



papiNet

**Global Transaction Standards
for the Paper Supply Chain**

ProductAttributes Documentation

Papinet Standard - Version 2.10

April 2003

ProductAttributes Documentation - papiNet Standard - Version 2.10

Copyright

Copyright 2000 – 2002 papiNet G.I.E (“papiNet”), International Digital Enterprise Alliance, Inc. (“IDEAlliance”), and American Forest & Paper Association, Inc. (“AF&PA”), 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 (formerly known as the European Paper Consortium for e-business - EPC), IDEAlliance (formerly known as the Graphic Communications Association - GCA), the parent organisation of IDEAlliance the Printing Industries of America (PIA), the American Forest and Paper Association (AF&PA), and the members of the papiNet Working Group (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.

Additional Copyright Information

Additional copyrights may be referenced throughout this document in the appropriate section.

ProductAttributes Documentation - papiNet Standard - Version 2.10

Table of Contents

Copyright	2
Use of Documents in papiNet Implementations	2
Additional Copyright Information	2
Message Documentation	5
The ProductAttributes Message	5
An Overview of the ProductAttributes Message	5
The Scope of the ProductAttributes Message.....	5
ProductAttributes Message Types.....	6
Business Rules for ProductAttributes Message.....	6
General ProductAttributes Business Rules.....	6
Processing the ProductAttributes Message	7
ProductAttributes Message types	7
Status Values Used When Processing the ProductAttributes Message.....	7
Possible Combination of Status Type on ProductAttributes Message	7
Message size	8
UML Diagram of Product Attributes.....	8
Overview of Product Attributes Schema structure.....	9
Graphical Representation of Product Attributes	10
ProductAttributes Schema Structure and Processing Logic	11
ProductAttributes Root	11
ProductAttributesHeader.....	12
ProductAttributesLineItem	13
Common Definition References - Elements	14
AdditionalText.....	14
EndUses.....	14
InformationalQuantity	14
MillParty.....	15
MonetaryAdjustment.....	16
OrderQuantities.....	16
OtherParty.....	17
PriceDetails.....	17
Product	18
ProductAttributesIssueDate.....	18
ProductAttributesProductURL.....	18
ProductAttributesReference.....	18
ReceiverParty.....	19
SenderParty.....	19
SenderProductAttributesDescription	19
SenderProductAttributesIdentifier	20
SenderProductAttributesName	20
ShipToCharacteristics	20
SupplierParty	20
TermsAndDisclaimers	21
TransactionHistoryNumber	21

ProductAttributes Documentation - papiNet Standard - Version 2.10

ValidityPeriod	21
Common Definition References – Attributes	22
AdjustmentType (Financial).....	22
MeasurementsAre	24
ProductAttributesLineItemStatusType	24
statusType.....	25
ProductAttributes Business Use Cases.....	27
Scenario A.....	28
Scenario B.....	30
Scenario C.....	32
Scenario D.....	34
Scenario E	36
Scenario F	37
Scenario G.....	39

ProductAttributes Documentation

papiNet Standard - Version 2.10

Message Documentation

The ProductAttributes Message

An Overview of the ProductAttributes Message

The purpose of the ProductAttributes message is to provide the means to communicate information about products between business partners.

The ProductAttributes message 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 message is also designed to enable exchange of product codes used in electronic trading. Once a buyer has received the information from a supplier he returns the information updated with the cross reference to his product codes.

The ProductAttributes message 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 messages, content detail, units of measure, and other message aspects.

The Scope of the ProductAttributes Message

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

The ProductAttributes message hierarchy consists of:

- The header level to define the issue date, identifier for the message (SenderProductAttributesIdentifier), sender, receiver, necessary additional parties, free text description, etc.
- The product attributes 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),

ProductAttributes Documentation

papiNet Standard - Version 2.10

- ◇ 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 Message Types

There are three message types for the ProductAttributes message:

- ProductProperties, which is used when sending a list of ProductAttributes
- PriceList, which is used when sending a pricelist
- ProductCrossReference, which is used in the exchange of ProductIdentifier cross-references for the product codes used by two trading partners

The use of these types is explained in more detail in the business scenarios later in this document.

Business Rules for ProductAttributes Message

General ProductAttributes Business Rules

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

Reference	Rule
PA001	The frequency and triggering of ProductAttributes message exchange is determined by agreement between trading partners.
PA002	The SenderProductAttributesIdentifier identifies the ProductAttributes message 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 Documentation

papiNet Standard - Version 2.10

Processing the ProductAttributes Message

ProductAttributes Message types

Level	Attribute	Value
Message	ProductAttributesType	<p>ProductProperties - This is used when sending a list of products.</p> <p>PriceList - This is used in the exchange of products together with related price information between business parties.</p> <p>ProductCrossReference - This is used in the exchange of ProductIdentifier cross reference information for the product codes used in electronic communication of business transactions between two trading partners.</p>

Status Values Used When Processing the ProductAttributes Message

The ProductAttributes message includes ProductAttributesStatusType, which indicates whether the message is:

- Original, which means that the message includes a new list of product attributes identifies by SenderProductAttributesIdentifier.
- Amended, which means that the message includes amendments to a list of products defined by SenderProductAttributesIdentifier (additions, amendments, or cancellations).
- Replace, which indicates that the message replaces the product information sent an earlier communication with the same SenderProductAttributesIdentifier .
- Cancelled, which means that the message includes a cancelled list of product attributes defined by the SenderProductAttributesIdentifier.

Possible Combination of Status Type on ProductAttributes Message

ProductAttributes-StatusType	ProductAttributesLineItem-StatusType
Original	Original
Amend	Amend Cancel Original
Cancel	N/A
Replace	Original

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

ProductAttributes Documentation

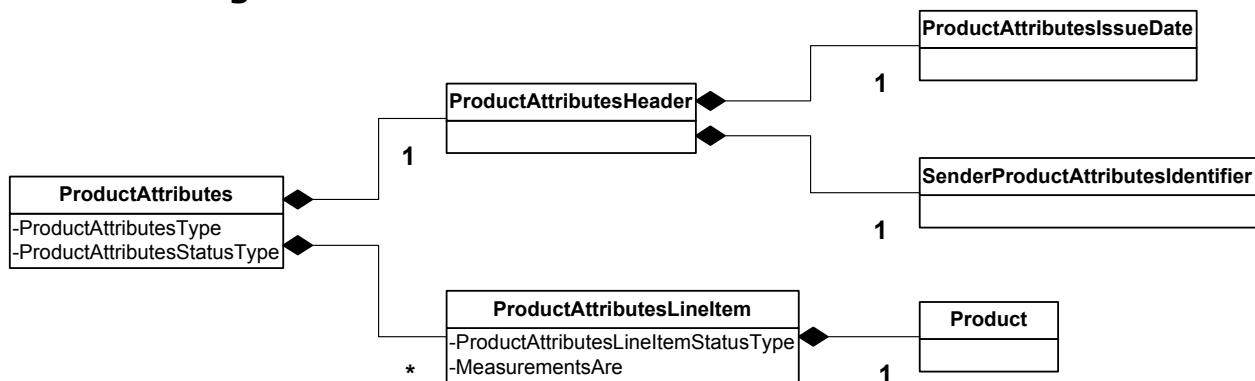
papiNet Standard - Version 2.10

Message size

It should be noted that sending many products in the same message may result in very large messages, which in turn may lead to problems when the receiver processes the message. It is therefore advisable to estimate the message size and check the message size during testing to find out what the maximum message 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 messages, using the same ProductAttributesIdentifier, 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.

