PrintTalk Specification

2.1-DRAFT-IP-2

8 January 2020

Legal Notice

Use of this document is subject to the following conditions which are deemed accepted by any person or entity making use hereof.

Table of Contents

1 Introduction

PrintTalk is an XML standard that is maintained by **CIP4** and provides a single format for *Print Providers* to collabora-tively communicate *Business Transactions* and specifications of *Print Products* both with their *Print Buyers* and among themselves. Print industry professionals from a wide range of organizations contributed to the design of **PrintTalk**. Such organizations include eCommerce companies, print business management systems, print providers, and print equip-ment manufacturers.

The **PrintTalk** standard embraces ▸ [XJDF 2.1] and ▸ [cXML 1.2]. These bracketed references are defined in ▸ Appendix A References. Whereas **XJDF** describes the printed product to be produced, **PrintTalk** specifies the business data and con-text between the *Print Provider* and *Print Buyer*.

PrintTalk 2.1 is a minor update of **PrintTalk** and is backwards compatible with **PrintTalk** 2.0. Thus any valid **PrintTalk** 2.0 document is also a valid **PrintTalk** 2.1 document. **PrintTalk** 2.1 adds some additional functionality and clarifications to **PrintTalk** 2.0. The major difference is that the referenced **XJDF** standard has been updated from ▸ [XJDF 2.0] to ▸ [XJDF 2.1]. Since the **XJDF** revision follows the same backwards compatibility policy, i.e. any valid ▸ [XJDF 2.0] is also a valid ▸ [XJDF 2.1], **XJDF** 2.0 elements MAY be provided in **PrintTalk** 2.1.

Although **PrintTalk** includes many concepts from cXML, they have been incorporated into the **PrintTalk** standard which is a complete, independent XML standard, with its own namespace and schema that define all the elements and attri-butes declared in this specification. **XJDF**, although also maintained by **CIP4**, is an independent standard complete with its own namespace and schema and as such **PrintTalk** uses **XJDF** by importing definitions from the **XJDF** schema.

This specification details the requirements of the XML data in ▸ Chapter 2 Structure, ▸ Chapter 3 Business Objects and ▸ Chapter 4 Subelements. The requirements for the transport protocol and security are defined in ▸ Chapter 5 Building a System. The use of **PrintTalk** to construct business workflows is described in ▸ Chapter 6 Business Object Workflows.

1.1 Intended Audience

While print industry professionals may find parts of this specification to be useful, our primary goal is to describe the **PrintTalk** protocol to programmers who wish to implement the **PrintTalk** interface specification within their software products. A working knowledge of XML and familiarity with **XJDF** is required to fully understand this document.

1.2 Use of XML

PrintTalk is encoded as XML and SHALL be a valid XML document according to ▸ [XML].
Note: Most data in **PrintTalk** is encoded in XML attributes; XML elements provide the hierarchical structure of the data.
Note: The data model does not require use of XML. Conceptually, any hierarchical data syntax could be used. XML was chosen to align with the **XJDF** specification and because it is in widespread use. In addition, leaving the choice of an un-derlying grammar open would lead to non interoperable implementations.

1.2.1 Use of XML namespaces

A **PrintTalk** implementation SHALL support standard namespace handling as defined in ▸ [XMLNS].

The **PrintTalk** namespace shall be declared in the *PrintTalk* element using the standard @*xmlns* attribute. For version 2.1 of **PrintTalk** the value SHALL be "http://www.printtalk.org/schema_20".
Note: It is anticipated that the value of the **PrintTalk** namespace will remain constant for all future versions of **PrintTalk** 2. This reflects the backwards compatibility of minor revisions for a major version of **PrintTalk**.

The namespace prefix for items that are defined in the **PrintTalk** namespace SHOULD be one of 'ptk' or no prefix, i.e. the default namespace.

If a **PrintTalk** instance contains items from the **XJDF** namespace, then the **XJDF** namespace SHOULD be declared in the *PrintTalk* element using the standard @*xmlns* attribute and SHOULD conform to the recommended use of prefix identi-fiers as specified in ▸ [XJDF 2.1].
Note: It is not recommended to use the empty, i.e. default, namespace for declaring **XJDF** items.

If a **PrintTalk** instance contains items from namespaces other than **PrintTalk** or **XJDF**, then the namespace for those items SHOULD be declared in the *PrintTalk* element using the standard @*xmlns* attribute and SHALL NOT use a prefix that is reserved for either **PrintTalk** or **XJDF**.
Note: Such namespaces are referred to as 'foreign namespaces' throughout this document.

1.2.2 Use of XML Schema

The XML schema for **PrintTalk** is designed to ensure that **PrintTalk** documents are syntactically valid, thus **PrintTalk** documents that are successfully validated against the **PrintTalk** schema SHALL be considered conformant to the syntax requirements described in this specification.

All elements that are defined in this specification are defined in the XML schema for this namespace, including those derived directly from ▸ [cXML 1.2], but excluding those defined in ▸ [XJDF 2.1].

1.3 Data Structures

Unless stated otherwise, this specification uses XML data types as used by **XJDF** and defined by ▸[XMLSchema]. For more details on data types, units of measurement etc, see the 'Data Structures' section in ▸[XJDF 2.1].

1.3.1 Currency

In addition to the data types defined by **XJDF**, **PrintTalk** requires the means to convey the currency of a transaction and therefore defines its own data type for this purpose. In the definition tables in this specification, attributes that use this have a value of 'Currency' in the DataType column.

Attributes of type 'Currency' SHALL contain a valid uppercase alphabetic currency code as defined in ▸[ISO4217].
Note: ▸[ISO4217] permits the use of numeric currency codes, however these are intended for use in systems that do not use Latin scripts.

1.4 Conventions Used in this Specification

This specification uses the same conventions and formatting as ▸[XJDF 2.1]. See the appropriate section in the introduction in ▸[XJDF 2.1] for details.

1.5 Glossary

This section defines terminology used throughout this document. References to other documents are indicated with square brackets, e.g. ▸[XJDF 2.1]. For most terms, see the terminology section in ▸[XJDF 2.1].

Table 1.1: Glossary

TERM	DEFINITION
Business Object	The portion of a **PrintTalk** document that describes a *Business Transaction*. For example an *RFQ Business Object* describes a request for quote *Business Transaction* and a *PurchaseOrder Business Object* describes a purchase order *Business Transaction*.
Business Transaction	When a *Print Buyer* sends a **PrintTalk** document to a *Print Provider*, or a *Print Provider* sends a **PrintTalk** document to a *Print Buyer*, the *Business Object* within the **PrintTalk** document describes a business transaction between the *Print Buyer* and *Print Provider*.
Catalog Item	A preprinted or non-printed item that can be ordered.
Change Order	A *Business Transaction* that requests changes to a *Confirmed Business Object*.
Confirmed Business Object	A *Business Object* that the other party has accepted by sending an appropriate *Business Object* as a response, e.g. a *PurchaseOrder* that references a *Quotation*. A *Confirmed Business Object* is no longer a *Pending Business Object*.
Item	Something that is explicitly priced. This can be a physical product, a service or anything that has a price.
Master Contract	A pre-negotiated existing contract whose terms and details apply to the *Business Objects* that reference it.
Negotiation Phase	The phase during which a *Print Buyer* and *Print Provider* negotiate a contract for a *Print Product* by exchanging *Business Objects*. The *Production Phase* follows the *Negotiation Phase*.
Pending Business Object	A *Business Object* that other *Business Objects* can reference via its *Request*/@*BusinessRefID*. A *Business Object* is pending from the time a *Print Buyer* or *Print Provider* creates it until: • It is confirmed by a *Confirmation* or other appropriate *Business Object*, see *Confirmed Business Object*. • It is refused by an explicit *Refusal*. • It expires. • Its creator sends a *Cancellation* whose *Request*/@*BusinessRefID* references it.
Print Buyer	The customer that is buying a *Print Product*.
Print Product	The printed matter produced according to a contract negotiated between the *Print Buyer* and *Print Provider* using the **PrintTalk** negotiation process.

Table 1.1: Glossary

TERM	DEFINITION
Print Provider	The producer of the *Print Product.*
PrintTalk Document	An XML document whose root element is *PrintTalk* and which follows the rules de–scribed in this specification.
PrintTalk Project	The *Business Transactions* that apply to one set of *Items* that are negotiated between the *Print Buyer* and the *Print Provider.*
Production Phase	The phase during which a *Print Provider* produces and ships a *Print Product.*
Requote	An additional *Quotation* for a variation in a job, for which the *Print Provider* has already issued a *Quotation.*

2 Structure

A **PrintTalk** document applies concepts that are defined in ▸[cXML 1.2] to the graphic arts. All **PrintTalk** elements SHALL be declared in the **PrintTalk** namespace. See ▸Table 2.1 PrintTalk Element for details. Like cXML, the top two elements of a **PrintTalk** document SHALL be a *Header* element and a *Request* element. The header concept comes directly from cXML, and identifies the parties involved in this correspondence. In some cases the definitions of *Header* and its subelements have been clarified for usage in business transactions that are unique to the graphic arts such as web to print. The *Request* element SHALL contain exactly one **PrintTalk** *Business Object*.

Within this document, whenever elements are specified with an "xjdf" namespace prefix, this prefix specifies the presence of that element from the **XJDF** namespace, i.e. http://www.CIP4.org/JDFSchema_2_0.

2.1 PrintTalk

The root element in a **PrintTalk** document SHALL be an element whose name is *PrintTalk*. An example is:

```
<PrintTalk xmlns="http://www.printtalk.org/schema_20"
  payloadID="P_000427" timestamp="2019-09-30T13:42:14+02:00">
</PrintTalk>
```

Table 2.1: PrintTalk Element

NAME	DATA TYPE	DESCRIPTION
ICSVersions ?	NMTOKENS	@*ICSVersions* SHALL list all CIP4 Interoperability Conformance Specification (ICS) Versions for **PrintTalk** that this **PrintTalk** complies with. The value of @*ICSVersions* SHALL conform to the value format described in ▸Section 2.1.1 ICS Versions Value. Any **XJDF** ICS versions that the **XJDF** complies with SHALL be specified in the respective *xjdf:XJDF*/@*ICSVersions*.
payloadID ?	NMTOKEN	@*payloadID* is a unique identifier with respect to space and time, used for logging purposes to identify documents. This value SHOULD NOT change for retry attempts. See ▸[cXML 1.2].
timestamp	dateTime	The date and time the message was sent. @*timestamp* SHALL NOT change for retry attempts.
Version ? New in PrintTalk 2.1	enumeration	@*Version* SHALL define the version of the **PrintTalk** document. The value of @*Version* SHALL be "2.1" for documents that comply with this specification. **Allowed values are:** 2.0 – Prior version. 2.1 – This version.
Header	element	The header defined in ▸[cXML 1.2] and as further specified in this document.
Request	element	The request defined in ▸[cXML 1.2] and as further specified in this document.

2.1.1 ICS Versions Value

To assist with interoperability conformance the **PrintTalk** can refer to one or more **CIP4** Interoperability Conformance Specification documents. Each document is referenced by using an NMTOKEN that complies with the following:

Value format: <ICSName>_L<ICSLevel>-<ICSVersion>

Example: "Cus-APP_L1-2.0" for the Automated Print Procurement ICS.

2.2 Header

Header is defined in cXML. Refer to the cXML users guide ▸[cXML 1.2].

The *From*, *To* and *Sender* elements allow systems to identify and authorize parties. The explicit or implied value of *Sender*/*Credential*[@*domain*="URL"] also provides the receiver with the URL where it should send any asynchronous requests to the originator.

The *From* and *To* elements in a document SHALL NOT change when a **PrintTalk** document is forwarded by an intermediate system. The *Sender* element SHALL be provided by any system other than the original creator of the document that is specified in *From*. For example, in an *RFQ*, the *From* party is the *Print Buyer*. The *To* party is the *Print Provider*. In the subsequent *Quotation*, those roles are reversed.

Referenced by: *PrintTalk*

Table 2.2: Header Element

NAME	DATA TYPE	DESCRIPTION
From	element	*From* SHALL identify the original sender of the **PrintTalk** transaction. This will typically be the *Print Buyer* in a transaction from *Print Buyer* to *Print Provider*. In a transaction from *Print Provider* to *Print Buyer* it will be typically the *Print Provider*. See ▸ [cXML 1.2].
Sender ?	element	*Sender* SHALL identify the current sender of the **PrintTalk** transaction. This will typically be an intermediate broker such as a web to print system that connects *Print Buyer* and *Print Provider*. If no intermediate actor is involved *Sender* SHALL NOT be provided and *From* SHALL contain the details of the sender. See ▸ [cXML 1.2].
To	element	*To* SHALL identify the final recipient of the **PrintTalk** transaction. This will typically be the *Print Provider* in a transaction from *Print Buyer* to *Print Provider* and will be the *Print Buyer* in a transaction from *Print Provider* to *Print Buyer*. See ▸ [cXML 1.2].

Example 2.1: Header

The following is an example of a *Header* taken from a *PurchaseOrder*. In this example, the *To* party is a *Print Provider* company called "Polkadot Printers". The DUNS number uniquely identifies them. The *From* party is a *Print Buyer*. It too, is uniquely identified by a DUNS number. The *From* is the party sending this **PrintTalk** document. There is no intermediate actor, therefore the *Sender* element is omitted.

```xml
<Header>
  <From>
    <Credential domain="URL">
      <Identity>https://customer.com</Identity>
    </Credential>
    <Credential domain="DUNS">
      <Identity>123456789</Identity>
    </Credential>
    <Credential domain="DescriptiveName">
      <Identity>Peter Pan</Identity>
    </Credential>
    <UserAgent>MIS 2.0</UserAgent>
  </From>
  <To>
    <Credential domain="URL">
      <Identity>https://printer.com</Identity>
    </Credential>
    <Credential domain="DescriptiveName">
      <Identity>Polkadot Printers</Identity>
    </Credential>
    <Credential domain="DUNS">
      <Identity>93009388</Identity>
    </Credential>
  </To>
</Header>
```

2.2.1 From

From SHALL identify the original sender of the **PrintTalk** transaction.

Table 2.3: From Element

NAME	DATA TYPE	DESCRIPTION
Credential +	element	A *Credential* identifies the various parties involved in a transaction.
UserAgent ?	element	*UserAgent* SHOULD contain a combination of agent name and agent version of the software agent that originally sent this **PrintTalk**. See ▸ [cXML 1.2].

2.2.2 Sender

Sender SHALL identify the current sender of the **PrintTalk** transaction.

Table 2.4: Sender Element

NAME	DATA TYPE	DESCRIPTION
Credential +	element	A *Credential* identifies the various parties involved in a transaction.
UserAgent ?	element	*UserAgent* SHOULD contain a combination of agent name and agent version of the software agent that finally sent this **PrintTalk**. See ▸ [cXML 1.2].

2.2.3 To

To SHALL identify the final recipient of the **PrintTalk** transaction.

Table 2.5: To Element

NAME	DATA TYPE	DESCRIPTION
Credential +	element	A *Credential* identifies the various parties involved in a transaction.

2.3 Request

Request is originally defined in cXML. Refer to the cXML users guide ▸ [cXML 1.2]. It has been enhanced in **PrintTalk** to be the container for generic data that is related to a *Business Object*, but independent of the specific type of *Business Object*.

Referenced by: *PrintTalk*

Table 2.6: Request Element

NAME	DATA TYPE	DESCRIPTION
BusinessID	NMTOKEN	The unique identifier for this *Request* and for the *Business Object* it contains.
BusinessRefID ?	NMTOKEN	The value of @*BusinessRefID* SHALL be the same as the @*BusinessID* of some other *Business Object* which acts as the primary parameter to the *Business Transaction* that this **PrintTalk** document represents. Prior to a *PurchaseOrder*, a @*BusinessRefID* SHALL refer to a *Business Object* that was received from the other party. Once a *PurchaseOrder* has been placed, *Request*/@*BusinessID* of the *PurchaseOrder* SHALL be used as the @*BusinessRefID* of all following *Business Objects.* See ▸ Table 6.1 Business Object References.

Table 2.6: Request Element

NAME	DATA TYPE	DESCRIPTION
CustomerProjectID ?	string	The internal order number of the entire order in the system of the customer. This identifier MAY be provided by the *Print Buyer* in the initial **PurchaseOrder** or **RFQ**. If provided by the *Print Buyer* in the initial **PurchaseOrder** or **RFQ**, it SHALL be provided by the *Print Provider* and the *Print Buyer* in all subsequent business objects that directly or indirectly reference the initial **PurchaseOrder** or **RFQ** in @*BusinessRefID*. @*CustomerProjectID* SHALL match *xjdf:CustomerInfo*/@*CustomerProjectID* if *xjdf:CustomerInfo*/@*CustomerProjectID* is provided. **Note:** See also *xjdf:CustomerInfo*/@*CustomerOrderID* for the internal identifier of an individual item.
DescriptiveName ?	string	Human readable summary description of the *Business Object* in this **Request**.
<**BusinessObject**>	element	A **PrintTalk Request** SHALL contain exactly one *Business Object*. See ▸ Chapter 3 Business Objects.
<foreign namespace elements> *	element	Any elements in a foreign namespace. These elements SHOULD NOT duplicate any **PrintTalk** functionality.

