



**papiNet**

**OrderStatus**

**papiNet Standard - Version 2.31**

**Documentation**

**Global Standard for the Paper and Forest  
Products Supply Chain**

**Build V2R31\_20211013**

**Date 2021-10-22**

**Production Release**

# OrderStatus

## papiNet Standard - Version 2.31

---

### Copyright

Copyright 2000 - 2021 papiNet G.I.E ("papiNet") and International Digital Enterprise Alliance, Inc. ("IDEAlliance") collectively "Copyright Owner". All rights reserved by the Copyright Owner under the laws of the United States, Belgium, the European Economic Community, and all states, domestic and foreign. This document may be downloaded and copied provided that all copies retain and display the copyright and any other proprietary notices contained in this document. This document may not be sold, modified, edited, or taken out of context such that it creates a false or misleading statement or impression as to the purpose or use of the papiNet specification, which is an open standard. Use of this Standard, in accord with the foregoing limited permission, shall not create for the user any rights in or to the copyright, which rights are exclusively reserved to the Copyright Owner.

papiNet, IDEAlliance, and the members of all papiNet Groups (collectively and individually, "Presenters") make no representations or warranties, express or implied, including, but not limited to, warranties of merchantability, fitness for a particular purpose, title, or non-infringement. The presenters do not make any representation or warranty that the contents of this document are free from error, suitable for any purpose of any user, or that implementation of such contents will not infringe any third party patents, copyrights, trademarks or other rights. By making use of this document, the user assumes all risks and waives all claims against Presenters.

In no event shall Presenters be liable to user (or other person) for direct, indirect, special or consequential damages arising from or related to any use of this document, including, without limitation, lost profits, business interruption, loss of programs, or other data on your information handling system even if Presenters are expressly advised of the possibility of such damages.

---

### Use of Documents in papiNet Implementations

Documents may be used as templates for a papiNet implementation. The Presenters grant the right to modify and edit them to fit an actual implementation project provided all copies display the copyright and any other proprietary notices contained in this document. Such modified documents must not be distributed beyond the trading partners implementing or maintaining a papiNet connection.

# OrderStatus

## papiNet Standard - Version 2.31

---

---

### Table of Contents

Copyright .....	2
Use of Documents in papiNet Implementations .....	2
Table of Contents .....	3
OrderStatus Documentation.....	4
OrderStatus e-Document Overview .....	4
The Scope of the OrderStatus e-Document .....	4
OrderStatusType [attribute] .....	4
Business Rules for OrderStatus.....	5
Processing the OrderStatus e-Document .....	5
Understanding the Diagrams and Content .....	6
OrderStatus Root Element .....	8
OrderStatus .....	8
Primary Elements .....	9
OrderStatusHeader .....	9
OrderStatusDetail .....	11
OrderStatus Business Scenarios .....	14
OrderStatus Scenario Listing.....	14
Scenario A .....	14
Scenario B .....	15
Scenario C .....	17
Scenario D .....	18
Scenario E .....	19
Scenario F .....	20

# OrderStatus

## papiNet Standard - Version 2.31

---

### OrderStatus Documentation

---

#### OrderStatus e-Document Overview

The OrderStatus e-Document permits the sender to report the current status of an order, specific order line items, or to report status of multiple orders based upon some specified criteria. The e-Document enables the sender to indicate a primary status as well as an additional secondary status at the order level as well as for each line and component.

Prior to implementing an OrderStatus e-Document it is assumed that the parties involved have already opened a trading partner relationship and a collaborative agreement has been reached. Such an agreement might include frequency of communication, content details, etc.

A trading partner sends an OrderStatus e-Document to another trading partner based on a schedule or on an event basis agreed between them. The event that triggers an OrderStatus e-Document might be the receipt of an InfoRequest e-Document, or a manufacturing stage.

---

#### The Scope of the OrderStatus e-Document

The OrderStatus e-Document includes:

- A specific date upon which the OrderStatus is generated;
- Purchase order information such as PO number, release number, PO date;
- SenderParty

The OrderStatus e-Document may include:

- Quantity information such as the original order quantity as well as the current OrderStatus quantities (such as the number of reels in a certain manufacturing stage; the number of reels shipped, etc.)
- Order product details such as item numbers, and paper characteristics
- Dates, such as press date, last date of change, and estimated delivery date
- A order and order line level status code
- Carrier and transportation information

It is recommended that the sender of the OrderStatus e-Document include the OrderStatusCode for the purchase order or supplier order in the e-Document.

---

#### OrderStatusType [attribute]

Communicates the method in which the OrderStatus has been, or should be, summarized.

*This item is restricted to the following list.*

##### **ByProduct**

By product

##### **ByPurchaseOrder**



# OrderStatus

## papiNet Standard - Version 2.31

---

---

By the purchase order the material was ordered

### **BySupplierOrderNumber**

By the order the material was manufactured

---

## **Business Rules for OrderStatus**

### **General Business Rules**

The following table lists the business rules that apply to OrderStatus e-Document.

