

papiNet

papiNet Envelope – Version 1.0

The papiNet Standard

Documentation

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

December 2009

2009-12-01

**Interoperability and the papiNet
Envelope**

papiNet Envelope Version 1.0

The papiNet Standard

Copyright

Copyright 2000 – 2009 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

papiNet Envelope Version 1.0

The papiNet Standard

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.

papiNet Envelope Version 1.0

The papiNet Standard

Table of Contents

Envelope Documentation.....	5
Envelope Introduction.....	5
Interoperability Vision.....	5
Aspects of Interoperability.....	5
The Need for Coordination between Trading Partners.....	7
Selecting the Proper Environment.....	7
papiNet XML Business Document Creation	9
Creating Well-formed papiNet XML.....	9
Sample XML - Document in Envelope	10
papiNet Envelope Root Element	13
papiNetEnvelope	13
Primary Elements.....	14
PayloadInfo	14
Payload	15
Envelope Use Cases	16
XML Transmission Use Cases	16
Envelope Parsing Usage Scenario	19
papiNetEnvelope Supporting Elements	23
Attachment	23
AttachmentIdentifier [attribute]	23
Build [attribute]	23
BusinessDocument	23
BusinessReceiver	24
BusinessSender	24
Date.....	24
Day.....	25
Document	25
DocumentDate.....	29
DocumentHistoryNumber	29
DocumentName [attribute]	29
DocumentNumber	30
DocumentType [attribute]	30
ID.....	30
Month.....	30
NumberOfAttachments.....	30
PartyIdentifier	30
PartyIdentifierType [attribute]	31
Schema	32
SequenceNumber	32
Signature	32
TestFlag [attribute]	32
Time	33
TimeStamp	33
TransmissionInfo.....	33
TransmissionReceiver	34
TransmissionSender	34
Version [attribute].....	35
Year	35

papiNet Envelope Version 1.0

The papiNet Standard

Envelope Documentation

Envelope Introduction

The papiNet envelope is a light-weight wrapper that provides a common starting point for message services wishing to parse a papiNet message. The papiNet envelope does not replace the message service envelope but works in concert with all message service envelopes to expose business routing and identification information.

Interoperability Vision

The papiNet Standards Group has the vision of enterprises of any size and in any geographical location meeting and conducting the business of paper, printing, and publishing with each other through the exchange of XML based e-business documents. The intent is to define a neutral method (one that is open and non proprietary) for exchanging these electronic business documents. In addition to being neutral the exchange process has to guarantee safe, secure delivery.

This interoperability vision can be summarized in the following way:

- Participants in the messaging transfer process should be able to choose the technology they desire to use to communicate messages independent of other participants in the communications network.
 - Errors in transmission to the destination must be communicated.
 - Security must be assured:
 - Privacy - Protect against information being disclosed or revealed to any entity not authorized to have that information by permitting the use of encryption techniques.
 - Authentication - Authenticate the claimed identity of the originator.
 - Authorization - Protect against the threat that unknown entities enter into the system and ensures that an entity performs only authorized actions within the system.
 - Integrity - Protect against the threat that the value of a data item might be changed en route.
 - Non-repudiation - Protect against one party to a transaction or communication later falsely denying that the transaction or communication occurred.
-

Aspects of Interoperability

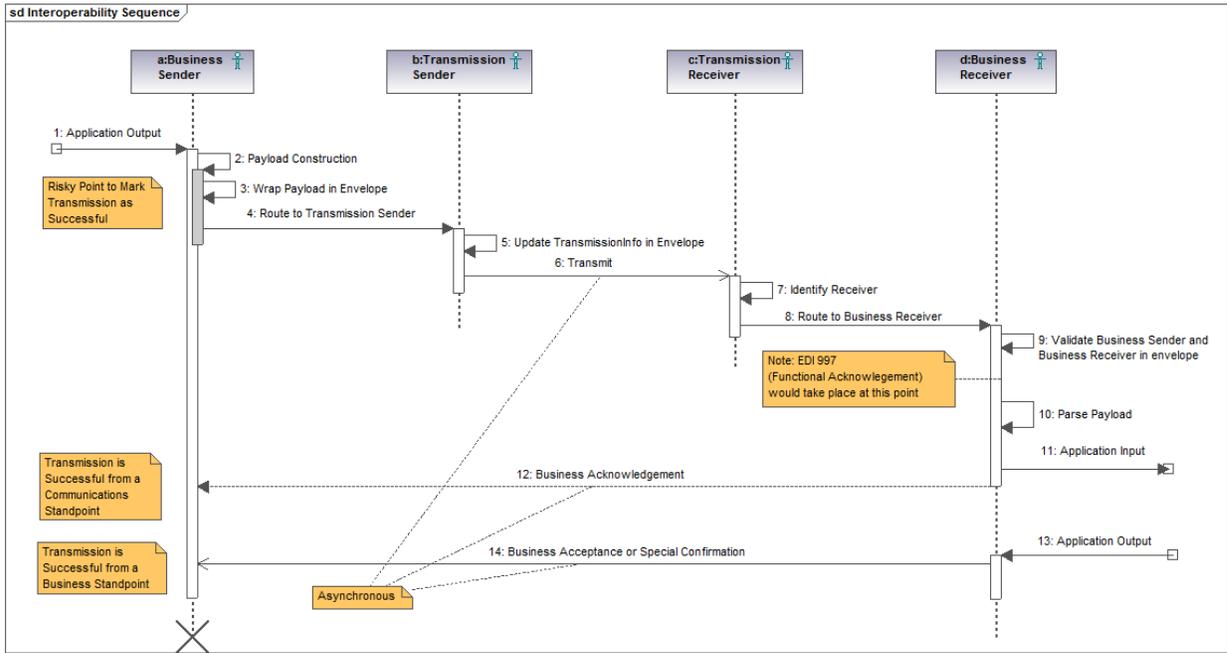
When we talk about e-business document communication there are many aspects to consider. The process starts with an individual reviewing the output from a business application and initiating some sort of action that can be communicated using a papiNet e-business document.

The papiNet e-business document is sent from your location to the recipient's location over the internet via one of several possible transfer methods. The payload is wrapped in an electronic envelope that provides

papiNet Envelope Version 1.0

The papiNet Standard

identification and supports security features.



The papiNet Payload

In the diagram above the papiNet e-business document, the papiNet payload, is created in Step 2, read in Step 10, and processed in Step 11. The standardized format of the papiNet payload provides a common view of the business document across trading partner boundaries.

The papiNet Envelope

The papiNet Standards Committee has found that there is a consistent group of information that is beneficial for XML processing programs to have readily available. Grouping this information into an envelope that contains the papiNet payload means that this information is always in the same location and provides a hook for the XML processing programs to use when processing a transmission.

Message Service and Transport Protocol Envelopes

We'll cover these two items together because, in many cases, the transport protocol envelope and the message service envelope are so closely related that it is hard to distinguish them.