Example 2.2:   Request

The following is an example of a **Request**.

```
<PrintTalk xmlns="http://www.printtalk.org/schema_20"
   payloadID="P_000431" timestamp="2019-09-30T13:42:14+02:00"
   xmlns:xjdf="http://www.CIP4.org/JDFSchema_2_0">
   <Header>
     <From>
       <Credential domain="URL">
         <Identity>https://printer.com</Identity>
       </Credential>
       <UserAgent>MIS 2.0</UserAgent>
     </From>
     <To>
       <Credential domain="URL">
         <Identity>https://customer.com</Identity>
       </Credential>
     </To>
   </Header>
   <Request BusinessID="Confirmation_000432" BusinessRefID="PO_ID">
     <Confirmation>
       <xjdf:Comment>Thank you for your order</xjdf:Comment>
     </Confirmation>
   </Request>
</PrintTalk>
```

2.4 PrintTalk Extensibility

The **PrintTalk** specification aims to support plug-and-play as much as possible. Nonetheless, **PrintTalk** is meant to be flexible and therefore, as each vendor may have private data to include in the **PrintTalk** files, it allows for this data to be added using a foreign namespace. See ▸ Section 1.2 Use of XML. Foreign namespace extensions SHOULD NOT duplicate functionality of attributes and elements defined by **PrintTalk**.

2.4.1 Elements

Elements in a foreign namespace SHALL NOT be specified in any **PrintTalk** element other than the **Request** element. The children of the **Request** element SHALL be ordered so that all elements in a foreign namespace follow all of the elements

in the **PrintTalk** namespace. The valid locations of foreign namespace elements in **XJDF** are also valid in **XJDF** descendents of **PrintTalk**.

2.4.2 Attributes

Attributes in a foreign namespace MAY be added to any **PrintTalk** element.

2.4.3 Attribute values

Throughout the specification NMTOKEN is used as a data type that allows for an unbounded set of values; the writer is free to use any value that conforms to the NMTOKEN requirements.

In a number of cases, e.g. *Credential*/@*domain*, the specification lists a set of values for specific uses. The writer is free to use other values as long as they do not duplicate the functionality of those already listed. In these cases it is RECOMMENDED that the value uses a namespace like prefix as an indication of the source namespace, e.g. "pfx:Value".

3 Business Objects

This chapter specifies the syntax of the individual *Business Objects.* See ▸ Chapter 6 Business Object Workflows for a discussion of the interactions of *Business Objects.*

3.1 Cancellation

A *Print Buyer* or a *Print Provider* SHALL send a *Cancellation* to cancel a previously sent *PurchaseOrder*.
Note: A *Print Buyer* will typically cancel a *PurchaseOrder* if the *Print Product* is no longer required, whereas a *Print Provider* will typically cancel a *PurchaseOrder* if no useable content has been provided within a deadline.

For a *Print Buyer* or a *Print Provider* to cancel a *PurchaseOrder*, it SHALL send a *Cancellation* whose *Request*/@*BusinessRefID* references the *PurchaseOrder*.

If a *Print Buyer* sends a *Cancellation*, the *Print Provider* SHALL respond with a *Confirmation* or *Refusal*. If a *Print Provider* sends a *Cancellation*, the *Print Buyer* MAY respond with a *Confirmation*, but SHALL NOT respond with a *Refusal*.
Note: If a *Cancellation* of a *PurchaseOrder* is accepted, production of the job will cease. Any work that has already been accrued can still be invoiced. If a *Cancellation* is refused, e.g. if the product has already been produced and shipped, it can still be delivered.

If no *PurchaseOrder* has been sent and a *Print Provider* intends to cancel a pending *Quotation*, the *Print Provider* SHALL send a *Cancellation* that references the *Quotation*.

References: *PurchaseOrder*, *Quotation*

Flow: *Print Buyer* to *Print Provider*, *Print Provider* to *Print Buyer*

Table 3.1: Cancellation Element

NAME	DATA TYPE	DESCRIPTION
JobIDRef ?	NMTOKEN	Reference to *xjdf:XJDF*/@*JobID* of the cancelled job. If not present, all jobs of the *PurchaseOrder* referred to in *Request*/@*BusinessRefID* SHALL be cancelled.
xjdf:Comment ?	element	Human readable comments about this cancellation.

3.2 Confirmation

If a *Print Provider* accepts a *PurchaseOrder* or *Cancellation*, it SHOULD send a *Confirmation* whose *Request*/@*BusinessRefID* SHALL reference the accepted *PurchaseOrder* or *Cancellation*. If a *Print Buyer* receives a *Cancellation* for a *PurchaseOrder*, the *Print Buyer* MAY send a *Confirmation* whose *Request*/@*BusinessRefID* SHALL reference the cancelled *PurchaseOrder*.

References: *Cancellation*, *PurchaseOrder*

Flow: *Print Buyer* to *Print Provider*, *Print Provider* to *Print Buyer*

Table 3.2: Confirmation Element

NAME	DATA TYPE	DESCRIPTION
xjdf:Comment ?	element	Human readable comments about this confirmation.

3.3 ContentDelivery

A *ContentDelivery* allows the *Print Buyer* to upload any type of content to the *Print Provider*, e.g. PDF files, variable data lists, barcodes etc.

The *Request*/@*BusinessRefID* of the *ContentDelivery* *Business Object* SHALL reference the order's *PurchaseOrder* if one has been sent. The order's *PurchaseOrder* is either the initial *PurchaseOrder* if no *Change Orders* have occurred, or the latest accepted *PurchaseOrder* if a *Change Order* has been made. If the content is delivered while an *RFQ* is pending, the *RFQ* SHALL be referenced.

References: *PurchaseOrder*, *RFQ*

Flow: *Print Buyer* to *Print Provider*

Table 3.3: ContentDelivery Element

NAME	DATA TYPE	DESCRIPTION
UpdateMethod	enumeration	@*UpdateMethod* SHALL define the usage of the content. **Allowed values are:** Add – Add the content to the product. Replace – Replace existing content in the same scope with the delivered content.
xjdf:XJDF	element	*xjdf:XJDF* SHALL include an *xjdf:ResourceSet*/[@*Name*="RunList"] that references the content. **Note:** An entire *xjdf:XJDF* is provided in order to provide any required product part context in a multi-part product for the content such as cover or body.

Example 3.1:   Content Delivery

```
<ContentDelivery UpdateMethod="Add">
  <xjdf:XJDF JobID="cart1.item1" Types="Product Delivery">
    <xjdf:ResourceSet Name="RunList" Usage="Input">
      <xjdf:Resource>
        <xjdf:RunList>
          <xjdf:FileSpec MimeType="application/pdf" URL="https://myFileSource/pdfs/file1.pdf"/>
        </xjdf:RunList>
      </xjdf:Resource>
    </xjdf:ResourceSet>
  </xjdf:XJDF>
</ContentDelivery>
```

3.4 ContentDeliveryResponse

A *ContentDeliveryResponse* allows the *Print Provider* to provide the *Print Buyer* with information about previously uploaded content deliveries. *Request*/@*BusinessRefID* of the *ContentDeliveryResponse* *Business Object* SHALL reference the *Business Object* that provided the content. This SHALL be either a *PurchaseOrder* or a *ContentDelivery*.

Exactly one *ContentDeliveryResponse* SHALL be provided for a content delivery that was provided in a *ContentDelivery* *Business Object*, even if the *Business Object* contains multiple content items.

At most one *ContentDeliveryResponse* SHALL be provided for a content delivery that was provided in a *PurchaseOrder* *Business Object*, even if the *Business Object* contains multiple content items. *ContentDeliveryResponse* SHALL NOT be provided for a *PurchaseOrder* *Business Object* after a *Confirmation* or *Refusal* that references the *PurchaseOrder* has been sent.

References: *ContentDelivery*, *PurchaseOrder*

Flow: *Print Provider* to *Print Buyer*

Table 3.4: ContentDeliveryResponse Element (Sheet 1 of 2)

Result	enumeration	Result of the entire content delivery. **Allowed values are:** Accepted – All of the content items were accepted and all of the content for the job is now in place. Warnings may have occurred. AcceptedWaiting – All of the content items were accepted. There is still one or more missing content items for the *PurchaseOrder*. Warnings may have occurred. Rejected – Rejected SHALL be specified if one or more of the content items was rejected. Additional details SHOULD be provided in an *xjdf:PreflightReport* entry in the *xjdf:AuditPool*.

Table 3.4: ContentDeliveryResponse Element (Sheet 2 of 2)

xjdf:AuditPool ?	element	If present, *xjdf:AuditPool* SHALL contain at least one *xjdf:PreflightReport* that provides details of the result of the upload. The selection context of any *xjdf:AuditResource* in the *xjdf:AuditPool* SHALL match one of the selection contexts of the corresponding *xjdf:RunList* from the *ContentDelivery*. **Note: XJDF** selection context is defined by the values of the *xjdf:Part* elements of a resource. See ▸ [XJDF 2.1].

3.5 Invoice

To request payment, the *Print Provider* SHALL send an *Invoice* whose *Request*/@*BusinessRefID* references the *PurchaseOrder*.

References: *PurchaseOrder*

Flow: *Print Provider* to *Print Buyer*

Table 3.5: Invoice Element

NAME	DATA TYPE	DESCRIPTION
DueDate	dateTime	@*DueDate* shall specify the due date when payment for this *Invoice* is due and after which a subsequent *Invoice* MAY be sent and MAY contain late payment fees.
MasterContract ?	element	*MasterContract* SHALL specify a *Master Contract* that has been applied to calculate the *Invoice*.
Pricing	element	*Pricing* SHALL specify the price of the products or services described by this *Invoice*.

3.6 OrderStatusRequest

An *OrderStatusRequest* allows the *Print Buyer* to query the *Print Provider* for details of the order status.

If a *Print Buyer* wants the status of an order, it SHALL send to the *Print Provider* an *OrderStatusRequest* whose *Request*/@*BusinessRefID* references the order's *PurchaseOrder*. The order's *PurchaseOrder* is defined as either the initial *PurchaseOrder* if no *Change Orders* have occurred or the latest accepted *PurchaseOrder* if a *Change Order* has been made.

References: *PurchaseOrder*

Flow: *Print Buyer* to *Print Provider*

Table 3.6: OrderStatusRequest Element

NAME	DATA TYPE	DESCRIPTION
Subscribed	boolean	If @*Subscribed* = "true", then the child *StatusRequest* elements are subscriptions for multiple *OrderStatusResponse* messages; else the *StatusRequest* elements are a request for one individual *OrderStatusResponse* message. The trigger conditions for subscribed *OrderStatusResponse* messages are implementation dependent. *OrderStatusResponse* messages SHOULD be sent whenever a relevant status change of the subscribed order takes place, i.e. whenever a new *xjdf:Milestone* is reached.
StatusRequest +	element	Individual status request description.

3.7 OrderStatusResponse

An *OrderStatusResponse* is a direct or triggered response to an *OrderStatusRequest*/*StatusRequest* or a triggered response to a *PurchaseOrder*/*StatusRequest*.

If a *Print Provider* receives an *OrderStatusRequest*[@*Subscribed*="true"], it SHALL immediately send to the *Print Buyer* one *OrderStatusResponse* with the latest reached milestone for each *OrderStatusRequest*/*StatusRequest*.

A *Print Provider* SHOULD send to a *Print Buyer* an automatically generated *OrderStatusResponse* for each *StatusRequest*[@*Subscribed*="true"] whenever a condition, such as the availability of a tracking identifier or the reaching of a new milestone, matches the subscription defined by *StatusRequest*.

The amounts specified in the scope of *xjdf:AuditResource* SHALL be the total amounts at the time of sending including any amounts sent in previous *OrderStatusResponse* business objects that refer to the same entity.

Request/@*BusinessRefID* of the *OrderStatusResponse* SHALL reference *Request*/@*BusinessID* of the *OrderStatusRequest* or *PurchaseOrder*.

References: *OrderStatusRequest*, *PurchaseOrder*

Flow: *Print Provider* to *Print Buyer*

Table 3.7: OrderStatusResponse Element

NAME	DATA TYPE	DESCRIPTION
JobIDRef	NMTOKEN	@*JobIDRef* SHALL identify the job that this *OrderStatusResponse* applies to.
xjdf:AuditPool	element	*xjdf:AuditPool* SHALL contain at least one *xjdf:AuditNotification* or *xjdf:AuditResource* that contains information that was requested in a *StatusRequest* element. *xjdf:AuditNotification/xjdf:Notification* elements SHOULD contain an *xjdf:Milestone* child element that describes the current status of the job. See ▸ [XJDF 2.1].

Example 3.2: Order Status Response

```
<OrderStatusResponse JobIDRef="j1">
  <xjdf:AuditPool>
    <xjdf:AuditNotification>
      <xjdf:Header AgentName="MIS" AgentVersion="2.0" DeviceID="MIS"
        Time="2019-09-30T13:42:14+02:00"/>
      <xjdf:Notification Class="Event" JobID="j1">
        <xjdf:Milestone MilestoneType="ShippingInProgress"/>
      </xjdf:Notification>
    </xjdf:AuditNotification>
    <xjdf:AuditResource>
      <xjdf:Header AgentName="MIS" AgentVersion="2.0" DeviceID="MIS"
        Time="2019-09-30T13:42:14+02:00"/>
      <xjdf:ResourceInfo>
        <xjdf:ResourceSet Name="DeliveryParams">
          <xjdf:Resource>
            <xjdf:DeliveryParams TrackingID="T123"/>
          </xjdf:Resource>
        </xjdf:ResourceSet>
      </xjdf:ResourceInfo>
    </xjdf:AuditResource>
  </xjdf:AuditPool>
</OrderStatusResponse>
```

3.8 ProofApprovalRequest

If a *Print Provider* wants a *Print Buyer* to approve a soft or hard copy order, the *Print Provider* SHALL send to the *Print Buyer* a *ProofApprovalRequest* whose *Request*/@*BusinessRefID* references the order's *PurchaseOrder*.
Note: *xjdf:ContentCheckIntent* allows the *Print Buyer* to define the desired proofing and preflighting workflow for a specific print job.
Note: Multiple *ProofApprovalRequest* elements need to be sent if multiple proofs need to be approved.

References: *PurchaseOrder*

Flow: *Print Provider* to *Print Buyer*

PROOFAPPROVALRESPONSE

Table 3.8: ProofApprovalRequest Element

NAME	DATA TYPE	DESCRIPTION
JobIDRef	NMTOKEN	Reference to the @*JobID* this proof corresponds to.
xjdf:ProofItem	element	Details of the item to be proofed. See ▸ [XJDF 2.1].

3.9 ProofApprovalResponse

A ***ProofApprovalResponse*** is the response to a ***ProofApprovalRequest***.

If a *Print Buyer* receives a ***ProofApprovalRequest***, it SHOULD send to the *Print Provider* a ***ProofApprovalResponse*** whose ***Request***/@*BusinessRefID* SHALL reference the ***ProofApprovalRequest*** and contains the *Print Buyer's* approval or rejection of a proof.

References: ***ProofApprovalRequest***

Flow: *Print Buyer* to *Print Provider*

Table 3.9: ProofApprovalResponse Element

NAME	DATA TYPE	DESCRIPTION
JobIDRef	NMTOKEN	Reference to the @*JobID* this proof corresponds to.
xjdf:ApprovalDetails	element	Describes the overall result of the approval. See ▸ [XJDF 2.1].
xjdf:ProofItem ?	element	Details of the item that has been proofed. See ▸ [XJDF 2.1].

3.10 PurchaseOrder

The *Print Buyer* SHALL accept a ***Quote*** of the ***Quotation*** for purchase of a *Print Product* by sending a ***PurchaseOrder*** that references the ***Quote***. A *Print Buyer* MAY initiate a *Change Order* by sending a ***PurchaseOrder*** that references an existing confirmed ***PurchaseOrder***. However, a ***PurchaseOrder*** MAY also be the first *Business Object* in a *Negotiation Phase*, especially for a reorder of a previously produced *Print Product*.

If a *Print Provider* accepts a ***PurchaseOrder***, it SHALL send a ***Confirmation*** that references the ***PurchaseOrder*** prior to the deadline specfied in @*Expires*. If a *Print Provider* does not accept a ***PurchaseOrder***, it SHALL send a ***Refusal*** that references the ***PurchaseOrder***.

