

**ProductAttributes**

**papiNet Standard - Version 2.31**

**Documentation**

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

**Build V2R31\_20240321**

**Date 2024-04-14**

**Production Release**

# **ProductAttributes**

## **papiNet Standard - Version 2.31**

---

### **Copyright**

Copyright 2000 - 2024 Papinet SNC ("papiNet") the "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, 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.

# ProductAttributes

## papiNet Standard - Version 2.31

---

---

### Table of Contents

Copyright .....	2
Use of Documents in papiNet Implementations .....	2
Table of Contents .....	3
ProductAttributes Documentation .....	4
ProductAttributes e-Document Overview .....	4
The Scope of the ProductAttributes e-Document .....	4
ProductAttributesType [attribute] .....	5
Business Rules for ProductAttributes .....	5
Processing the ProductAttributes e-Document .....	6
Understanding the Diagrams and Content .....	6
ProductAttributes Root Element .....	8
ProductAttributes .....	8
Primary Elements .....	10
ProductAttributesHeader .....	10
ProductAttributesLineItem .....	12
ProductAttributes Business Scenarios .....	16
ProductAttributes Scenario Listing .....	16
Scenario A .....	16
Scenario B .....	18
Scenario C .....	19
Scenario D .....	20

# ProductAttributes

## papiNet Standard - Version 2.31

---

---

### ProductAttributes Documentation

---

#### ProductAttributes e-Document Overview

The ProductAttributes e-Document provides the means to communicate information about products between business partners.

The ProductAttributes e-Document gives a supplier the ability to send information about available products to a buyer or other business partner, so that the receiving party can create and maintain a database of the products available from the supplier. The product definitions consist of one or more product codes and enough product attribute values to define the product, such as product codes, brand name, supplier, producing mill, basis weight, and whether it's supplied in reels or sheets, sizes, as well as many more possibilities.

The ProductAttributes e-Document is also designed to enable exchange of product codes used in electronic trading. Once a buyer has received the information from a supplier the buyer can respond with the information updated with a cross reference to the buyer's product codes.

The ProductAttributes e-Document assumes that a previous agreement between the parties exchanging the information has taken place. The parties exchange the information on a frequency or event basis agreed between them. The agreement would include frequency, the event that triggers communication, content detail, units of measure, and other aspects.

---

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

Most of the information in the e-Document is optional, which gives the parties involved in the information exchange the ability to tailor make the e-Document contents to suit their needs.

The ProductAttributes e-Document hierarchy consists of:

- The header level to define the issue date, identifier for the e-Document, sender, receiver, necessary additional parties, free text description, etc.
- The ProductAttributes line item level defining ProductIdentifier (product codes, product names, brand codes, brand names), suppliers, mills, price details, breakdown of order quantities, validity period, end uses
- the product level to define the product characteristics like
  - classification, product free text description (with the possibility to include several languages),
  - paper properties (basis weight, caliper, colour, gloss, opacity, etc),
  - conversion characteristics (reel/sheet, core details, grain direction, winding direction, size) and
  - packaging characteristics (single/multi-packed reel, ream, number of sheets, pallet description)

# ProductAttributes

## papiNet Standard - Version 2.31

### ProductAttributesType [attribute]

The type of information being communicated in the ProductAttributes e-Document



*This item is restricted to the following list.*

#### PriceList

The information represents a price list.

#### ProductCrossReference

The information contains product cross reference information

#### ProductProperties

The information is a representation of the properties of the products.

#### UniqueProductIdentifier

The information contains unique product identifiers.

## Business Rules for ProductAttributes

### General Business Rules

The following tables list the business rules that apply to ProductAttributes.

Identifier	Business Rule
PA001	The frequency and triggering of ProductAttributes e-Document exchange is determined by agreement between trading partners.
PA002	The SenderProductAttributesIdentifier identifies the ProductAttributes e-Document and is unique between the trading partners, identifying a collection of products communicated between trading partners. The trading partners must define how they divide the entire set of products communicated between themselves in manageable portions (see warning later in this document about risks associated with large messages).
PA003	A ProductIdentifier must identify the product between two trading partners, although it might not be unique within the ProductAttributesLineItems sent, allowing the sender to repeat ProductAttributesLineItems with the same product identifier detailing different product characteristics. The ProductIdentifier together with key product attributes to establish uniqueness are defined in the agreement between the trading partners.