<b>Identifier</b>	<b>Business Rule</b>
<b>OS001</b>	The OrderPrimaryStatus must apply to the order as a whole (i.e., an order cannot be considered to be "Complete" unless each line item is "Complete")
<b>OS002</b>	If the OrderStatus e-Document is in response to an InfoRequest e-Document, the RequestNumber must be included and reflect the RequestNumber of the InfoRequest.
<b>OS003</b>	If the InfoRequest e-Document does not indicate a specific line item, the response must include each line item and its corresponding status.
<b>OS004</b>	If the InfoRequest e-Document OrderStatus request type indicates a response type of "ByPurchaseOrder", then the OrderStatusType must also be "ByPurchaseOrder". Similarly for InfoRequest OrderStatus request types of "ByProduct" and "BySupplierOrderNumber". The OrderStatusType must be of "ByProduct" and "BySupplierOrderNumber" respectively.

---

## **Processing the OrderStatus e-Document**

- The OrderStatus e-Document is the proper response to an InfoRequest e-Document containing an InfoRequestType of "OrderStatus". Alternatively, it may be published on a previously agreed upon schedule based on time intervals or process manufacturing stages. Under this second scenario, the supplier would publish the e-Document at the agreed upon schedule without requiring an InfoRequest e-Document as the trigger.
- It is possible that the OrderStatus e-Document would not be received and processed by the recipient's procurement or order generation system. In this scenario, the OrderStatus e-Document would be received and printed out for distribution to interested parties or alternatively published 'on line' and viewed via a URL or some form of website access designed to display the status.

# OrderStatus

## papiNet Standard - Version 2.31

---

- The OrderStatus e-Document is an information-only e-Document. It does not alter the legal agreement between the parties regarding the order submission and fulfilment. If the OrderStatus e-Document indicates a serious problem or issue with the ability of the supplier to fulfil the order, then the parties involved must resolve the issue. If the OrderStatus e-Document indicates that the PurchaseOrder cannot be fulfilled, then the Parties to this transaction must resolve this open transaction through other means. (It is recommended that an Amended OrderConfirmation e-Document be sent when a PurchaseOrder cannot be fulfilled.)
- Based on the e-Document information requested by the customer, the OrderStatus is processed automatically in the supplier's system.

---

### Understanding the Diagrams and Content

This section provides a graphical view of the schema structures, a discussion of the item's children. You can find additional information about papiNet and the standard at [www.papiNet.org](http://www.papiNet.org).

The graphics contain content model indicators, cardinality indicators, and data type information.

Associated with each graphic are the definitions for the parent item and any associated child items. All attributes are listed first, followed by the elements.

The following information should help you interpret and understand this standard. Please note the following:

- Content Model and Cardinality operate together to determine if the element or attribute are required in the instance document.
- The same attribute can never appear multiple times in the same element so, you will never see a multiple cardinality indicator.

#### Content model indicators:

There are three possible types of content: "sequence", "choice", and "all". The papiNet standard currently does not use the "all" construct.

- (sequence)

The sequence of the items to the right of the graphic (or below the text) is required.

- (choice)

A choice of the items to the right of the graphic (or below the text) is permitted.

- (all)

All the items to the right of the graphic are required.

#### Cardinality indicators:

- Dotted line around element or attribute.

A single instance of the item can optionally exist.

- Dotted line around item with range indicated below.

Multiple instances of the item can optionally exist.

- Solid line around item.

# OrderStatus

## papiNet Standard - Version 2.31

---

---

A single instance of the item must exist.

- Solid line around item with range indicated below

At least one instance must exist; multiple instances can optionally exist.

### **Datatype indication:**

When a data type is assigned to an element (either a simple type or complex type) the name of the data type is presented beneath the item name in the graphic.

- In some cases additional information about the data type is presented (the default value).

Elements can either have content that is textual/numeric in nature or content that is made up of additional elements and/or attributes.

- When the content is textual/numeric in nature “three straight horizontal lines” will appear in the upper left-hand corner of the graphic. Pay attention to these elements because they are where you will be entering your information.
- When the content is made up of additional elements and/or attributes a “gray-box” will appear on the right-hand side of the graphic.
- If the graphic shows both the horizontal lines and the gray-box then, in the papiNet standard, the content below the element are attributes.

# OrderStatus

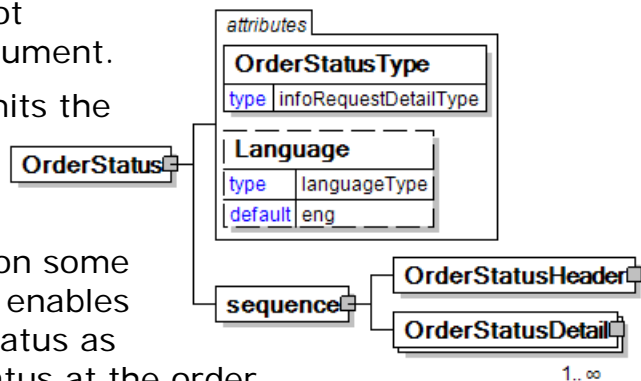
## papiNet Standard - Version 2.31

### OrderStatus Root Element

#### OrderStatus

The OrderStatus element is the root element for the OrderStatus e-Document.