References: N/A (***PurchaseOrder*** can be the initiating transaction), ***PurchaseOrder***, ***Quotation***

Flow: *Print Buyer* to *Print Provider*

Table 3.10: PurchaseOrder Element (Sheet 1 of 2)

NAME	DATA TYPE	DESCRIPTION
Expires	dateTime	Date/time when this ***PurchaseOrder*** becomes invalid. If no ***Confirmation*** has been sent by *Print Provider* within the time specified, the *Print Buyer* MAY assume that the ***PurchaseOrder*** has been refused.
QuoteID ?	NMTOKEN	If this ***PurchaseOrder*** references a ***Quotation*** then @*QuoteID* SHALL be used by the *Print Buyer* to select which of the existing quotes is to be used for the purchase order and SHALL contain one of the values of ***Quote***/@*QuoteID* from the referenced ***Quotation***. If this ***PurchaseOrder*** does not reference a ***Quotation*** then @*QuoteID* SHALL NOT be specified.
ReorderID ?	NMTOKENS	Whitespace separated list of unique values that refer to ***Request***/@*BusinessID* values of **PrintTalk** ***PurchaseOrder*** documents that are the basis for this (collected) re-order, if the re-order is the starting point.

Table 3.10: PurchaseOrder Element (Sheet 2 of 2)

NAME	DATA TYPE	DESCRIPTION
MasterContract ?	element	If **Pricing** is present, **MasterContract** SHALL specify a *Master Contract* that has been applied to calculate the **Pricing**. If **Pricing** is not present, **MasterContract** SHALL specify a *Master Contract* that SHALL be applied to calculate the price.
Pricing ?	element	**Pricing** SHALL specify the binding price of the purchase order. **Note:** This price will typically originate from a web to print or similar e-commerce system.
StatusRequest *	element	Each **StatusRequest** defines a subscription for automated status updates which relate to production orders that are specified in *xjdf:XJDF* of this **PurchaseOrder**. If present, the value of **StatusRequest**/@*JobIDRef* SHALL be the same as the value of one of the *xjdf:XJDF*/@*JobID* values in this **PurchaseOrder**.
xjdf:Comment ?	element	Human readable comments about this purchase order.
xjdf:XJDF +	element	Description of one or more *Print Products* or *Catalog Items*. The **XJDF** elements MAY be used to specify very detailed *Print Products*, or they MAY also be used to describe the procurement of finished goods in catalog based environments. **Note:** Multiple *xjdf:XJDF* in a **PurchaseOrder** are an explicit request to purchase multiple items, e.g. when describing a shopping cart in a web to print application. See ▸ [XJDF 2.1].

Example 3.3:   Purchase Order

```
<PurchaseOrder Expires="2019-09-30T13:42:14+02:00">
  <xjdf:XJDF JobID="cart1.item1" Types="Product">
    <xjdf:ProductList>
      <xjdf:Product Amount="100" DescriptiveName="BusinessCards" IsRoot="true"/>
    </xjdf:ProductList>
  </xjdf:XJDF>
  <xjdf:XJDF JobID="cart1.item2" Types="Product">
    <xjdf:ProductList>
      <xjdf:Product Amount="500" DescriptiveName="Brochures" IsRoot="true"/>
    </xjdf:ProductList>
  </xjdf:XJDF>
</PurchaseOrder>
```

3.11 Quotation

A **Quotation** is a list of **Quotes**, each of which is an offer to provide a variation of a set of printed products or catalog items. Usually a *Print Provider* creates a **Quotation** as a response to an **RFQ**.

If the *Print Provider* chooses to respond to an **RFQ**, it SHALL send a **Quotation** whose **Request**/@*BusinessRefID* references the received **RFQ**. A **Quotation** MAY contain multiple **Quote** elements that describe options that the *Print Provider's* estimator generated.

If a *Print Provider* chooses to initiate a *Change Order* for a *Confirmed Business Object*, it SHALL send a **Quotation** whose **Request**/@*BusinessRefID* references the **PurchaseOrder** for the *Confirmed Business Object*. If such a **Quotation** for a *Confirmed Business Object* expires, and the *Print Provider* cannot produce the job as quoted due to issues wihin the responsiblilty of the *Print Buyer*, the *Print Provider* MAY send a **Cancellation** that references the original **PurchaseOrder**.

When a *Print Buyer* receives a **Quotation**, the *Print Buyer* MAY

- accept the **Quotation** by sending a **PurchaseOrder** that references the **Quote** that the *Print Buyer* wishes to accept from the list of **Quote** elements in the **Quotation**.
- decline the **Quotation** by either sending a **Refusal** or letting it expire. The *Print Buyer* MAY choose to create a new **RFQ** that requests a *Requote*.

References:	**PurchaseOrder**, **RFQ**
Flow:	*Print Provider* to *Print Buyer*

Table 3.11: Quotation Element

NAME	DATA TYPE	DESCRIPTION
Estimate	boolean	For each *Quote* element inside this *Quotation*, the price SHALL be a binding amount if @*Estimate* = "false", otherwise, the *Print Buyer* SHALL treat the price as an estimate only.
Expires	dateTime	Date/time when this *Quotation* becomes invalid.
MasterContract ?	element	*MasterContract* SHALL specify a *Master Contract* that has been applied to calculate the *Quote*/*Pricing* definitions.
Quote +	element	Each *Quote* element SHALL describe a complete distinct variation of all *Print Products* quoted. A *Quote* SHALL NOT be a quote for an individual part or *Item* of the *Order*.
xjdf:Comment ?	element	Human readable comments about this quotation.

Example 3.4:   Quotation

```xml
<Quotation Estimate="false" Expires="2019-10-16T18:00:00+02:00">
  <Quote QuoteID="q1">
    <Pricing Currency="GBP">
      <Price Amount="500"
        DescriptiveName="500 simple business cards" Price="250"
        PriceType="Total" TaxType="Net">
        <Additional Amount="100" BaseAmount="500" BasePrice="250" Price="40"/>
        <Additional Amount="500" BaseAmount="1000" BasePrice="400" Price="150"/>
      </Price>
    </Pricing>
    <xjdf:XJDF DescriptiveName="simple business cards" JobID="jobid" Types="Product">
    </xjdf:XJDF>
  </Quote>
  <Quote QuoteID="q2">
    <Pricing Currency="GBP">
      <Price Amount="500"
        DescriptiveName="500 varnished business cards" Price="350"
        PriceType="Total" TaxType="Net">
        <Additional Amount="100" BaseAmount="500" BasePrice="350" Price="50"/>
        <Additional Amount="500" BaseAmount="1000" BasePrice="500" Price="200"/>
      </Price>
    </Pricing>
    <xjdf:XJDF DescriptiveName="varnished business cards"
      JobID="jobid" Types="Product">
    </xjdf:XJDF>
  </Quote>
</Quotation>
```

3.11.1 Quote

A *Quote* element describes a quote for a particular variation of a product or products.

Table 3.12: Quote Element

NAME	DATA TYPE	DESCRIPTION
DeviationCause ?	enumeration	@*DeviationCause* specifies the reason for providing a deviating *Quote*. @*DeviationCause* SHOULD NOT be specified unless @*Deviations* is specified. If @*Deviations* is specified, @*DeviationCause* SHOULD be specified. **Note:** @*DeviationCause* specifies the reason for the deviation, whereas @*Deviations* specifies what deviates. **Allowed values are:** ContentMismatch – The content data that was provided by the *Print Buyer* does not meet the specifications of the *RFQ* or *PurchaseOrder*. Efficiency – The suggested product is similar and can be produced more economically. Other – Any reason not covered by the other values. Production – A product cannot be produced as requested, e.g a two sided brochure with three pages. Quality – The suggested product has higher quality.
Deviations ?	NMTOKENS	If this *Quote* differs from the original request, @*Deviations* SHOULD specify the general areas where the quote differs. **Note:** @*Deviations* specifies what deviates, whereas @*DeviationCause* specifies the reason for the deviation. **Values include:** Assembling – The composite product is produced with a different assembly. Binding – The product is produced with a different binding. Color – The product is produced with a different set of colorants. ContentCheck – The product is produced with a different proofing or preflighting. DeliveryDate – The delivery date differs. Embossing – The product is produced with a different embossing. Folding – The product is produced with a different folding, creasing or perforating. HoleMaking – The product is produced with different holes in the media. Laminating – The product is produced with different laminating. Layout – The product is produced with different layout. Media – The product is produced with a different substrate. Production – The product is produced with different production techniques. ShapeCutting – The product is produced with different shape cutting. Technology – The product is produced with a different print technology. Variable – The product is produced with different variable content settings. **Note:** See ▸ Section 2.4.3 Attribute values for extending this list.
QuoteID	NMTOKEN	The unique identifier for this *Quote* element. @*QuoteID* SHALL be unique only in the context of a *Quotation*.
Pricing	element	*Pricing* SHALL be used to indicate the list of price *Items* that are associated with this quote.
xjdf:Comment [@*Type*= "Deviation"] ?	element	If this *Quote* does not match the requirements of the *RFQ*, then this element SHOULD contain a human readable description of the deviations.
xjdf:Comment [@*Type*= "JobDescription"] ?	element	Additional human readable comments about this quote.
xjdf:XJDF *	element	**XJDF** representing the *Print Products* or *Catalog Items* that this *Quote* has been made for. Potentially this MAY differ from the *RFQ* that is referenced by *Request*/@*BusinessRefID*, in which case details of any deviations and their magnitude SHALL be described in *xjdf:Comment*[@*Type*="Deviation"]. If a *Quote* represents the product or *Catalog Item* defined in the *RFQ* without deviation, then the **XJDF** NEED NOT be provided. See ▸ [XJDF 2.1].

3.12 Refusal

A *Print Buyer* or a *Print Provider* sends a **Refusal** to decline receipt of a **Cancellation**, **PurchaseOrder**, **Quotation** or **RFQ**.

If a *Print Provider* chooses to decline either a **Cancellation** or a **PurchaseOrder** sent by a *Print Buyer*, it SHALL send a **Refusal** whose **Request**/@*BusinessRefID* references the declined *Business Object*. If a *Print Buyer* chooses to decline a **Quotation** sent by a *Print Provider* or a *Print Provider* chooses to decline an **RFQ** sent by a *Print Buyer*, it SHOULD send a **Refusal** whose **Request**/@*BusinessRefID* references the declined *Quotation* or **RFQ** or let the *Business Object* expire.

References: **Cancellation**, **PurchaseOrder**, **Quotation**, **RFQ**

Flow: *Print Buyer* to *Print Provider*, *Print Provider* to *Print Buyer*

Table 3.13: Refusal Element

NAME	DATA TYPE	DESCRIPTION
Reason	enumeration	Machine readable general reason for declining the request. **Allowed values are:** Busy – No production resources are available to produce the request within the required deadline. InvalidPrice – When the **Refusal** comes from the *Print Provider*, the price is too low. When the **Refusal** comes from the *Print Buyer*, the price is too high. Milestone – A production milestone has passed and changes can no longer be accepted. Other – Any other reason. If 'Other' is specified @*ReasonDetails* SHOULD be specified. WrongProduct – The printer is not capable of producing the requested type of product.
ReasonDetails ?	NMTOKENS	Additional details to describe the reason for refusal. **Values include those from** ▸ Refusal codes.
xjdf:Comment ?	element	Human readable comments about this refusal.

3.12.1 Refusal codes

The following table defines a list of suggested values for **Refusal**/@*ReasonDetails*.

Note: See ▸ Section 2.4.3 Attribute values for extending this list.

Table 3.14: Refusal codes (Sheet 1 of 2)

REASON	REASON DETAILS	DESCRIPTION
Busy	Deadline	The requested delivery date cannot be met.
Busy	OverCapacity	The requested amount is too high.
Busy	UnderCapacity	The requested amount is too low.
InvalidPrice	NonStandard	The requested products do not fit the contract for the assumed price.
InvalidPrice	WrongPricing	The contractual price lists have been updated and the price is no longer valid.
WrongProduct	Assembling	The requirements specified in *xjdf:AssemblingIntent* cannot be achieved.
WrongProduct	Binding	The requirements specified in *xjdf:BindingIntent* cannot be achieved.
WrongProduct	Color	The requirements specified in *xjdf:ColorIntent* cannot be achieved.
WrongProduct	ContentCheck	The requirements specified in *xjdf:ContentCheckIntent* cannot be achieved.
WrongProduct	Embossing	The requirements specified in *xjdf:EmbossingIntent* cannot be achieved.
WrongProduct	Folding	The requirements specified in *xjdf:FoldingIntent* cannot be achieved.
WrongProduct	HoleMaking	The requirements specified in *xjdf:HoleMakingIntent* cannot be achieved.
WrongProduct	Laminating	The requirements specified in *xjdf:LaminatingIntent* cannot be achieved.

Table 3.14: Refusal codes (Sheet 2 of 2)

REASON	REASON DETAILS	DESCRIPTION
WrongProduct	Layout	The requirements specified in *xjdf:LayoutIntent* cannot be achieved.
WrongProduct	Media	The requirements specified in *xjdf:MediaIntent* cannot be achieved.
WrongProduct	Production	The requirements specified in *xjdf:ProductionIntent* cannot be achieved.
WrongProduct	ShapeCutting	The requirements specified in *xjdf:ShapeCuttingIntent* cannot be achieved.
WrongProduct	Variable	The requirements specified in *xjdf:VariableIntent* cannot be achieved.

3.13 RFQ

A *Print Buyer* sends an *RFQ* *Business Object*, a 'Request For Quote', to a *Print Provider* to request a *Quotation*. The *Negotiation Phase* starts with an *RFQ*.

An *RFQ* SHALL convey the unambiguous intentions of the *Print Buyer* to the *Print Provider*.

A *Print Buyer* sends an *RFQ* whose semantics vary depending on the value of *Request*/@*BusinessRefID* which defines the context of the *RFQ*:

- **Not specified:** If a *Print Buyer* chooses to initiate a new negotiation by making a request for quote to a *Print Provider*, it SHALL send an *RFQ* with no *Request*/@*BusinessRefID*.
- **Quotation:** If a *Print Buyer* receives a *Quotation* for a job and wants a *Requote* for a variation of the job, it SHALL send an *RFQ* whose *Request*/@*BusinessRefID* references the received *Quotation*.
- **PurchaseOrder:** If a *Print Buyer* wants to initiate a *Change Order* for a *Confirmed Business Object* and the *PurchaseOrder* form of a *Change Order* cannot be used because there are changes that don't have quotes, the *Print Buyer* SHALL send an *RFQ* whose *Request*/@*BusinessRefID* references the confirmed *PurchaseOrder*. A *Change Order RFQ* asks what the consequences would be for a certain change of the *Print Product*. See *PurchaseOrder* for an alternate *Change Order* that a *Print Buyer* can send.

When a *Print Provider* receives an *RFQ*, the *Print Provider* SHALL either

- accept the *RFQ* by sending a *Quotation* to the *Print Buyer*.
- decline the *RFQ* by either sending a *Refusal* to the *Print Buyer* or letting the *RFQ* expire.

References: N/A (*RFQ* can be the initiating transaction), *PurchaseOrder*, *Quotation*

Flow: *Print Buyer* to *Print Provider*

Table 3.15: RFQ Element (Sheet 1 of 2)

NAME	DATA TYPE	DESCRIPTION
AmountPrices ?	IntegerList	@*AmountPrices* SHALL specify a list of base quantities, for each of which the resulting *Quotation* SHOULD provide in each *Quote*/*Pricing* element a *Price* element with the cost in @*Price* and quantity in @*Amount*. If @*AmountPrices* is specified then the *Quote*/*Pricing*/*Price* elements SHOULD also contain *Additional* elements. If @*AmountPrices* is not specified the requested amounts SHALL be extracted from *xjdf:XJDF*/*ProductList*/*Product*/@*Amount*.
Currency ?	Currency	The value of @*Currency* SHALL be copied to all returned *Quotation*/*Quote*/*Pricing*/@*Currency* attributes. **Note:** See ▸ Section 1.3.1 Currency for definition of valid currency values, and refer to ▸ [ISO4217].
Estimate ?	boolean	If "false", the *Print Provider* SHALL provide prices in the responding *Quotation* that will be binding and not altered in any subsequent *Invoice* that is raised as a result of this *Quotation*. If not "false", the *Print Provider* SHOULD provide prices in the responding *Quotation* that are estimates; consequently the prices in any subsequent *Invoice* may vary from those in the *Quotation*. However, the *Print Provider* MAY provide prices that are binding. In all cases the *Print Provider* SHALL clearly indicate whether the responding *Quotation* is a binding amount or an estimate in *Quotation*/@*Estimate*.
Expires	dateTime	Date/time when this *RFQ* becomes invalid.

Table 3.15: RFQ Element (Sheet 2 of 2)

NAME	DATA TYPE	DESCRIPTION
QuoteIDs ? New in PrintTalk 2.1	NNTOKENS	If this *RFQ* references a *Quotation* then @*QuoteIDs* MAY be used by the *Print Buyer* to select which of the existing quotes are to be requoted. In this case @*QuoteIDs* SHALL contain one or more of the values of *Quote*/@*QuoteID* from the referenced *Quotation*. If this *RFQ* does not reference a *Quotation* then @*QuoteIDs* SHALL NOT be specified.
ReorderID ?	NMTOKENS	Whitespace separated list of unique values that refer to *Request*/@*BusinessID* values of **PrintTalk** *PurchaseOrder* documents that are the basis for this (collected) *RFQ*. The *RFQ* is intended to lead to a re-order.
MasterContract ?	element	*MasterContract* SHALL specify a *Master Contract* that SHALL be applied to calculate the prices in the returned *Quotation*.
xjdf:Comment ?	element	Human readable comments about this request for quote.
xjdf:XJDF +	element	Description of one or more *Print Products* or *Catalog Items* that a *Quotation* is requested for. See ▸ [XJDF 2.1].