In the papiNet world we deal with the HTTP, FTP, and SMTP transport protocols. From an allegorical standpoint you could think of the transport protocols in the following way:

- SMTP – as secure as your postal mail and with the same issue of junk mail.
- FTP – a package delivery service that delivers the package to your front door step but does not wait for someone to provide a signature of receipt. Sometimes they ring the doorbell. You can secure your front door step but you cannot ensure that a receipt is provided.

papiNet Envelope Version 1.0

The papiNet Standard

- HTTP/HTTPS – (think of HTTPS as an armoured car delivery) Depending on the nature of the message service you use HTTP can be implemented so that it provides delivery similar to a personal high-security courier who personally delivers the message to you, provides the security cipher, and guarantees that the message has not been tampered with and is from the party indicated. You can also run HTTP/HTTPS in such a way that it provides the same lower level of capability as with SMTP or FTP.

In the HTTP/HTTPS transport protocol environment there are several message protocols that you need to be aware of they are SOAP, AS2, and SOAP-ebXML. The level of functionality provided by the HTTP/HTTPS requires more coordination between the sender and recipient than is required by SMTP and FTP.

The Need for Coordination between Trading Partners

When two trading partners decide to exchange electronic business documents they must agree on the parameters for the exchange. Be aware that all communication and all messages do not need to be sent in the same way between all trading partners. You could be receiving messages in one form from a trading partner and sending them in another form. However, both of you must at least agree that you will be “looking out” for messages of a certain type.

Most companies like to reduce the complexity of their infrastructure by reducing the number of communications approaches they have to support. There are two competing factors that prevent a single uniform approach for all companies.

- First, there is the drive to the “lowest common denominator” which would be the solution that is easiest and cheapest to implement. Depending on who you are this could be email, ftp, phone, or fax.
- Secondly, there is the drive to the “highest level of security” which would be the HTTP/HTTPS with AS2, SOAP, or ebXML.

So, coordination is required between two trading partners. The first time you set-up your environment for one of the more robust solutions there will be some initial steps that you need to perform. These will be mostly one-time steps that will be applicable for subsequent implementations.

Selecting the Proper Environment

The number one or two question that we are continually asked is, “What message service should I use when communicating with my trading partners?” The answer depends on many different factors.

In order to achieve the security considerations mentioned at the beginning of this document our first recommendation is always going to be one of the HTTP oriented protocols. However, if security is less of a concern then other choices may be possible.

papiNet Envelope Version 1.0

The papiNet Standard

- Are you a large company, a medium sized company, or a small company?

Security considerations aside, a small company may feel very comfortable in using SMTP (e-mail) to send papiNet messages. In general, most medium and large companies use one of the HTTP oriented protocols. FTP is used by some medium and large companies as it provides a bit more reliability and security than e-mail however; it is not as robust as the HTTP oriented protocols.

- What are your trading partners doing? What are your customers requiring?

This point gets to the coordination that is required between the sender and receiver and the amount of investment that is required to implement any solution. Obviously you do not want to implement a solution that does not meet your security requirements but all the approaches require a degree of coordination.

- How sensitive is the information you are communicating?

While all the approaches provide a certain degree of security SMTP provides the lowest and HTTPS with AS2, SOAP, or ebXML provides the highest. Of course the coordination requirements are more extensive with AS2, SOAP, or ebXML.

papiNet Envelope Version 1.0

The papiNet Standard

papiNet XML Business Document Creation

Creating Well-formed papiNet XML

This document reviews a typical XML document, the sample XML that immediately follows. You might want to place this discussion next to the sample XML so that you can follow the discussion line-by-line. If you have any questions about this discussion you may address them to the papiNet Technical Director (<mailto:acolman@idealliance.org>).

Line 1:

The required XML element with attributes for version and encoding is the first line of the XML document. There is currently only a single version of xml and while there are several encoding approaches UTF-8 is the approach that is prescribed by the Internet Engineering Task Force (IETF). UTF-8 can handle every character, pictograph languages may take up more space but they can be represented.

Line 2:

The root element of the papiNet envelope (papiNetEnvelope) is shown on line 2. If you scan down to line 44 you will find a second "root element". The second root element is the root element of the papiNet payload, the business document, which is, in this case, "DeliveryMessage". These root elements share many characteristics, we'll cover them when we discuss lines 3, 4, 5, and 6. A similar discussion will take place with the DeliveryMessage root element. Note that line 2 starts with "<" and that a closing ">" does not show until line 6. Line 2 is an element and the entries on line 3, 4, 5, and 6 are attributes.

Line 3:

Line 3 indicates that a namespace (without a prefix) is being defined. By looking at the attribute value you should be able to determine that this is the namespace for the papiNet envelope.

Line 4:

Another namespace, this one with the prefix of "xsi" is being defined.

Line 5 & 6:

The namespace of "xsi" is used along with one of its attributes to communicate the location for the schema. The value of the xsi:schemaLocation (what's between the quotation marks) is entries, the first shows on line 5 and the second on line 6. In programming terminology this would be a name-value pair. The entry on line 5 is the name, which in this case is a namespace, and is the namespace that we defined on line 3. The entry on line 6 is the value, which in this case is a location on the internet, and is the location where papiNet stores the schema for the papiNetEnvelope. Unfortunately, to the uninitiated, line 5 & 6 are extremely confusing because while it is easy to understand that the schema location (the portion on line 6) would be a location on the internet it is very confusing as to why a namespace would look like a location on the internet. The reason why many namespaces look like web-addresses is because web-addresses are associated with domain names and domain names are a very easy way to ensure uniqueness and control

papiNet Envelope Version 1.0

The papiNet Standard

(there are other ways, as well).

While this example shows the use of the `xsi:schemaLocation` attribute you can still choose to use a schema stored at a different location. This is common practice in a production environment where you want to use a locally stored version (refer to Use Case Section). Regardless, it is strongly recommended by papiNet to include a `schemaLocation` in your XML document.

Line 7 through 43 and 94 through 97:

The `papiNetEnvelope` structure. The closing elements of the envelope are shown on lines 83, 84, and 85, thus the envelope "wraps around" the papiNet payload (the `BusinessDocument` or, in this case, a `DeliveryMessage`).

Line 44 through 94:

The papiNet payload, the `BusinessDocument` section, or in this case a `DeliveryMessage`.

Line 45 through 48:

The namespace and `schemaLocation` for the `DeliveryMessage` is shown in these lines.

Line 45 indicates that this `DeliveryMessage` lies in the `v2r40` namespace which is managed by papiNet. The manner in which this namespace is applied to the document means that this namespace definition applies to those items between lines 44 and 82 (replacing the envelope namespace). The reason this happens is because the namespace is specified on an element.