UML Diagram of Product Attributes



ProductAttributes Documentation

papiNet Standard - Version 2.10

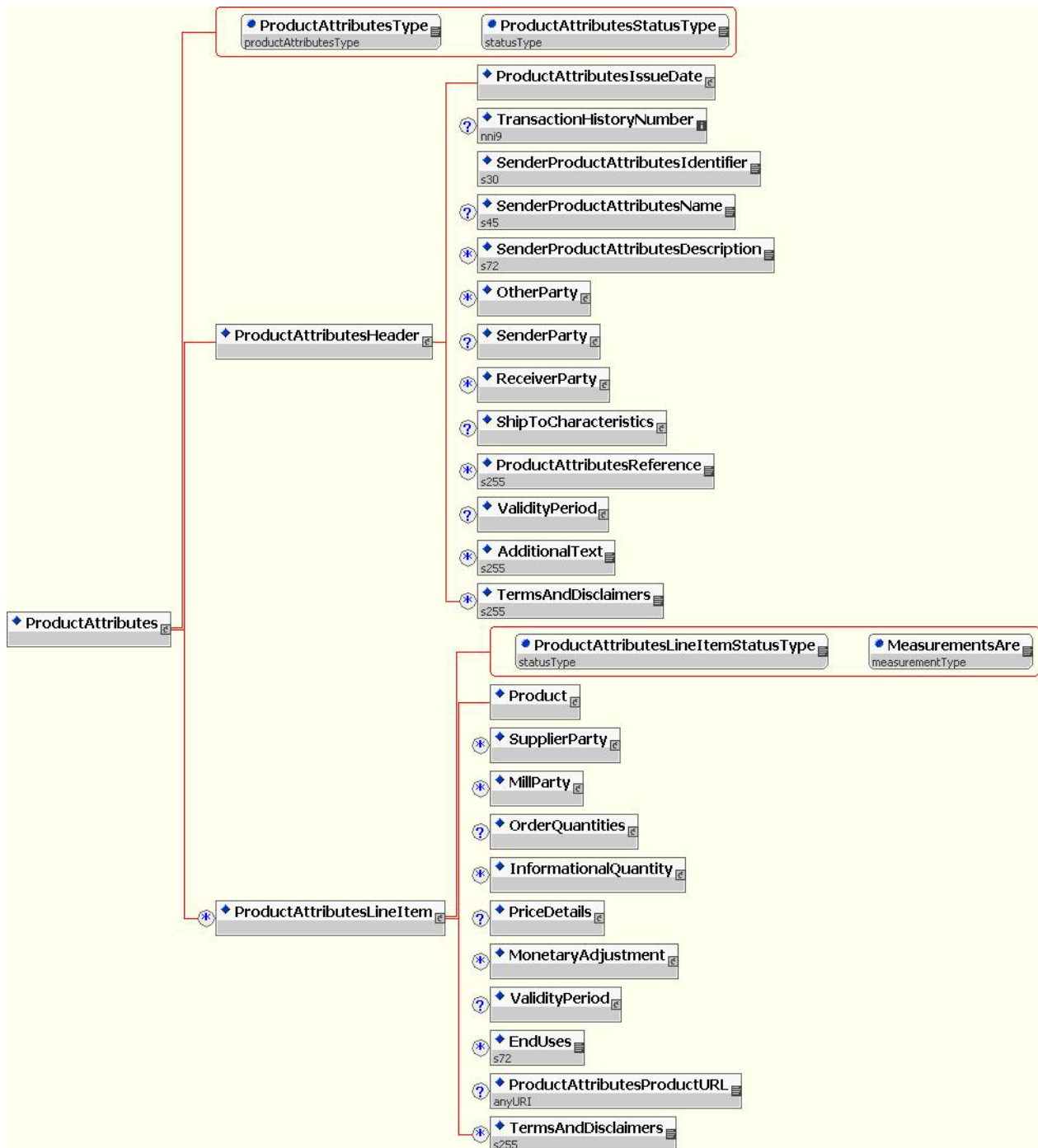
Overview of Product Attributes Schema structure

ProductAttributes	Root	Required	Single
@ProductAttributesType	Attribute	Required	Single
@ProductAttributesStatusType	Attribute	Required	Single
ProductAttributesHeader	Element	Required	Single
ProductAttributesIssueDate	Element	Required	Single
TransactionHistoryNumber	Element	Optional	Single
SenderProductAttributesIdentifier	Element	Required	Single
SenderProductAttributesName	Element	Optional	Single
SenderProductAttributesDescription	Element	Optional	Multiple
OtherParty	Element	Optional	Multiple
SenderParty	Element	Optional	Single
ReceiverParty	Element	Optional	Multiple
ShipToCharacteristics	Element	Optional	Single
ProductAttributesReference	Element	Optional	Multiple
ValidityPeriod	Element	Optional	Single
AdditionalText	Element	Optional	Multiple
TermsAndDisclaimers	Element	Optional	Multiple
ProductAttributesLineItem	Element	Required	Multiple
@ProductAttributesLineItemStatusType	Attribute	Required	Single
@MeasurementsAre	Attribute	Required	Single
Product	Element	Required	Multiple
SupplierParty	Element	Optional	Multiple
MillParty	Element	Optional	Multiple
OrderQuantities	Element	Optional	Single
InformationalQuantity	Element	Optional	Multiple
PriceDetails	Element	Optional	Single
MonetaryAdjustment	Element	Optional	Multiple
ValidityPeriod	Element	Optional	Single
EndUses	Element	Optional	Multiple
ProductAttributesProductURL	Element	Optional	Single
TermsAndDisclaimers	Element	Optional	Multiple

ProductAttributes Documentation

papiNet Standard - Version 2.10

Graphical Representation of Product Attributes



ProductAttributes Documentation

papiNet Standard - Version 2.10

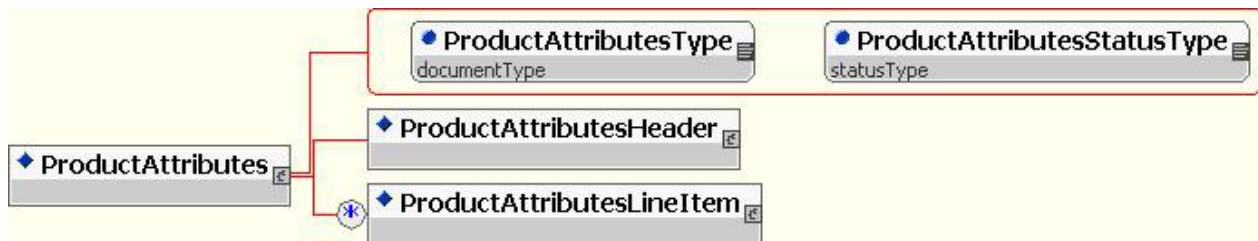
ProductAttributes Schema Structure and Processing Logic

This section provides a detailed graphical view of the Product Attributes Schema structure; the ProductAttributes root element, the ProductAttributesHeader, and the ProductAttributesLineItem. Discussions of the subordinates to ProductAttributes can be found in the Glossary document and a review of the data-types can be found in the Design document (available at www.papiNet.org). This section also contains a review of the processing logic that is special to the Product Attributes message.