# ProductAttributes

## papiNet Standard - Version 2.31

---

---

### Processing the ProductAttributes e-Document

#### Status Values Used When Processing the ProductAttributes e-Document

The ProductAttributes e-Document includes ProductAttributesStatusType, which indicates whether the e-Document is:

- Original, which means that the e-Document includes a new list of product attributes.
- Amended, which means that the e-Document includes amendments to a list of products (additions, amendments, or cancellations).
- Replaced, which indicates that the e-Document replaces the product information sent an earlier communication.
- Cancelled, which means that the e-Document includes a cancelled list of product attributes.

#### Possible Combination of Status Type on ProductAttributes e-Document

It is highly recommended that the Replace status type be only used to update previously sent ProductAttributes information. Replaced should not be used to resend the original information; in this case the Original status type should be used.

#### e-Document size

It should be noted that sending many products in the same e-Document may result in very large messages, which in turn may lead to problems when the receiver processes the e-Document. It is therefore advisable to estimate the e-Document size and check the e-Document size during testing to find out what the maximum e-Document size is likely to be. If necessary the exchange of the product information between the business parties should be divided into manageable portions.

We recommend that when dividing the information into several e-Documents, using the same SenderProductAttributesIdentifier, it is recommended that the first communication be sent with status type Original and subsequent ones with the status type Amend. Please refer to the use cases for a more detailed description. An alternative solution to the above-described problem is to divide the set of products to be communicated into manageable portions, assigning each of the portions their own SenderProductAttributesIdentifier.

---

### 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:

# ProductAttributes

## papiNet Standard - Version 2.31

---

- 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.

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.

# ProductAttributes

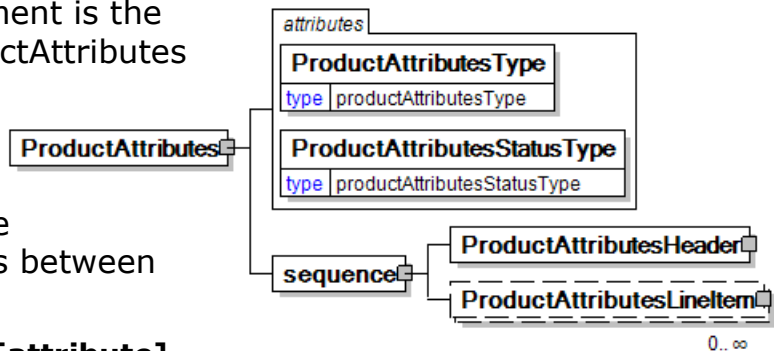
## papiNet Standard - Version 2.31

### ProductAttributes Root Element

#### ProductAttributes

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

The ProductAttributes e-Document provides the means to communicate information about products between business partners.



#### ProductAttributesType [attribute]

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

The type of information being communicated in the ProductAttributes e-Document

*This item is restricted to the following list.*

##### PriceList

The information represents a price list.

##### ProductCrossReference

The information contains product cross reference information

##### ProductProperties

The information is a representation of the properties of the products.

##### UniqueProductIdentifier

The information contains unique product identifiers.

#### ProductAttributesStatusType [attribute]

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

Defines the status of the entire ProductAttributes e-Document, in other words, at the root level

*This item is restricted to the following list.*

##### Amended

The supplied information is changed.

##### Cancelled

The supplied information is cancelled. Items that have been cancelled are not included in totals on the summary levels of the e-Document.

##### Original

The supplied information is the first version of that information.

##### Replaced

The supplied information is replacing earlier supplied information. The receiver should revalidate the information in their system based upon the entire information received.

#### (sequence)



# ProductAttributes

## papiNet Standard - Version 2.31

---

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

### **ProductAttributesHeader**

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

Information that applies to the entire ProductAttributes e-Document.

