

**eHealthBox v.3
Publication WS
Cookbook
Version 3.13**

This document is provided to you free, of charge, by the

eHealth platform

**Willebroekkaai 38 – 1000 Brussel
38, Quai de Willebroek – 1000 Bruxelles**

Anyone is free to distribute this document, referring to the URL source.

Table of contents

Contents

Table of contents	2
1. Document management	4
1.1 Document history	4
2. Introduction	5
2.1 Goal of the service	5
2.2 Goal of the document	5
2.3 New in version 3.0	5
2.4 eHealth platform document references	6
2.5 External document references	6
2.6 Service history	6
3. Support	8
3.1 Helpdesk eHealth platform	8
3.1.1 Certificates	8
3.1.2 For issues in production	8
3.1.3 For issues in acceptance	8
3.1.4 For business issues	8
3.2 Status	8
4. Global overview	9
5. Step-by-step	10
5.1 Technical requirements	10
5.1.1 Use of the eHealth SSO solution	10
5.1.2 End-to-end encryption for known recipient	10
5.1.3 WS-I Basic Profile 1.1	10
5.1.4 Tracing	10
5.2 Process overview	11
5.2.1 WS WSDL (Web Service Definition Language)	11
5.3 eHealthBox Publication WS	11
5.3.1 Lifetime of a message	11
5.3.2 Treatment of NIHL-11 numbers	11
5.3.3 Out-of-Office system	12
5.3.4 Mass mailings	12
5.3.5 SendMessage Method	12
5.3.6 Used Types	18
5.3.7 Extensions and Encoding (BODY vs ANNEX)	25
5.3.8 Txt Size in XML tags	26
6. Risks and security	27
6.1 SOAP with attachments	27
6.2 SecurityBusiness security	27



6.2.1	Web service	27
6.2.2	The use of username, password and token.....	27
7.	Test and release procedure.....	28
7.1	Procedure	28
7.1.1	Initiation	28
7.1.2	Development and test procedure.....	28
7.1.3	Create test cases.....	28
7.1.4	Release procedure	28
7.1.5	Operational follow-up.....	29
7.2	Test cases	29
8.	Error and failure messages.....	30
8.1	Send Message Response Status Codes.....	30
8.2	Soap Fault Error Codes	31
8.2.1	Schema Validation Errors.....	32
8.2.2	Technical Errors	33
8.3	Error codes originating from the eHealth platform:	34
9.	Annex 1 – Publish a message to a list of professionals	35

To the attention of: “IT expert” willing to integrate this web service.

1. Document management

1.1 Document history

Version	Date	Author	Description of changes / remarks
3.0	06/05/2013	eHealth platform	eHealthBox v3.0 WS
3.1	31/08/2017	eHealth platform	Update list of qualities in annex 2
3.2	16/02/2018	eHealth platform	New logo links
3.3	30/04/2018	eHealth platform	Extract Annex
3.4	21/01/2020	eHealth platform	Update
3.5	20/04/2020	eHealth platform	WS-I Org Compliance
3.6	29/04/2020	eHealth platform	Anonymization of data examples
3.7	20/04/2021	eHealth platform	§ 5.1.3 Tracing
3.8	13/10/2021	eHealth platform	Number of annexes reduced to 25 maximum.
3.9	22/02/2022	eHealth platform	§ 5.3.1: Lifetime of a message
3.10	01/07/2022	eHealth platform	§ 3.2 Status (added) § 5.1.3 Tracing (updated)
3.11	20/07/2022	eHealth platform	§ 2.4 eHealth platform document references (updated)
3.12	20/09/2022	eHealth platform	§ 5.3.7 Extensions and Encoding BODY vs ANNEX (added)
3.13	27/07/2023	eHealth platform	§ 5.3.8 Txt size in XML tag

2. Introduction

2.1 Goal of the service

The eHealthBox Publication web service (WS) allows an authenticated user to publish an (encrypted) eHealthBox message (Document or News) for different addressees, with optional elements such as annexes and metadata.

The eHealthBox central systems receive and process the publication request asynchronously.

This means that a successful response does not guarantee the message will be correctly published at the end of the process. A publication failure can occur later on due to the behaviour of external systems. In this case, a structured notification message (an eHealthBox message) is sent back to the sender.

A successful response message only guarantees that the message will be processed.

Fields indicated as 'obsolete' are old fields that are still in use by some systems and kept for backward compatibility. They are out-of-date and must not be used by new partners for they do provide no 'extra' feature.

The size of a message is currently limited to 30MB. The global size of an eHealthBox is limited to 10MB by default (including inbox and bin folder). The maximum number of annexes is currently limited to 25. Note that an encrypted message weighs more due to the encryption overhead.

2.2 Goal of the document

This document provides functional and technical information about calling the eHealthBox Publication WS, as provided by the eHealth platform.

In this service specification document, we will explain the structure and content aspects of the possible requests, as well as the replies of the eHealth platform WS. An example illustrates each of those messages. This document shows also a list of possible errors.

This information should allow (the IT department of) an organization to integrate and use the WS call.

Some technical and legal requirements must be met in order for the eHealth platform WS to be integrated in client applications; this document is meant to provide you with an overview of these requirements.

This document is neither a development nor a programming guide for internal applications. The eHealth platform partners always have a total freedom within those fields. Nevertheless, in order to interact in a smooth, homogeneous and risk controlled way with a maximum of partners, eHealth platform partners must commit to comply with specifications, data format, and release processes described in this document.

In addition, our partners in the healthcare sector must also comply with the business rules of validation and integration of data within their own applications in order to minimize errors and incidents..

2.3 New in version 3.0

- A new "Out-of-Office" system (OoO) has been implemented. It will enable the sender to know whether one of the recipients is absent and send his message to a substitute or substitutes to be treated. For example, physicians on holiday may want to ensure continuity of healthcare services for their patients. To do this, they can automatically transfer their messages to another colleague responsible during their holidays thanks to the "Out-of-Office" system.
- If a recipient is absent, a special SendMessageResponse, which contains the substitutes, is sent back. See further details below.
- SOAP with Attachments (SwA) standard is now being used instead of MTOM. The Digest element must contain the SHA-256 hashing of the content in order to validate integrity of the content. See further details below.
- News and Document types were merged. A News is now a special kind of document.
- ContentSpecification contains a ContentType element now, specifying the type of message.
- eHealthBoxes are now limited to 10MB on inbox and trash bin folder

2.4 eHealth platform document references

On the portal of the eHealth platform, you can find all the referenced documents.¹ These versions, or any following ones, can be used for the eHealth platform service.