The graphical display of the Schema contains occurrence indicators and data type information. These indicators appear to the left of the boxes in the schema graphic and they have the following meanings:

- (Blank) Required, single instance
- (+) Required, multiple instances
- (?) Optional, single instance
- (*) Optional, multiple instances

ProductAttributes Root



ProductAttributes has two required attributes:

- ProductAttributesType, with choices of:
 - ◇ PriceList
 - ◇ ProductCrossReference
 - ◇ ProductProperties
- ProductAttributesStatusType, with choices of:
 - ◇ Original
 - ◇ Amended
 - ◇ Replace
 - ◇ Cancelled

ProductAttributes contains the elements:

- [ProductAttributesHeader](#)
- [ProductAttributesLineItem](#)

ProductAttributes Documentation

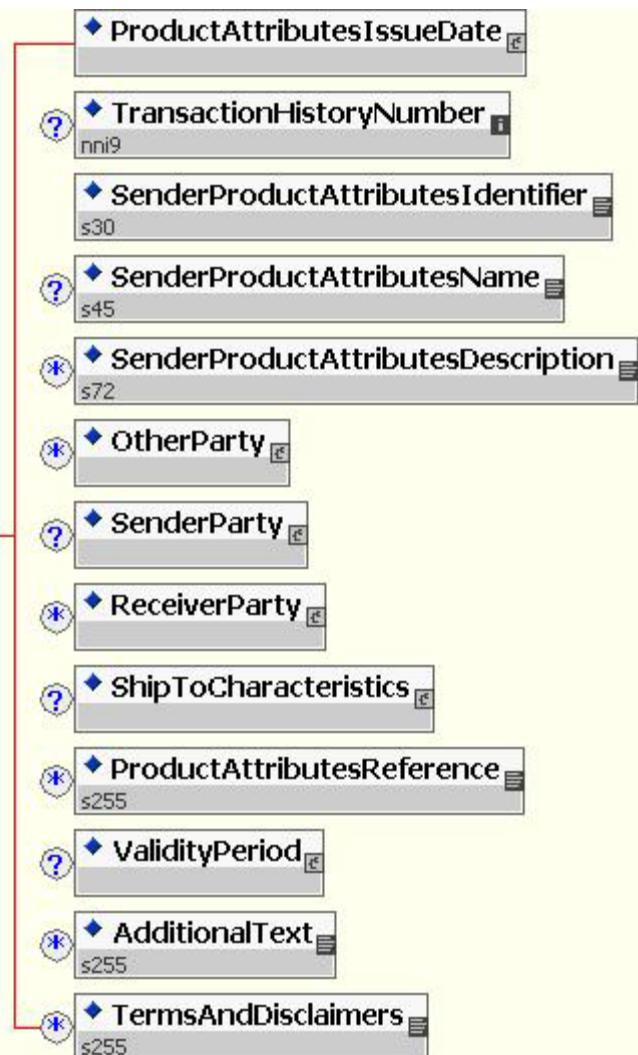
papiNet Standard - Version 2.10

ProductAttributesHeader

Information that applies to the entire ProductAttributes message.

ProductAttributesHeader contains the following elements:

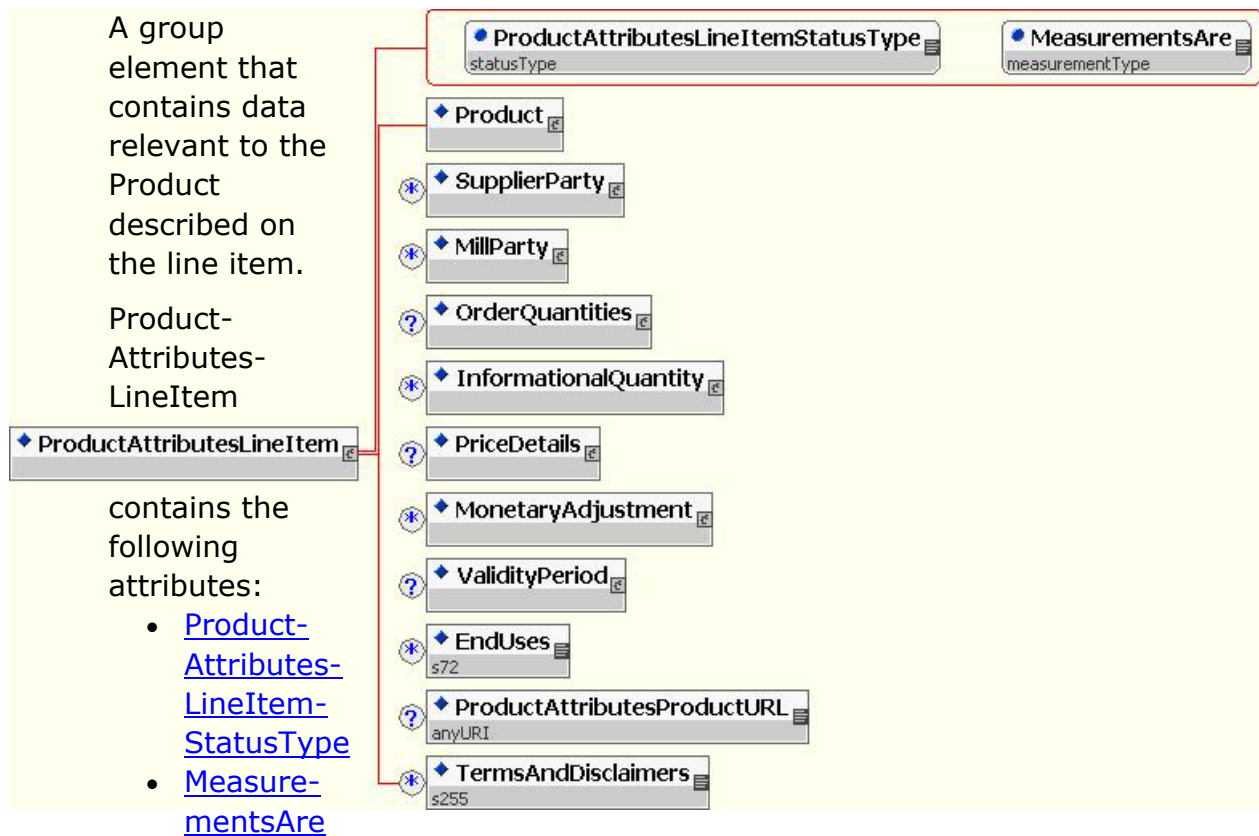
- [ProductAttributesIssueDate](#)
- [TransactionHistoryNumber](#)
- [SenderProductAttributesIdentifier](#)
- [SenderProductAttributesName](#)
- [SenderProductAttributesDescription](#)
- [OtherParty](#)
- [SenderParty](#)
- [ReceiverParty](#)
- [ShipToCharacteristics](#)
- [ProductAttributesReference](#)
- [ValidityPeriod](#)
- [AdditionalText](#)
- [TermsAndDisclaimers](#)