Example 3.5: Request For Quote

```
<RFQ AmountPrices="1000 5000" Expires="2019-10-05T12:00:00+02:00">
  <xjdf:XJDF JobID="J1" Types="Product">
    <xjdf:ProductList>
      <xjdf:Product DescriptiveName="BusinessCards" IsRoot="true"/>
    </xjdf:ProductList>
  </xjdf:XJDF>
</RFQ>
```

3.14 StockLevelRequest

A *StockLevelRequest* allows the *Print Buyer* to query the *Print Provider* for the available stock levels of preprinted or non-printed items.

References: N/A (*StockLevelRequest* is the initiating transaction)

Flow: *Print Buyer* to *Print Provider*

Table 3.16: StockLevelRequest Element (Sheet 1 of 2)

NAME	DATA TYPE	DESCRIPTION
Availability ?	enumerations	@*Availability* SHALL specify a list of availabilities for the items whose stock levels are being requested. **Allowed values are from:** ▸ Table 3.17 Availability Attribute Values.
Currency ?	Currency	The value of @*Currency* SHALL be copied to all returned *StockLevelResponse*/*StockLevel*/*Pricing*/@*Currency* attributes. **Note:** See ▸ Section 1.3.1 Currency for definition of valid currency values, and refer to ▸ [ISO4217].
DisplayPrice ?	boolean	If true then the response SHALL include a *StockLevel*/*Pricing* element and all *StockLevel*/*Pricing*/*Price* elements SHALL have a value in the @*Price* attribute.
ExternalID ?	regexp	Regular expression that filters the returned list of *StockLevel* elements to only include elements whose *StockLevel*/@*ExternalID* matches @*ExternalID*. If @*ExternalID* is not specified, *StockLevel* elements SHALL be returned for all products.

Table 3.16: StockLevelRequest Element (Sheet 2 of 2)

NAME	DATA TYPE	DESCRIPTION
MasterContract ?	element	*MasterContract* SHALL specify a *Master Contract* that SHALL be applied to calculate the price in the returned *StockLevel* elements. *MasterContract* SHALL NOT be present if @*DisplayPrice*="false".

Table 3.17: Availability Attribute Values

VALUE	DESCRIPTION
Available	Defines stocked items that are currently available. *StockLevel*/@*Amount* SHALL show the number of items currently available. *StockLevel*/@*ProductionDuration* SHALL NOT be specified.
Deliverable	Defines stocked items that are currently not available, but are expected to be available at some future time. *StockLevel*/@*Amount* SHALL show the amount and *StockLevel*/@*ProductionDuration* SHALL show when the items are expected to be available.
Undeliverable	Defines either stocked items that are out of stock and for whom no future availability can be estimated, or items that are no longer stocked and may never be available. *StockLevel*/@*Amount* and *StockLevel*/@*ProductionDuration* SHALL NOT be specified.

3.15 StockLevelResponse

A *StockLevelResponse* is the response to a *StockLevelRequest*.

If a *Print Provider* receives a *StockLevelRequest*, it SHALL send to the *Print Buyer* a *StockLevelResponse* whose *Request*/@*BusinessRefID* references the *StockLevelRequest*.

References: *StockLevelRequest*

Flow: *Print Provider* to *Print Buyer*

Table 3.18: StockLevelResponse Element

NAME	DATA TYPE	DESCRIPTION
MasterContract ?	element	*MasterContract* SHALL specify a *Master Contract* that has been applied to calculate the price in the returned *StockLevel* elements.
StockLevel *	element	Level of available stock.

3.15.1 StockLevel

Table 3.19: StockLevel Element (Sheet 1 of 2)

NAME	DATA TYPE	DESCRIPTION
Amount	integer	Number of items that are currently available or will be available within the time defined by @*ProductionDuration*.
Availability	enumeration	@*Availability* SHALL specify the availability of the item. **Allowed values are from:** ▸ Table 3.17 Availability Attribute Values.
DescriptiveName	string	Human-readable descriptive name of the item that this *StockLevel* relates to. @*DescriptiveName* SHALL be provided to allow the item to be successfully identified.
ExternalID	NMTOKEN	The *xjdf:Product*/@*ExternalID* of the respective item.
LotID ?	NMTOKEN	Production lot of this item. Used to separate available stock with different pricing based on production lots. **Note:** See ▸ [XJDF 2.1] and *xjdf:Part*/@*LotID* for a discussion of production lots.
ProductionDuration ?	duration	Estimated time until item will be available. **Note:** @*ProductionDuration* excludes shipping times to the recipient.

Table 3.19: StockLevel Element (Sheet 2 of 2)

NAME	DATA TYPE	DESCRIPTION
Pricing ?	element	List of *Price* definitions for the *Item* specified by this *StockLevel*. Multiple *Price* elements in this *Pricing* element specify the price for orders with an order amount greater than or equal to *Price*/@*Amount*. *Price*/@*DescriptiveName* SHALL be identical for all *Pricing*/*Price* elements in the scope of one *StockLevel*.
xjdf:Contact ?	element	If present, *xjdf:Contact* SHALL specify the location of the item in stock. Both *xjdf:Contact/xjdf:Company* and *xjdf:Contact/xjdf:Address* SHOULD be specified.
xjdf:XJDF	element	Detailed description of the *Print Product* or *Catalog Item* that is described by this *StockLevel*.

4 Subelements

The elements in this chapter are subelements that are used as children of other elements or *Business Objects.*

4.1 Additional

The price for orders of at least the nominal delivery quantity specified in @*BaseAmount*. The price for a given amount, AMT, SHALL be calculated as:

$$\textbf{\textit{Additional}}/@\textit{BasePrice} + \frac{(AMT - \textbf{\textit{Additional}}/@\textit{BaseAmount}) \times \textbf{\textit{Additional}}/@\textit{Price}}{\textbf{\textit{Additional}}/@\textit{Amount}}$$

Note: Some *Print Providers* only allow purchasing of specific amounts rather than ranges, e.g. when printing on gang forms. In this case **Additional** refers to exactly one amount rather than a range of possible amounts.

Referenced by: ***Price***

Table 4.1: Additional Element

NAME	DATA TYPE	DESCRIPTION
Amount ?	double	@*Amount* SHALL specify the additional number of *Items* that @*Price* refers to. If @*Amount* is present, **PurchaseOrder** SHALL specify an amount that can be calculated as @*BaseAmount* + N x @*Amount* where N is a positive integer, else this **Additional** SHALL apply only to the exact amount and price as specified in @*BasePrice* and @*BaseAmount*.
BaseAmount	double	Minimum ordered quantity for which this **Additional** is applicable.
BasePrice	double	Price for the quantity specified in @*BaseAmount*.
Price ?	double	The price of the additional number of *Items* as specified in @*Amount*. @*Price* SHALL be specified if @*Amount* is present.

SUBELEMENTS

4.2 Credential

A *Credential* identifies the various parties involved in a transaction.

Referenced by: *From*, *Sender*, *To*

Table 4.2: Credential Element

NAME	DATA TYPE	DESCRIPTION
domain	NMTOKEN	From cXML: Specifies the type of credential. This attribute allows documents to contain multiple types of credentials for multiple authentication domains. The value of @*domain* SHALL be unique for all sibling *Credential* elements, i.e. within the context of the parent *From*, *To* or *Sender* element. When using a proprietary value for @*domain* it SHOULD use a form similar to 'prefix:ProprietaryValue', i.e. declare a namespace and use the namespace prefix as an indication for the source of the value. **Values include:** CompanyID – Company identifier as specified in *xjdf:Company*/@*CompanyID*. CustomerID – Customer identifier in the print provider system. If an **XJDF** is provided in the *Business Object*, the value of *Identity* SHALL match a *xjdf:CustomerInfo*/@*CustomerID* in the referenced **XJDF**. DescriptiveName – Human-readable description of the person or organization of the parent *From*, *To* or *Sender* element. DUNS – The D-U-N-S number, see ▸[DUNS]. Email – The email address as a URL according to ▸[RFC6068]. ResponseURL – The URL to which asynchronous responses to the business object in this *Request* SHALL be sent. A *Credential*[@*domain*="ResponseURL"] SHALL NOT be provided more than once in one *Header* and SHALL only be provided in the context of either *From* or *Sender*. If *Credential*[@*domain*="ResponseURL"] is not present it defaults to the explicit or implied value of *Sender*/ *Credential*[@*domain*="URL"]. ShopID – Identifier of the web shop if the *Business Object* has been submitted in a web-to-print environment. If an **XJDF** is provided in the *Business Object*, the value of *Identity* SHALL match a *xjdf:XJDF*/*xjdf:GeneralID*[@*IDUsage*="ShopID"]/@*IDValue* in the referenced **XJDF**. URL – The URL of the receiving web server (*To*) or sending web server (*From*, *Sender*). UserID – Identifier of the individual user in the *Print Buyer* system. If an **XJDF** is provided in the *Business Object*, the value of *Identity* SHALL match a *xjdf:Contact*/@*UserID* in the referenced **XJDF**. **Note:** See ▸Section 2.4.3 Attribute values for extending this list.
Identity	element	The text in *Identity* states who the *Credential* represents. For example, the text MAY be an email address or some other unique string of characters. See ▸[cXML 1.2].

4.2.1 Identity

The *Identity* element is used to provide the value of the parent *Credential*.

Table 4.3: Identity Element

NAME	DATA TYPE	DESCRIPTION
	text	The text body SHALL contain the value of the parent *Credential*. The body of *Credential* SHOULD be specified as a text string with no leading or trailing whitespace and with no control characters such as LF, CR, etc. **Note:** *Identity* is defined as an element with body text for compatibility with cXML.

4.3 MasterContract

MasterContract SHALL specify the details of a pre-negotiated *Master Contract* that the *Business Object* applies to.

Referenced by: *Invoice*, *PurchaseOrder*, *Quotation*, *RFQ*, *StockLevelRequest*, *StockLevelResponse*

Table 4.4: MasterContract Element

NAME	DATA TYPE	DESCRIPTION
ContractID	string	Identifier of the *Master Contract*. The value of @*ContractID* SHALL always remain constant for the same *Master Contract*.
DescriptiveName ?	string	Human readable summary description of the *Master Contract*.

4.4 Payment

Payment contains additional details about a payment.

Referenced by: *Pricing*

Table 4.5: Payment Element (Sheet 1 of 2)

NAME	DATA TYPE	DESCRIPTION
Authorization ?	string	Authorization code for this *Business Object*.
AuthorizationExpires ?	dateTime	Expiration date of the @*Authorization*.
Expires ?	dateTime	@*Expires* contains a future date regarding the *Payment*. For payment types that require an expiry date as part of the payment validation, e.g. "CreditCard", "DebitCard", @*Expires* SHALL contain the expiry date. If the expiration date for the *Payment* requires less detail than that given by dateTime, then the writer SHALL set @*Expires* to a valid dateTime value and the reader SHALL use only those details of @*Expires* that are required to construct a valid expiration for the payment type of the *Payment*. In either case the value SHALL NOT be rounded. For payment types that involve a transfer, e.g. "BankTransfer", "DigitalCurrency", @*Expires* MAY be used to define when the sending system will cease attempting to make the payment should it be rejected for any reason. For ongoing payment types, e.g. "Contract", @*Expires* MAY be used to indicate when the ongoing arrangement expires. For other payment types the use of @*Expires* is implementation dependent.
IsPaid ?	boolean	If @*IsPaid* is "true", the payment transaction has been finalized and no further payment is due. If @*IsPaid* is "false", then there remain outstanding payments due. @*IsPaid* SHALL NOT be provided in the context of a quotation.
PaymentType	enumeration	If @*IsPaid* is "true", @*PaymentType* SHALL specify the method of payment. If @*IsPaid* is "false", @*PaymentType* SHALL specify an acceptable payment type. Additional details SHOULD be provided in @*PaymentTypeDetails*. **Allowed values are:** BankTransfer – A bank transaction such as a SEPA payment. Contract – A general negotiated contract between *Print Buyer* and *Print Provider*. The contract SHOULD be identified in a *MasterContract* in the parent *Business Object*. CreditCard – A credit card. The credit card type SHOULD be identified in @*PaymentTypeDetails*. DebitCard – A debit card. The debit card type SHOULD be identified in @*PaymentTypeDetails*. DigitalCurrency – A digital currency. The currency type SHOULD be identified in @*PaymentTypeDetails*. Invoice – An invoice that is provided with or after delivery. Invoice SHALL NOT be specified in the context of an *Invoice*. Other – Any payment type that cannot be described by one of the other values in this list. If @*PaymentType*="Other", @*PaymentTypeDetails* SHALL be specified. PaymentProvider – An Internet payment provider. The payment provider brand SHOULD be identified in @*PaymentTypeDetails*.

Table 4.5: Payment Element (Sheet 2 of 2)

NAME	DATA TYPE	DESCRIPTION
PaymentTypeDetails ?	string	Additional details of the payment type such as provider brand.
xjdf:Comment [@*Type*= "PaymentTerms"] ?	element	Human readable text that SHALL specify the terms and conditions of the payment.
xjdf:Contact ?	element	Contact details of the person such as the credit card holder or bank account holder that this *Payment* refers to.
xjdf:GeneralID *	element	Additional identifiers related to the *Payment*.

4.4.1 xjdf:GeneralID – Payment

This section specifies particular values for *xjdf:GeneralID*/@*IDUsage* in the context of *Payment*.

Referenced by: *Payment*

Table 4.6: xjdf:GeneralID Element (For use with Payment)

NAME	DATA TYPE	DESCRIPTION
IDUsage	NMTOKEN	**Values include:** Bank – The name of the bank. BankAccount – The local bank account number. BIC – The SWIFT bank identifier code. CardNumber – The identifying number from a credit card, debit card etc. IBAN – The international bank account number. **Note:** See ▸ Section 2.4.3 Attribute values for extending this list.
IDValue	string	@*IDValue* SHOULD be machine readable. See ▸ [XJDF 2.1].

4.5 Price

The *Price* element specifies the price of an *Item* which inclues the price, a description of the *Item* and additional metadata.

Referenced by: *Pricing*

Table 4.7: Price Element (Sheet 1 of 2)

NAME	DATA TYPE	DESCRIPTION
Amount ?	double	Amount of items that this *Price* refers to. If @*UnitPrice* is specified, @*Amount* SHALL be specified.
DescriptiveName	string	The description of the *Item*.
DropID ?	NMTOKEN	@*DropID* SHALL reference the delivery that this *Price* applies to and SHALL match *xjdf:ResourceSet*[@*Name*="DeliveryParams"]/*xjdf:Resource*/*xjdf:Part*/ @*DropID* from the **XJDF**.
ItemRefs ?	NMTOKENS	Each value in @*ItemRefs* SHALL refer to an *xjdf:Product*/@*ExternalID* within an associated *xjdf:XJDF* of the *Business Object* that this *Price* relates to.
LineID ?	ID	The unique identifier for the *Item* that this *Price* represents. If this *Price* represents a product or product part, that is defined in an *xjdf:Product* that is itself in the scope of the enclosing business object, *xjdf:Product*/ *xjdf:GeneralID*[@*IDUsage*="LineID"]/@*IDValue* SHOULD be set to the value of @*LineID*.

Table 4.7: Price Element (Sheet 2 of 2)

NAME	DATA TYPE	DESCRIPTION
LineIDRefs ?	IDREFS	References to *Price*/@*LineID* of the *Items* that have been added to calculate @*Price*. *Price* elements that have been referenced by @*LineIDRefs* SHALL be ignored when searching for the total price. **Note:** Due to differing tax regulations for private and business transactions, there can be multiple total values. **Note:** Due to rounding, @*Price* may vary from the sum of @*Price* values of each of the referenced *Price* elements in @*LineIDRefs*.
Price	double	@*Price* SHALL specify the price of the entire amount as specified by @*Amount* of the *Item*.
PriceType	enumeration	@*PriceType* SHALL specify the machine readable type of the price. **Allowed values are:** Discount – Discount prices SHALL be negative. DownPayment – The amount that SHALL be paid up front. Production will not commence before this amount is received by the *Print Provider.* Handling – This price relates to handling charges. If shipping and handling are combined in one *Item* the value "Shipping" SHOULD be used. Markup – This refers to additional charges that have been incurred. Markup prices SHALL be positive. Other – Any price type not covered by other values in this list. Prepaid – The amount has already been paid and SHALL be deducted from the total. Prepaid prices SHALL be negative. Product – The requested products. If @*PriceType* = "Product", then @ItemRefs SHOULD reference the products that this *Price* applies to. Shipping – This price relates to shipping charges. Subtotal – This refers to a partial sum of other prices. The value is for information only and SHOULD NOT be used to calculate the total price. Total – Total.
TaxType	enumeration	@*TaxType* SHALL specify the taxation of the *Price*. **Allowed values are:** Gross – The price includes taxes. Net – The price excludes taxes. Tax – The price is only the tax. **Note:** The "Gross" price MAY differ from the sum of "Net" price and "Tax" due to rounding.
UnitPrice ?	double	Price for one unit of the *Item*, i.e. the @*Price* of the *Item* if @*Amount* = "1".
Additional *	element	Price for orders of at least the nominal delivery quantity specified in *Additional*/@*BaseAmount*. *Additional* SHALL NOT be specified in an *Invoice* or *PurchaseOrder*.