ID	Title	Version	Date	Author
1	Secure Token Service Holder-of-Key profile (STS) – Cookbook	1.6	25/01/2023	eHealth platform
2	End-to-End Encryption Known recipient – Cookbook	2.9	18/07/2022	eHealth platform
3	eHealthBox v3 Consultation - Cookbook	3.12	26/07/2023	eHealth platform
4	eHealthBox Supported Qualities	1.4	22/12/2021	eHealth platform
5	eHBox SSO	1.2	22/04/2020	eHealth platform

2.5 External document references

All documents can be found through the internet. They are available to the public, but not supported by the eHealth platform.

ID	Title	Date
1.	http://www.w3.org/TR/SOAP-attachments	11/12/2000
2.	http://www.ws-i.org/Profiles/AttachmentsProfile-1.0.html	20/04/2006
3.	http://ws-i.org/profiles/basic/1.1/swaref.xsd	24/08/2004
4.	http://docs.oasis-open.org/ws-sx/ws-securitypolicy/200702/	NA
5.	http://www.w3.org/standards/xml/	24/03/11
6.	http://www.oasis-open.org/specs/	24/03/11
7.	http://www.w3.org/TR/soap12-mtom/	NA

2.6 Service history

This chapter contains the list of changes made to the service with respect to the previous version.

Previous version	Previous release date	Changes
1.0 DEPRECATED	03/02/2011	Major changes: Encryption, Multi-box, Publication Web Application, Consultation Web Application reviewed, general reliability ...
2.0 DEPRECATED	29/04/2012	Major changes: SOAP with Attachment, Out-of-Office system, DeleteMessage, MoveMessage, structured notification messages, performances.
3.0	06/05/2013	Major changes : new Out-of-Office system (insertOoO, getOoO, deleteOoO) , OoO SendMessageResponse with substitutes, MTOM, News and Document merged.

¹ www.ehealth.fgov.be/ehealthplatform

3.1	31/01/2020	Major changes : No more News, No more PublicationID, no more publications to a whole quality (impossible to send a message to all people how have a same quality).
-----	------------	--

3. Support

3.1 Helpdesk eHealth platform

3.1.1 Certificates

In order to access the secured eHealth platform environment you have to obtain an eHealth platform certificate, used to identify the initiator of the request. In case you do not have one, please consult the chapter about the eHealth Certificates on the portal of the eHealth platform

- <https://www.ehealth.fgov.be/ehealthplatform/nl/ehealth-certificaten>
- <https://www.ehealth.fgov.be/ehealthplatform/fr/certificats-ehealth>

For technical issues regarding eHealth platform certificates

- Acceptance: acceptance-certificates@ehealth.fgov.be
- Production: support@ehealth.fgov.be

3.1.2 For issues in production

eHealth platform contact centre:

- Phone: 02 788 51 55 (on working days from 7 am till 8 pm)
- Mail: support@ehealth.fgov.be
- Contact Form :
 - <https://www.ehealth.fgov.be/ehealthplatform/nl/contact> (Dutch)
 - <https://www.ehealth.fgov.be/ehealthplatform/fr/contact> (French)

3.1.3 For issues in acceptance

Integration-support@ehealth.fgov.be

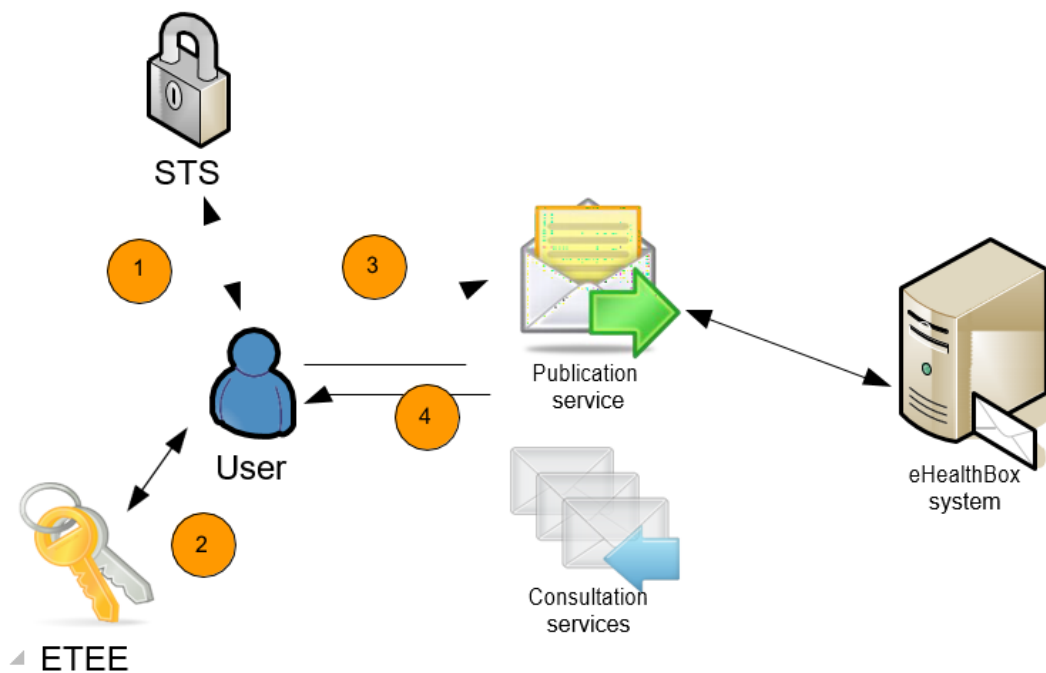
3.1.4 For business issues

- regarding an existing project: the project manager in charge of the application or service
- regarding a new project or other business issues: info@ehealth.fgov.be

3.2 Status

The website <https://status.ehealth.fgov.be> is the monitoring and information tool for the ICT functioning of the eHealth services that are partners of the Belgian eHealth system.

4. Global overview



This global overview aims to show how the publication WS is used.

- Step 1. To use the Publication WS, you have to contact the WS STS to get a secure token containing the identification of the user (see 5.1.1 and CookBook STS).
- Step 2. Optionally, if you want to encrypt the content of your message, you have to call method Get ETK of the ETK Depot WS (synonymous to the ETEE WS, see the cookbook: "ETEE for known recipient"¹) in order to get the public key(s) of the recipient(s) and use the Crypto Library.
- Step 3. Once you have your secure token, you are able to use and contact the "Publication service" to publish your message(s).
- Step 4. When your message has been sent, the system will respond to you with a response message.