Sample XML - Document in Envelope

```
1. <?xml version="1.0" encoding="UTF-8"?>
2. <papiNetEnvelope
3.     xmlns="http://www.papinet.org/env/v1r00"
4.     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5.     xsi:schemaLocation="http://www.papinet.org/env/v1r00
6.         http://www.papinet.org/env/v1r00/papiNetEnvelope.xsd">
7. <PayloadInfo TestFlag="false">
8.     <TransmissionInfo>
9.         <SequenceNumber>0</SequenceNumber>
10.        <ID>123</ID>
11.        <TimeStamp>2006-02-03T15:00:00</TimeStamp>
12.        <TransmissionSender>
13.            <PartyIdentifier PartyIdentifierType="papiNetGlobalPartyIdentifier">
14.                1.3.6.1.4.1.13099
15.            </PartyIdentifier>
16.        </TransmissionSender>
17.        <TransmissionReceiver>
18.            <PartyIdentifier PartyIdentifierType="DunsNumber">
19.                14-315-8939</PartyIdentifier>
20.        </TransmissionReceiver>
```

papiNet Envelope Version 1.0

The papiNet Standard

```
20. </TransmissionInfo>
21. <BusinessSender>
22.   <PartyIdentifier PartyIdentifierType="papiNetGlobalPartyIdentifier">
23.     1.3.6.1.4.1.14100</PartyIdentifier>
24. </BusinessSender>
25. <BusinessReceiver>
26.   <PartyIdentifier PartyIdentifierType="DunsNumber">
27.     15-415-9939</PartyIdentifier>
28. </BusinessReceiver>
29. <Document DocumentName="DeliveryMessage">
30.   <DocumentNumber>001002456</DocumentNumber>
31.   <DocumentDate>
32.     <Date>
33.       <Year>2008</Year>
34.       <Month>03</Month>
35.       <Day>26</Day>
36.     </Date>
37.   </DocumentDate>
38. </Document>
39. <NumberOfAttachments>0</NumberOfAttachments>
40. <Schema Version="V2R40">
41.   DeliveryMessageV2R40.xsd
42. </Schema>
43. </PayloadInfo>
44. <Payload>
45.   <BusinessDocument>
46.     <DeliveryMessage
47.       xmlns=http://www.papinet.org/v2r40
48.       xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance
49.       xsi:schemaLocation="http://www.papinet.org/v2r40
50.         http://www.papinet.org/v2r40/DeliveryMessageV2R40.xsd"
51.       DeliveryMessageStatusType="Original"
52.       DeliveryMessageType="DeliveryMessage">
53.     <DeliveryMessageHeader>
54.       <DeliveryMessageNumber>001002456</DeliveryMessageNumber>
55.       <DeliveryMessageDate>
56.         <Date>
57.           <Year>2008</Year>
58.           <Month>03</Month>
59.           <Day>06</Day>
60.         </Date>
61.       </DeliveryMessageDate>
62.       <ShipToCharacteristics>
```

papiNet Envelope Version 1.0

The papiNet Standard

```
61.      <ShipToParty
62.          PartyType="Buyer">
63.          <NameAddress>
64.              <Name1>myBuyerName</Name1>
65.          </NameAddress>
66.      </ShipToParty>
67. </ShipToCharacteristics>
68. <DeliveryLeg>
69.     <DeliveryLegSequenceNumber>
70.         1</DeliveryLegSequenceNumber>
71.     <DeliveryOrigin>
72.         <LocationParty
73.             PartyType="Supplier">
74.             <NameAddress>
75.                 <Name1>mySupplierName</Name1>
76.             </NameAddress>
77.         </LocationParty>
78.     </DeliveryOrigin>
79. </DeliveryLeg>
80. </DeliveryMessageHeader>
81. <DeliveryMessageLineItem>
82.     <DeliveryMessageLineItemNumber>
83.         1</DeliveryMessageLineItemNumber>
84.     <Product>
85.         <ProductIdentifier
86.             Agency="Buyer"
87.             ProductIdentifierType="SKU">
88.                 1234567890</ProductIdentifier>
89.         </Product>
90.     <Quantity
91.         QuantityType="Count">
92.             <Value
93.                 UOM="Reel">5</Value>
94.         </Quantity>
95.     </DeliveryMessageLineItem>
96. </DeliveryMessage>
97. </BusinessDocument>
98. </Payload>
99. </papiNetEnvelope>
```

papiNet Envelope Version 1.0

The papiNet Standard

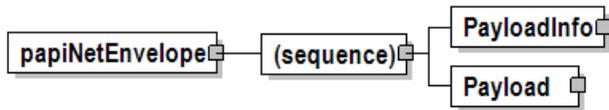
papiNet Envelope Root Element

papiNetEnvelope

The root element of the papiNet Envelope.

(sequence)

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



PayloadInfo

PayloadInfo is mandatory. A single instance is required.

A grouping element for items describing the e-business document and parameters related to its communication.

Payload

Payload is mandatory. A single instance is required.

A grouping element for payload information of the papiNet e-business document envelope.

papiNet Envelope Version 1.0

The papiNet Standard

Primary Elements

PayloadInfo

A grouping element for items describing the e-business document and parameters related to its communication.

TestFlag [attribute]

TestFlag is mandatory. A single instance is required.

If "true" or "1" then this is a test transmission. If "false" or "0" then this transmission is for production use.

(sequence)

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

TransmissionInfo

TransmissionInfo is mandatory. One instance is required, multiple instances might exist.

Information about the transmission. Each participant in the transmission process appends a new TransmissionInfo element indicating that they are the TransmissionSender and to who, the TransmissionReceiver, the XML message is being transmitted to.

BusinessSender

BusinessSender is mandatory. A single instance is required.

The party who issued the papiNet e-document.

BusinessReceiver

BusinessReceiver is mandatory. A single instance is required.

The business entity for whom the business 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.

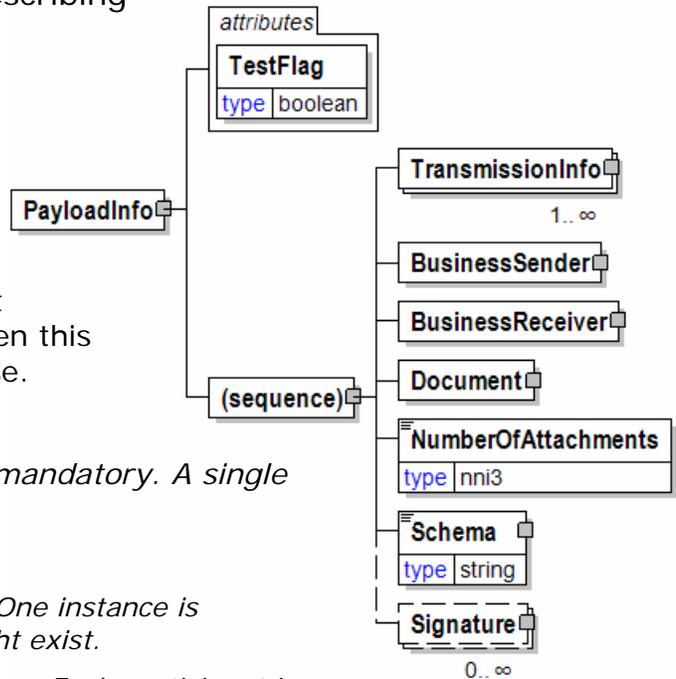
Document

Document is mandatory. A single instance is required.

A group item containing information that describes the document contained in the envelope payload.

NumberOfAttachments

NumberOfAttachments is mandatory. A single instance is required.



papiNet Envelope Version 1.0

The papiNet Standard

The number of Attachment element instances in the payload.

Schema

Schema is mandatory. A single instance is required.

The name of the schema to use for validation. This can be a base URI or a complete pointer to the schema to use.