### **ProductAttributesLineItem**

*ProductAttributesLineItem is optional. Multiple instances might exist.*

A group element that contains data relevant to the Product described on the line item.

# ProductAttributes papiNet Standard - Version 2.31

## Primary Elements

### ProductAttributesHeader

Information that applies to the entire ProductAttributes e-Document.

#### (sequence)

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

#### ProductAttributesNumber

ProductAttributesNumber is optional. A single instance might exist.

The unique identifier of the ProductAttributes e-Document.

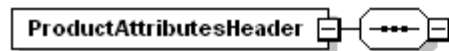
#### ProductAttributesIssueDate

ProductAttributesIssueDate is mandatory. A single instance is required.

The Date and Time the ProductAttributes are issued. (Compare to ValidityPeriod.)

#### TransactionHistoryNumber

TransactionHistoryNumber is optional. A single instance might exist.



A sequential number that keeps track of the version of a document.

However when the document is a confirmation document, in which case the TransactionHistoryNumber refers to the trigger transaction for the confirmation.

#### 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.

#### SenderProductAttributesIdentifier

SenderProductAttributesIdentifier is mandatory. A single instance is required.

A unique identifier assigned to the combination of products, properties, and ReceivingParty that makes this list of ProductAttributes unique from the SendingParty's point of view. Subsequent ProductAttributes with updates will use this same identifier.



# ProductAttributes

## papiNet Standard - Version 2.31

---

### **SenderProductAttributesName**

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

The name given to the set of products sent in the ProductAttributes e-Document.

### **SenderProductAttributesDescription**

*SenderProductAttributesDescription is optional. Multiple instances might exist.*

An element containing free text used to describe the set of products sent in the ProductAttributes.

### **OtherParty**

*OtherParty is optional. Multiple instances might exist.*

An organisation or business entity other than those specifically detailed within a e-Document.

### **SenderParty**

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

The business entity issuing the e-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 e-Document is intended, the destination of 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.

### **ShipToCharacteristics**

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

A group item that provides information important for the Ship-To Party.

### **ProductAttributesReference**

*ProductAttributesReference is optional. Multiple instances might exist.*

References that apply to the ProductAttributes e-Document, contains the ProductAttributesReferenceType attribute.

### **ValidityPeriod**

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

The validity period for a specific item, e.g. the validity period for an e-Document.