5. Step-by-step

5.1 Technical requirements

All the xml requests that are submitted to the WS must be encoded in the UTF-8 format.

5.1.1 Use of the eHealth SSO solution

This section specifies how the call to the Secure Token Service (STS) must be done in order to access the WS. You must precise several attributes in the request. The details on the identification attributes and the certification attributes can be found in the separate document eHealth eHBox_SSO.

To access the eHealth WS, the response token must contain “true” for the ‘BOOLEAN’ certification attribute.

If you obtain “false”, contact the eHealth contact center to verify that the requested test cases were correctly configured.

5.1.2 End-to-end encryption for known recipient

The message to send to the WS may be encrypted.

To encrypt the message, you should retrieve the public key of all recipients plus yours from the eHealth Token Key (ETK) depot. Then, you should encrypt the message using the obtained public keys via the eHealth platform Crypto Libraries.

All the information about the use of the encryption libraries and how to call the ETK depot is described in the ETEE for known recipient cookbook available in the eHealth platform cookbook on the eHealth platform website¹.

Encrypted message convention: If an encrypted message is to be sent, ALL “Encryptable” fields MUST contain (all) encrypted content. In other words, first encrypt the content of each of those fields separately and set IsEncrypted to True. If IsEncrypted is set to True, and non-encrypted content is being transported, unexpected errors can occur. The other way around, if IsEncrypted is set to False, and encrypted content is being transported, unexpected errors can also occur. You cannot choose to encrypt some “Encryptable” fields solely.

5.1.3 WS-I Basic Profile 1.1

Your request must be WS-I compliant (See Chap 2.4 - External Document Ref).

5.1.4 Tracing

To use this service, the request SHOULD contain the following two http header values (see RFC

<https://datatracker.ietf.org/doc/html/rfc7231#section-5.5.3>):

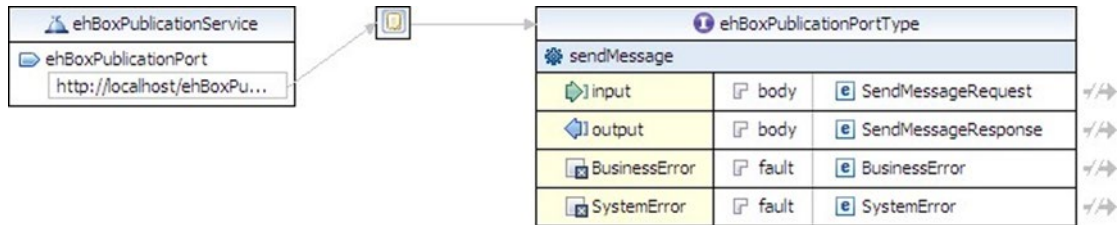
1. User-Agent: information identifying the software product and underlying technical stack/platform. It MUST include the minimal identification information of the software such that the emergency contact (see below) can uniquely identify the component.
 - a. Pattern: {minimal software information}/{version} {minimal connector information}/{connector-package-version}
 - b. Regular expression for each subset (separated by a space) of the pattern: `[[a-zA-Z0-9-√]]*√[[0-9azA-Z-_.]]*`
 - c. Examples:
User-Agent: myProduct/62.310.4 Technical/3.19.0
User-Agent: Topaz-XXXX/123.23.X freeconnector/XXXXX.XXX
2. From: email-address that can be used for emergency contact in case of an operational problem.
Examples:
From: info@mycompany.be

5.2 Process overview

5.2.1 WS WSDL (Web Service Definition Language)

You can find more information on the Registry website of the eHealth platform:

<https://portal.api.ehealth.fgov.be>



The important sections of the WSDL (Web Service Definition Language) of the Publication WS are:

- The applicable **Policies**, which cover the **security** aspects.
- The types (**SendMessageRequest** and **SendMessageResponse**) used by the **sendMessage** method. The fault message is also defined.
- The **sendMessage** method.

5.3 eHealthBox Publication WS

5.3.1 Lifetime of a message

A message will automatically be deleted when:

- the message is in the folder "IN" for more than one year. The exact date can be retrieved in the key "expirationDate".
- the message is in the folder "SENT" for more than 3 days and does no longer exist in the folder "INBOX", "BININBOX" or "STANDBY" of any box or the message is in the folder "SENT" for more than one year.
- the message is in the folder "BININBOX " for more than 3 months and has already been read. The exact date can be calculated by adding three months to the PublicationDate. Unread messages older than three months will remain in the folder until they are read or the expirationDate has passed.
- the message is in the folder "BINSENTBOX" for more than 3 months. The exact date can be calculated by adding three months to the PublicationDate.
- The message is of type 'ACKNOWLEDGMENT' (see ContentType) and older than 30 days.

5.3.2 Treatment of NIHI-11 numbers

When sending a message via the Publication WS, the identification of the sender is by default, retrieved from the STS Token. The recipient will see the identifier of the sender when retrieving the message via the Consultation WS. If the sender is a healthcare professional, his SSIN can be retrieved from the STS token and used as identifier for the sender. If the sender is a hospital or some other healthcare organization, the identifier in the STS token will be of the type NIHI. Finally, if the sender is an organization, the identifier type will be a CBE. Exception is made if BoxId (optional element) is filled out. In that case, the person connected to the WS is sending the message "in name of the person or organization" specified in the element BoxId. Thus, if a healthcare professional (a person) wants to send a message using his NIHI (INAMI) number instead of his SSIN number, he has to submit the request and fill out his NIHI number in the BoxId.

For example, the sender is connected as a doctor with an STS token containing his INSS number as attribute:

```
Id = 12345678910  
Type = INSS  
Quality = DOCTOR
```

But he does not want to display his SSIN number to the recipient, since he prefers to display his NIHI number; so he uses the BoxId element in the request, and fills it out as follows:

```
<BoxId>
  <Id>123456789</Id>
  <Type>NIHII</Type>
  <Quality>DOCTOR</Quality>
</BoxId>
```

The sender will then be stored in the application with the data found in BoxId (after access validation).

Note: a NIHI number consists out of 11 numbers (with the 3 qualification numbers) or 8 numbers (without the 3 qualification numbers).