Signature

Signature is optional. Multiple instances might exist.

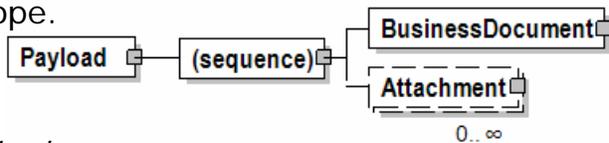
A signature for the BusinessDocument or an Attachment included in the Payload that conforms to the W3C XML Digital Signature specification.

Payload

A grouping element for payload information of the papiNet e-business document envelope.

(sequence)

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



BusinessDocument

BusinessDocument is mandatory. A single instance is required.

This element contains the actual papiNet e-business document payload content.

Attachment

Attachment is optional. Multiple instances might exist.

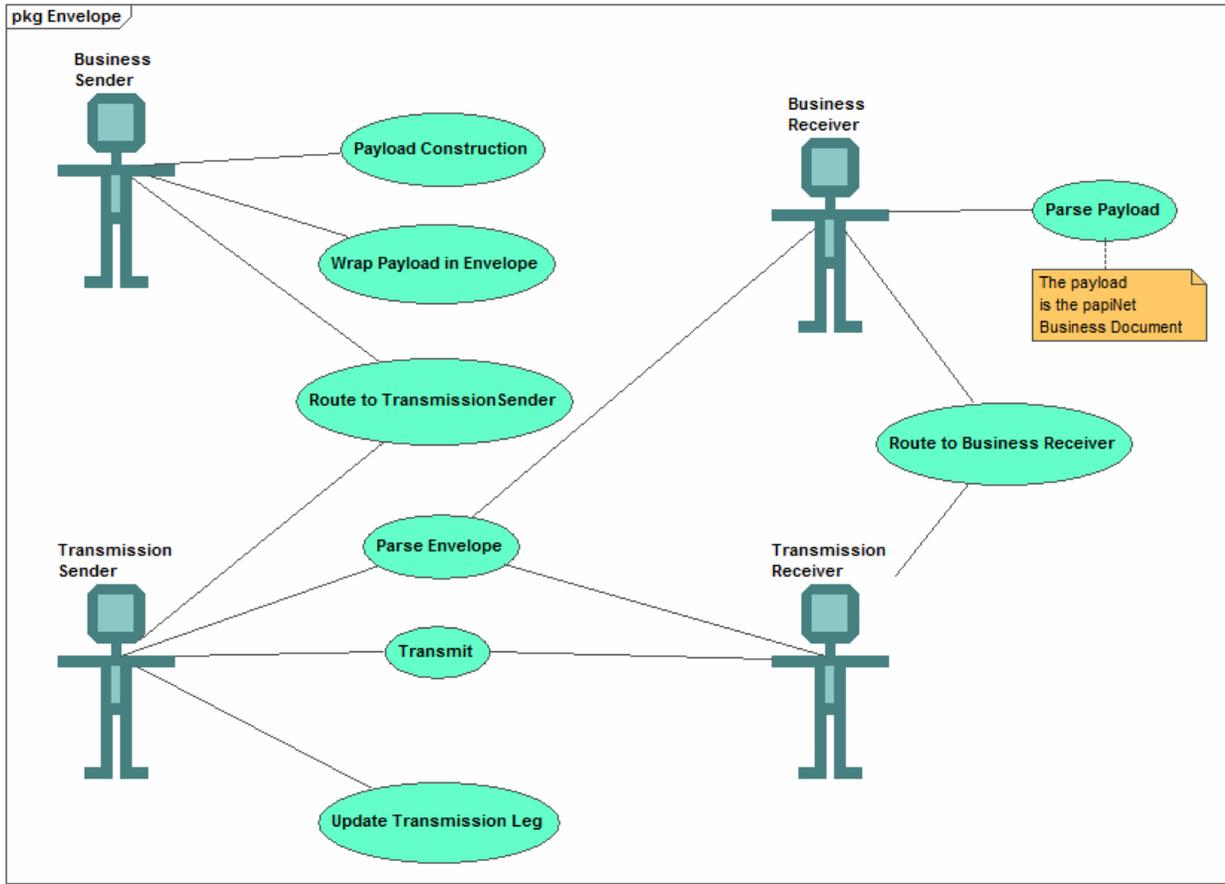
An attachment to the e-business document in Base-64-encoded binary format.

papiNet Envelope Version 1.0

The papiNet Standard

Envelope Use Cases

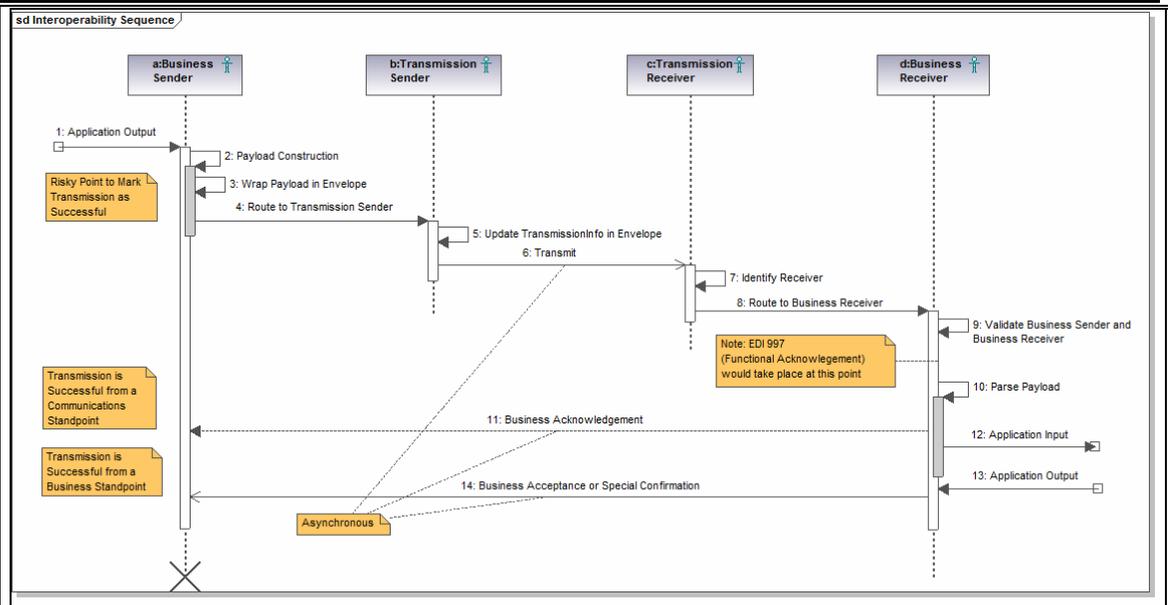
XML Transmission Use Cases



Goal	Receipt of papiNet payload by BusinessReceiver
Actors	BusinessSender, BusinessReceiver, TransmissionSender, TransmissionReceiver.
System	We are dealing with a message service and its associated components.
Discussion	

papiNet Envelope Version 1.0

The papiNet Standard



Discussion

The Business Sender is responsible for constructing the papiNet payload, wrapping it in the papiNet envelope, and routing the information to Transmission Sender.