4.6 Pricing

Pricing SHALL specify a list of *Price* elements, each of which in turn specifies the price for a single *Item*.
Note: The usage of *Pricing* depends upon its parent *Business Object* and is clarified for each occurrence.
Referenced by: *Invoice*, *PurchaseOrder*, *Quote*, *StockLevel*

Table 4.8: Pricing Element (Sheet 1 of 2)

NAME	DATA TYPE	DESCRIPTION
Currency	Currency	@*Currency* SHALL identify the currency of all *Price*/@*Price* values. **Note:** See ▸ Section 1.3.1 Currency for definition of valid currency values, and refer to ▸ [ISO4217].

Table 4.8: Pricing Element (Sheet 2 of 2)

NAME	DATA TYPE	DESCRIPTION
Payment *	element	Details of the method of payment. Multiple *Payment* elements SHALL specify multiple payment options. At most one *Payment* element SHALL be specified in the context of a purchase order. If *Payment*/@*IsPaid*="true", exactly one *Payment* element SHALL be specified. *Payment* SHALL be unique and specify the selected methods of payment in the context of a *PurchaseOrder* and SHALL specify the list of allowed methods of payments in all other contexts.
Price +	element	Each *Price* element represents the price for a single *Item* or service.
xjdf:Contact ?	element	Contact details of the person that receives the invoice. **Note:** *Payment*/*xjdf:Contact* specifies the details of an account or card holder, which can be different than the formal recipient of an invoice.

Example 4.1: Pricing

```
<Pricing Currency="CAD">
  <Price DescriptiveName="Item #1" ItemRefs="product1"
    LineID="L_1" Price="42" PriceType="Product" TaxType="Gross"/>
  <Price DescriptiveName="Item #2" ItemRefs="product2"
    LineID="L_2" Price="21" PriceType="Product" TaxType="Gross"/>
  <Price DescriptiveName="Total" LineIDRefs="L_1 L_2" Price="63"
    PriceType="Total" TaxType="Gross"/>
  <Price DescriptiveName="Included tax" LineIDRefs="L_1 L_2"
    Price="12.6" PriceType="Total" TaxType="Tax"/>
</Pricing>
```

4.7 StatusRequest

Referenced by: *OrderStatusRequest*, *PurchaseOrder*

Table 4.9: StatusRequest Element

NAME	DATA TYPE	DESCRIPTION
JobIDRef ?	NMTOKEN	@*JobIDRef* SHALL reference an *xjdf:XJDF*/@*JobID* that was present in the parent *PurchaseOrder*, or the *PurchaseOrder* that is referenced by *Request*/@*BusinessRefID* when the parent *Request* is an *OrderStatusRequest*. If not specified, this *StatusRequest* refers to all jobs specified in the corresponding *PurchaseOrder*.
MilestoneAmount ?	enumeration	If @*MilestoneAmount* is present, *OrderStatusResponse* SHOULD contain *xjdf*:*AuditPool*/*xjdf*:*AuditNotification*/*xjdf*:*Notification*/*xjdf*:*Milestone* elements. Exactly one of @*MilestoneAmount* and @*ResourceAudit* SHALL be present. **Allowed values are:** Brief – Only the latest *xjdf:Milestone* of the current job status SHALL be returned. Completed – All completed *xjdf:Milestone* elements of the current job and the *xjdf:Milestone* elements of steps that have begun processing SHALL be returned. "Completed" SHALL NOT be specified if *OrderStatusRequest*/@*Subscribed*="true" or if this *StatusRequest* is a child of *PurchaseOrder*.
ResourceAudit ?	NMTOKEN	@*ResourceAudit* SHALL contain the requested value of @*Name* of the requested *OrderStatusResponse*/*xjdf*:*AuditPool*/*xjdf*:*AuditResource*/*xjdf*:*ResourceInfo*/*xjdf*:*ResourceSet*. If @*ResourceAudit* is present, *OrderStatusResponse* SHOULD contain an *xjdf*:*ResourceSet* element. Exactly one of @*MilestoneAmount* and @*ResourceAudit* SHALL be present. Typical values include *DeliveryParams* for delivery tracking information or *QualityControlResult* for quality reporting.

4.8 UserAgent

See ▸ [cXML 1.2].

UserAgent SHOULD contain a combination of agent name and agent version of the software agent. See the definition of *xjdf:Header*/@*AgentName* and *xjdf:Header*/@*AgentVersion* for details.

Referenced by: *From*, *Sender*

Table 4.10: UserAgent Element

NAME	DATA TYPE	DESCRIPTION
	text	The text body SHALL contain the value of the user agent. The body of *UserAgent* SHOULD be specified as a text string with no leading or trailing whitespace and with no control characters such as LF, CR, etc. **Note:** *UserAgent* is defined as an element with body text for compatibility with cXML.

5 Building a System

A system that uses **PrintTalk** will typically be deployed over the internet and therefore security and authentication need to be addressed. Since **PrintTalk** can contain sensitive data, it is strongly encouraged to use encryption technology whenever possible.

5.1 Bidirectional Transport Protocol

PrintTalk SHALL be implemented using the request-response model by sending an HTTP or HTTPS POST request that contains the **PrintTalk** XML or packaged **PrintTalk** ZIP. See ▸ Section 5.5 PrintTalk Packaging.

If no error occurred then the synchronous protocol response SHALL be an empty HTTP or HTTPS response with an HTTP status code of 200. If an error occurred then the HTTP status code SHALL be present and SHALL NOT be 200, and the HTTP body MAY be non-empty, e.g. with a human readable text or html representation of the error.

5.2 Unidirectional Transport Protocol

In order to enable an implementation that does not require an http server on the side of the *Print Buyer*, the *Print Provider* or neither, **PrintTalk** MAY be implemented by polling a URL with an HTTP or HTTPS GET request with additional HTTP header fields as defined in ▸ Table 5.1 Additional HTTP header fields encoded in the HTTP Header. Some of these additional header fields are required for this type of **PrintTalk** implementation as shown in the 'Use' column.

If data is available then the synchronous protocol response SHALL be a response with an HTTP status code of 200 that contains the **PrintTalk** XML or packaged **PrintTalk** ZIP. See ▸ Section 5.5 PrintTalk Packaging. If no data is available then the synchronous protocol response SHALL be an empty response with an HTTP status code of 204.

If an error occurred then the HTTP status code SHALL be present and SHALL NOT be in the range of 200-299, and the HTTP body MAY be non-empty, e.g. with a human readable text or html representation of the error.

Table 5.1: Additional HTTP header fields

HEADER FIELD NAME	USE	DESCRIPTION
printtalk-from-url	Optional	The value of *PrintTalk*/*Header*/*From*/*Credential*[@*domain*="URL"] of the requested **PrintTalk** element. If not specified, the value is the same as the request URL.
printtalk-last-payload	Optional	The value of *PrintTalk*/@*payloadID* of the last successfully processed **PrintTalk** element that matches printtalk-from-url and printtalk-to-url.
printtalk-to-url	Required	The value of *PrintTalk*/*Header*/*To*/*Credential*[@*domain*="URL"] of the requested **PrintTalk** element.

5.3 Encryption

All **PrintTalk** transactions SHOULD be encrypted using a secure transport protocol. The details and setup of encryption, including the exchange of certificates, are implementation dependent.

5.4 Authentication

Whereas encryption allows for secure communication in the internet by hiding the contents of transactions from third parties, authentication is used to verify that the sender of a message is actually the entity that it claims to be. Authentification is therefore extremely important when fulfilling contracts in order to prevent fraud and misuse of the interface.

It is strongly encouraged to only allow access to authenticated clients. If authentication is implemented, it SHOULD follow the directions shown in ▸ [RFC6750]. All implementations SHOULD support at least section "2.1 Authorization Request Header Field" from ▸ [RFC6750].

▸ [RFC6750] provides methods to verify the identity of a message sender. **PrintTalk** defines no proprietary authentication methods but it is emphasized that authentication is important in all business transactions.

The initial exchange of the token and renewal, as well as the format of the token is not part of this specification and is implemention dependent.

5.5 PrintTalk Packaging

Additional assets such as PDF files or images MAY be provided in a **PrintTalk** transaction. These assets can be referenced in the web as URLs or can be provided in a single ZIP package that also contains the **PrintTalk**. In this case the ZIP package consists of a single **PrintTalk** transaction and the digital assets to which the transaction refers.

5.5.1 ZIP Packaging

Zip is a de facto industry standard for packaging and compressing data. Directory structures can be encoded in a zip package. For details see ▶[ZIP].

5.5.1.1 Identifying the PrintTalk transaction

The **PrintTalk** transaction SHALL be named PrintTalk.ptk and SHALL reside in the root directory of the zip package.

5.5.1.2 Referencing Digital Assets within a ZIP Package

Referenced digital assets that reside in the zip package, for instance those that are referenced with @*URL*, SHALL be referenced as local URLs. The base URL for calculating local URLs SHALL be the root of the zip package.

Digital assets other than the PrintTalk MAY be placed in a directory tree structure within the zip file.

5.5.1.3 ZIP File Name Encoding

All file and directory names in a zip package SHALL be encoded in UTF-8.
Note: Zip allows any encoding but provides no method to declare the encoding.

5.6 MIME Types and File Extensions

The following MIME types and extensions SHOULD be used when storing **PrintTalk** as a file or when a MIME type is required, e.g. when setting the http Content-Type header.

Table 5.2: MIME Types and File Extensions

MIME TYPE	EXTENSION	USAGE
application/ vnd.cip4-ptk+xml	ptk	Unpackaged **PrintTalk**.
application/ vnd.cip4-ptk+zip	ptk.zip	Zip packaged **PrintTalk**. A double extension of .ptk.zip SHOULD be used.

6 Business Object Workflows

6.1 Lifetime of Business Objects

Some *Business Objects* such as *Quotation* or *PurchaseOrder* may only be valid for a given period of time. These *Business Objects* have an *@Expires* attribute, that defines the date and time when a proposed contract is no longer binding for the sending party unless it is either confirmed or refused. A *Business Object* that has neither expired nor been confirmed is a *Pending Business Object.* A *Pending Business Object* can be confirmed by sending the appropriate *Business Object* as defined in ▶ Table 6.1 Business Object References that references the *Pending Business Object.* A *Pending Business Object* can be explicitly refused by sending a *Refusal* that references the *Pending Business Object.*

6.2 Identifiers in PrintTalk

This section summarizes the usage of the various IDs that are defined in **PrintTalk** in typical workflow scenarios.

6.2.1 BusinessID

Every *Business Object* SHALL be uniquely identified by specifying *Request*/@*BusinessID* in the parent *Request*. The values of multiple *Request*/@*BusinessID*'s SHALL ONLY be identical if the values of their respective parent *PrintTalk*/@*payloadID*'s are identical.

6.2.2 BusinessRefID

Request/@*BusinessRefID* references a *Business Object* that identifies the *PrintTalk Project*. See ▶ Table 6.1 Business Object References for a summary of the *Request*/@*BusinessID* values that SHALL be referenced by *Request*/@*BusinessRefID* of the various business objects.

6.2.3 ContractID

A *Print Buyer* or *Print Provider* MAY use *MasterContract*/@*ContractID* to reference a pre-negotiated contract that SHOULD be used to define pre-negotiated terms and conditions such as discounts.

6.2.3.1 ContractID provided by the Print Buyer

A *Print Buyer* should provide its preferred *Master Contract* ID as early in the *PrintTalk Project* as possible. This will typically be in an *RFQ* or an initial *PurchaseOrder*.

6.2.3.2 ContractID provided by the Print Provider

If terms from a *Master Contract* were applied, the *Print Provider* SHALL provide the ID of the *Master Contract* used to calculate a *Quotation* or *Invoice*.

6.2.4 CustomerProjectID

The *Request*/@*CustomerProjectID* is the ID of the entire order, which contains one or more jobs or individual products in the context of the *Print Buyer's* MIS or ERP system. Once specified, *Request*/@*CustomerProjectID* SHALL remain constant for all *Business Objects* of a *PrintTalk Project*. If present, *xjdf*:*CustomerInfo*/@*CustomerProjectID* SHALL match *Request*/@*CustomerProjectID*.

6.2.5 DropID

Price/@*DropID* is used to identify individual deliveries of a split delivery. This can be either a different delivery address, a different delivery time and date, or both.

6.2.6 ExternalID

StockLevel/@*ExternalID* is a generic identifier for products in a shop system. @*ExternalID* allows standard products to be selected by identifier rather than complete descriptions.
Note: @*ExternalID* values will have to be synchronized using proprietary synchronization which is not covered by the **PrintTalk** specification.

6.2.7 LineID

Price/@*LineID* is used to identify the line *Item.*

6.2.8 LotID

StockLevel/@*LotID* is an identifier that differentiates products that were produced in different production runs.

6.2.9 payloadID

Every *PrintTalk* element SHOULD be uniquely identified by specifying *PrintTalk*/@*payloadID*.

6.2.10 QuoteID

@*QuoteID* is used to identify (*Quote*/@*QuoteID)* and select (*PurchaseOrder*/@*QuoteID*) individual *Quotes* within a *Quotation Business Object*.

6.2.11 ReorderID

@*ReorderID* can be used by the *Print Buyer* in an *RFQ* or *PurchaseOrder* to provide the business identifiers (i.e. the values of *Request*/@*BusinessID*) of previous completed *PurchaseOrders*.

6.3 XJDF Identifiers used by PrintTalk

This section summarizes the usage of IDs that are defined in **XJDF** and used in **PrintTalk** workflows.

6.3.1 CustomerOrderID

The *xjdf:CustomerInfo*/@*CustomerOrderID* is the ID of the job or individual product in the context of the *Print Buyer*'s MIS or ERP system.

6.3.2 GeneralID

xjdf:GeneralID provides a generic mechanism to define identifiers. In **PrintTalk** it is used to define *Payment* specifics, which can vary greatly from country to country. It is also used to link partial products in **XJDF** to price lines.

6.3.3 JobID

The *xjdf:XJDF*/@*JobID* is the ID of the job within the *Print Provider*'s workflow system.

When an **XJDF** is created by a *Print Buyer*, e.g. in an *RFQ*, then the *xjdf:XJDF*/@*JobID* SHOULD be specified. If a *Print Provider* cannot maintain an external @*JobID*, the @*JobID* SHOULD be copied to *xjdf:CustomerInfo*/@*CustomerOrderID*.

All interactions between the *Print Buyer* and *Print Provider* SHALL be based on the initial @*JobID* as specified by the *Print Buyer*.
Note: This implies that the *xjdf:XJDF*/@*JobID* within a production job of the *Print Provider*'s workflow system NEED NOT be identical to *xjdf:XJDF*/@*JobID* when communicating with the *Print Buyer*.

6.4 Interrelation of Business Objects

PrintTalk supports a variety of procurement, negotiation, change order, proofing and content delivery workflows. Not every **PrintTalk** aware application is required to support all workflows. CIP4 is developing interoperability conformance specifications that define the requirements for these individual workflows.

The following table summarizes the interactions of the various business objects.

Table 6.1: Business Object References

BUSINESS OBJECT	REFERENCED BUSINESS OBJECT	NOTES
Cancellation	*PurchaseOrder* (Current)	Cancellation of a *PurchaseOrder* can be sent by either *Print Provider* or *Print Buyer*.
Cancellation	*Quotation*	Cancellation of a *Quotation* can only be sent by the *Print Provider*.
Confirmation	*Cancellation*	The *Print Provider* or *Print Buyer* acknowledges that a *PurchaseOrder* has been cancelled.
Confirmation	*PurchaseOrder* (Current)	Confirmation of any *PurchaseOrder* including change orders.
ContentDelivery	*PurchaseOrder* (Current)	Delayed delivery of content after a confirmed *PurchaseOrder*.
ContentDelivery	*RFQ* (Current)	Delivery of content prior to a *PurchaseOrder*.
ContentDeliveryResponse	*ContentDelivery*	*ContentDeliveryResponse* in the case where content is delivered in a separate *ContentDelivery*.