When authenticated as an organization, it is not possible to use the BoxId.

5.3.3 Out-of-Office system

This system enables the sender to know whether one of the recipients is absent and send his message to a substitute or substitutes to be treated. For example, physicians on holiday may want to ensure continuity of healthcare services for their patients. To do this, they can automatically transfer their messages to another colleague responsible during their holidays thanks to the “Out-of-Office” system. If a recipient is absent, a special SendMessageResponse with the substitutes is returned.

When sending a message, the system checks if some of the recipients of the message have an active OoO. If so, a specific OoO Business error is returned (in SendMessageResponse). This response contains the period during which the original recipient is absent, the substitutes of the recipient (if specified) and their qualities.

It is possible that the absentee has not provided a replacement. In this case, the same OoO Business error will be returned but without a substitute.

SendMessageRequest must then be called again and substitutes should be added next to the original recipient.

OoOProcessed under DestinationContext is set to true for the original recipient. For this recipient, there will be no more OoO verification during the second request. If the message is to be sent encrypted, the message must be encrypted again for the added substitute(s).

If the substitute himself is absent, a new OoO Business error will be returned, and so on. The OoO activates automatically when the time is reached.

OoO system is inactive for a mass mailing (described below): no notification will be sent.

5.3.4 Mass mailings

You can send a message to a specific person or to multiple persons by defining one DestinationContext per person. You can also send a message to a list of persons sharing the same “Quality”. Please refer to Annex 1 – Publish a message to a list of professionals. In order to be able to send a message to all persons sharing the same quality (all nurses, or all doctors), prior authorization from the eHealth platform (info@eHealth.fgov.be) is required. The eHealth platform will investigate your request, and then allow you to publish this type of message.

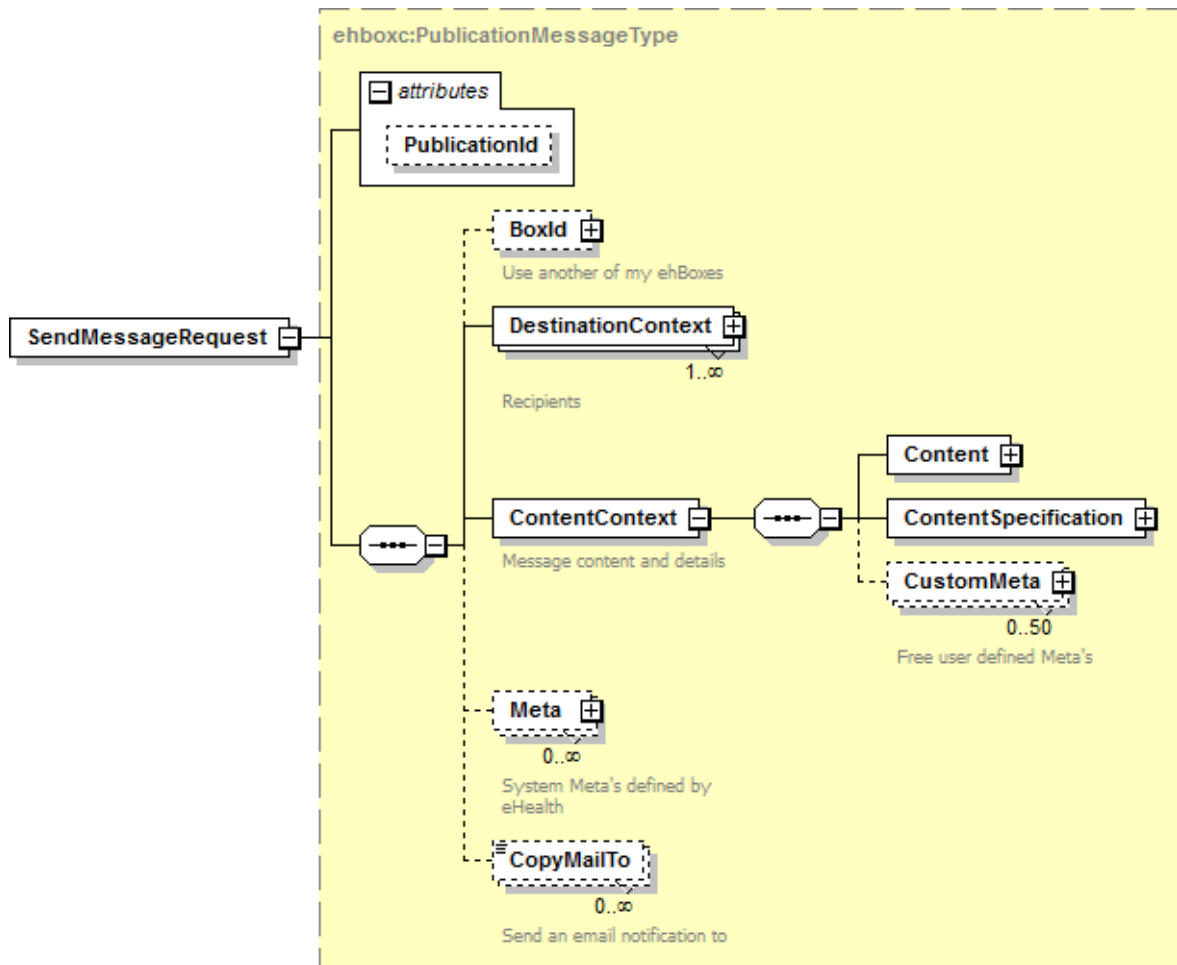
5.3.5 SendMessage Method

The request that must be sent in order to publish the eHealthBox content for one or more recipients will be discussed here. You can send a message of the type “Document” or “News item”. A document contains a PublicationId that you define and that cannot be used twice.

A News item, on the other hand, can be updated and this is done by sending a new news item with the same PublicationId. You cannot update a “Document” type.

5.3.5.1 SendMessage Request

You will find all mandatory information about the allowed combinations Id-Type- Quality (See eHealthBox – Supported qualities v1.2).