- The Business Sender and Transmission Sender can be the same entity.
- The creation of the digital signature by the Business Sender and the insertion into the papiNet Envelope means that the Business Sender is responsible for the creation of the papiNet envelope.

The Transmission Sender will parse the papiNet envelope to identify the Business Receiver and determine the appropriate transmission route, update the envelope with a new TransmissionInfo element instance, and then transmit.

The Transmission Receiver will parse the papiNet envelope to identify the Business Receiver and either route directly to the Business Receiver or the Transmission Receiver will take the role of Transmission Sender if another intermediate transmission leg is required.

The Business Receiver will parse the papiNet envelope to validate the Business Receiver and

papiNet Envelope Version 1.0

The papiNet Standard

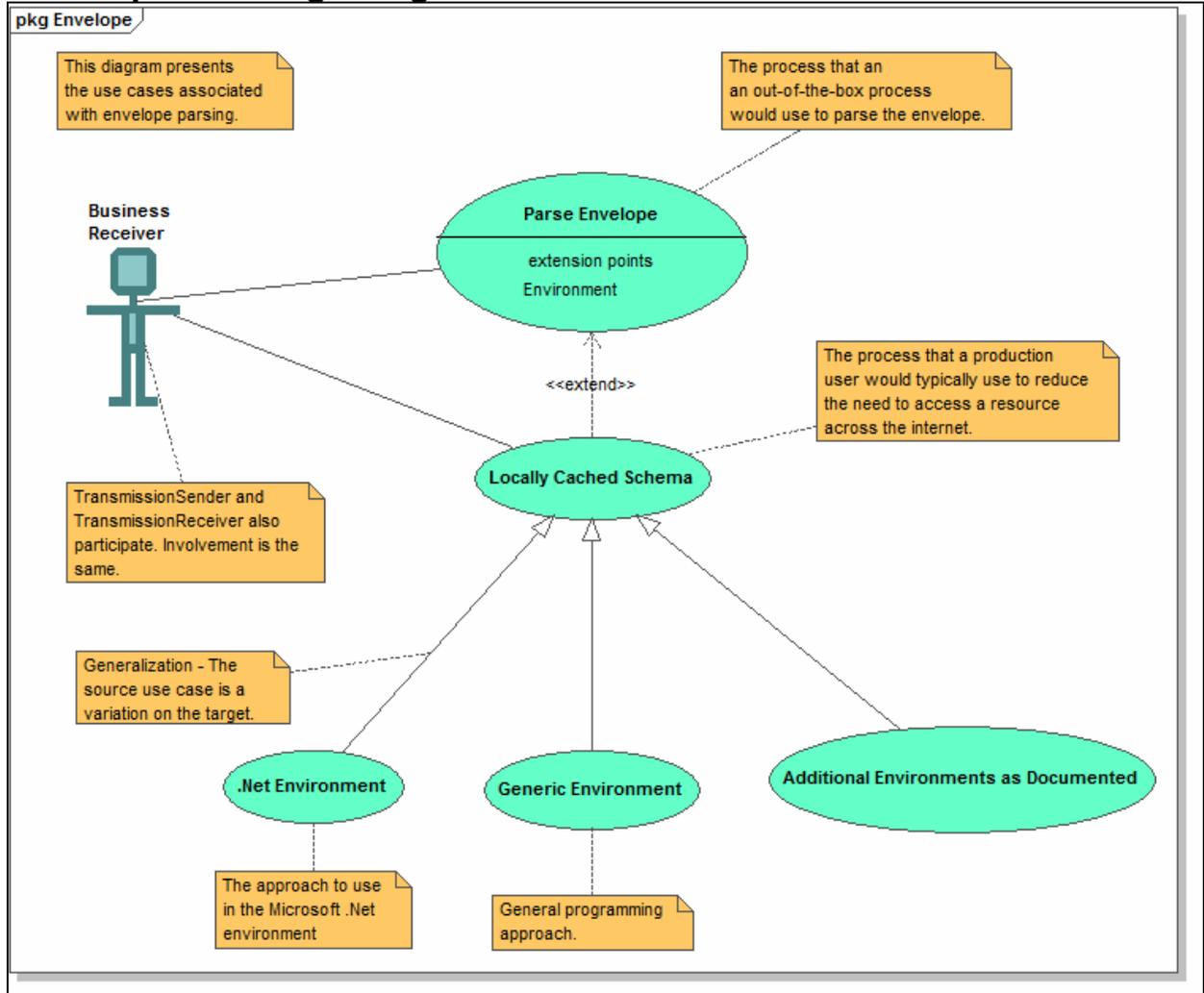
Business Sender identification, signatures, and schema information to use when parsing the payload and attachments.

- In some environments it may be possible to separate the Business Document, Attachments, and Signatures from the papiNetEnvelope wrapper. However, it is expected that, in order to assure the validity of all information, in many environments the Business Receiver will process the XML information from within the papiNetEnvelope.

papiNet Envelope Version 1.0

The papiNet Standard

Envelope Parsing Usage Scenario



Parse Envelope

Goal	Read the papiNetEnvelope for information associated routing and payload processing
Actors	BusinessReceiver, TransmissionSender, TransmissionReceiver. <ul style="list-style-type: none"> For clarity purposes only BusinessReceiver is shown.
System	In an environment where routing is managed outside of a formal message service (e-mail or sneaker-net) the system would typically be the XML parser that is within the Actors browser. Note: In environments where the routing process is

papiNet Envelope Version 1.0

The papiNet Standard

	automated the system events take place within the message service (MS) or a service that is tightly coupled with the MS.
Discussion	The parser will use the what it is defined to use to locate the schema for validation. This could be the xs:schemaLocation or hints provided by the namespace.

Locally Cached Schema

Goal	Permit the Actors to use schema that are locally stored instead of using schema that are indicated in the papiNet envelope. The schema indicated in the papiNet envelope would typically be stored at a location on the internet that would result in processing.
Actors	Refer to "Parse Envelope" use case.
System	In environments where the routing process is automated the system events take place within the message service (MS) or a service that is tightly coupled with the MS.
Discussion	<p>papiNet examples indicate a schemaLocation. We do this to permit validation of the XML information in low-tech environments. You are not required to use schemaLocation information, even if it is included. There are several reasons why you would disregard the schemaLocation. The reason that we will be using for demonstrative purposes is the desire to use a locally stored schema, to improve the responsiveness of your production environment.</p> <p>Depending on your programming environment the technique for managing locally cached schema will be different. Refer to the specific use case for your environment for recommendations. If your environment is not listed contact the papiNet Technical Director (acolman@idealliance.org) and he will work with you to determine the approach.</p>