The OrderStatus e-Document permits the sender to report the current status of an order, specific order line items, or to report status of multiple orders based upon some specified criteria. The e-Document enables the sender to indicate a primary status as well as an additional secondary status at the order level as well as for each line and component.



#### OrderStatusType [attribute]

*OrderStatusType is mandatory. A single instance is required.*

Communicates the method in which the OrderStatus has been, or should be, summarized.

*This item is restricted to the following list.*

##### **ByProduct**

By product

##### **ByPurchaseOrder**

By the purchase order the material was ordered

##### **BySupplierOrderNumber**

By the order the material was manufactured

#### Language [attribute]

*Language is optional. A single instance might exist.*

The valid Alpha 2- and Alpha 3-character list of language codes in the ISO 639-1 and 639-2 international standards.

Information on the content of this attribute is available at:  
[https://www.loc.gov/standards/iso639-2/php/code\\_list.php](https://www.loc.gov/standards/iso639-2/php/code_list.php)

#### (sequence)

*The contents of (sequence) are mandatory. A single instance is required.*

##### **OrderStatusHeader**

*OrderStatusHeader is mandatory. A single instance is required.*

Information that applies to the entire OrderStatus e-Document.

##### **OrderStatusDetail**

*OrderStatusDetail is mandatory. One instance is required, multiple instances might exist.*

The specifics of the information being communicated in the OrderStatus e-Document.



# OrderStatus

## papiNet Standard - Version 2.31

### Primary Elements

#### OrderStatusHeader

Information that applies to the entire OrderStatus e-Document.

##### (sequence)

*The sequence of items below is mandatory. A single instance is required.*

##### OrderStatusNumber

*OrderStatusNumber is mandatory. A single instance is required.*

A number used to identify the OrderStatus report.

##### OrderStatusResponseDate

*OrderStatusResponseDate is mandatory. A single instance is required.*

The Date and Time that the OrderStatus e-Document was created.

##### RequestNumber

*RequestNumber is optional. A single instance might exist.*

A unique tracking number specifically identifying the InfoRequest e-Document to the originator. The tracking number is returned with the "information", the answer, to help match the answer to the request.

##### RequestingParty

*RequestingParty is optional. A single instance might exist.*

The party requesting the information.

##### RespondToParty

*RespondToParty is optional. Multiple instances might exist.*

The party the document should be responded to.

##### SenderParty

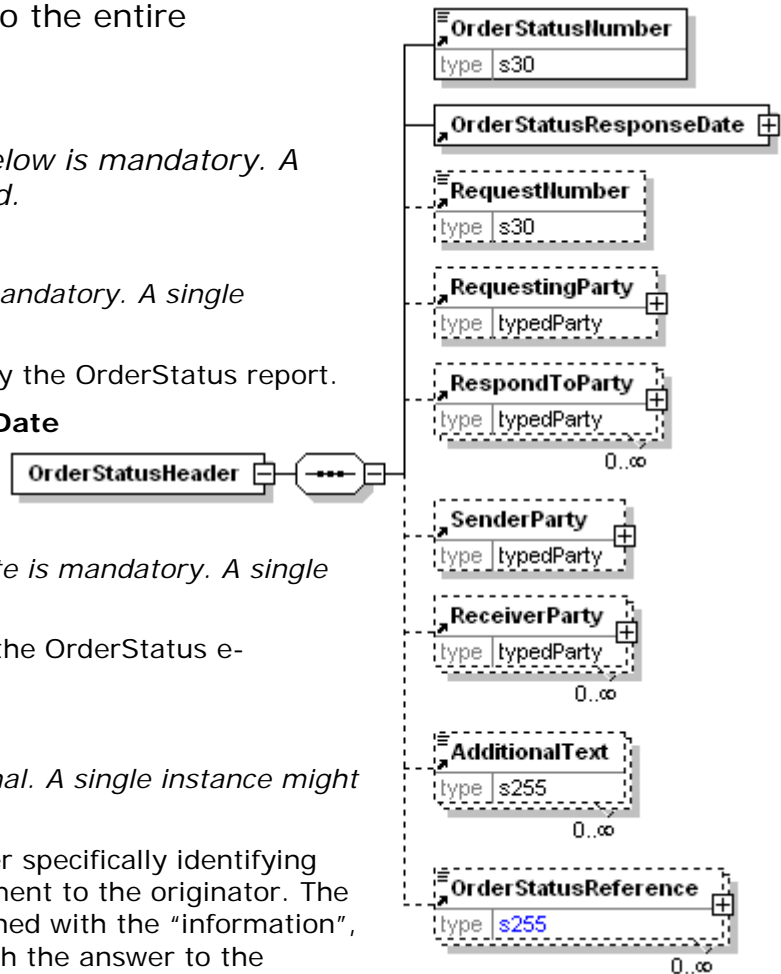
*SenderParty is optional. A single instance might exist.*

The business entity issuing the business document, the source of the document.  
The entity responsible for the content. If the sender party has out sourced the message service to a third party the SenderParty is the issuer of the e-Document and not the party performing the transmission service of the electronic message.