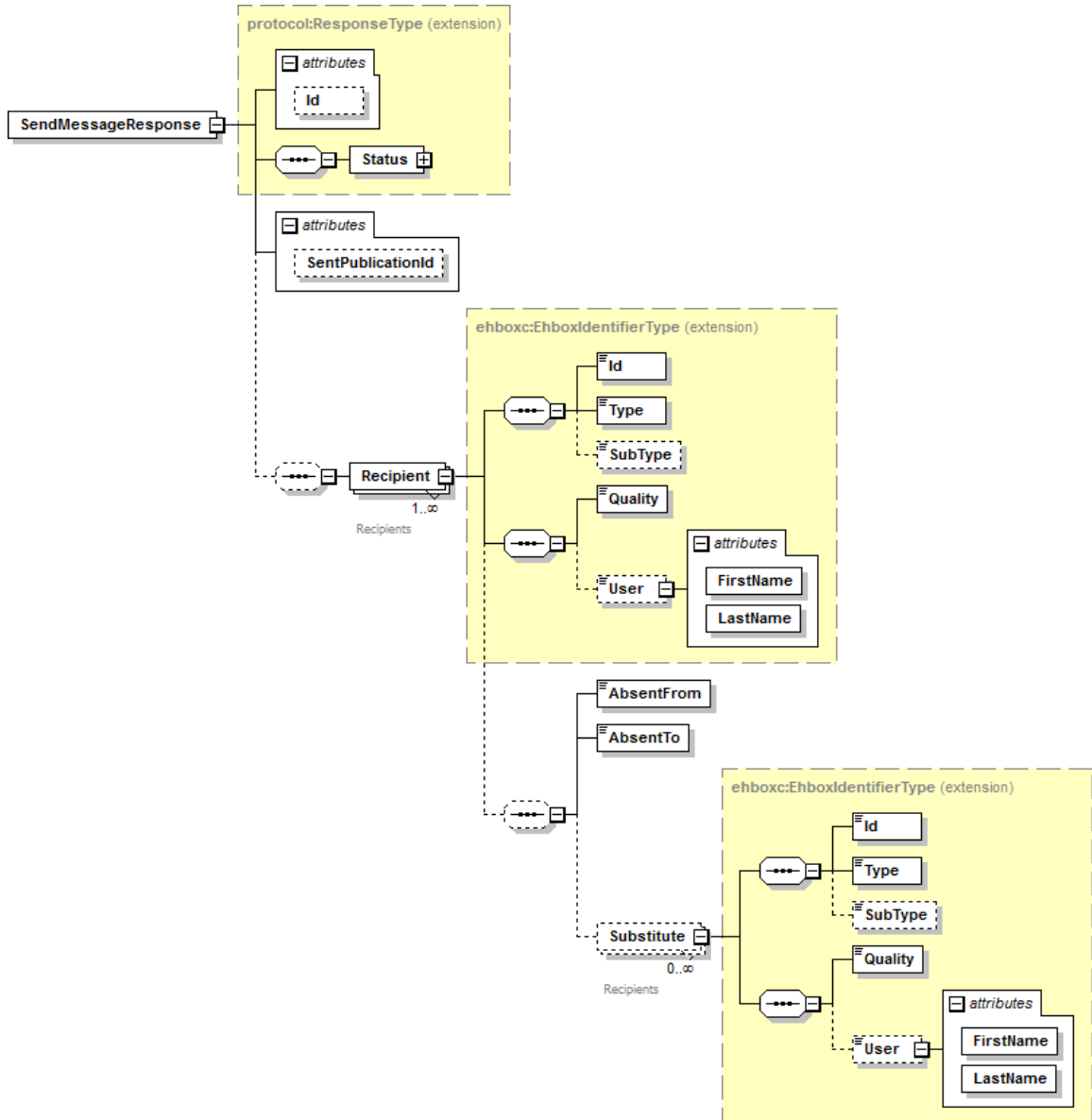


Field name	Description
PublicationId	<p>The identification of the publication as defined to the client. Any alphanumeric string (minimum 1, maximum 13) can be used. The use of the PublicationId depends on the content type: document or news item (see further).</p> <ul style="list-style-type: none"> • If the content is a “document”, the PublicationId must be unique. If you try to publish two documents with the same PublicationId, the second one will not be published. • If the content is a “news item”, you can use the same PublicationId to update your news content. When you publish two news with the same PublicationId, the first will be added as history and replaced by the second in the inbox folder. You have to be careful, if you first send a news item to x recipients, and then update your news item by sending a new content with the same PublicationId, all the recipients will receive the updated content, even if you specify other recipients.
BoxId	<p>If the client wants to use another of his eHealthBoxes, he can specify it here. The message will then be sent as if the authentication was made with this eHealthBox. This avoids the client having to re-authenticate himself each time.</p>

DestinationContext	<p>The <i>DestinationContext</i> is a complex type that contains information about the recipients.</p> <p>See section 5.3.6.7.</p> <p>A <i>SendMessageRequest</i> can have numerous <i>DestinationContexts</i> (numerous recipients).</p>
ContentContext	<p>The <i>ContentContext</i> is a complex type that contains the message content.</p> <p>See section 5.3.6.4.</p>
Meta	<p>Currently, no meta information is defined.</p> <p>Additional system meta information can be defined by the eHealth platform and used in agreement with the client (for future needs). The type of meta information must be defined in the eHealthBox system before it can be used (see section 5.3.6.10).</p>
CopyMailTo	<p>One or more email address(es) that will receive a notification when the message has been published (optional, string minimum 1, maximum 80). If you would like to notify more than one recipient, you can add each e-mail address in a separate <i>CopyMailTo</i> element.</p> <p>By default, a notification will be sent to the hospital's security manager (registered in the user management of the social security) or in case of a publication, to an individual person (doctor, citizen...): the person will receive a notification if he has updated his email address on the web application UPPAD.</p>

5.3.5.2 SendMessage Response

We describe here how the response returns for the request as described above. The response contains a success status code if no major error could be found in the request. Still, remember that the system is asynchronous: a successful response does not guarantee that the message will be published correctly at the end of the process. In chapter 8, you will find the possible business errors returned by the method.



Field name	Description
Id	The <i>Id</i> is attributed to the eHealthBox message by the eHealthBox system. This is a unique number, used to identify the message in the eHealthBox system. This Id will be provided to the recipient by the Consultation WS.