ProductAttributes Documentation

papiNet Standard - Version 2.10

ProductAttributesLineItem



ProductAttributesLineItem contains the following elements:

- [Product](#)
- [SupplierParty](#)
- [MillParty](#)
- [OrderQuantities](#)
- [InformationalQuantity](#)
- [PriceDetails](#)
- [MonetaryAdjustment](#)
- [ValidityPeriod](#)
- [EndUses](#)
- [ProductAttributesProductURL](#)
- [TermsAndDisclaimers](#)

ProductAttributes Documentation

papiNet Standard - Version 2.10

Common Definition References - Elements

AdditionalText

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. This element is repeatable and limited to the number of characters that would normally fit across a page.



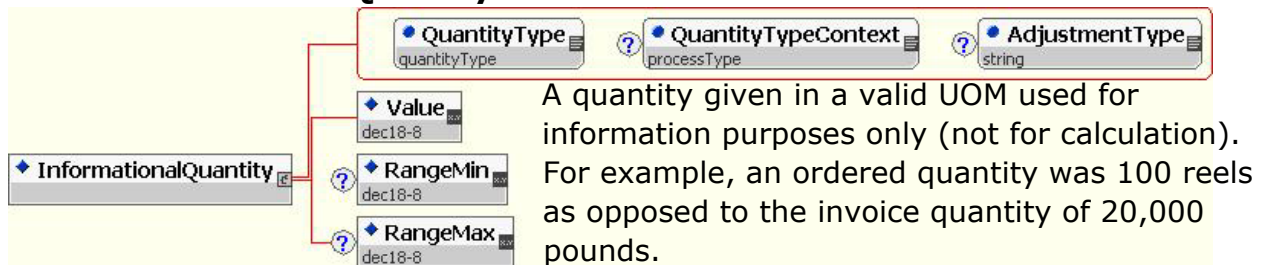
EndUses

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



EndUses contains the [Language](#) attribute.

InformationalQuantity



InformationalQuantity contains the following attributes:

- [QuantityType](#)
- [QuantityTypeContext](#)
- [AdjustmentType](#)

And the following elements:

- [Value](#)
- [RangeMin](#)
- [RangeMax](#)

ProductAttributes Documentation

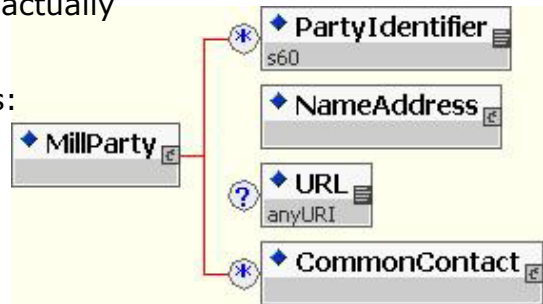
papiNet Standard - Version 2.10

MillParty

The organisation or business entity that actually produces the paper product.

MillParty contains the following elements:

- [PartyIdentifier](#)
- [NameAddress](#)
- [URL](#)
- [CommonContact](#)



ProductAttributes Documentation

papiNet Standard - Version 2.10

MonetaryAdjustment

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 they type of adjustment being communicated:

- [AdjustmentType](#)

MonetaryAdjustment contains the following elements:



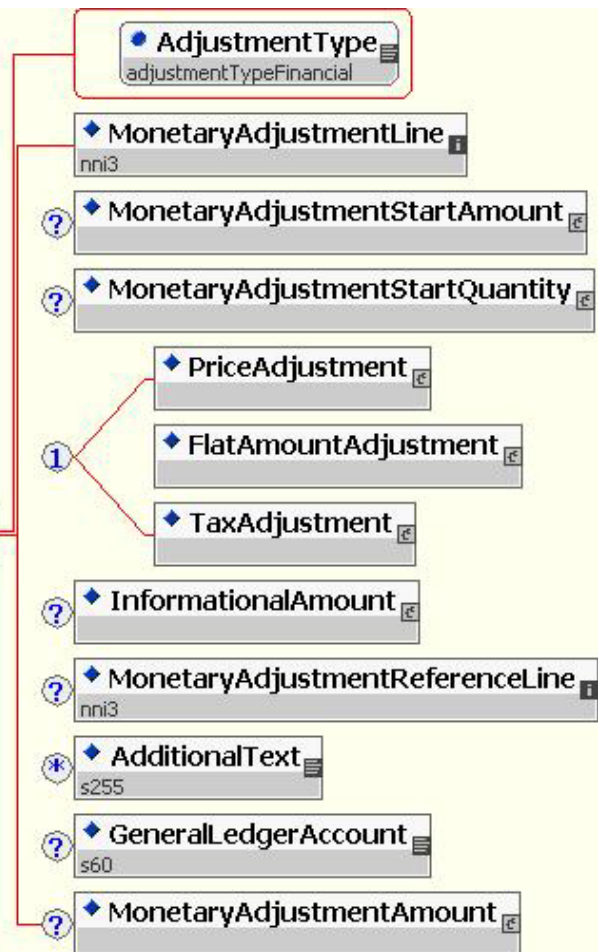
- [MonetaryAdjustmentLine](#)
- [MonetaryAdjustmentStartAmount](#)
- [MonetaryAdjustmentStartQuantity](#)

A choice of one of the following:

- [PriceAdjustment](#)
- [FlatAmountAdjustment](#)
- [TaxAdjustment](#)

And then the following elements:

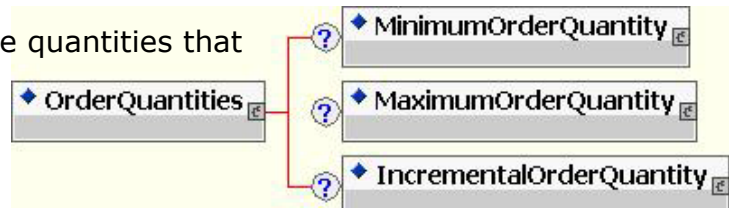
- [InformationalAmount](#)
- [MonetaryAdjustmentReferenceLine](#)
- [AdditionalText](#)
- [GeneralLedgerAccount](#)
- [MonetaryAdjustmentAmount](#)



OrderQuantities

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

OrderQuantities contains the following elements:



- [MinimumOrderQuantity](#)
- [MaximumOrderQuantity](#)
- [IncrementalOrderQuantity](#)