##### ReceiverParty

*ReceiverParty is optional. Multiple instances might exist.*

The business entity for whom the business document is intended, the destination of



# OrderStatus

## papiNet Standard - Version 2.31

---

---

the document.

The entity interested in the content. If the receiver party has outsourced the message service to a third party the ReceiverParty is the intended party for the e-Document and not the party performing the receiving service of the electronic message.

### **AdditionalText**

*AdditionalText is optional. Multiple instances might exist.*

A text field that is used to communicate information not previously defined or for special instructions. To be used only for circumstances not covered by specific elements.

### **OrderStatusReference**

*OrderStatusReference is optional. Multiple instances might exist.*

An item detailing relevant references (such as contract number) pertaining to the OrderStatus. The type of reference is identified by the attribute OrderStatusReferenceType.

# OrderStatus

## papiNet Standard - Version 2.31

### OrderStatusDetail

The specifics of the information being communicated in the OrderStatus e-Document.

**(sequence)**

*The sequence of items below is mandatory. A single instance is required.*

**PurchaseOrderInformation**

*PurchaseOrderInformation is mandatory. A single instance is required.*

A group item containing information unique to this purchase order, which is provided by the buyer. PurchaseOrderInformation can be optional in the supply chain. Invoices are created without having a Purchase Order in Vendor Managed Inventory. Freight invoices also will not have a Purchase Order number.

**PurchaseOrderLineItemNumber**

*PurchaseOrderLineItemNumber is mandatory. A single instance is required.*

The sequential number that uniquely identifies the purchase order line item.

**(sequence)**

*The sequence of items below is optional. Multiple instances might exist.*

**SupplierOrderNumber**

*SupplierOrderNumber is mandatory. A single instance is required.*

The number of the supplier order.

**SupplierOrderLineItemNumber**

*SupplierOrderLineItemNumber is optional. A single instance might exist.*

The number of the line item on the supplier order.

**LocationParty**

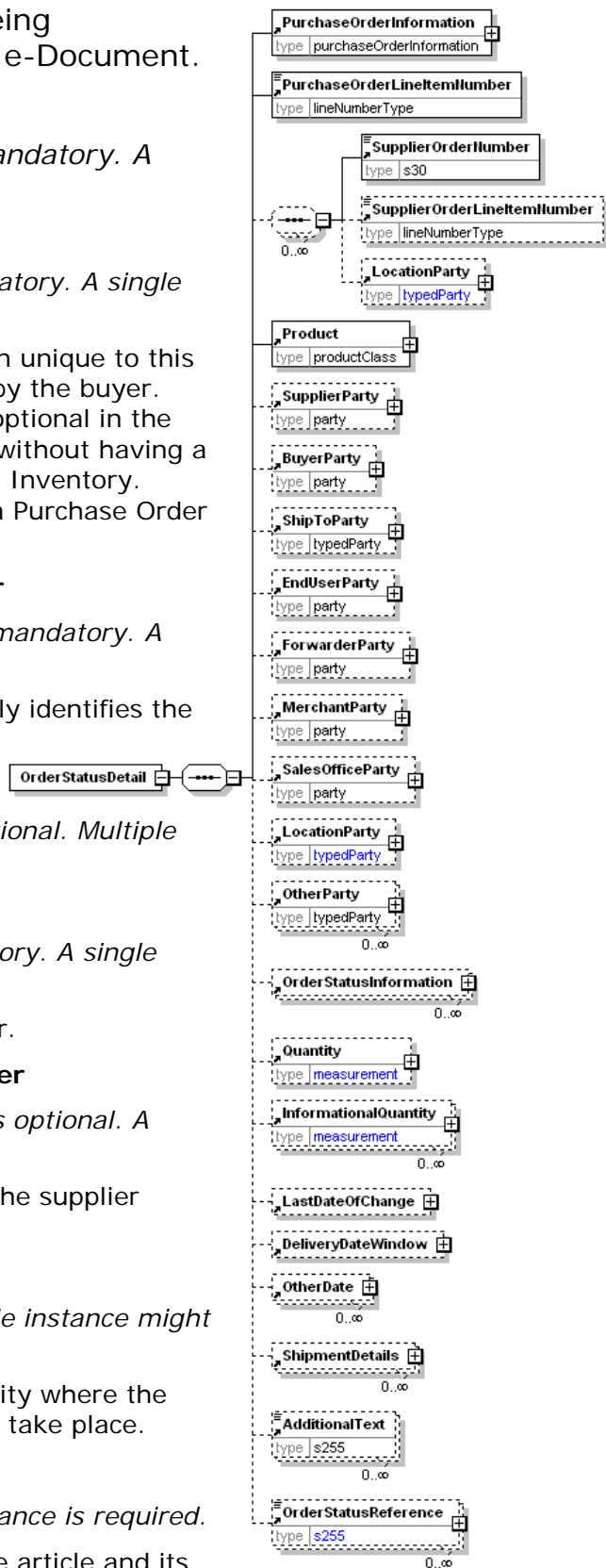
*LocationParty is optional. A single instance might exist.*