papiNet Envelope Version 1.0

The papiNet Standard

Locally Cached Schema in a .Net Environment

Goal	Refer to "Locally Cached Schema" use case.
Actors	Refer to "Parse Envelope" use case.
System	Refer to "Locally Cached Schema" use case.
Discussion	<p>If you are using the .Net Framework version 1.0 then refer to System.Xml.XmlSchemaCollection for instructions.</p> <p>System.Xml.Schema.XmlSchemaSet is the approach recommended in .Net Framework version 2.0 as follows:</p> <ul style="list-style-type: none"> • External namespaces or locations referenced in include, import, and redefine elements of a schema are resolved with respect to the base URI of the schema that includes or imports them. For example, if the base URI of the including or importing schema is empty or a null reference ("Nothing" in Visual Basic), the external locations are resolved with respect to the current directory. The XmlResolver class is used to resolve external schemas by default. To disable resolution of include, import, and redefine elements of a schema, set the XmlResolver property to a null reference ("Nothing" in Visual Basic). <p>System.Xml.Schema.XmlSchemaSet is applied to a System.Xml.XmlReader using the XmlReaderSettings object:</p> <ul style="list-style-type: none"> • The ProcessInlineSchema and ProcessSchemaLocation validation flags of an XmlReaderSettings object are not set by default. When these flags are set, the XmlResolver of the XmlReaderSettings object is used to resolve schema locations encountered in the instance document in the XmlReader. If the XmlResolver object is a null reference ("Nothing" in Visual Basic), schema locations are not resolved even if the

papiNet Envelope Version 1.0 The papiNet Standard

	ProcessInlineSchema and ProcessSchemaLocation validation flags are set.
References	<p>XmlSchemaSet:</p> <ul style="list-style-type: none"> • http://msdn2.microsoft.com/en-us/library/system.xml.schema.xmlschemaset.aspx <p>XmlReader:</p> <ul style="list-style-type: none"> • http://msdn2.microsoft.com/en-us/library/system.xml.xmlreader.aspx <p>XmlReaderSettings:</p> <ul style="list-style-type: none"> • http://msdn2.microsoft.com/en-us/library/system.xml.xmlreadersettings.aspx

Locally Cached Schema in a Generic Environment

Goal	Refer to "Locally Cached Schema" use case.
Actors	Refer to "Parse Envelope" use case.
System	Refer to "Locally Cached Schema" use case.
Discussion	<p>Generally speaking when you want to use a local version of a schema instead of the schema referenced in the communicated XML information you will follow the follow steps:</p> <ol style="list-style-type: none"> 1. Determine if there is functionality within your environment to manage this particular scenario. Refer to the use case extensions for examples. 2. Perform a non-validating read of the papiNet envelope to identify the /papiNetEnvelope/PayloadInfo/Schema element content along with its attributes Use the above information to populate the validating xml reader.

papiNet Envelope Version 1.0

The papiNet Standard

papiNetEnvelope Supporting Elements

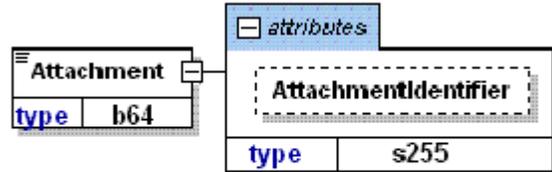
Attachment

An attachment to the e-business document in Base-64-encoded binary format.

AttachmentIdentifier [attribute]

AttachmentIdentifier is optional. A single instance might exist.

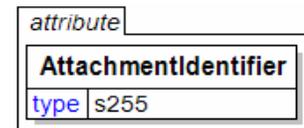
The identifier for the attachment. Permits referencing the attachment from within the BusinessDocument payload or for signature purposes.



The AttachmentIdentifier links to any type of content for inclusion

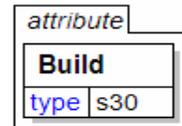
AttachmentIdentifier [attribute]

The identifier for the attachment. Permits referencing the attachment from within the BusinessDocument payload or for signature purposes.



Build [attribute]

The build number of the schema. The build number is comprised of the version number associated with a date. The format is vNrNN_yyyymmdd.



BusinessDocument

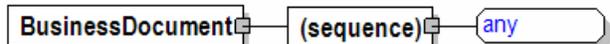
This element contains the actual papiNet e-business document payload content.

(sequence)

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

[Any Type]

Any type of content is permitted at this point.

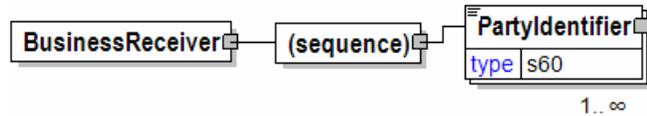


papiNet Envelope Version 1.0

The papiNet Standard

BusinessReceiver

The business entity for whom the business 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.

(sequence)

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

PartyIdentifier

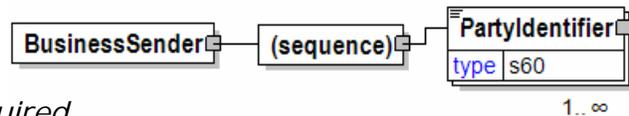
PartyIdentifier is mandatory. One instance is required, multiple instances might exist.

A unique identifier of a specific party. This element contains an attribute PartyIdentifierType that indicates the type of party.

BusinessSender

The party who issued the papiNet e-document.

(sequence)



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

PartyIdentifier

PartyIdentifier is mandatory. One instance is required, multiple instances might exist.

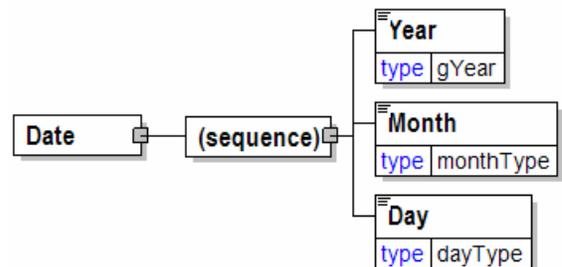
A unique identifier of a specific party. This element contains an attribute PartyIdentifierType that indicates the type of party.

Date

A group element that contains the specification of Year, Month, and Day.

(sequence)

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



Year

Year is mandatory. A single instance is required.

papiNet Envelope Version 1.0

The papiNet Standard

Identifies the year number when communicating a calendar date.

Month

Month is mandatory. A single instance is required.

Identifies the month number when communicating a calendar date.

Day

Day is mandatory. A single instance is required.

Identifies the day of the month when communicating a calendar date. Day is an integer between 1 and 31.

Day

Identifies the day of the month when communicating a calendar date. Day is an integer between 1 and 31.

Day	
type	dayType

Document

A group item containing information that describes the document contained in the envelope payload.

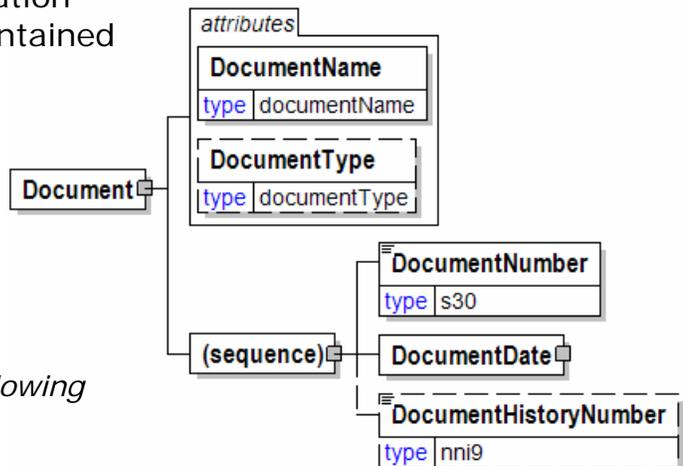
DocumentName [attribute]

DocumentName is mandatory. A single instance is required.