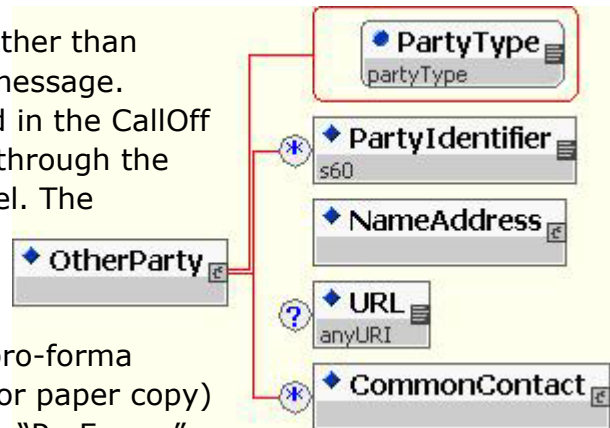
ProductAttributes Documentation

papiNet Standard - Version 2.10

OtherParty

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

- CallOff - If a carrier is specified in the CallOff message, this should be done through the CarrierParty on the header level. The PartyType "Carrier" should not be used.
- Invoice - The seller may generate multiple copies of a pro-forma invoice and a copy (electronic or paper copy) sent to the OtherParty that has "ProForma" assigned to the PartyType attribute.



OtherParty contains the [PartyType](#) attribute.

OtherParty contains the following elements:

- [PartyIdentifier](#)
- [NameAddress](#)
- [URL](#)
- [CommonContact](#)
 - ✧ Refer to [Party](#) for additional information

PriceDetails

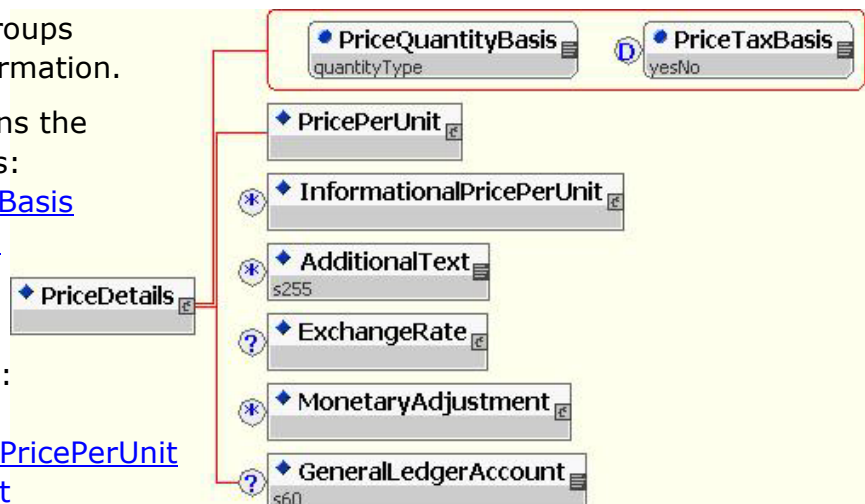
An element that groups together price information.

PriceDetails contains the following attributes:

- [PriceQuantityBasis](#)
- [PriceTaxBasis](#)

PriceDetails contains the following elements:

- [PricePerUnit](#)
- [InformationalPricePerUnit](#)
- [AdditionalText](#)
- [ExchangeRate](#)



ProductAttributes Documentation

papiNet Standard - Version 2.10

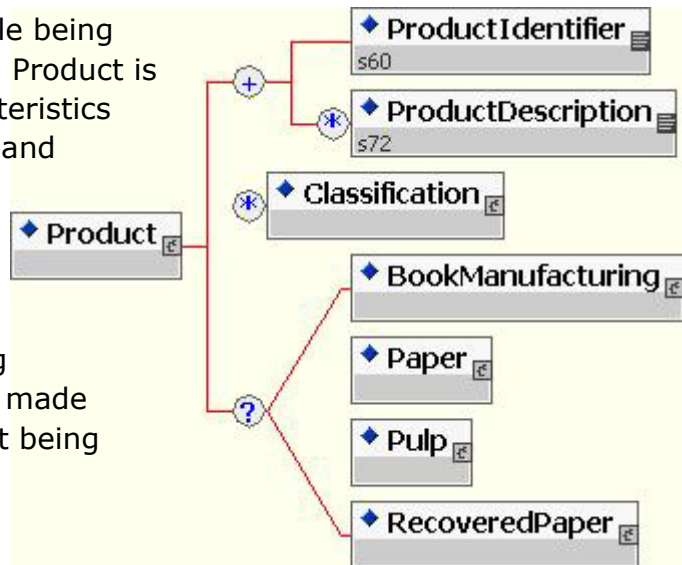
Product

A group item defining the article being ordered and its characteristics. Product is used to specify product characteristics organized by ProductIdentifier and Classification.

- [ProductIdentifier](#)
- [ProductDescription](#)
- [Classification](#)

A choice of one of the following "market-segment" elements is made based upon the type of product being ordered.

- [BookManufacturing](#)
- [Paper](#)
- [Pulp](#)
- [RecoveredPaper](#)



Pulp and Paper are the only products whose characteristics and conversion features have been implemented in papiNet. Recovered Paper and Book Manufacturing are in process and additional product types are under consideration based upon user input.

ProductAttributesIssueDate

The [Date](#) and [Time](#) the Product Attributes are issued. (Compare to [ValidityPeriod](#).)

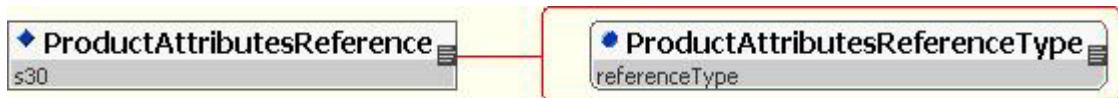


ProductAttributesProductURL

The web address of information pertinent to a particular product.



ProductAttributesReference



References that apply to the ProductAttributes message, contains the [ProductAttributesReferenceType](#) attribute.

ProductAttributes Documentation

papiNet Standard - Version 2.10

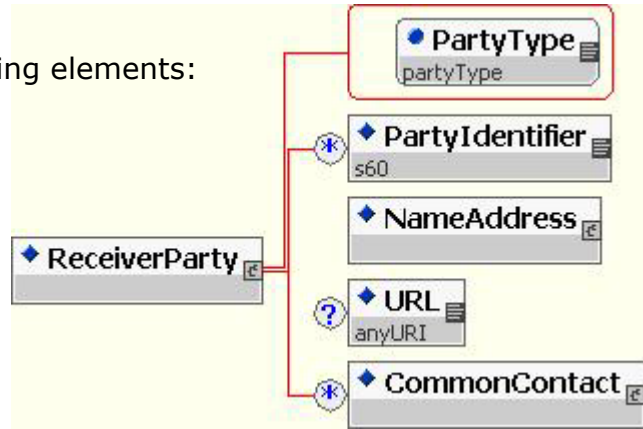
ReceiverParty

The party for whom the information (message) is intended, typed by the attribute [PartyType](#).