The organization or business entity where the business event took place or will take place.

**Product**

*Product is mandatory. A single instance is required.*

Product is a group item defining the article and its characteristics. Product is used to specify product characteristics organized by ProductIdentifier, ProductDescription, and Classification. Book Manufacturing, Label Stock, Paper, Pulp, Recovered Paper, Wood Products, and Virgin Fibre market segments have defined their product



# OrderStatus

## papiNet Standard - Version 2.31

---

characteristics and conversion features for implementation in papiNet.

### **SupplierParty**

*SupplierParty is optional. A single instance might exist.*

The organisation or business entity responsible for providing the product. SupplierParty is also the seller of the product, if Seller is not specified as OtherParty = Seller.

### **BuyerParty**

*BuyerParty is optional. A single instance might exist.*

The legal entity to which the product is sold. Also commonly referred to as the sold-to party or customer. If no OtherParty is defined as the Payer, the Buyer is the Payer.

### **ShipToParty**

*ShipToParty is optional. A single instance might exist.*

The name and/or address to which the goods should be delivered with the party type indicated by the PartyType attribute.

### **EndUserParty**

*EndUserParty is optional. A single instance might exist.*

The party using, consuming, or converting the product. For example, a printer using paper reels for a print job for a publisher. The final ShipTo destination for a product is normally to the end user's facilities.

### **ForwarderParty**

*ForwarderParty is optional. A single instance might exist.*

The trading partner involved in the forwarding of the shipment.

### **MerchantParty**

*MerchantParty is optional. A single instance might exist.*

This named party represents the merchant involved in the business transaction. This party is only used in the communication of OrderStatus otherwise it is handled via OtherParty.

### **SalesOfficeParty**

*SalesOfficeParty is optional. A single instance might exist.*

This named party represents the sales office involved in the business transaction. This party is only used in the communication of OrderStatus otherwise it is handled via OtherParty.

### **LocationParty**

*LocationParty is optional. A single instance might exist.*

The organization or business entity where the business event took place or will take place.

### **OtherParty**

*OtherParty is optional. Multiple instances might exist.*

An organisation or business entity other than those specifically detailed within a business document.

### **OrderStatusInformation**

*OrderStatusInformation is optional. Multiple instances might exist.*

# OrderStatus

## papiNet Standard - Version 2.31

---

A group element that stores two levels of OrderStatus codes.

### **Quantity**

*Quantity is optional. A single instance might exist.*

The Quantity element contains attributes that provide information about the type of quantity that is being communicated, the context in which the particular quantity is to be viewed, and (if the quantity represents an adjustment) an adjustment type.

The Quantity element contains three child elements that enable you to communicate a range of values for the quantity and a target or actual value. It is at this level (Value, RangeMin, and RangeMax) that the unit of measure is specified. This permits the range to be specified in a different unit of measure than the target.

### **InformationalQuantity**

*InformationalQuantity is optional. Multiple instances might exist.*

A quantity given in a valid UOM used for information purposes only (not for calculation). For example, an ordered quantity was 100 reels as opposed to the invoice quantity of 20,000 pounds.

### **LastDateOfChange**

*LastDateOfChange is optional. A single instance might exist.*

The last date for which changes to the line item may occur before it is "locked" in the production process. LastDateOfChange encapsulates Date.

### **DeliveryDateWindow**

*DeliveryDateWindow is optional. A single instance might exist.*

A group item defining the date/time interval for delivery to take place. An element which may contain the estimated date for which delivery is expected. This date is not absolute.

### **OtherDate**

*OtherDate is optional. Multiple instances might exist.*

A date that may not be specifically detailed within a document (example: print date at the PurchaseOrderLineItem).

### **ShipmentDetails**

*ShipmentDetails is optional. Multiple instances might exist.*

A group element containing information that relates to the shipment details for the line item.

### **AdditionalText**

*AdditionalText is optional. Multiple instances might exist.*

A text field that is used to communicate information not previously defined or for special instructions. To be used only for circumstances not covered by specific elements.

### **OrderStatusReference**

*OrderStatusReference is optional. Multiple instances might exist.*

An item detailing relevant references (such as contract number) pertaining to the OrderStatus. The type of reference is identified by the attribute OrderStatusReferenceType.