### **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.

### **TermsAndDisclaimers**

*TermsAndDisclaimers is optional. Multiple instances might exist.*

# ProductAttributes

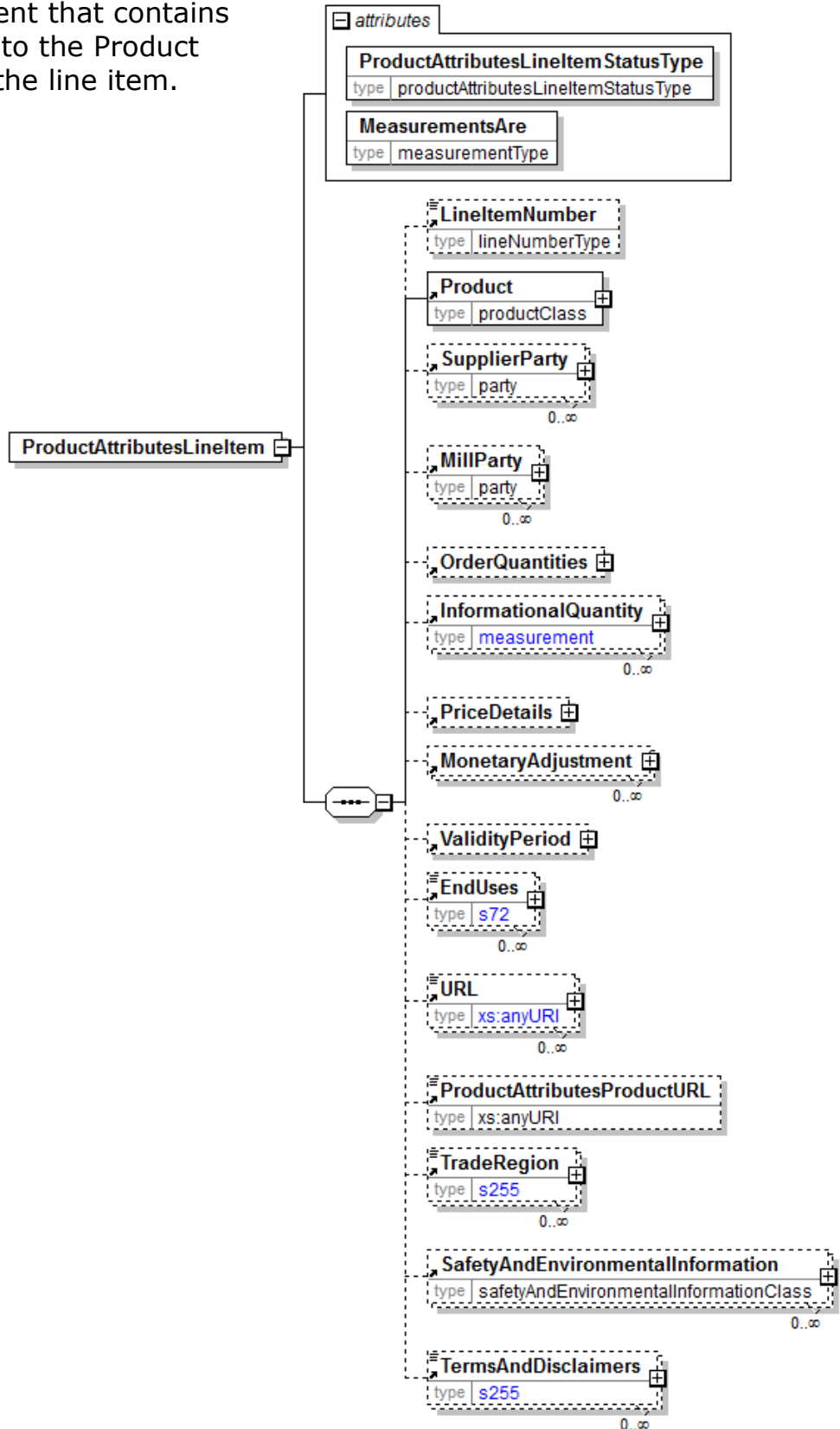
## papiNet Standard - Version 2.31

---

An element that contains legal information with an indication of what the Language is.

### ProductAttributesLineItem

A group element that contains data relevant to the Product described on the line item.



# ProductAttributes

## papiNet Standard - Version 2.31

---

### **ProductAttributesLineItemStatusType [attribute]**

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

Defines the status of the ProductAttributesLineItem

*This item is restricted to the following list.*

#### **Amended**

The supplied information is changed.

#### **Cancelled**

The supplied information is cancelled. Items that have been cancelled are not included in totals on the summary levels of the e-Document.

#### **Original**

The supplied information is the first version of that information.

### **MeasurementsAre [attribute]**

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

Used to define for a product whether the details provided are ranges and lists of properties, or whether they are discrete, single properties of the product.

*This item is restricted to the following list.*

#### **Discreet**

For a product, elements with measurements are observed or target values, and elements that allow a list of attributes for the product to be specified must have zero or one occurrence.

#### **Range**

For a product, elements with measurements specify a range, and elements that allow a list of attributes for the product to be specified may have zero or more occurrences.

### **(sequence)**

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

#### **LineItemNumber**

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

A sequential number that uniquely identifies the line item of an e-Document.

#### **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 characteristics and conversion features for implementation in papiNet.

#### **SupplierParty**

*SupplierParty is optional. Multiple instances 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.

# ProductAttributes

## papiNet Standard - Version 2.31

---

### **MillParty**

*MillParty is optional. Multiple instances might exist.*

The organisation or business entity that actually produces the product.

### **OrderQuantities**

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

An element that contains the quantities that can be used for ordering.

### **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.

### **PriceDetails**

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

An element that groups together price information.