Table 6.1: Business Object References

| BUSINESS OBJECT	REFERENCED BUSINESS OBJECT	NOTES
ContentDeliveryResponse	*PurchaseOrder* (Current)	*ContentDeliveryResponse* in the case where content is delivered in a *PurchaseOrder*.
Invoice	*PurchaseOrder* (Current)	*Invoice* always references the last confirmed *PurchaseOrder*, regardless of any *Cancellation*.
OrderStatusRequest	*PurchaseOrder* (Current)	*OrderStatusRequest* always references the last confirmed *PurchaseOrder*.
OrderStatusResponse	*OrderStatusRequest*	An *OrderStatusResponse* that is generated from an explicit *OrderStatusRequest* SHALL reference that *OrderStatusRequest*.
OrderStatusResponse	*PurchaseOrder* (Current)	An *OrderStatusResponse* that is generated from *StatusRequest* elements that were provided in the *PurchaseOrder* SHALL reference that *PurchaseOrder*.
ProofApprovalRequest	*PurchaseOrder* (Current)	*ProofApprovalRequest* always references the last confirmed *PurchaseOrder*.
ProofApprovalResponse	*ProofApprovalRequest*	*ProofApprovalResponse* always references the *ProofApprovalRequest*.
PurchaseOrder	–	The initial *PurchaseOrder*. **Note:** An initial *PurchaseOrder* implies known prices from a grid or dynamic website.
PurchaseOrder	*PurchaseOrder* (Previous)	*Change Order PurchaseOrder* initiated by *Print Buyer*. If the *Change Order* is refused, the original *PurchaseOrder* is still in place.
PurchaseOrder	*Quotation*	*PurchaseOrder* that is based on an *RFQ* / *Quote*.
PurchaseOrder	*Quotation*	*Change Order PurchaseOrder* response to a *Print Provider* initiated *Quotation*.
Quotation	*PurchaseOrder* (Current)	*Print Provider* based *Change Order Quotation* e.g. due to content mismatch.
Quotation	*RFQ*	Initial *Quotation* based on an *RFQ*.
Refusal	*Cancellation*	*Print Provider* refused cancellation of a *PurchaseOrder*. A *Print Buyer* cannot refuse a *Print Provider* initiated cancellation.
Refusal	*PurchaseOrder* (Current)	Refusal of any *PurchaseOrder* including *Change Orders*.
Refusal	*Quotation*	*Print Buyer* refuses an initial *Quotation* or *Print Provider* suggested modification of a *PurchaseOrder*.
Refusal	*RFQ*	*Print Provider* refuses an initial *RFQ* or *Print Buyer* requested modification of a *PurchaseOrder*.
RFQ	–	An initial *RFQ* does not reference any other *Business Object*.
RFQ	*PurchaseOrder* (Current)	*Change Order RFQ* initiated by the *Print Buyer*. If the *Change Order* is refused, the original *PurchaseOrder* is still in place.
RFQ	*Quotation*	Modified *RFQ* based on an existing *Quotation*.
StockLevelRequest	–	A *StockLevelRequest* is always an initial *Business Object*.
StockLevelResponse	*StockLevelRequest*	*StockLevelResponse* always references the *StockLevelRequest*.

6.5 Workflows

6.5.1 Procurement

The following workflows outline the steps required to negotiate a contract that results in a *PurchaseOrder*. Additional workflows such as proofing, content delivery or invoicing may also be required.

6.5.1.1 Fixed Price Procurement

If a *Print Buyer* requests standardized products with known prices, e.g. from a web to print site, then no **PrintTalk** based negotiation is required. The *Print Buyer* initiates the procurement with a *PurchaseOrder*. The *Print Provider* SHALL confirm the *PurchaseOrder* by sending a *Confirmation*. If the *Print Provider* does not accept the *PurchaseOrder*, it SHALL send a *Refusal*.

Figure 6-1: Fixed Price Procurement
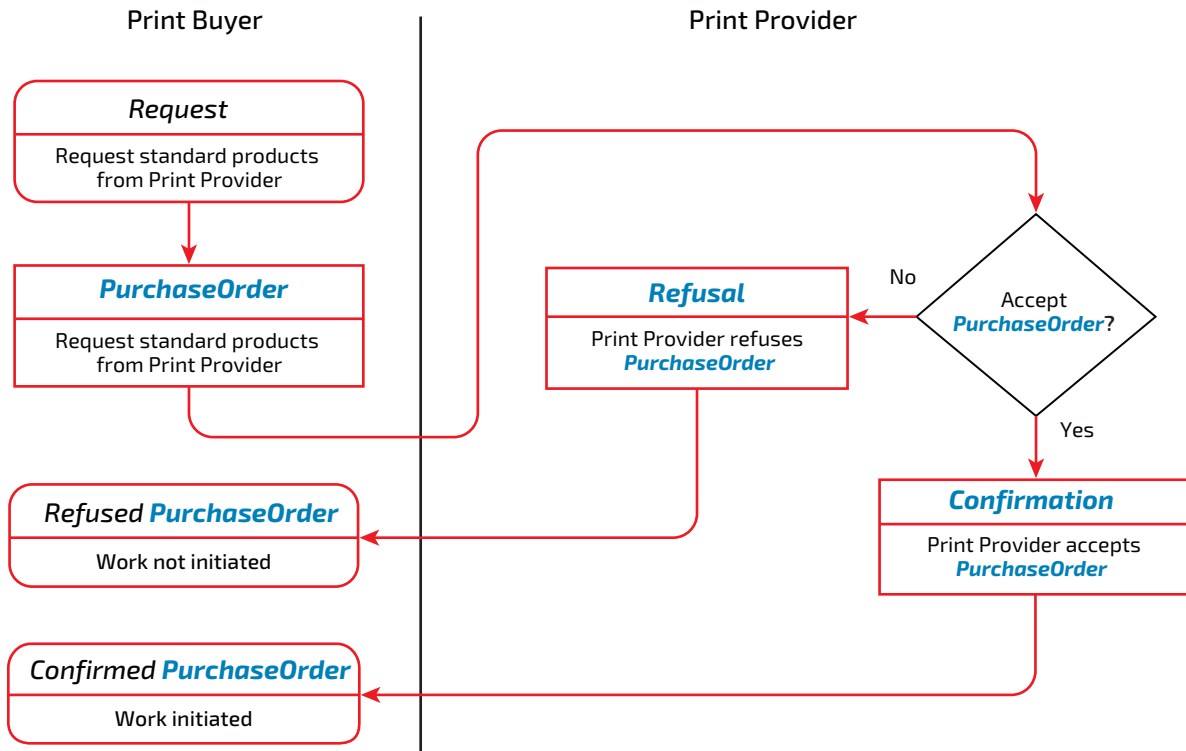
6.5.1.2 Request For Quote Procurement

If a *Print Buyer* requests a customized product, the *Print Buyer* initiates the procurement with an *RFQ* that provides details of the desired product. The *Print Provider* SHOULD respond with a *Quotation* that references the *RFQ*. The *Print Provider* MAY decline the *RFQ* by either sending a *Refusal* or letting the *RFQ* expire. If the *Print Buyer* decides to accept one of the *Quotation*/*Quote* items, it SHALL send a *PurchaseOrder* that references the *Quotation* and *Quote*. For further processing of the *PurchaseOrder* by the *Print Provider*, see ▸ Section 6.5.1.1 Fixed Price Procurement.

The *Print Provider* MAY send a *Cancellation* that references a pending *Quotation*.

Figure 6-2: Request For Quote Procurement
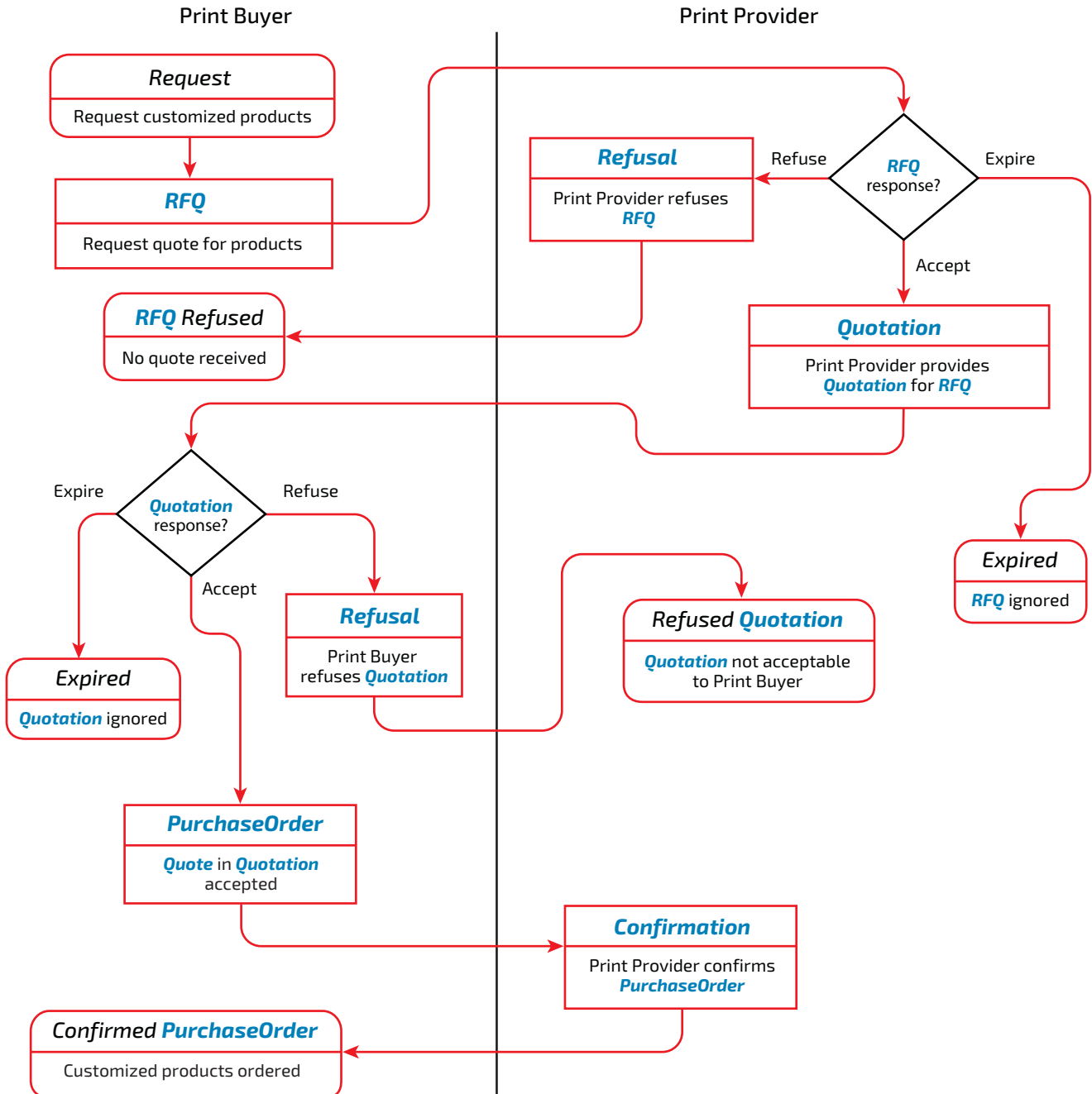
6.5.2 Purchase Order Modification

Some situations arise, where details of a confirmed *Business Object* need to be modified. These modifications can be initiated by either the *Print Buyer* or *Print Provider*.
Note: Some ICSs and workflows explicitly prohibit any modification to a confirmed *PurchaseOrder*.

6.5.2.1 Print Buyer Initiated Purchase Order Modification

A *Print Buyer* MAY decide to modify or cancel a confirmed *PurchaseOrder*.

6.5.2.1.1 Purchase Order Cancellation

A *Print Buyer* can attempt to cancel a *PurchaseOrder* by sending a *Cancellation* that references the *PurchaseOrder*. The *Print Provider* SHALL send either a *Confirmation* or *Refusal* that references the *Cancellation*.

Note: A confirmed *Cancellation* can still require payments such as cancellation fees or even the entire *Invoice*, depending on the terms of the contract.
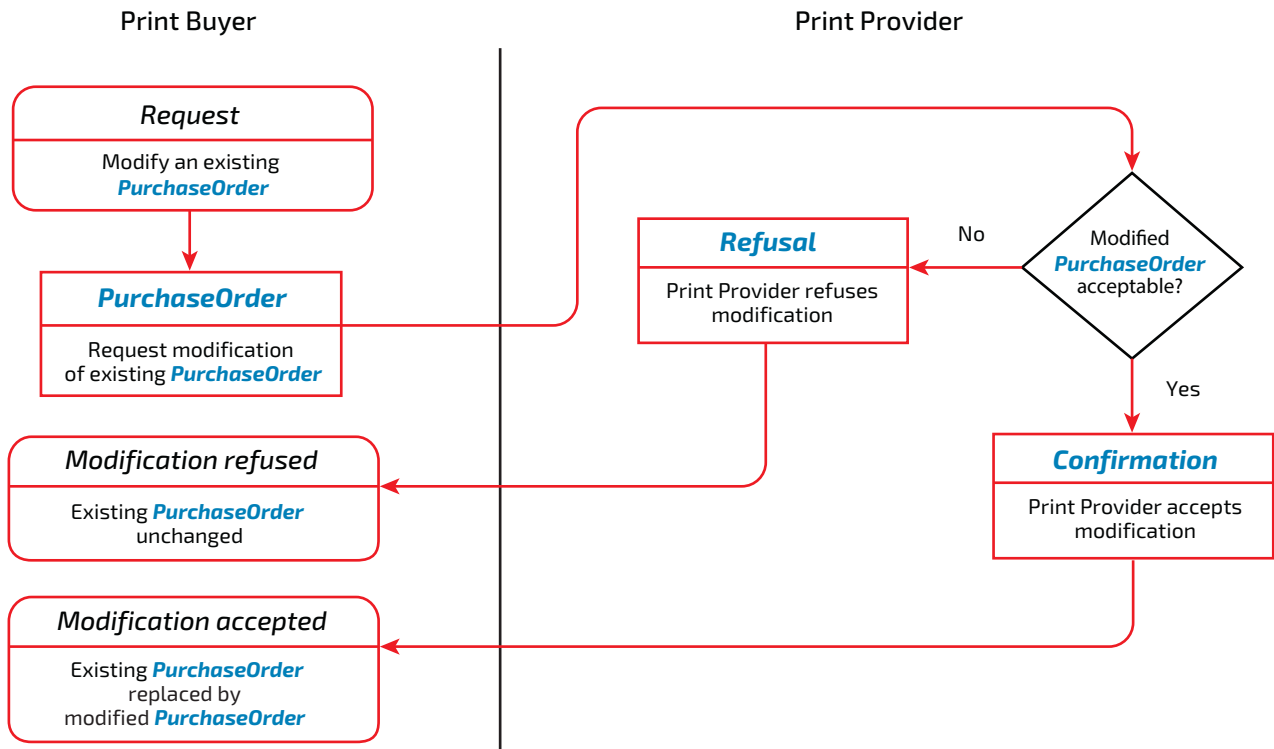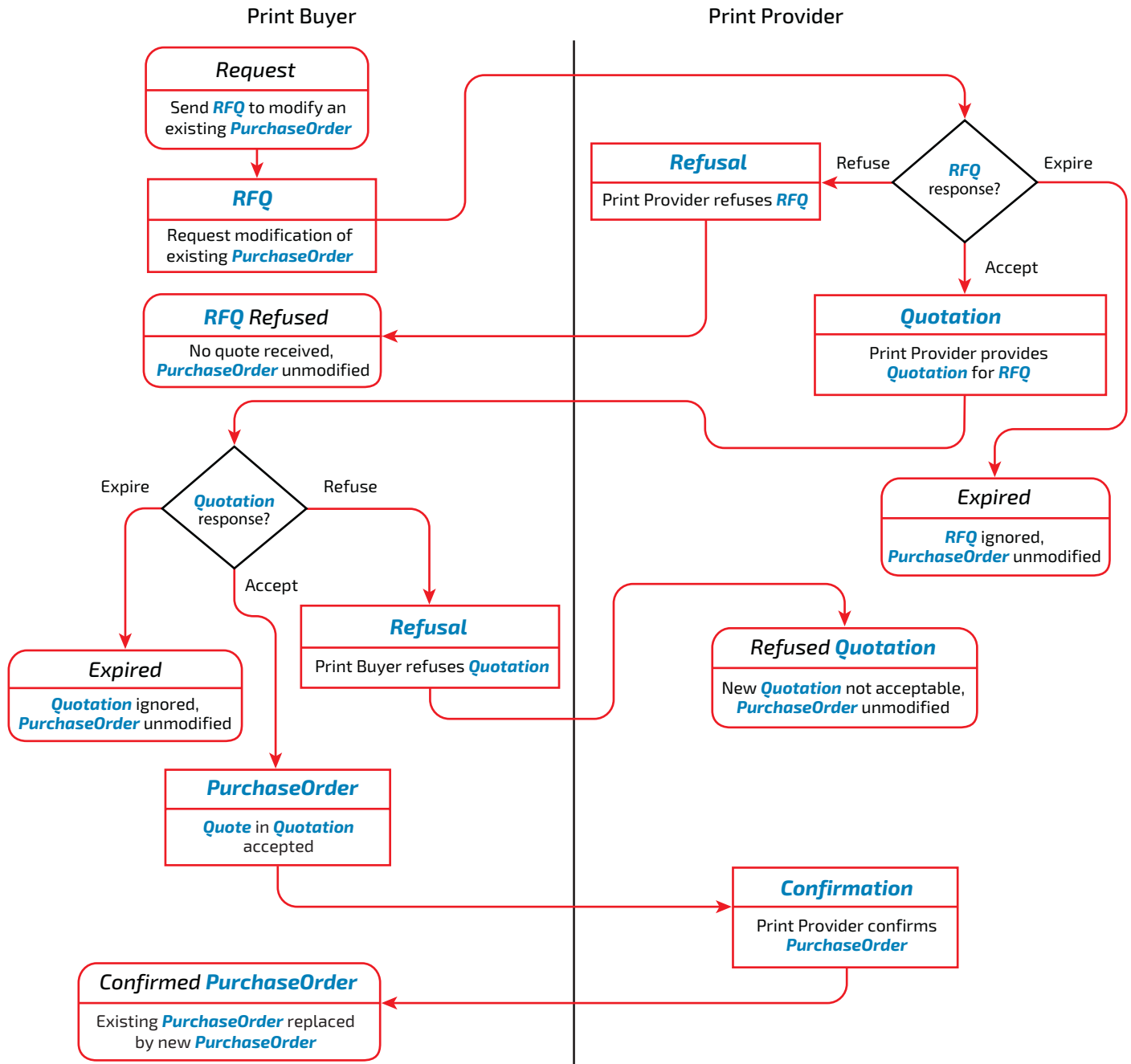
Figure 6-3: Print Buyer Initiated Purchase Order Cancellation
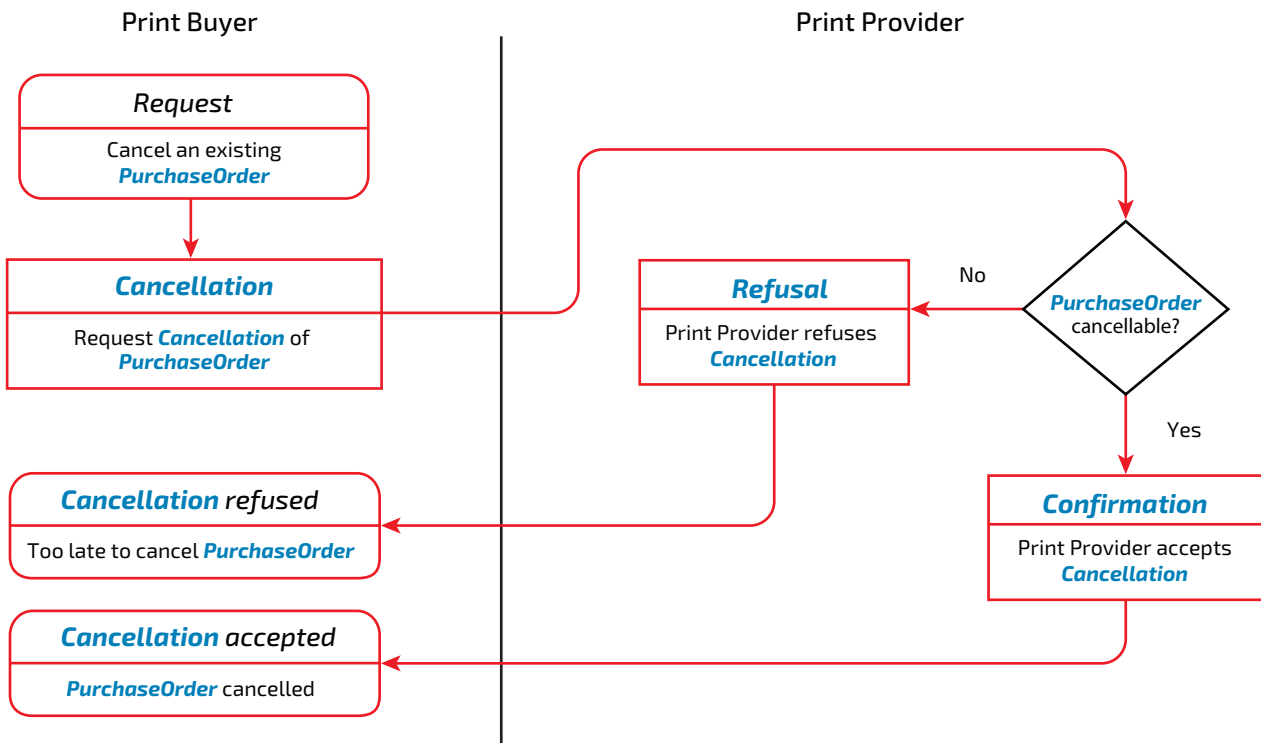
6.5.2.1.2 Purchase Order Modification