# OrderStatus

## papiNet Standard - Version 2.31

### OrderStatus Business Scenarios

#### OrderStatus Scenario Listing

<b>Scenario A</b>	Buyer/Publisher issues a Purchase Order to a Supplier and issues an InfoRequest to learn about the status of the order within the supplier's system. The InfoRequestType is "OrderStatus".
<b>Scenario B</b>	Buyer/Publisher issues a Purchase Order and later issues an InfoRequest to learn about the status of a specific product on that PO which has been ordered. The InfoRequestType is "OrderStatus".
<b>Scenario C</b>	Buyer/Publisher issues a Purchase Order and later issues an InfoRequest to learn about the status of a specific product on that PO which has been ordered. The InfoRequestType is "OrderStatus".
<b>Scenario D</b>	Partners have previously agreed upon the Supplier publishing a periodic OrderStatus update on a specific schedule. There is no InfoRequest e-Document.
<b>Scenario E</b>	A small enterprise wants to check the status of an order via a web browser at an online marketplace. The Supplier publishes the current complete OrderStatus. The InfoRequestType is "OrderStatus".
<b>Scenario F</b>	Printer/Binder issues an InfoRequest to a Component Vendor to learn about the status of a Purchase Order issued by a Publisher to the Component Vendor for a product to be shipped to the Printer/Binder.

#### Scenario A

<b>e-Document</b>	OrderStatus
<b>Type</b>	ByPurchaseOrder
<b>Scenario</b>	Buyer/Publisher issues a Purchase Order to a Supplier and issues an InfoRequest to learn about the status of the order within the Mill system. The InfoRequestType is "OrderStatus" and the OrderStatusRequestDetailType is "ByPurchaseOrder".
<b>Outcome</b>	An InfoRequest is generated by the Buyer's system

## OrderStatus

### papiNet Standard - Version 2.31

	and received into the Supplier's system.
<b>Initiator</b>	Buyer
<b>Receiver</b>	Supplier
<b>Trigger</b>	None
<b>Step 1.</b>	<p>Buyer records an original request into their system then sends it to the Supplier. At a minimum, all required elements and corresponding attributes are recorded:</p> <ul style="list-style-type: none"> <li>• InfoRequestType = OrderStatus</li> <li>• RequestNumber = unique number</li> <li>• SenderParty = publisher</li> <li>• RequestingParty = publisher</li> <li>• RespondToParty = publisher</li> <li>• OrderStatusRequestDetailType = ByPurchaseOrder</li> <li>• ByPurchaseOrder = specified PO</li> </ul>
<b>Step 2.</b>	<p>Seller receives an InfoRequest and responds with OrderStatus. At a minimum, all required elements and corresponding attributes are recorded:</p> <ul style="list-style-type: none"> <li>• OrderStatusType = ByPurchaseOrder</li> <li>• OrderStatusNumber = unique number</li> <li>• OrderStatusResponseDate = response date</li> <li>• RequestNumber = unique number</li> <li>• SenderParty = supplier</li> <li>• RequestingParty = publisher</li> <li>• RespondToParty = publisher</li> <li>• For each PO line item: <ul style="list-style-type: none"> <li>• PurchaseOrderInformation = specified PO</li> <li>• PurchaseOrderLineItemNumber = PO line item</li> <li>• OrderStatusCode = status of PO line item</li> <li>• Product = product identifier</li> </ul> </li> </ul> <p>It is likely that many of the optional elements in the OrderStatus e-Document will also be populated in the response e-Document.</p>

#### Scenario B

<b>e-Document</b>	OrderStatus
<b>Type</b>	ByPurchaseOrder
<b>Scenario</b>	Buyer/Publisher issues a Purchase Order and later issues an InfoRequest to learn about the status of two specific line items contained within

## OrderStatus

### papiNet Standard - Version 2.31

	the Order. The InfoRequestType is "OrderStatus" and the OrderStatusRequestDetailType is "ByPurchaseOrder".
<b>Outcome</b>	An InfoRequest is generated by the Buyer's system and received into the Supplier's system.
<b>Initiator</b>	Buyer
<b>Receiver</b>	Supplier
<b>Preconditions</b>	What exists prior to the start?
<b>Trigger</b>	None
<b>Step 1.</b>	<p>Buyer records an original request into their system then sends it to the Supplier. At a minimum, all required elements and corresponding attributes are recorded:</p> <ul style="list-style-type: none"> <li>• InfoRequestType = OrderStatus</li> <li>• RequestNumber = unique number</li> <li>• SenderParty = publisher</li> <li>• RequestingParty = publisher</li> <li>• RespondToParty = publisher</li> <li>• OrderStatusRequestDetailType = ByPurchaseOrder</li> <li>• ByPurchaseOrder = specified PO</li> <li>• PurchaseOrderLineItemNumber = specified PO line item 1</li> <li>• PurchaseOrderLineItemNumber = specified PO line item 2</li> </ul>
<b>Step 2.</b>	<p>Seller receives an InfoRequest and responds with OrderStatus. At a minimum, all required elements and corresponding attributes are recorded:</p> <ul style="list-style-type: none"> <li>• OrderStatusType = ByPurchaseOrder</li> <li>• OrderStatusNumber = unique number</li> <li>• OrderStatusResponseDate = response date</li> <li>• RequestNumber = unique number</li> <li>• SenderParty = supplier</li> <li>• RequestingParty = publisher</li> <li>• RespondToParty = publisher</li> <li>• For each PO line item: <ul style="list-style-type: none"> <li>• PurchaseOrderInformation = specified PO</li> <li>• PurchaseOrderLineItemNumber = PO line item</li> <li>• OrderStatusCode = status of PO line item</li> </ul> </li> </ul>