### **MonetaryAdjustment**

*MonetaryAdjustment is optional. Multiple instances might exist.*

The element containing the information necessary for the understanding, calculation, and treatment of an adjustment to a currency amount. MonetaryAdjustment contains an attribute that indicates the type of adjustment being communicated.

### **ValidityPeriod**

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

The validity period for a specific item, e.g. the validity period for an e-Document.

### **EndUses**

*EndUses is optional. Multiple instances might exist.*

A text element used to express in human readable form a list of applicable end uses for a product. Examples of end uses are:

- Magazine
- Book
- Commercial print
- etc

### **URL**

*URL is optional. Multiple instances might exist.*

Universal Resource Locator. While typically a web address you could use this field to hold an email address.

### **ProductAttributesProductURL**

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

The web address of information pertinent to a particular product.

### **TradeRegion**

*TradeRegion is optional. Multiple instances might exist.*

A geographic area for product availability communication.

# ProductAttributes

## papiNet Standard - Version 2.31

---

### **SafetyAndEnvironmentalInformation**

*SafetyAndEnvironmentalInformation is optional. Multiple instances might exist.*

Name of certification type, if any, on the goods (For example, FSC, PEFC). SafetyAndEnvironmental needs a value or measurement to communicate the percentage of the product is certified (for example, 75% is certified by the indicated agency).

### **TermsAndDisclaimers**

*TermsAndDisclaimers is optional. Multiple instances might exist.*

An element that contains legal information with an indication of what the Language is.

# ProductAttributes papiNet Standard - Version 2.31

## ProductAttributes Business Scenarios

### ProductAttributes Scenario Listing

<b>Scenario A</b>	Supplier sends an e-Document with information about products supplied to a buyer. <ul style="list-style-type: none"> <li>Supplier later sends an e-Document to update product information previously sent to a buyer including new products, amended products and cancelled products.</li> </ul>
<b>Scenario B</b>	Supplier sends e-Document to a buyer about grade code products with key product attributes to establish uniqueness. <ul style="list-style-type: none"> <li>Supplier later sends an e-Document to update these products including new products, amended products and cancelled products.</li> </ul>
<b>Scenario C</b>	Supplier sends e-Document to a buyer about grade code products with key product attributes to establish uniqueness. <ul style="list-style-type: none"> <li>Supplier later sends an e-Document to update these products using the ProductAttributesStatusType 'Replace'.</li> </ul>
<b>Scenario D</b>	Sending large amounts of product information using several ProductAttributes messages.
<b>Scenario E</b>	Buyer organisation sends information about their product codes to a supplier defining how they cross reference to the supplier's product codes after receiving the product information from the supplier.
<b>Scenario F</b>	Product cross-reference done by third party after receiving the product information from the buyer and the supplier. Third party does the cross-reference and sends the information to the buyer and the supplier.
<b>Scenario G</b>	Supplier sends a price list to a buyer.

### Scenario A

<b>e-Document</b>	ProductAttributes
<b>Type</b>	ProductProperties
<b>Status</b>	Original
<b>Scenario</b>	Supplier sends e-Document with information about SKU products supplied to a buyer and



## ProductAttributes papiNet Standard - Version 2.31

	Supplier later sends an e-Document to update SKU product information previously sent to a buyer including new products, amended products and cancelled products.
<b>Outcome</b>	Products are stored in the buyer's ERP system. Product information in buyer's system is updated in the buyer's ERP system.
<b>Initiator</b>	Supplier
<b>Receiver</b>	Buyer
<b>Preconditions</b>	This Use case assumes that the Supplier and Buyer have previously agreed to exchange product information using the ProductAttributes e-Document, including also the details on how an SKU is defined.
<b>XML File</b>	The name of any sample file.
<b>Trigger</b>	The trigger for this e-Document has to be agreed in advance between the Supplier and Buyer.
<b>Step 1.</b>	Supplier sends a ProductAttributes e-Document to the buyer.  Buyer extracts the information he requires from the e-Document and stores it in his ERP system.  Statuses sent within this e-Document: <ul style="list-style-type: none"> <li>• ProductAttributesType = 'ProductProperties'</li> <li>• ProductAttributesStatusType = 'Original'</li> <li>• ProductAttributesLineItem - ActionType = 'Original'</li> </ul>
<b>Step 2</b>	Buyer updates his information system.  Buyer looks up the information using ProductAttributesIdentifier together with SKU and updates the product information depending on ProductAttributesLineItem->ActionType: <ul style="list-style-type: none"> <li>• 'Original' - Buyer adds the SKU to his ERP system</li> <li>• 'Amend' - Buyer amends the SKU in his ERP system</li> <li>• 'Cancel' - Buyer deletes the SKU from his ERP system</li> <li>• Statuses sent within this e-Document:</li> <li>• ProductAttributesType = 'ProductProperties'</li> </ul>