A *Print Buyer* can attempt to modify a *PurchaseOrder* by sending a new *PurchaseOrder* that references the current *PurchaseOrder*. The *Print Provider* SHOULD send either a *Confirmation* or *Refusal* that references the new *PurchaseOrder*.

Figure 6-4: Print Buyer Initiated Purchase Order Modification

6.5.2.1.3 Purchase Order Modification with RFQ

A *Print Buyer* can request a quote for modifications to a *PurchaseOrder* by sending a new *RFQ* that references the current *PurchaseOrder*. The *Print Provider* SHOULD send a *Quotation* that references the new *RFQ*. If the *Print Provider* sends a *Quotation*, the following steps are identical to ▶ Section 6.5.1.2 Request For Quote Procurement. If no *PurchaseOrder* is sent by the *Print Buyer*, the original *PurchaseOrder* is still valid.

Figure 6-5: Print Buyer Initiated Purchase Order Modification with RFQ

6.5.2.2 Print Provider Initiated Purchase Order Modifications

A *Print Provider* MAY be forced to modify or cancel a confirmed *PurchaseOrder*. In many cases this will be due to a mismatch of content and the product description that was provided in the *PurchaseOrder*.

6.5.2.2.1 Purchase Order Cancellation

In some rare circumstances, a *Print Provider* may be forced to cancel a confirmed *PurchaseOrder* by sending a *Cancellation* that references the *PurchaseOrder*. The *Print Buyer* MAY send a *Confirmation* that references the *Cancellation*. The *Print Buyer* SHALL NOT send a *Refusal* that references the *Cancellation*.

Figure 6-6: Print Provider Initiated Purchase Order Cancellation
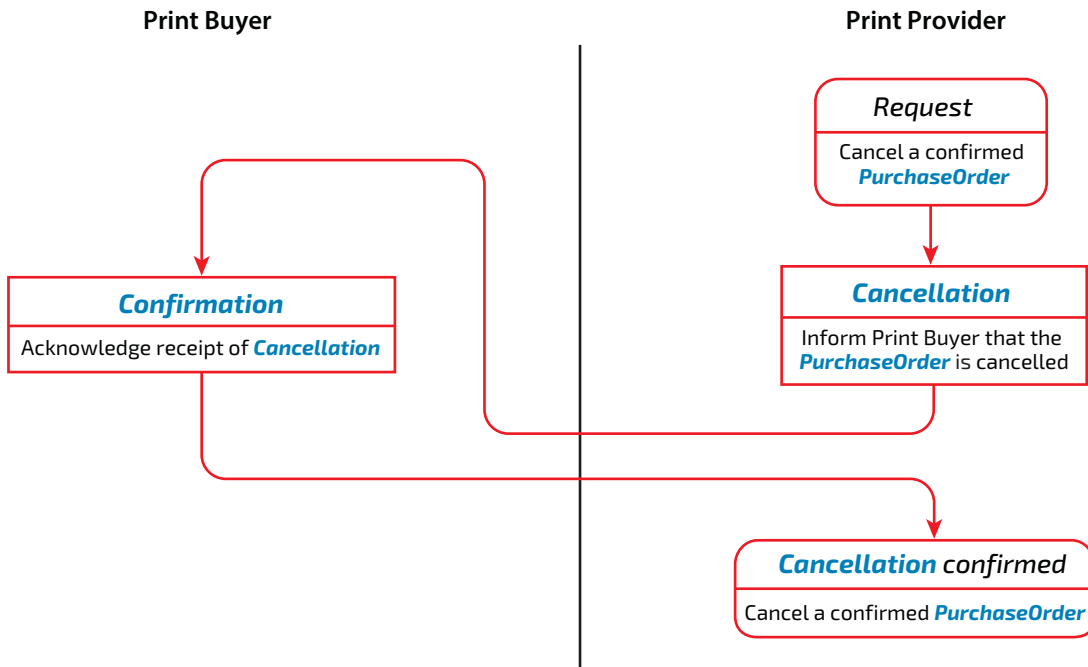
6.5.2.2.2 Quotation Modification

If the content and the details of the *PurchaseOrder* do not match, the *Print Provider* SHOULD send a *Quotation* that match-es the content. The *Print Buyer* SHOULD respond with a *PurchaseOrder* that references the *Quotation* and *PurchaseOrder*/ @*QuoteID* that references one of the *Quotation*/*Quote* items. If the *Print Buyer* sends a *Refusal* or lets the *Quotation* expire, the *Print Provider* MAY send a *Cancellation* that references the the original *PurchaseOrder*.

Figure 6-7: Print Provider Initiated Quotation Modification
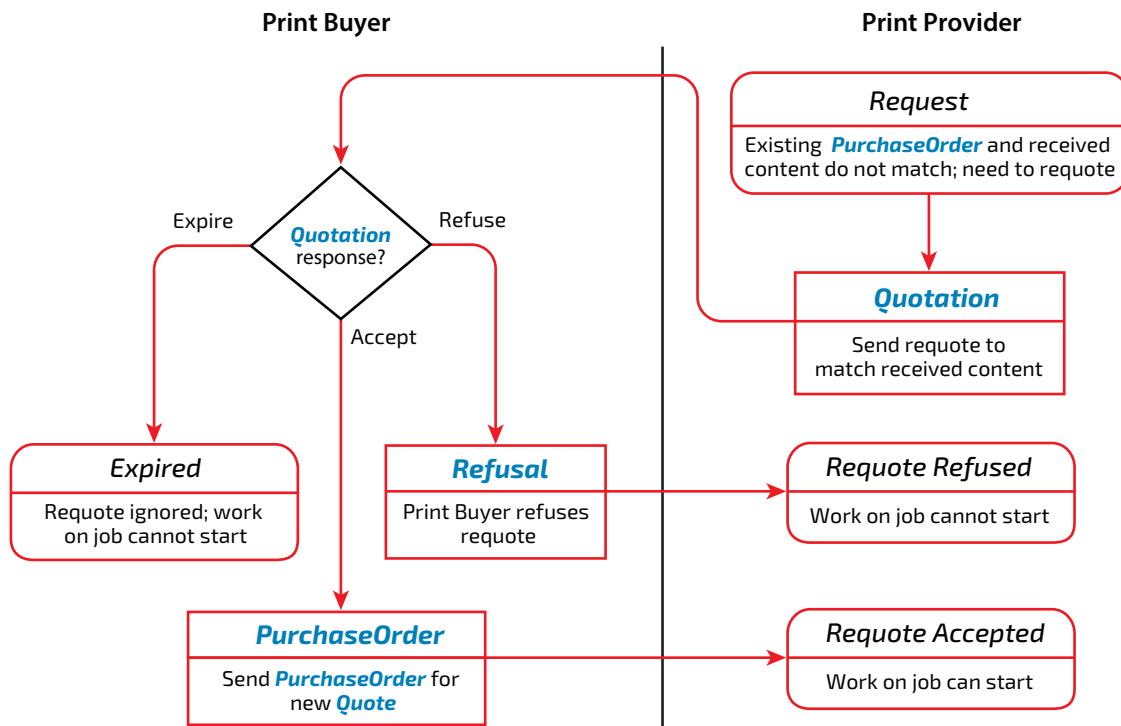
6.5.3 Job Tracking

A *Print Buyer* MAY request details of a job such as milestones, tracking identifiers or color reports.

There are two options for requesting details of a job:

- · A request for the current state of the job, see ▸ Section 6.5.3.1 Job Tracking without Subscription.
- · A subscription for future updates of the state of a job, see ▸ Section 6.5.3.2 Job Tracking with Subscription.

6.5.3.1 Job Tracking without Subscription

A *Print Buyer* MAY request the *Print Provider* to provide details of the current state of a job by sending an *OrderStatusRequest* with *OrderStatusRequest*/@*Subscribed*="false" that references a *PurchaseOrder*.

The *Print Provider* SHALL respond with an *OrderStatusResponse* that references the *OrderStatusRequest* and is evaluated in the context of the *PurchaseOrder* that is referenced by *OrderStatusRequest*.

Figure 6-8: Job Tracking without Subscription
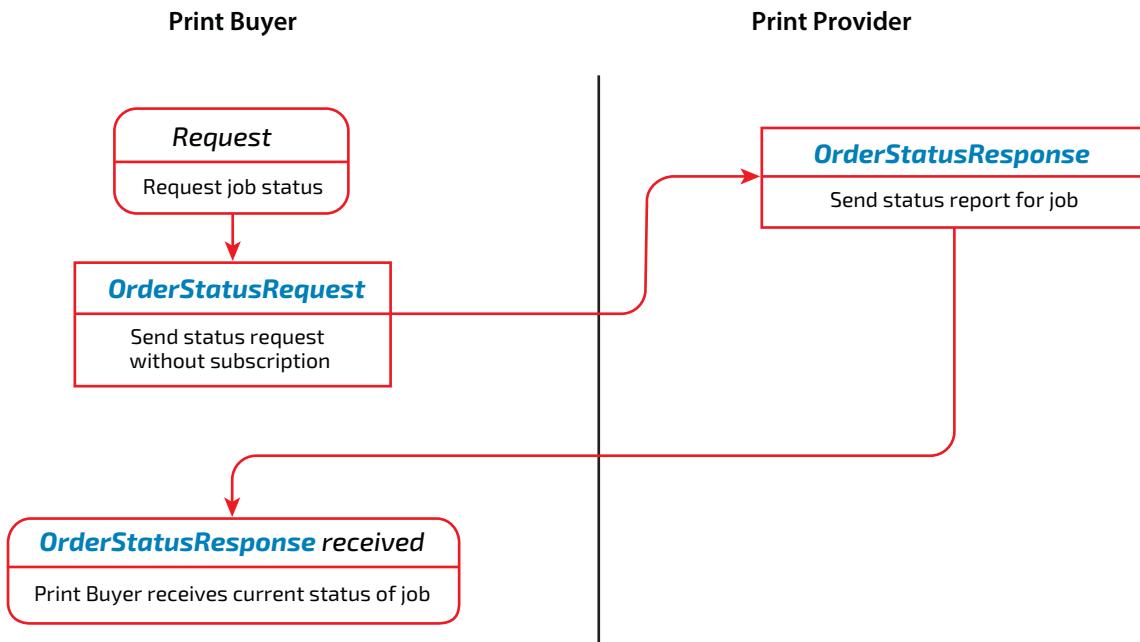
6.5.3.2 Job Tracking with Subscription

A *Print Buyer* MAY subscribe with the *Print Provider* to provide details of the state either by sending an explicit *OrderStatusRequest* with *OrderStatusRequest*/@*Subscribed*="true" that references a *PurchaseOrder* or by providing one or more *PurchaseOrder*/*StatusRequest* elements.

The *Print Provider* SHALL NOT respond with an *OrderStatusResponse* until one of the the trigger conditions defined in a *StatusRequest* is met. The *Print Provider* SHALL send details of the status of a job to a *Print Buyer* by sending an *OrderStatusResponse* that references the *PurchaseOrder* whenever one of the conditions defined by a *StatusRequest* is met.