## OrderStatus

### papiNet Standard - Version 2.31

	<ul style="list-style-type: none"> <li>• Product = product identifier</li> </ul> <p>It is likely that many of the optional elements in the OrderStatus e-Document will also be populated in the response e-Document.</p>
--	--

#### Scenario C

<b>e-Document</b>	OrderStatus
<b>Type</b>	ByPurchaseOrder
<b>Scenario</b>	Buyer/Publisher issues a PurchaseOrder and later issues an InfoRequest to learn about the status of a specific product on that PO which has been ordered. The InfoRequestType is "OrderStatus" and the OrderStatusRequestDetailType is "ByPurchaseOrder".
<b>Outcome</b>	An InfoRequest is generated by the Buyer's system and received into the Supplier's system.
<b>Initiator</b>	Buyer
<b>Receiver</b>	Supplier
<b>Trigger</b>	None
<b>Step 1.</b>	<p>Buyer records an original request into their system then sends it to the Supplier. At a minimum, all required elements and corresponding attributes are recorded:</p> <ul style="list-style-type: none"> <li>• InfoRequestType = OrderStatus</li> <li>• RequestNumber = unique number</li> <li>• SenderParty = publisher</li> <li>• RequestingParty = publisher</li> <li>• RespondToParty = publisher</li> <li>• OrderStatusRequestDetailType = ByPurchaseOrder</li> <li>• ByPurchaseOrder = specified PO</li> <li>• Product = specified product identifier</li> </ul>
<b>Step 2.</b>	<p>Seller receives InfoRequest and responds with OrderStatus. At a minimum, all required elements and corresponding attributes are:</p> <ul style="list-style-type: none"> <li>• OrderStatusType = ByPurchaseOrder</li> <li>• OrderStatusNumber = unique number</li> <li>• OrderStatusResponseDate = response date</li> <li>• RequestNumber = unique number</li> </ul>

## OrderStatus

### papiNet Standard - Version 2.31

	<ul style="list-style-type: none"> <li>• SenderParty = supplier</li> <li>• RequestingParty = publisher</li> <li>• RespondToParty = publisher</li> </ul> <p>For each PO line item:</p> <ul style="list-style-type: none"> <li>• PurchaseOrderInformation = specified PO</li> <li>• PurchaseOrderLineItemNumber = PO line item</li> <li>• OrderStatusCode = status of PO line item</li> <li>• Product = product identifier</li> </ul> <p>It is likely that many of the optional elements in the OrderStatus e-Document will also be populated in the response e-Document.</p>
--	---

#### Scenario D

<b>e-Document</b>	OrderStatus
<b>Type</b>	ByPurchaseOrder or ByProduct
<b>Scenario</b>	Partners have previously agreed upon the Supplier publishing a periodic OrderStatus update on a specific schedule.
<b>Outcome</b>	An OrderStatus is generated by the Supplier's system and received into the Buyer's system.
<b>Initiator</b>	by schedule
<b>Receiver</b>	Buyer
<b>Preconditions</b>	What exists prior to the start?
<b>XML File</b>	The name of any sample file.
<b>Trigger</b>	Prearranged schedule
<b>Step 1.</b>	<p>Supplier initiates an OrderStatus e-Document at predefined intervals. At a minimum, all required elements and corresponding attributes are recorded:</p> <ul style="list-style-type: none"> <li>• OrderStatusType = ByPurchaseOrder or ByProduct</li> <li>• OrderStatusNumber = unique number</li> <li>• OrderStatusResponseDate = response date</li> <li>• SenderParty = supplier</li> <li>• RequestingParty = publisher</li> <li>• RespondToParty = publisher</li> </ul> <p>For each PO line item:</p> <ul style="list-style-type: none"> <li>• PurchaseOrderInformation = specified PO</li> <li>• PurchaseOrderLineItemNumber = PO line</li> </ul>

## OrderStatus

### papiNet Standard - Version 2.31

	<p>item</p> <p>and/or</p> <ul style="list-style-type: none"> <li>• Product = product identifier</li> <li>• OrderStatusCode = status of PO line item</li> </ul> <p>It is likely that many of the optional elements in the OrderStatus e-Document will also be populated in the response e-Document.</p>
--	--