## ProductAttributes papiNet Standard - Version 2.31

	<ul style="list-style-type: none"> <li>ProductAttributesStatusType = 'Amend'</li> <li>ProductAttributesLineItem -&amp;gt; ActionType = 'Original', 'Amend' or 'Cancel'</li> </ul>
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### Scenario B

<b>e-Document</b>	ProductAttributes
<b>Type</b>	ProductProperties
<b>Status</b>	Original
<b>Scenario</b>	<p>Supplier sends e-Document with product information using grade codes and with key characteristics to establish uniqueness to a buyer.</p> <ul style="list-style-type: none"> <li>Supplier later sends an e-Document to update this product information including new products, amended products and cancelled products.</li> </ul>
<b>Outcome</b>	<p>Products are stored in the buyer's ERP system.</p> <p>Product information in buyer's system is updated in the buyer's ERP system.</p>
<b>Initiator</b>	Supplier
<b>Receiver</b>	Buyer
<b>Preconditions</b>	<p>This Use case assumes that the Supplier and Buyer have previously agreed to exchange product information using the ProductAttributes e-Document. The products are defined using grade codes with key attributes to establish uniqueness. The grade codes and key attributes have to be defined in the trading partner agreement to make sure the buyer knows how to update the information.</p>
<b>Trigger</b>	<p>The trigger for this e-Document has to be agreed in advance between the Supplier and Buyer.</p>
<b>Step 1.</b>	<p>Supplier sends a ProductAttributes e-Document to the buyer.</p> <p>Buyer extracts the information he requires from the e-Document and stores it in his ERP system.</p> <p>Statuses sent within this e-Document:</p> <ul style="list-style-type: none"> <li>ProductAttributesType = 'ProductProperties'</li> <li>ProductAttributesStatusType = 'Original'</li> </ul>

## ProductAttributes papiNet Standard - Version 2.31

	<ul style="list-style-type: none"> <li>ProductAttributesLineItem -&amp;gt; ActionType = 'Original'</li> </ul>
<b>Step 2.</b>	<p>Supplier sends product information updates using the ProductAttributes e-Document to the Buyer who updates his information.</p> <p>Buyer looks up the product information using ProductAttributesIdentifier together with grade code and agreed key attributes and updates the information depending on ProductAttributesLineItem - ActionType</p> <ul style="list-style-type: none"> <li>'Original' - Buyer adds the SKU to his ERP system</li> <li>'Amend' - Buyer amends the SKU in his ERP system</li> <li>'Cancel' - Buyer deletes the SKU from his ERP system</li> </ul> <p>Statuses sent within this e-Document:</p> <ul style="list-style-type: none"> <li>ProductAttributesType = 'ProductProperties'</li> <li>ProductAttributesStatusType = 'Amend'</li> <li>ProductAttributesLineItem -&amp;gt; ActionType = 'Original', 'Amend' or 'Cancel'</li> </ul>