ReceiverParty contains the following elements:

- [PartyIdentifier](#)
- [NameAddress](#)
- [URL](#)
- [CommonContact](#)

Refer to [Party](#) (in the Glossary) for additional information.



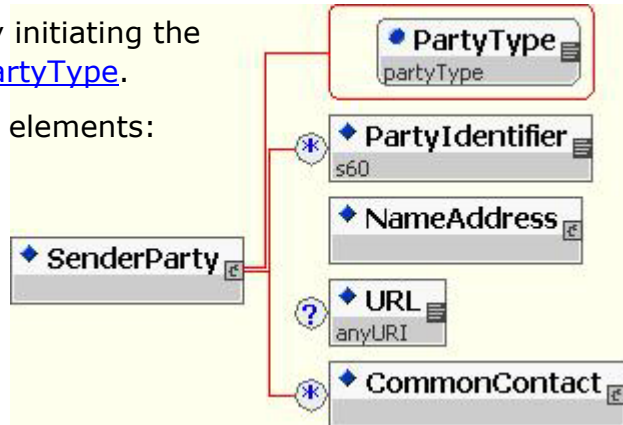
SenderParty

The organisation or business entity initiating the message, typed by the attribute [PartyType](#).

SenderParty contains the following elements:

- [PartyIdentifier](#)
- [NameAddress](#)
- [URL](#)
- [CommonContact](#)

Refer to [Party](#) (in the Glossary) for additional information.



SenderProductAttributesDescription

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

SenderProductAttributesDescription contains the [Language](#) attribute.



ProductAttributes Documentation

papiNet Standard - Version 2.10

SenderProductAttributesIdentifier

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 messages with updates will use this same identifier.



SenderProductAttributesName

The name given to the set of products sent in the ProductAttributes message.



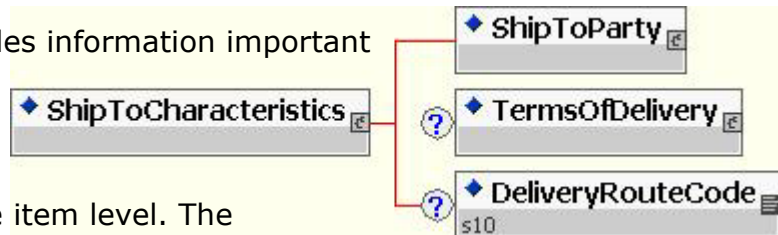
ShipToCharacteristics

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

ShipToCharacteristics may be referenced at

both the header and line item level. The reference at the header is required and acts as a default for the value at the line level, unless overridden at the line level. ShipToCharacteristics contains the following elements:

- [ShipToParty](#)
- [TermsOfDelivery](#)
- [DeliveryRouteCode](#)

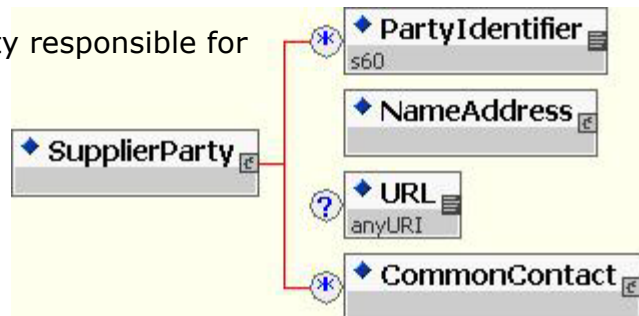


SupplierParty

The organisation or business entity responsible for providing the product.

SupplierParty contains the following elements:

- [PartyIdentifier](#)
- [NameAddress](#)
- [URL](#)
- [CommonContact](#)



Refer to [Party](#) for additional information

ProductAttributes Documentation

papiNet Standard - Version 2.10

TermsAndDisclaimers

A repeatable element that contains legal information with an indication of what the [Language](#) is.



TransactionHistoryNumber

A sequential number that keeps track of the version of any document being sent by the message originator.



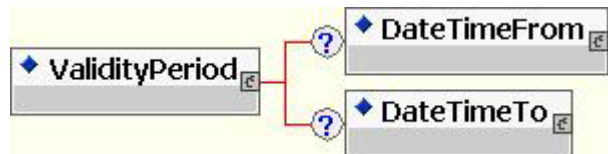
OrderConfirmation– Used to associate an OrderConfirmation message with a specific PurchaseOrder message. This is the same number that was sent in the original purchase order.

ValidityPeriod

The validity period for a blanket purchase order or product attributes definition.

ValidityPeriod contains the following elements:

- [DateTimeFrom](#)
- [DateTimeTo](#)



ProductAttributes Documentation

papiNet Standard - Version 2.10

Common Definition References – Attributes

AdjustmentType (Financial)

Describes the type of adjustment applied to an element. An adjustment can be either a value expressed in a currency or a percentage.

Used in:

- MonetaryAdjustment

Choices:

- **CancellationCharge** - A charge applied because a cancellation was late (after the last date of change) and all or part of the ordered product has already been made.
- **CashDiscount** - A reduction due to a discount applied.
- **ChargesForward** - A pending invoice amount brought forward from a previous invoice.
- **ClaimAdjustment** - An adjustment related to a filed complaint.
- **Commission** - Commission costs paid to the trading agency or trader.
- **ContractAllowance** - An allowance for a volume discount, usually because the agreed amount has been exceeded.
- **DeliveryNonConformanceAllowance** - An allowance due to non-conformance with the purchase agreement regarding a delivery.
- **EarlyShipAllowance** - An allowance due to a delivery that took place earlier than planned. The seller may be charged a penalty. If this is found before issuing the invoice, it can be handled through this allowance. If found later, it will be handled through a credit note.
- **Environmental** - An allowance applied due to environmental regulations.
- **ExpeditedShipmentCharge** - An extra cost charged due to unexpected or emergency requirements (for example, breaking off a run in the mill to reach the production date for a rush order).
- **FreightAllowance** - The freight cost allowed by the mill. This amount is negative, and will be deducted from the invoice (U.S. business practice, in the European Union (EU), use only Freight Charge).
- **FreightCharge** - The actual cost incurred to deliver the product. This amount is positive, and will be added to the invoice.
- **Inspection** - The cost associated with performing an inspection of the product.
- **InterestCharge** - A charge applied for interest on overdue payments.