#### Scenario E

<b>e-Document</b>	OrderStatus
<b>Type</b>	ByPurchaseOrder
<b>Scenario</b>	A small enterprise wants to check the status of an order via a web browser at an online marketplace. The Supplier publishes the current complete OrderStatus. The InfoRequestType is "OrderStatus" and the OrderStatusRequestDetailType is "ByPurchaseOrder".
<b>Outcome</b>	An OrderStatus is generated by the Supplier's system and received into the Buyer's/Marketplace's system.
<b>Initiator</b>	Buyer/website
<b>Receiver</b>	Supplier
<b>Preconditions</b>	What exists prior to the start?
<b>Trigger</b>	InfoRequest
<b>Step 1.</b>	<p>The Buyer logs on to the website and views open orders online and indicate orders for which status is requested. Website creates InfoRequest and sends it to the Supplier. At a minimum, all required elements and corresponding attributes are recorded:</p> <ul style="list-style-type: none"> <li>• InfoRequestType = OrderStatus</li> <li>• RequestNumber = unique number</li> <li>• SenderParty = website or e-marketplace</li> <li>• RequestingParty = publisher</li> <li>• RespondToParty = website or e-marketplace</li> <li>• OrderStatusRequestDetailType = ByPurchaseOrder</li> <li>• ByPurchaseOrder = specified PO</li> </ul>

## OrderStatus

### papiNet Standard - Version 2.31

<b>Step 2.</b>	<p>Seller receives InfoRequest and responds with OrderStatus. At a minimum, all required elements and corresponding attributes are recorded:</p> <ul style="list-style-type: none"> <li>• OrderStatusType = ByPurchaseOrder</li> <li>• OrderStatusNumber = unique number</li> <li>• OrderStatusResponseDate = response date</li> <li>• RequestNumber = unique number</li> <li>• SenderParty = supplier</li> <li>• RequestingParty = website or e-marketplace</li> <li>• RespondToParty = publisher</li> </ul> <p>For each PO line item:</p> <ul style="list-style-type: none"> <li>• PurchaseOrderInformation = specified PO</li> <li>• PurchaseOrderLineItemNumber = PO line item</li> <li>• OrderStatusCode = status of PO line item</li> <li>• Product = product identifier</li> </ul> <p>It is likely that many of the optional elements in the OrderStatus e-Document will also be populated in the response e-Document.</p>
----------------	---

#### Scenario F

<b>e-Document</b>	OrderStatus
<b>Type</b>	ByPurchaseOrder
<b>Scenario</b>	Buyer/Publisher issues a PurchaseOrder to a Component Vendor for a cover to be shipped to a printer/binder and transmits a copy of the PurchaseOrder to the printer/binder. The Printer/Binder later issues an InfoRequest to learn about the status of the covers on that PO which had been sent to the component vendor. The InfoRequestType is "OrderStatus" and the OrderStatusRequestDetailType is "ByPurchaseOrder".
<b>Outcome</b>	An InfoRequest is generated by the Printer/Binder's system and received into the Component Vendor's system.
<b>Initiator</b>	Printer/Binder
<b>Receiver</b>	Supplier (Component Vendor)

## OrderStatus

### papiNet Standard - Version 2.31

<b>Trigger</b>	None
<b>Step 1.</b>	<p>Printer/Binder records an original request into their system then sends it to the Supplier (component vendor). At a minimum, all required elements and corresponding attributes are recorded:</p> <ul style="list-style-type: none"> <li>• InfoRequestType = OrderStatus</li> <li>• RequestNumber = unique number</li> <li>• SenderParty = printer/binder</li> <li>• RequestingParty = printer/binder</li> <li>• RespondToParty = printer/binder</li> <li>• OrderStatusRequestDetailType = ByPurchaseOrder</li> <li>• ByPurchaseOrder = specified PO</li> <li>• Product = specified product identifier</li> </ul>
<b>Step 2.</b>	<p>Supplier (Component Vendor) receives InfoRequest and responds with OrderStatus. At a minimum, all required elements and corresponding attributes are recorded:</p> <ul style="list-style-type: none"> <li>• OrderStatusType = ByPurchaseOrder</li> <li>• OrderStatusNumber = unique number</li> <li>• OrderStatusResponseDate = response date</li> <li>• RequestNumber = unique number</li> <li>• SenderParty = supplier</li> <li>• RequestingParty = publisher</li> <li>• RespondToParty = publisher</li> </ul> <p>For each PO:</p> <ul style="list-style-type: none"> <li>• PurchaseOrderInformation = specified PO</li> <li>• OrderStatusCode = status of PO line item</li> <li>• Product = product identifier</li> </ul> <p>It is likely that many of the optional elements in the OrderStatus e-Document will also be populated in the response e-Document.</p>