Status	<p>The <i>Status</i> block contains a code and a message. If no error has occurred during the transaction, the <i>Code</i> will be '100' and the <i>Message</i> 'SUCCESS'. Remember that the system is asynchronous: a successful response does not guarantee that the message will be published correctly at the end of the process.</p> <p>Otherwise:</p> <p>In case of a business error:</p> <ul style="list-style-type: none"> • The <i>Code</i> will be an error code, which unequivocally identifies the problem (see further). • The <i>Message</i> will be a description of the error. Each Message has a <i>Lang</i> (language) characteristic : <ul style="list-style-type: none"> - "FR": French - "NL": Dutch - "EN": English - "DE": German - "NA": Not applicable <p>In case of technical errors, you will receive a Soap Fault message (see chapter 8).</p>
SentPublicationId	The PublicationId provided in the SendMessageRequest message allows grouping of requests/responses. String 13 digits.
Recipient	This optional <i>Recipient</i> element is a complex type containing information about the absent recipients (if any). This type extends <i>EhboxIdentifierType</i> . For detailed information about the content: see section 5.3.6.7.
AbsentFrom/To	Contains the period of absence of the recipient. The start and end date are inclusive. The element type is xs:date.
Substitute	This optional <i>Substitute</i> element is a complex type containing information about the colleagues replacing the absentee (if any defined by the absent person). For detailed information about the content: see section 5.3.6.7.

5.3.5.3 Example

The following example does not contain the SAML assertion.

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:urn="urn:be:fgov:ehhealth:ehbox:publication:protocol:v3">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:SendMessageRequest PublicationId="1009927801393">
      <BoxId>
        <Id>12345678910</Id>
        <Type>INSS</Type>
        <Quality>DOCTOR</Quality>
      </BoxId>
      <DestinationContext>
        <Id>123456789</Id>
        <Type>NIHII</Type>
        <SubType>HOSPITAL</SubType>
        <Quality>HOSPITAL</Quality>
      </DestinationContext>
    </urn:SendMessageRequest>
  </soapenv:Body>
</soapenv:Envelope>
```



```

        </DestinationContext>
        <DestinationContext>
            <Id>987654321</Id>
            <Type>NIHII</Type>
            <Quality>DOCTOR</Quality>
        </DestinationContext>
        <ContentContext>
            <Content>
                <Document>
                    <Title>DocumentTitle</Title>
                    <EncryptableBinaryContent>
cid:4906633928</EncryptableBinaryContent>
                    <DownloadFileName>test.txt</DownloadFileName>
                    <MimeType>application/octet-stream</MimeType>
                    <Digest>MzQ5ZmE3ODlhNzEyNTdlYTZiOTQ2Y2Q3NTVk
ZDkyMzUxZDMwNjc4Y2FjMGJkMDgwYTI3Zjc5ZjJm OTQ2YzY3Mw==</Digest>
                </Document>
                <FreeInformations>
            </Content>
        </ContentContext>
        <EncryptableFreeText>SW5mb3JtYXRpb24=</EncryptableFreeText>
        </FreeInformations>
        <EncryptableINSSPatient>ODQxMjA4MjI3NjI=</EncryptableINSSPatient>
        <Annex>
            <EncryptableTitle>YW5uZXggMQ==</EncryptableTitle>
        </Annex>
        <EncryptableTextContent>YW5uZXg=</EncryptableTextContent>
            <DownloadFileName>annexname.txt</DownloadFileName>
            <MimeType>text/plain</MimeType>
        </Annex>
        </Content>
        <ContentSpecification>
            <ContentType>DOCUMENT</ContentType>
            <IsImportant>true</IsImportant>
            <IsEncrypted>true</IsEncrypted>
            <PublicationReceipt>true</PublicationReceipt>
            <ReceivedReceipt>true</ReceivedReceipt>
            <ReadReceipt>true</ReadReceipt>
        </ContentSpecification>
        </ContentContext>
        <CopyMailTo>me@test.be</CopyMailTo>
    </urn:SendMessageRequest>
</soapenv:Body>
</soapenv:Envelope>

```

Response:

```

<ehboxp:SendMessageResponse Id="5H0111043267W"
SentPublicationId="1009927801393"
xsi:schemaLocation="urn:be:fgov:ehhealth:ehbox:publication:protocol:v3
ehhealth-ehBox-publication-schema-protocol-2_0.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ehboxp="urn:be:fgov:ehhealth:ehbox:publication:protocol:v3">
    <Status>
        <Code>826</Code>
        <Message Lang="EN">One or more recipients have an Out-Of-Office
active.</Message>
    </Status>
    <Recipient>

```

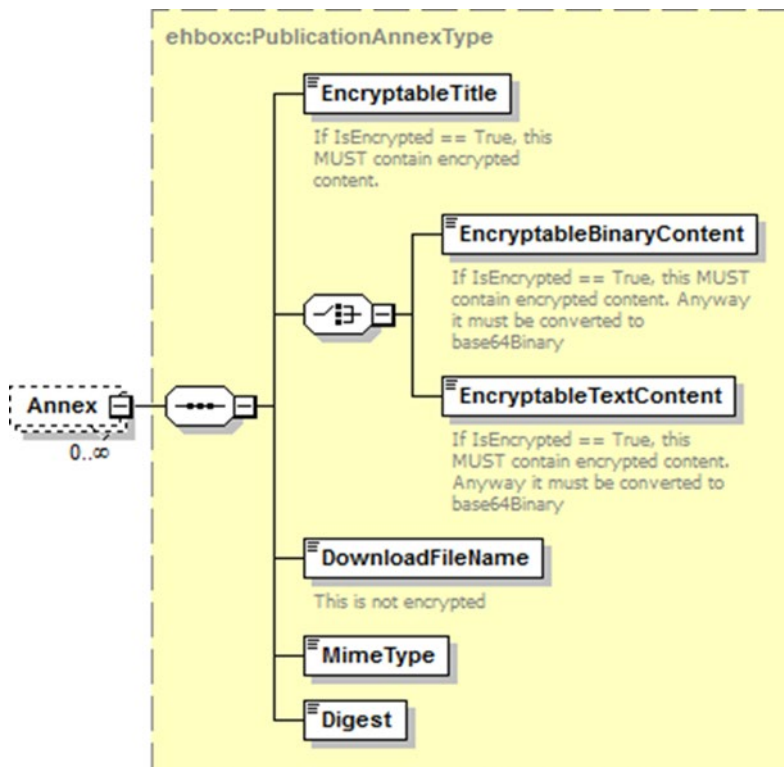
```