### Scenario C

<b>e-Document</b>	ProductAttributes
<b>Type</b>	ProductProperties
<b>Status</b>	Replace
<b>Scenario</b>	<p>Supplier sends e-Document with product information using grade codes and with key characteristics to establish uniqueness to a buyer and</p> <ul style="list-style-type: none"> <li>Supplier sends an e-Document to replace product information previously sent to a buyer defined by SenderProductAttributesIdentifier</li> </ul>
<b>Outcome</b>	<p>Supplier sends e-Document with product information using grade codes and with key attributes to establish uniqueness to a buyer. The Buyer stores the information in his ERP system.</p> <ul style="list-style-type: none"> <li>Supplier sends an update to previously products using the replace function. The</li> </ul>

## ProductAttributes papiNet Standard - Version 2.31

	product information defined by SenderProductAttributesIdentifier is replaced in the buyer's ERP system.
<b>Initiator</b>	Supplier
<b>Receiver</b>	Buyer
<b>Preconditions</b>	This Use case assumes that the Supplier and Buyer have previously agreed to exchange product information using the ProductAttributes e-Document. The products are defined using grade codes with key attributes to establish uniqueness. The parties have also agreed to use the 'Replace' status code to handle updates. The grade codes and key attributes have to be defined in the trading partner agreement
<b>Trigger</b>	The trigger for this e-Document has to be agreed in advance between the Supplier and Buyer, e.g. time dependent, when changes are made to Supplier product data.
<b>Step 1.</b>	Supplier sends a ProductAttributes e-Document to the buyer.  Buyer extracts the information he requires from the e-Document and stores it in his ERP system.  Statuses sent within this e-Document: <ul style="list-style-type: none"> <li>• ProductAttributesType = 'ProductProperties'</li> <li>• ProductAttributesStatusType = 'Original'</li> <li>• ProductAttributesLineItem -&amp;gt; ActionType = 'Original'</li> </ul>
<b>Step 2.</b>	Buyer updates his information system.  Buyer looks up the product information using ProductAttributesIdentifier and deletes the product information from his ERP system. After this the buyer adds the products defined in ProductAttributesLineItem to his ERP system <ul style="list-style-type: none"> <li>• ProductAttributesType = 'ProductProperties'</li> <li>• ProductAttributesStatusType = 'Replace'</li> <li>• ProductAttributesLineItem -&amp;gt; ActionType = 'Original'.</li> </ul>

### Scenario D

<b>e-Document</b>	ProductAttributes
-------------------	-------------------

## ProductAttributes papiNet Standard - Version 2.31

<b>Type</b>	ProductProperties
<b>Status</b>	Original / Amended
<b>Scenario</b>	The number of product definitions that the Supplier has to send to the Buyer is so big that sending the information in only one e-Document would result in such a large e-Document that the Buyer's receiving system cannot handle it. It has thus been agreed that the information is divided into manageable portions using several messages referring the same ProductAttributesIdentifier.
<b>Outcome</b>	Supplier sends e-Document with product information using grade codes and with key attributes to establish uniqueness to a buyer. <ul style="list-style-type: none"> <li>The product information in subsequent messages identified by the same SenderProductAttributesIdentifier is added to the Buyer's ERP system.</li> </ul>
<b>Initiator</b>	Supplier
<b>Receiver</b>	Buyer
<b>Preconditions</b>	This Use case assumes that the Supplier and Buyer have previously agreed to exchange product information in this manner, i.e. all the product information is not sent in one e-Document, but divided into manageable portions as defined in the trading party agreement.
<b>Trigger</b>	The trigger for this e-Document has to be agreed in advance between the Supplier and Buyer, e.g. time dependent, when changes are made to Buyer's or Supplier's product data.
<b>Step 1.</b>	Buyer sends an e-Document including his product codes and also sends the product details that define the product.  Statuses sent within this e-Document: <ul style="list-style-type: none"> <li>ProductAttributesType = 'ProductProperties'</li> <li>ProductAttributesStatusType = 'Original'</li> <li>ProductAttributesLineItem -&amp;gt; ActionType = 'Original'</li> </ul>
<b>Step 2.</b>	Supplier updates his information system

## ProductAttributes papiNet Standard - Version 2.31

	<p>Supplier extracts the information he requires from the e-Document and stores it in his ERP system.</p> <p>Statuses sent within this e-Document:</p> <ul style="list-style-type: none"> <li>• ProductAttributesType = 'ProductProperties'</li> <li>• ProductAttributesStatusType = 'Amended'</li> <li>• ProductAttributesLineItem -&amp;gt; ActionType = 'Original'</li> </ul>
--	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### Scenario E