Defines the document name for documents defined by papiNet.

This item is restricted to the following list.

- Availability**
- BookSpecification**
- BusinessAcceptance**
- BusinessAcknowledgement**
- Calendar**
- CallOff**
- CallOffWood**
- Complaint**
- ComplaintResponse**
- CreditDebitNote**



papiNet Envelope Version 1.0

The papiNet Standard

DeliveryInstruction
DeliveryMessage
DeliveryMessageBook
DeliveryMessageWood
GoodsReceipt
InfoRequest
InventoryChange
InventoryDispositionInstructions
InventoryStatus
Invoice
InvoiceWood
LoadTender
LoadTenderResponse
OrderConfirmation
OrderConfirmationWood
OrderStatus
PackingList
Planning
ProductAttributes
ProductPerformance
ProductQuality
PurchaseOrder
PurchaseOrderWood
RFQ
RFQResponse
ScaleTicket
ShipmentStatus
ShippingInstructions
Statement
Usage
Other

DocumentType [attribute]

DocumentType is optional. A single instance might exist.

Identifies the type of a document.

This item is restricted to the following list.

papiNet Envelope Version 1.0

The papiNet Standard

Agreement
Approximate
AvailabilityStatus
BlanketOrder
Booking
BookingConfirmation
CallOff
CallOffConfirmation
Claim
Complaint
ConfirmingOrder
ConsumptionOrder
CreditNote
DebitNote
DeliveryInstruction
DeliveryMessage
Duplicate
Feedback
InitialShipmentAdvice
InventoryChange
InventoryStatus
Invoice
LoadAvailable
LoadAvailableConfirmation
LoadedSpecification
LoadingInstruction
LoadingOrder
LoadingSchedule
LoadTender
LoadTenderResponse
OfficeSchedule
OrderStatus
PreBooking
PreBookingConfirmation
PrePayment

papiNet Envelope Version 1.0

The papiNet Standard

PackingSpecification
PriceList
ProductCrossReference
ProductProperties
ProForma
ReleaseOrder
ReservationOrder
SelfBillingStatement
ShipmentAdvice
ShippingInstructionsConfirmation
ShippingInstructionsPackaging
ShippingInstructionsRouting
SpecOrder
StandardOrder
SpecRFQ
StandardRFQ
SupplierSpec
SupportSchedule
ThirdPartyShipmentAdvice
TrialOrder
UnloadingSchedule
Waybill

(sequence)

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

DocumentNumber

DocumentNumber is mandatory. A single instance is required.

The unique identifier of the document included in the BusinessDocument element of the papiNetEnvelope Payload.

DocumentDate

DocumentDate is mandatory. A single instance is required.

The Date and optionally Time the document that is contained in the BusinessDocument element of the papiNetEnvelope to was issued.

DocumentHistoryNumber

DocumentHistoryNumber is optional. A single instance might exist.

A sequential number that keeps track of the version of the e-document being sent by the document originator. The envelope should always contain info for the e-

papiNet Envelope Version 1.0

The papiNet Standard

document sent.

DocumentDate

The Date and optionally Time the document that is contained in the BusinessDocument element of the papiNetEnvelope to was issued.

(sequence)

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

Date

Date is mandatory. A single instance is required.

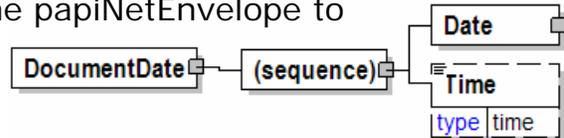
A group element that contains the specification of Year, Month, and Day.

Time

Time is optional. A single instance might exist.

Times are treated in a standard XML fashion. The standard XML approach to time is hh:mm:ss, hh:mm:ssZ, or hh:mm:ss±hh:mm.

- hh:mm:ss±hh:mm is used to represent a time with a time zone. The time zone is communicated as the number of hours offset from the Universal Time Coordinate (UTC), a.k.a. Greenwich Mean Time.
- hh:mm:ssZ indicates the time at Zulu (another name for UTC).
- hh:mm:ss or hh:mm indicates in the XML standard the time at UTC. PapiNet recommends that timezone should always be specified on Time in the papiNet envelope.



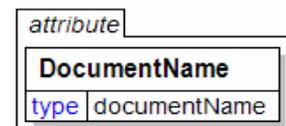
DocumentHistoryNumber

A sequential number that keeps track of the version of the e-document being sent by the document originator. The envelope should always contain info for the e-document sent.



DocumentName [attribute]

Defines the document name for documents defined by papiNet.



papiNet Envelope Version 1.0

The papiNet Standard

DocumentNumber

The unique identifier of the document included in the BusinessDocument element of the papiNetEnvelope Payload.

DocumentNumber	
type	s30

DocumentType [attribute]

Identifies the type of a document.

attribute	
DocumentType	
type	documentType

ID

The unique identifier for the transmission. You can develop this identifier independently from other systems or you can use a similar identifier to one that you are currently using. For example, you could use something similar to the ISA control number in X12.

ID	
type	string

Month

Identifies the month number when communicating a calendar date.

Month	
type	monthType

NumberOfAttachments

The number of Attachment element instances in the payload.

NumberOfAttachments	
type	nni3

PartyIdentifier

A unique identifier of a specific party. This element contains an attribute PartyIdentifierType that indicates the type of party.

PartyIdentifier	
type	s60

attributes	
PartyIdentifierType	
type	partyIdentifierType

PartyIdentifierType [attribute]

PartyIdentifierType is mandatory. A single instance is required.

Provides a contextual definition for the party identifier value. This party

papiNet Envelope Version 1.0

The papiNet Standard

identifier enables the trading partners to use ID codes for the different organisation versus business entities involved in the transaction.

This item is restricted to the following list.

DunsNumber

A 9-digit identifier maintained by Dun & Bradstreet to uniquely identify a commercial enterprise.

Duns4Number

Similar to the DunsNumber that defines a parent company/organization. The additional 4 digits enable the definition/identification of a sub-entity, such as a mill/plant of this parent company/organisation.

GlobalLocationNumber

The globally unique EAN.UCC System identification number for legal entities, functional entities and physical locations. The Global Location Number uses the EAN/UCC-13 Data Structure. Details about the structure and usage are available at the <http://www.gs1.org> website.

papiNetGlobalPartyIdentifier

A readily available and free identifier supported by IANA (the Internet Assigned Numbers Authority). Private Enterprise Numbers that conforms to proper URI naming conventions are available, free-of-charge from the Internet Assigned Numbers Authority (IANA). The application form is available at the

<http://www.iana.org/cgi-bin/enterprise.pl>

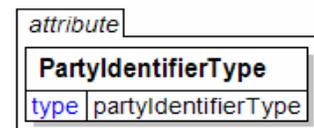
website. papiNet will be developing a broader recommendation for party naming in the future however, obtaining an IANA Private Enterprise Number (PEN) will permit globally unique trading partner identification.