<Id>987654321</Id>
<Type>NIHII</Type>
<Quality>DOCTOR</Quality>
<AbsentFrom>2012-01-01+01:00</AbsentFrom>
<AbsentTo>2012-01-04+01:00</AbsentTo>
<Substitute>
  <Id>1111111111</Id>
  <Type>INSS</Type>
  <Quality>DOCTOR</Quality>
</Substitute>
<Substitute>
  <Id>2222222222</Id>
  <Type>INSS</Type>
  <Quality>DOCTOR</Quality>
</Substitute>
</Recipient>
</ehboxp:SendMessageResponse>

```

5.3.6 Used Types

5.3.6.1 Annex



Field name	Descriptions
EncryptableTitle	An <i>Annex</i> has an <i>EncryptableTitle</i> , a human readable description of its intent (minimum 1, maximum 400 characters in binary form). If <i>IsEncrypted</i> is true (see Section 5.3.6.5), the title must be encrypted (see section 5.1.2).

EncryptableBinaryContent	Contains a SOAP reference to the attachment Eg. <EncryptableBinaryContent cid:4906633928/>. Type is xsd:anyURI. If <i>IsEncrypted</i> is true (see Section 5.3.6.5), the attachment must be encrypted (see section 5.1.2).
EncryptableTextContent	Text content, in binary form. If <i>IsEncrypted</i> is true (see Section 5.3.6.5), the text content must be encrypted (see section 5.1.2).
DownloadFileName	E.g. "principal.pdf" (string minimum 1, maximum 255).
MimeType	Represents the mime type of the content. E.g. "application/pdf", "text/plain", "application/octet-stream" (string minimum 1, maximum 255).
Digest	Mandatory element. The SHA-256 hash of the content after encryption (if encrypted) in binary form. E.g. <Digest>w1+mJdkfliLwJeo4ySUHDu/2nwuXJwBpj9atfTDZZig=</Digest>

5.3.6.2 BoxId

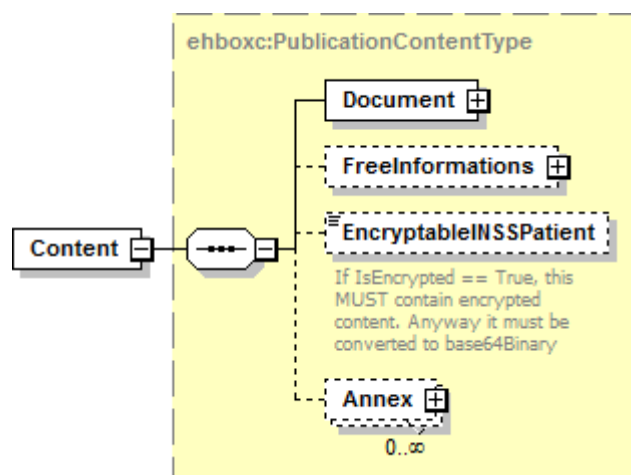
A *BoxId* contains all the information on the eHealthBox the client wants to use for the request.

BoxId cannot be used when authenticated as an organization. You will find all mandatory information about the allowed combinations Id-Type- Quality.

Field name	Description
ID	Your eHealthBox's identification number: a digital number representing an INSS, NIHII, FAMPH, or CBE. String.
Type	Your eHealthBox's ID type ("INSS", "NIHII", "FAMPH" or "CBE"). String.
Subtype (obsolete)	If the recipient is an organization, the <i>Subtype</i> allows (if necessary) further specification (such as "HOSPITAL" <i>SubType</i> for a Hospital <i>Quality</i> , or "GROUP" <i>SubType</i> for a Group <i>Quality</i>). String.
Quality	Your eHealthBox's <i>Quality</i> . String (see eHBox_Quality v1.01 20180227.docx)

5.3.6.3 Content

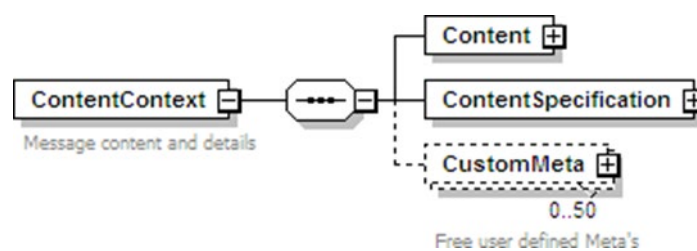
A *Content* contains the message content (a document or a news item) and optionally some free information, a Patient INSS and some annexes.



Field name	Description
Document	A news item is now a special kind of Document. See section 5.3.6.8
FreeInformations	See section 5.3.6.9