ProductAttributes Documentation

papiNet Standard - Version 2.10

- **Other** - Any other adjustment not covered by the attribute list.
- **PriceCorrection** - An adjustment associated with a correction of the product's price.
- **ProductionSetUpCharge** - A special charge applied due to specific buyer requests.
- **Provision** - The act of making ready or equipping for use.
- **Rebate** - A rebate accepted by the buyer and seller based on the agreed price and volume.
- **ReturnedLoadAllowance** - An allowance applied due to the return of the product.
- **ReturnLoadCharge** - A charge applied due to the return of the product.
- **SpecialDeliveryCharge** - A special charge for delivering the product differently from the usual or agreed-to method.
- **SpecialHandlingCharge** - A special charge for handling the product differently from the usual or agreed-to method.
- **SpecialPackagingCharge** - A special charge for packaging the product differently from the usual or agreed-to method, or if packaging as requested is out of spec.
- **StopOffAllowance** - An allowance for a delivery stop-off.
- **StopOffCharge** - A charge for a delivery stop-off.
- **StorageCharge** - An extra cost charged for storing the product on location (in the plant).
- **Tax** - An adjustment that reflects a tax item.
- **TestingCharge** - A charge related to testing a new paper grade.
- **TradeDiscount** - A discount based on the terms of the trade given on the agreed price. Usually a percentage.
- **TransferCharge** - A freight charge incurred by the goods transfer from one location to another location not previously agreed upon.
- **TrialDiscount** - A discount applied for a trial run of a new or improved product.
- **VolumeDiscount** - A rebate accepted by the buyer and the seller based on the agreed volume.

ProductAttributes Documentation

papiNet Standard - Version 2.10

MeasurementsAre

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.

Used in:

- ProductAttributesLineItem

Choices:

- **Discrete** – 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.

ProductAttributesLineItemStatusType

Defines the status of the ProductAttributesLineItem

Used in:

- ProductAttributesLineItem

Choices:

ProductAttributesLineItemStatusType uses a restricted set of the [statusType](#) data type.

- [Accepted](#)
- [Cancelled](#)
- [Original](#)

ProductAttributes Documentation

papiNet Standard - Version 2.10

statusType

Acts as a flag indicating what processing may be required.

- **Accepted** - Either the seller or the buyer accepts the header information.
- **Amended** - Either the seller or the buyer has changed the header information.
- **Cancelled** - The buyer has cancelled a line item that was sent in a previous message. Lines that have been cancelled are not included in Totals on the Summary levels of the message.
- **Counterproposal** - The sender is presenting a different proposal.
- **New** - This is the first time the message has been sent.
- **NoAction** - This call-off header has not been amended and thereby requires no action.
- **Original** - The message information is the first version of that information.
- **Pending** - The seller will update the call-off line item at a later time.
- **Rejected** - The header information is rejected.
- **Replaced** - The entire message referred to has been replaced and the receiver should revalidate the information in their system based upon the entire message contents received.

ProductAttributes Documentation
papiNet Standard - Version 2.10

ProductAttributes Documentation

papiNet Standard - Version 2.10

ProductAttributes Business Use Cases

<u>Scenario A</u>	Supplier sends a message with information about products supplied to a buyer. <ul style="list-style-type: none"> • Supplier later sends a message to update product information previously sent to a buyer including new products, amended products and cancelled products.
<u>Scenario B</u>	Supplier sends message to a buyer about grade code products with key product attributes to establish uniqueness. <ul style="list-style-type: none"> • Supplier later sends a message to update these products including new products, amended products and cancelled products.
<u>Scenario C</u>	Supplier sends message to a buyer about grade code products with key product attributes to establish uniqueness. <ul style="list-style-type: none"> • Supplier later sends a message to update these products using the ProductAttributesStatusType 'Replace'.
<u>Scenario D</u>	Sending large amounts of product information using several ProductAttributes messages.
<u>Scenario E</u>	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.
<u>Scenario F</u>	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.
<u>Scenario G</u>	Supplier sends a price list to a buyer.

ProductAttributes Documentation

papiNet Standard - Version 2.10

Scenario A

Message	ProductAttributes
Type	ProductProperties
Status	Original
Business scenario	<p>Supplier sends message with information about SKU products supplied to a buyer and Supplier later sends a message to update SKU product information previously sent to a buyer including new products, amended products and cancelled products.</p> <pre> sequenceDiagram participant S as Supplier : SenderParty participant B as Buyer : ReceiverParty S->>B: Receive ProductAttributes Message() B-->S: BusinessAcknowledgement S->>B: Receive Amended ProductAttributes Message() B-->S: BusinessAcknowledgement </pre>
Outcome	<p>Products are stored in the buyer's ERP system. Product information in buyer's system is updated in the buyer's ERP system.</p>
Initiator	Supplier
Receiver	Buyer
Pre-conditions	<p>This Use case assumes that the Supplier and Buyer have previously agreed to exchange product information using the ProductAttributes message, including also the details on how an SKU is defined.</p>
Trigger	<p>The trigger for this message has to be agreed in advance between the Supplier and Buyer.</p>
Step 1	Supplier sends a ProductAttributes message to the buyer.
	<p>Buyer extracts the information he requires from the message and stores it in his ERP system. Statuses sent within this message:</p> <ul style="list-style-type: none"> • ProductAttributesType = 'ProductProperties' • ProductAttributesStatusType = 'Original' • ProductAttributesLineItem -> ActionType = 'Original'
Step 2	Buyer updates his information system
	Buyer looks up the information using ProductAttributesIdentifier together with SKU and updates the product information depending on

ProductAttributes Documentation papiNet Standard - Version 2.10

	<p>ProductAttributesLineItem->ActionType:</p> <ul style="list-style-type: none">• 'Original' - Buyer adds the SKU to his ERP system• 'Amend' - Buyer amends the SKU in his ERP system• 'Cancel' - Buyer deletes the SKU from his ERP system <p>Statuses sent within this message:</p> <ul style="list-style-type: none">• ProductAttributesType = 'ProductProperties'• ProductAttributesStatusType = 'Amend'• ProductAttributesLineItem -> ActionType = 'Original', 'Amend' or 'Cancel'
--	---

ProductAttributes Documentation papiNet Standard - Version 2.10

Scenario B

Message	ProductAttributes
Type	ProductProperties
Status	Original
Business Scenario	<p>Supplier sends message with product information using grade codes and with key characteristics to establish uniqueness to a buyer.</p> <ul style="list-style-type: none"> Supplier later sends a message to update this product information including new products, amended products and cancelled products. <pre> sequenceDiagram participant S as Supplier : SenderParty participant B as Buyer : ReceiverParty S->>B: Receive ProductAttributes Message() B-->S: BusinessAcknowledgement S->>B: Receive Amended ProductAttributes Message() B-->S: BusinessAcknowledgement </pre>
Outcome	<p>Products are stored in the buyer's ERP system. Product information in buyer's system is updated in the buyer's ERP system.</p>
Initiator	Supplier
Receiver	Buyer
Pre-conditions	<p>This Use case assumes that the Supplier and Buyer have previously agreed to exchange product information using the ProductAttributes message. 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>
Trigger	<p>The trigger for this message has to be agreed in advance between the Supplier and Buyer.</p>
Step 1	<p>Supplier sends a ProductAttributes message to the buyer.</p> <p>Buyer extracts the information he requires from the message and stores it in his ERP system.</p> <p>Statuses sent within this message:</p> <ul style="list-style-type: none"> ProductAttributesType = 'ProductProperties' ProductAttributesStatusType = 'Original' ProductAttributesLineItem -> ActionType = 'Original'