StandardAddressNumber

A 7-digit code to uniquely identify an organisation and its location. The number consists of 6 digits with an 11-modulus check digit as the seventh digit. ISAN is the same with the ISO Country code as a prefix. This identifier is solely used by the book industry.

PartyIdentifierType [attribute]

Provides a contextual definition for the party identifier value. This party identifier enables the trading partners to use ID codes for the different organisation versus business entities involved in the transaction.



papiNet Envelope Version 1.0

The papiNet Standard

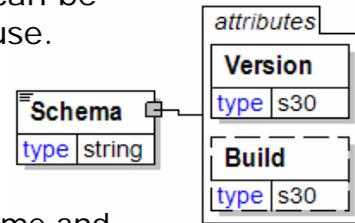
Schema

The name of the schema to use for validation. This can be a base URI or a complete pointer to the schema to use.

Version [attribute]

Version is mandatory. A single instance is required.

The version identifier for the schema. For papiNet schema the version is at the end of the schema file name and follows the format of VnRnn. The version can also be found in the version attribute of the root element as the beginning of the build number.



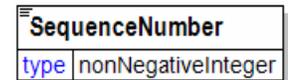
Build [attribute]

Build is optional. A single instance might exist.

The build number of the schema. The build number is comprised of the version number associated with a date. The format is vNrNN_yyyymmdd.

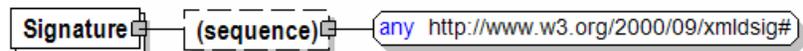
SequenceNumber

The sequence of the item. Provides a mechanism for sorting.



Signature

A signature for the BusinessDocument or an Attachment included in the Payload that conforms to the W3C XML Digital Signature specification.



(sequence)

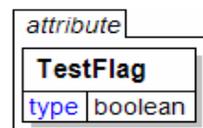
The sequence of items below is optional. A single instance might exist.

[Any Type]

Any type of content is permitted at this point.

TestFlag [attribute]

If "true" or "1" then this is a test transmission. If "false" or "0" then this transmission is for production use.



papiNet Envelope Version 1.0

The papiNet Standard

Time

Times are treated in a standard XML fashion. The standard XML approach to time is hh:mm:ss, hh:mm:ssZ, or hh:mm:ss±hh:mm.

Time
type time

- hh:mm:ss±hh:mm is used to represent a time with a time zone. The time zone is communicated as the number of hours offset from the Universal Time Coordinate (UTC), a.k.a. Greenwich Mean Time.
- hh:mm:ssZ indicates the time at Zulu (another name for UTC).
- hh:mm:ss or hh:mm indicates in the XML standard the time at UTC. PapiNet recommends that timezone should always be specified on Time in the papiNet envelope.

TimeStamp

The date and time e-business document, the payload, was prepared for transmission. The format and interpretation of Timestamp is the standard W3C xs:dateTime format.

TimeStamp
type dateTime

TransmissionInfo

Information about the transmission. Each participant in the transmission process appends a new TransmissionInfo element indicating that they are the TransmissionSender and to who, the TransmissionReceiver, the XML message is being transmitted to.

(sequence)

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

SequenceNumber

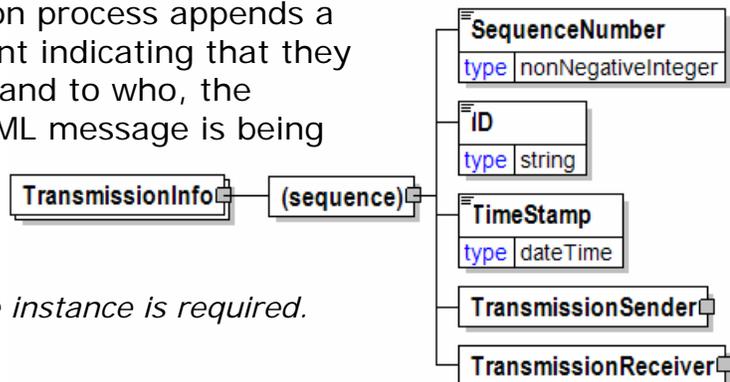
SequenceNumber is mandatory. A single instance is required.

The sequence of the item. Provides a mechanism for sorting.

ID

ID is mandatory. A single instance is required.

The unique identifier for the transmission. You can develop this identifier independently from other systems or you can use a similar identifier to one that you are currently using. For example, you could use something similar to the ISA control number in X12.



papiNet Envelope Version 1.0

The papiNet Standard

TimeStamp

TimeStamp is mandatory. A single instance is required.

The date and time e-business document, the payload, was prepared for transmission. The format and interpretation of Timestamp is the standard W3C xs:dateTime format.

TransmissionSender

TransmissionSender is mandatory. A single instance is required.

The party responsible for transmitting the papiNet e-document. This can be the same as the party issuing the e-document or any handling service for the BusinessSender such as a portal, a hub, etc. depending on the set-up of communication services.

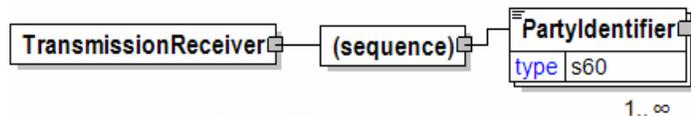
TransmissionReceiver

TransmissionReceiver is mandatory. A single instance is required.

The party responsible for the transmission receipt of the papiNet e-document. This can be the party who will eventually process the papiNet for the e-document contents or any handling service BusinessReceiver such as a portal, a hub etc. depending on the set-up of communication services.

TransmissionReceiver

The party responsible for the transmission receipt of the papiNet e-document. This can be the party who will eventually process the papiNet for the e-document contents or any handling service BusinessReceiver such as a portal, a hub etc. depending on the set-up of communication services.



(sequence)

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

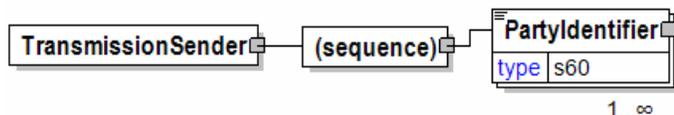
PartyIdentifier

PartyIdentifier is mandatory. One instance is required, multiple instances might exist.

A unique identifier of a specific party. This element contains an attribute PartyIdentifierType that indicates the type of party.

TransmissionSender

The party responsible for transmitting the papiNet e-document. This can be the same as the party issuing the e-document or any



papiNet Envelope Version 1.0

The papiNet Standard

handling service for the BusinessSender such as a portal, a hub, etc. depending on the set-up of communication services.

(sequence)

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

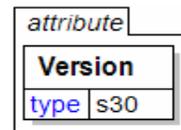
PartyIdentifier

PartyIdentifier is mandatory. One instance is required, multiple instances might exist.

A unique identifier of a specific party. This element contains an attribute PartyIdentifierType that indicates the type of party.

Version [attribute]

The version identifier for the schema. For papiNet schema the version is at the end of the schema file name and follows the format of VnRnn. The version can also be found in the version attribute of the root element as the beginning of the build number.



Year

Identifies the year number when communicating a calendar date.

