http://www.unicode.org/versions/Unicode10.0.0/

[UNISECFAQ] Unicode Consortium “Unicode Security FAQ”, November 2013, http://www.unicode.org/faq/security.html

[UTR17] Unicode Consortium “Unicode Character Encoding Model”, UTR#17, November 2008, http://www.unicode.org/reports/tr17/tr17-7.html

[UTR20] Unicode Consortium “Unicode in XML and other Markup Languages”, UTR#20, January 2013, http://www.unicode.org/reports/tr20/tr20-9.html

[UTR23] Unicode Consortium “Unicode Character Property Model”, UTR#23, November 2008, http://www.unicode.org/reports/tr23/tr23-9.html

[UTR33] Unicode Consortium “Unicode Conformance Model”, UTR#33, November 2008, http://www.unicode.org/reports/tr33/tr33-5.html[UTS10] Unicode Consortium, “Unicode Collation Algorithm”, UTS#10, June 2014, http://www.unicode.org/reports/tr10/tr10-30.html,

[UTS35] Unicode Consortium, “Unicode Locale Data Markup Language”, UTS#35, September 2014, http://www.unicode.org/reports/tr35/tr35-37/tr35.html

[UTS39] Unicode Consortium, “Unicode Security Mechanisms”, UTS#39, September 2014, http://www.unicode.org/reports/tr39/tr39-9.html

10. Authors' Addresses

Ira McDonald
High North
PO Box 221
Grand Marais, MI 49839
906-494-2434
blueroofmusic@gmail.com

Rick Yardumian
Canon Information & Imaging Solutions, Inc
15975 Alton Parkway
Irvine, CA 92618
ryardumian@ciis.canon.com

The authors would especially like to thank Rainer Prosi (Heidelberg, CIP4 CTO) for his extensive and authoritative contributions to this document.

The authors would also like to thank the following individuals for their contributions to this document:

Daniel Manchala	Xerox
Michael Sweet	Apple
Paul Tykodi	TCS
William Wagner	TIC
Pete Zehler	Xerox