ProductAttributes Documentation

papiNet Standard - Version 2.10

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

ProductAttributes Documentation papiNet Standard - Version 2.10

Scenario C

Message	ProductAttributes
Type	ProductProperties
Status	Replace
Scenario	<p>Supplier sends message with product information using grade codes and with key characteristics to establish uniqueness to a buyer and</p> <ul style="list-style-type: none"> Supplier sends a message to replace product information previously sent to a buyer defined by SenderProductAttributesIdentifier <pre> sequenceDiagram participant Supplier as Supplier : SenderParty participant Buyer as Buyer : ReceiverParty Supplier->>Buyer: Receive ProductAttributes Message() Buyer-->>Supplier: BusinessAcknowledgement Supplier->>Buyer: Receive Replaced ProductAttributes Message() Buyer-->>Supplier: BusinessAcknowledgement </pre>
outcome	<p>Supplier sends message 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"> Supplier sends an update to previously products using the replace function. The product information defined by SenderProductAttributesIdentifier is replaced in the buyer's ERP system.
Initiator	Supplier
Receiver	Buyer
Pre-conditions	<p>This Use case assumes that the Supplier and Buyer have previously agreed to exchange product information using the ProductAttributes message. 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</p>
Trigger	<p>The trigger for this message has to be agreed in advance between the Supplier and Buyer, e.g. time dependent, when changes are made to Supplier product data.</p>
Step 1	Supplier sends a ProductAttributes message to the buyer.
	Buyer extracts the information he requires from the message and stores it in his ERP system.

ProductAttributes Documentation papiNet Standard - Version 2.10

	Statues sent within this message: <ul style="list-style-type: none">• ProductAttributesType = 'ProductProperties'• ProductAttributesStatusType = 'Original'• ProductAttributesLineItem -> ActionType = 'Original'
Step 2	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">• ProductAttributesType = 'ProductProperties'• ProductAttributesStatusType = 'Replace'• ProductAttributesLineItem -> ActionType = 'Original'.

ProductAttributes Documentation papiNet Standard - Version 2.10

Scenario D

Message	ProductAttributes
type	ProductProperties
Status	Original / Amended
scenario	<p>The number of product definitions that the Supplier has to send to the Buyer is so big that sending the information in only one message would result in such a large message 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.</p> <pre> sequenceDiagram participant Buyer as Buyer : ReceiverParty participant Supplier as Supplier : SenderParty Buyer->>Supplier: Receive Amended ProductAttributes Message() Supplier->>Buyer: BusinessAcknowledgement Buyer->>Supplier: Receive ProductAttributes Message() Supplier->>Buyer: BusinessAcknowledgement </pre>
outcome	<p>Supplier sends message with product information using grade codes and with key attributes to establish uniqueness to a buyer and</p> <ul style="list-style-type: none"> The product information in subsequent messages identified by the same SenderProductAttributesIdentifier is added to the Buyer's ERP system.
Initiator	Supplier
Receiver	Buyer
Pre-conditions	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 message, but divided into manageable portions as defined in the trading party agreement.
Trigger	The trigger for this message 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.
Step 1	Buyer sends a message including his product codes and also sends the product details that define the product.
	<p>Statuses sent within this message:</p> <ul style="list-style-type: none"> ProductAttributesType = 'ProductProperties' ProductAttributesStatusType = 'Original'

ProductAttributes Documentation papiNet Standard - Version 2.10

	<ul style="list-style-type: none">ProductAttributesLineItem -> ActionType = 'Original'
Step 2	Supplier updates his information system
	Supplier extracts the information he requires from the message and stores it in his ERP system. Statuses sent within this message: <ul style="list-style-type: none">ProductAttributesType = 'ProductProperties'ProductAttributesStatusType = 'Amended'ProductAttributesLineItem -> ActionType = 'Original'

ProductAttributes Documentation papiNet Standard - Version 2.10

Scenario E

Message	ProductAttributes
type	ProductCrossReference
Status	Original
Scenario	<p>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.</p> <pre> sequenceDiagram participant Supplier as Supplier : SenderParty participant Buyer as Buyer : ReceiverParty Supplier->>Buyer: Receive ProductAttributes Message() Buyer-->>Supplier: BusinessAcknowledgement Buyer->>Supplier: Receive ProductAttributes Message() Buyer-->>Supplier: BusinessAcknowledgement </pre>
Outcome	The supplier updates his product code cross-reference information using the information sent by the buyer.
Initiator	Buyer
Receiver	Supplier
Pre-conditions	This Use case assumes that the Supplier and Buyer have previously agreed to exchange product code cross-reference information in this manner.
Trigger	The trigger for this message has to be agreed in advance e.g. time dependent, when changes are made to Buyer's or Supplier's product data.
Step 1	The Supplier sends his product information to the Buyer.
	The Buyer updates the information sent by the Supplier with his product codes.
Step 2	The Buyer sends the cross-reference information to the supplier. The Buyers and Supplier product codes are included in the message by repeating the ProductIdentifier element including both the Supplier's and the Buyer's product codes.
	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.

ProductAttributes Documentation

papiNet Standard - Version 2.10

Scenario F

Message	ProductAttributes
Type	ProductCrossReference
Status	Original
Scenario	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.
<pre> sequenceDiagram participant S as Supplier : SenderParty participant TP as 3rd Party : ReceiverParty participant B as Buyer : SenderParty S->>TP: Receive ProductAttributes Message B->>TP: Receive ProductAttributes Message() TP-->>S: BusinessAcknowledgement TP-->>B: BusinessAcknowledgement S->>TP: Receive ProductAttributes Message() B->>TP: Receive ProductAttributes Message() TP-->>S: BusinessAcknowledgement TP-->>B: BusinessAcknowledgement </pre>	
Use case outcome	The Supplier and Buyer update their product code cross-reference information using the information sent by the third party.
Initiator	Buyer
Receiver	Supplier
Pre-conditions	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.
Trigger	The trigger for this message 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.
Step 1	The Supplier and Buyer send their product information to the third party.
	The third party creates the cross-reference between the product codes.
Step 2	The third party sends the cross-reference information to the Supplier and Buyer. The Buyers and Supplier product codes are included in the message by repeating the ProductIdentifier element including both the Supplier's and the Buyer's product

ProductAttributes Documentation papiNet Standard - Version 2.10

	codes.
	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.

ProductAttributes Documentation

papiNet Standard - Version 2.10

Scenario G

Message	ProductAttributes
Type	PriceList
Status	Original
Business scenario	<p>Supplier sends a pricelist to a Buyer.</p> <pre> sequenceDiagram participant S as Supplier : SenderParty participant B as Buyer : ReceiverParty S->>B: Receive ProductAttributes Message B-->>S: BusinessAcknowledgement </pre>
Outcome	Buyer stores the price information in his ERP system.
Initiator	Supplier
Receiver	Buyer
Pre-conditions	This Use case assumes that the Supplier and Buyer have previously agreed to exchange price information in this manner.
Trigger	The trigger for this message has to be agreed in advance between the Supplier and Buyer, e.g. time dependent or when changes are made to price information.
Step 1	Supplier sends product information including enough key attributes to enable the supplier to also include the price information.
	The Buyer updates his ERP system with the price information.