A *Print Provider* MAY additionally send details of the status of a job to a *Print Buyer* by sending an *OrderStatusResponse* that references the *PurchaseOrder* in a system dependent manner.
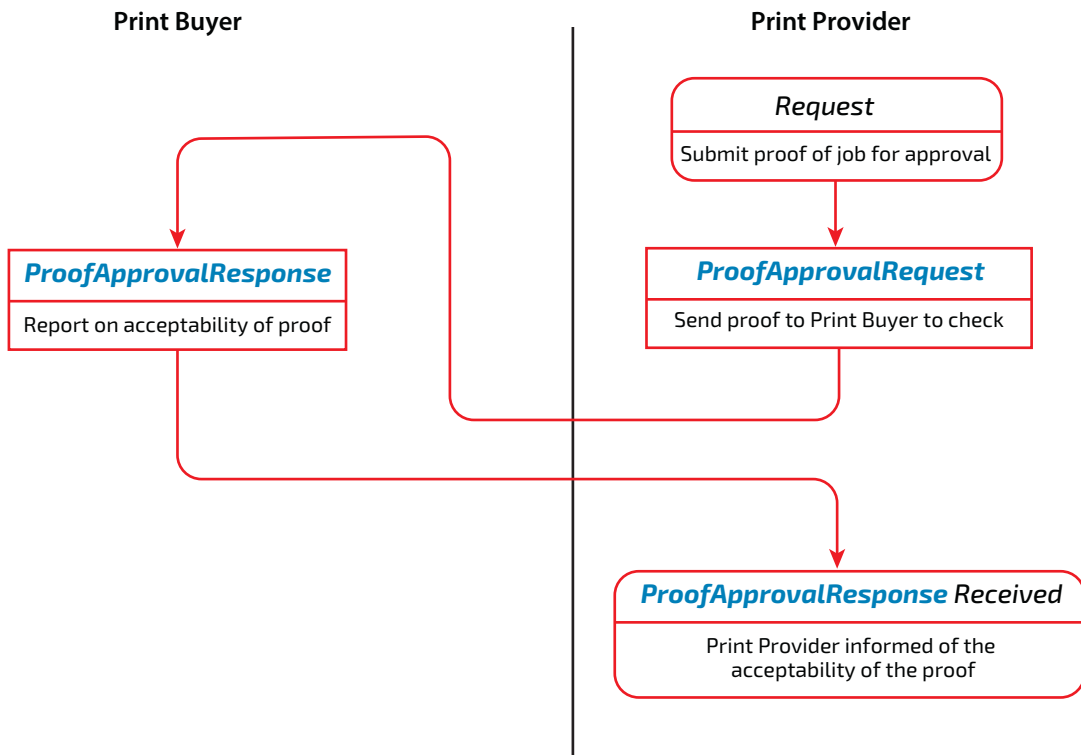
Figure 6-9: Job Tracking with Subscription

TBD

6.5.4 Proof Approval

A *Print Provider* MAY request the *Print Buyer* to approve a soft or hard proof of the content by sending a *ProofApprovalRequest* that references the *PurchaseOrder*.

The *Print Buyer* SHOULD respond with a *ProofApprovalResponse* that references the *ProofApprovalRequest*.

Figure 6-10: Proof Approval

6.5.5 Content Delivery

In most cases, content will be delivered after the *Print Buyer* has sent a *PurchaseOrder*, however content can also be delivered prior to a *PurchaseOrder*, after an *RFQ*.

In either case the *Print Provider* SHOULD check the files and SHALL send a *ContentDeliveryResponse* that SHALL reference the *ContentDelivery* and contains a report of the content check and the completeness of the job.

6.5.5.1 Content Delivery for a Purchase Order

If no content or insufficient content is provided in the *PurchaseOrder*, the *Print Buyer* SHOULD send a *ContentDelivery* to the *Print Provider* that SHALL reference the *PurchaseOrder* and that provides references to the files that SHALL be up-loaded. See ▸ Section 5.5 PrintTalk Packaging for details on how the content MAY be packaged in a single request.
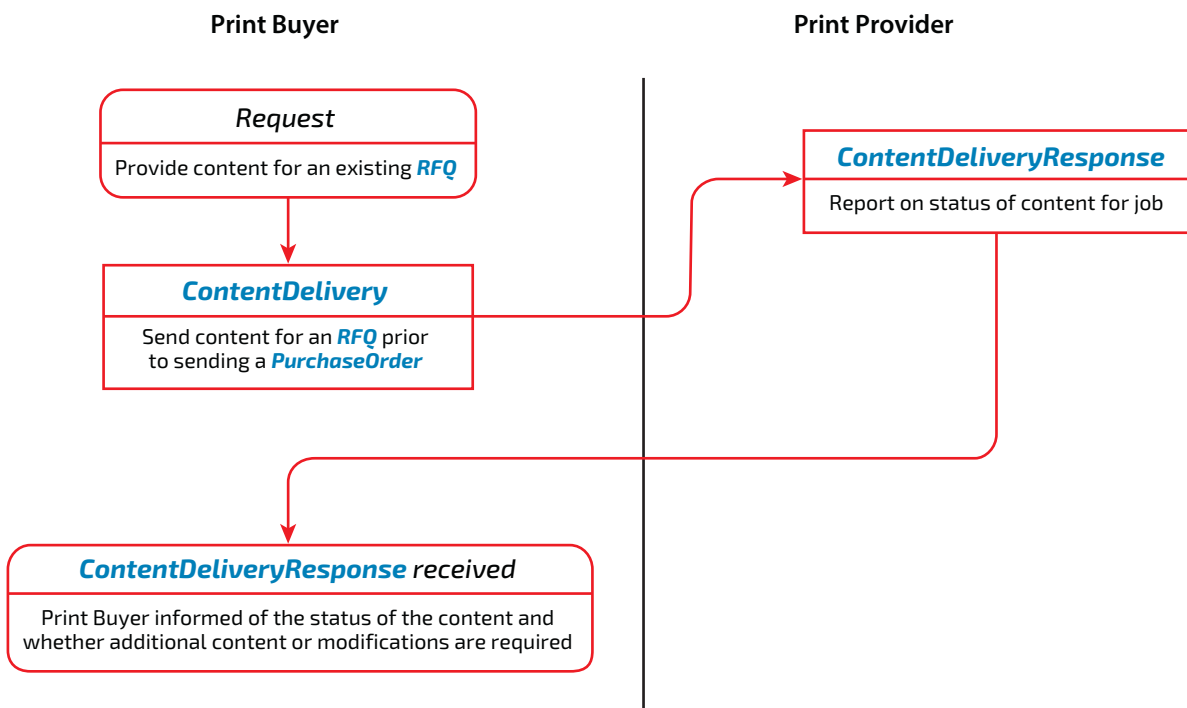
Figure 6-11: Content Delivery for a Purchase Order
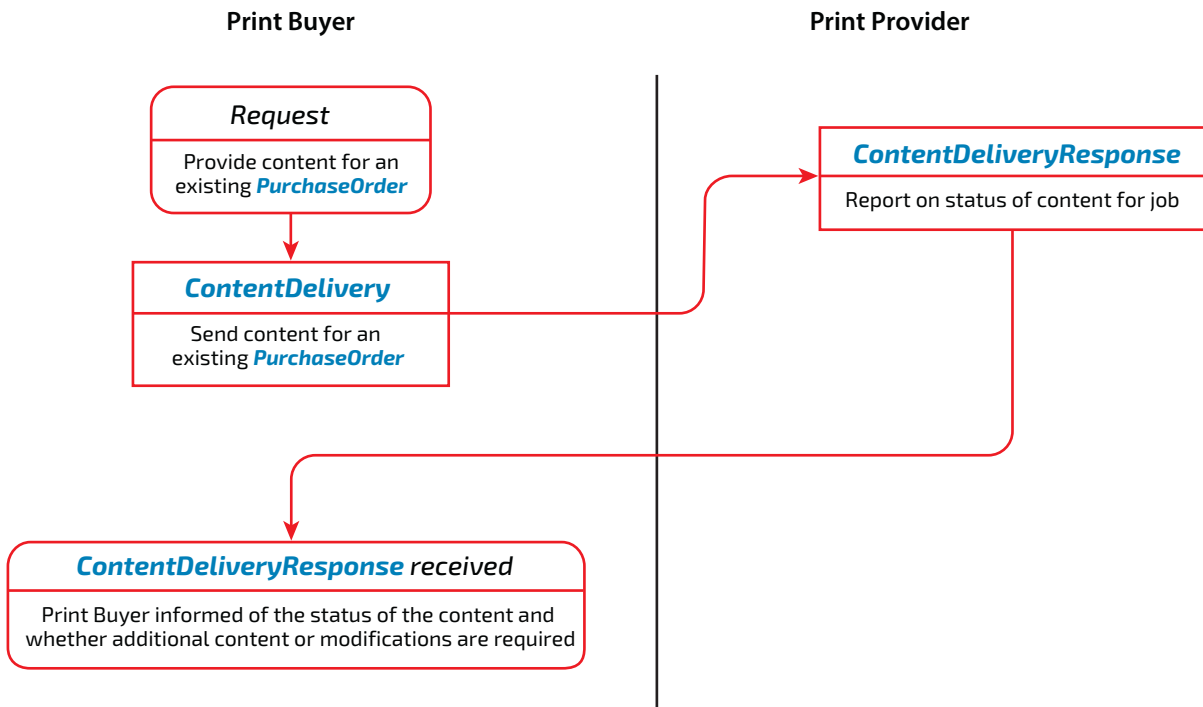
6.5.5.2 Content Delivery Prior to a Purchase Order

If a *Print Buyer* wants to provide content prior to sending a *PurchaseOrder*, it MAY send *ContentDelivery* business objects that SHALL reference an *RFQ*.

*Figure 6-12: Content Delivery Prior to a Purchase Order*

6.5.6 Catalog and Warehouse Exchange

A *Print Buyer* MAY request a report on available items by sending a **StockLevelRequest** to the *Print Provider*. Available items can include both preprinted items that are on stock and custom product types that will be printed individually.

The *Print Provider* SHALL respond with a list of items in a **StockLevelResponse** that references the **StockLevelRequest**.

Figure 6-13: Catalog and Warehouse Exchange
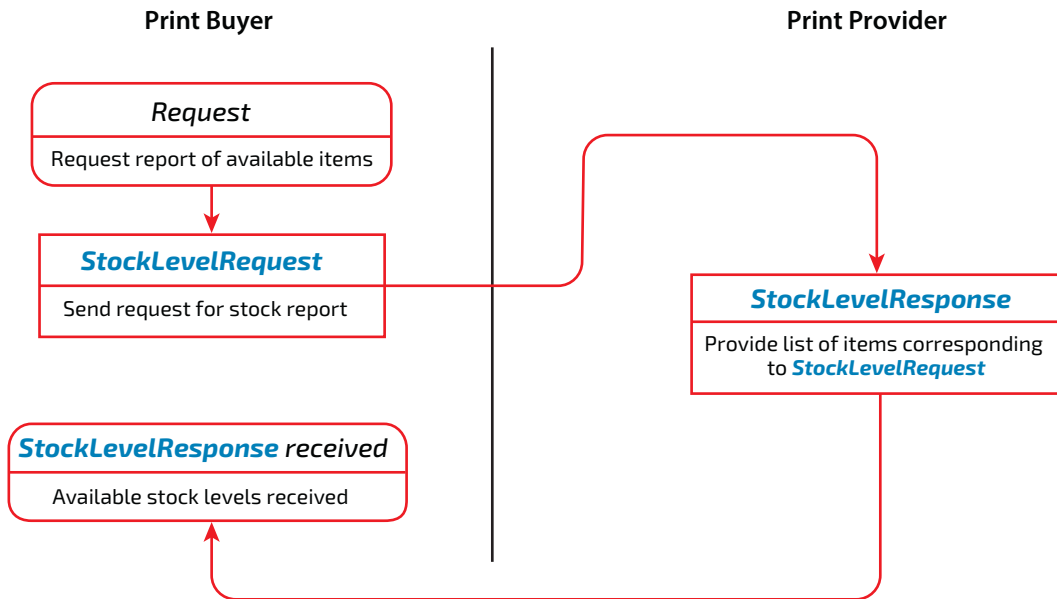
6.5.7 Invoicing

The *Print Provider* typically sends an **Invoice** to the *Print Buyer* after the job is shipped. It may also send an **Invoice** several times, e.g. when certain milestones during the *Production Phase* are reached or after a job has been cancelled. An **Invoice** MAY include additional charges or discounts.

Figure 6-14: Invoicing
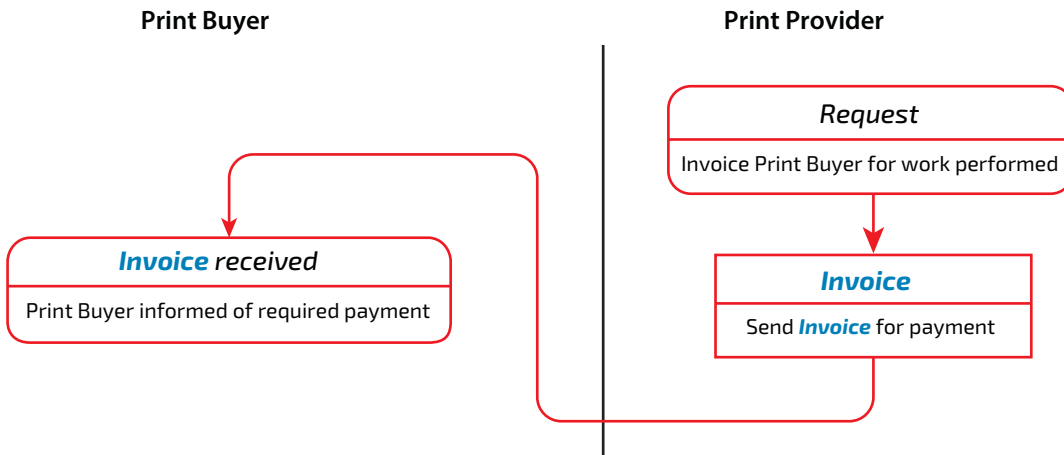
A References

Throughout this specification references to other documents are indicated by short symbolic names inside square brackets, (e.g., ▸[RFC6068]). Implementers need to read and conform to such referenced documents when implementing a part of this specification with such a reference. The reader is directed to this section to find the full title, date, source and availability of all such references.

Table A.1: References

TERM	DEFINITION
[cXML 1.2]	*cXML Reference Guide, version 1.2* Date: April 2013 Produced by: cXML Organization Available at: http://cxml.org/files/downloads.html
[DUNS]	The Data Universal Numbering System or D-U-N-S number is a unique nine-digit identifier for businesses. Produced by: Dun & Bradstreet Available at: https://www.dnb.com/
[ISO4217]	*ISO 4217:2015* *Codes for the representation of currencies* Date: 2015 Produced by: ISO Available at: https://www.iso.org/store.html
[RFC6068]	*RFC 6068* *The mailto URL scheme* Date: October 2010 Produced by: Internet Engineering Task Force (IETF), Network Working Group Available at: http://www.rfc-editor.org/rfcsearch.html
[RFC6750]	*RFC 6750* *The OAuth 2.0 Authorization Framework: Bearer Token Usage.* Date: October 2012 Produced by: Internet Engineering Task Force (IETF), Network Working Group Available at: http://www.rfc-editor.org/rfcsearch.html
[XJDF 2.0]	*Exchange Job Definition Format* *Version 2.0* Date: January 2018 Produced by: CIP4 Organization Available at: http://www.cip4.org
[XJDF 2.1]	*Exchange Job Definition Format* *Version 2.1* Date: June 2020 Produced by: CIP4 Organization Available at: http://www.cip4.org
[XML]	*Extensible Markup Language (XML) 1.0 (Fifth Edition)* *Version (W3C Recommendation of 26 November 2008)* Date: 26 November 2008 Produced by: World Wide Web Consortium (W3C) Available at: http://www.w3.org/TR/2008/REC-xml-20081126/

Table A.1: References

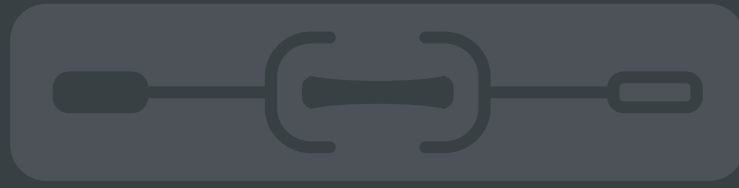| TERM	DEFINITION
[XMLNS]	*Namespaces in XML 1.0 (Third Edition)* *Version (W3C Recommendation of 8 December 2009)* Date: 8 December 2009 Produced by: World Wide Web Consortium (W3C) Available at: http://www.w3.org/TR/REC-xml-names/
[XMLSchema]	*XML Schema Part 0+1+2: Primer, Structures and Datatypes, Second Edition* *Version (W3C Recommendation of 28 October 2004)* Date: 28 October 2004 Produced by: World Wide Web Consortium (W3C) XML Schema working group Available at: http://www.w3.org/TR/xmlschema-0/ http://www.w3.org/TR/xmlschema-1/ http://www.w3.org/TR/xmlschema-2/
[ZIP]	*.ZIP Application Note* *.ZIP File Format Specification – Version 6.3.6* Date: 26 April 2019 Produced by: PKWARE Inc. Available at: https://support.pkware.com/display/PKZIP/APPNOTE

B Release Notes

This appendix contains a brief summary of items that have been changed in **PrintTalk** 2.1.

Refer to previous versions of the **PrintTalk** specification for a complete history of changes.

Table B.1: Release notes for PrintTalk 2.1

ITEM	ACTION	DESCRIPTION	LOCATION
@*Version*	New	Added attribute to *PrintTalk*.	▶ Table 2.1 PrintTalk Element

CIP4

ORGANIZATION

INTEGRATION THROUGH COOPERATION

Adobe

CloudLab

HEIDELBERG

hp

Kodak

MÜLLER MARTINI

RICOH

xerox

cip4.org