<b>e-Document</b>	ProductAttributes
<b>Type</b>	ProductCrossReference
<b>Status</b>	Original
<b>Scenario</b>	After receiving the product information from the supplier the buyer sends information about his product codes to the defining his product codes cross-reference to the supplier's product codes.
<b>Outcome</b>	The supplier updates his product code cross-reference information using the information sent by the buyer.
<b>Initiator</b>	Buyer
<b>Receiver</b>	Supplier
<b>Preconditions</b>	This Use case assumes that the Supplier and Buyer have previously agreed to exchange product code cross-reference information in this manner.
<b>Trigger</b>	The trigger for this e-Document has to be agreed in advance e.g. time dependent, when changes are made to Buyer's or Supplier's product data.
<b>Step 1.</b>	<p>The Supplier sends his product information to the Buyer.</p> <p style="text-align: center;">The Buyer updates the information sent by the Supplier with his product codes.</p>
<b>Step 2.</b>	The Buyer sends the cross-reference information to the supplier. The Buyers and Supplier product codes are included in the e-Document by repeating the ProductIdentifier element including both the Supplier's and the Buyer's product

## ProductAttributes papiNet Standard - Version 2.31

	<p>codes.</p> <p>The supplier updates his product code cross-reference information using the information sent by the buyer. Later, when receiving for instance orders, where the Buyer references his product code, the Supplier can use the cross-reference information to match the Buyer's product code to his own product code.</p>
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### Scenario F

<b>e-Document</b>	ProductAttributes
<b>Type</b>	ProductCrossReference
<b>Status</b>	Original
<b>Scenario</b>	Both the Supplier and the Buyer send their product information to a third party. The third party maps the product codes of the Supplier and Buyer to create the cross-reference between the product codes. The third party returns the cross-reference information to the Supplier and Buyer.
<b>Outcome</b>	The Supplier and Buyer update their product code cross-reference information using the information sent by the third party.
<b>Initiator</b>	Buyer
<b>Receiver</b>	Supplier
<b>Preconditions</b>	This Use case assumes that the Supplier, Buyer, and the third party have previously agreed to exchange product code cross-reference information in this manner.
<b>Trigger</b>	The trigger for this e-Document has to be agreed in advance between the Supplier, Buyer, and third party, e.g. time dependent or when changes are made to Buyer's or Supplier's product data.
<b>Step 1.</b>	The Supplier and Buyer send their product information to the third party. The third party creates the cross-reference between the product codes.
<b>Step 2.</b>	The third party sends the cross-reference

## ProductAttributes papiNet Standard - Version 2.31

	<p>information to the Supplier and Buyer. The Buyers and Supplier product codes are included in the e-Document by repeating the ProductIdentifier element including both the Supplier's and the Buyer's product codes.</p> <p>The Supplier and Buyer update their product code cross-reference information using the information sent by the third party. Later, when receiving for instance orders, where the Buyer references his product code, the Supplier can use the cross-reference information to match the Buyer's product code to his own product code.</p>
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### Scenario G

<b>e-Document</b>	ProductAttributes
<b>Type</b>	PriceList
<b>Status</b>	Original
<b>Scenario</b>	Supplier sends a pricelist to a Buyer.
<b>Outcome</b>	Buyer stores the price information in his ERP system.
<b>Initiator</b>	Supplier
<b>Receiver</b>	Buyer
<b>Preconditions</b>	This Use case assumes that the Supplier and Buyer have previously agreed to exchange price information in this manner.
<b>Trigger</b>	The trigger for this e-Document has to be agreed in advance between the Supplier and Buyer, e.g. time dependent or when changes are made to price information.
<b>Step 1.</b>	<p>Supplier sends product information including enough key attributes to enable the supplier to also include the price information.</p> <p style="text-align: center;">The Buyer updates his ERP system with the price information.</p>