EncryptableINSSPatient	This optional field allows specifying an INSS number of a patient concerned by the message content. If <i>IsEncrypted</i> is true (see Section 5.3.6.5), the INSS number must be encrypted (see section 5.1.2).
Annex	See section 5.3.6.1

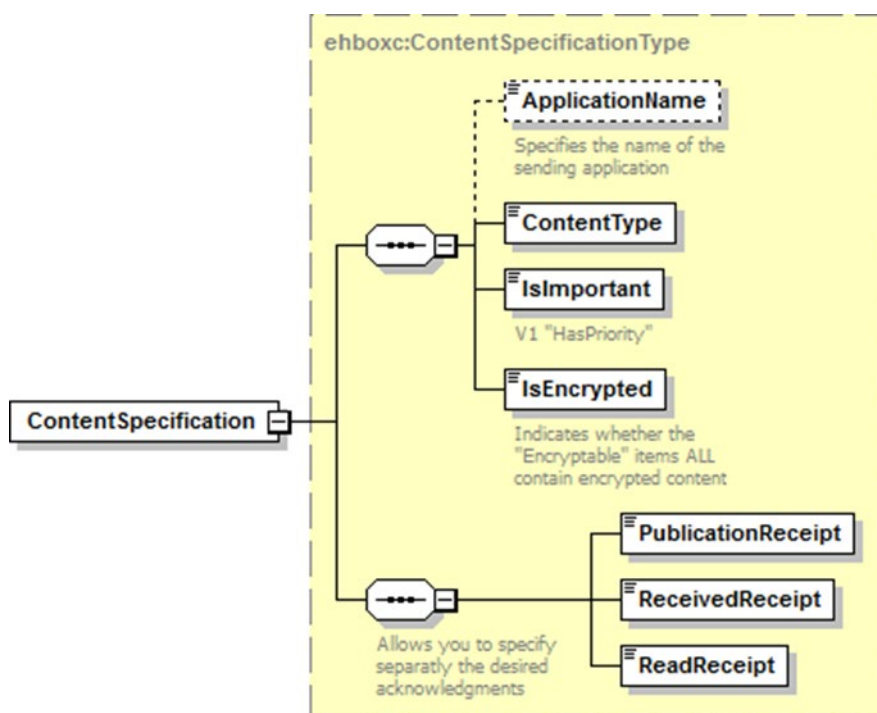
5.3.6.4 ContentContext

A *ContentContext* contains the message content and message details, as well as zero-or-more (50 maximum) free *CustomMetas*. The user can freely specify these *CustomMetas* for internal usage. You can define a Key and a value for each *CustomMeta* (see 5.3.6.6).



Field name	Descriptions
Content	See section 5.3.6.3.
ContentSpecification	See section 5.3.6.5.
CustomMeta	See section 5.3.6.6

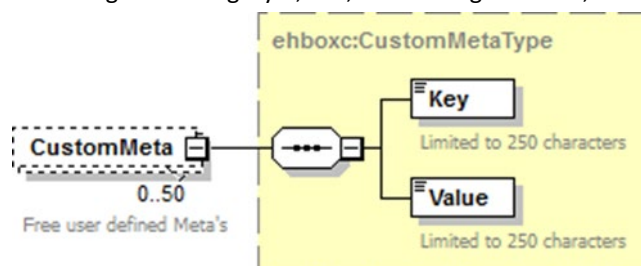
5.3.6.5 ContentSpecification



Field name	Description
ApplicationName	The application sending the message (optional, string minimum 1, maximum 25).
ContentType	The <i>Content Type</i> of the message ("DOCUMENT", "NEWS").
IsImportant	BOOLEAN (true or false) that indicates if the message is to be considered as important.
IsEncrypted	BOOLEAN (true or false) that indicates if the content has been encrypted.
PublicationReceipt	BOOLEAN (true or false) that indicates if a publication receipt is requested. A message is returned to the sender's eHealthBox when it is stored in the database. Still, the recipient may not be able to view it if his eHealthBox is full.
ReceivedReceipt	BOOLEAN (true or false) that indicates if a received receipt is requested. A message is returned to the sender's eHealthBox when the recipient has viewed the message. (Call <i>GetMessagesList</i>).
ReadReceipt	BOOLEAN (true or false) that indicates if a read receipt is requested. A message is returned to the sender's eHealthBox when the message is read by the recipient (Call <i>GetFullMessage</i>).

5.3.6.6 CustomMeta

CustomMeta was developed in order to enable the client to transport any Meta information relative to the message he wants. You can specify a maximum of 50 different pairs (key, value). The fields are each limited to 250 characters. Those CustomMetas will be transported from the sender to the recipient. You can e.g. example add a CustomMeta for internal usage as "CategoryId, 17", or "MessageContent, Blood analysis".

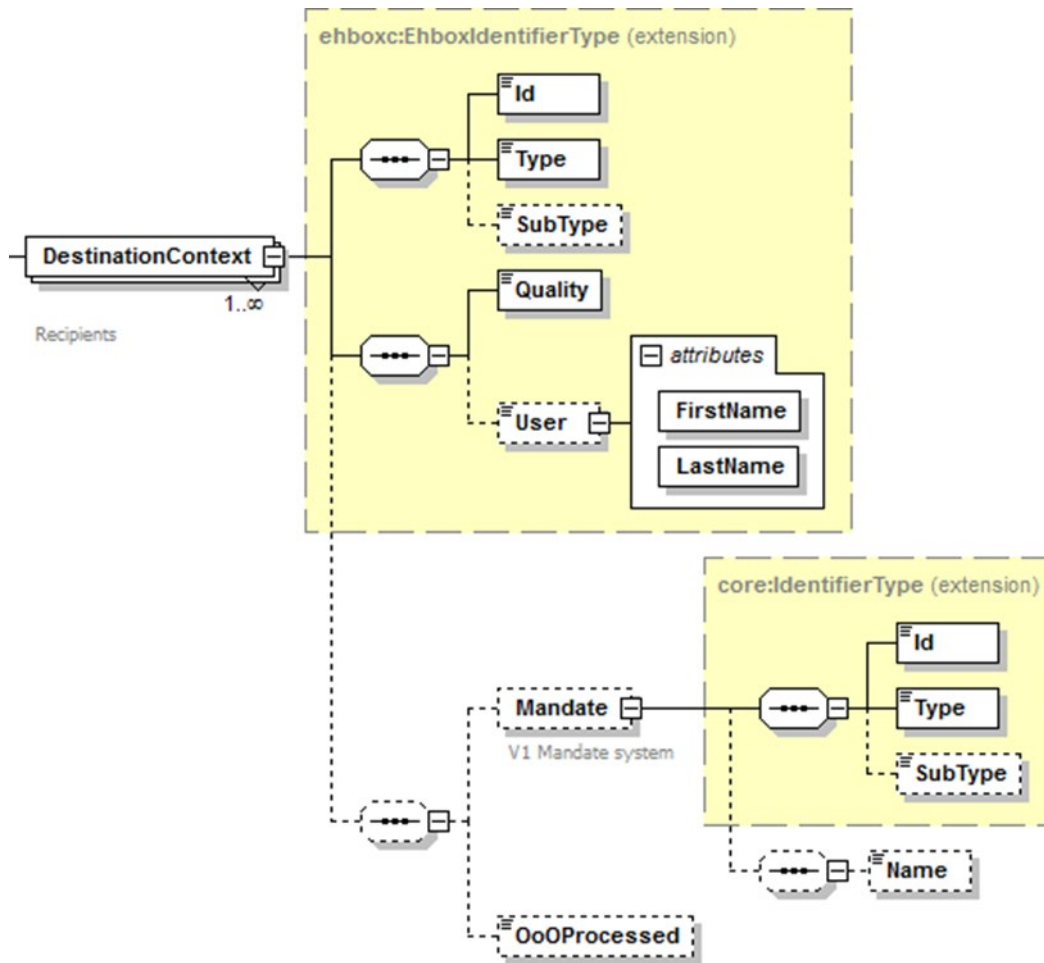


Field name	Description
Key	Alphanumeric string used as a key (string minimum 1, maximum 250).
Value	Alphanumeric string value corresponding to the Key (string minimum 1, maximum 250).

5.3.6.7 DestinationContext

A DestinationContext contains all the information on the recipient.

You will find all mandatory information about the allowed combinations Id-Type- Quality. If you would prefer to publish your document to a list of recipients sharing the same "Quality" (e.g. to all Nurses), please refer to Annex 1 – Publish a message to a list of professionals.



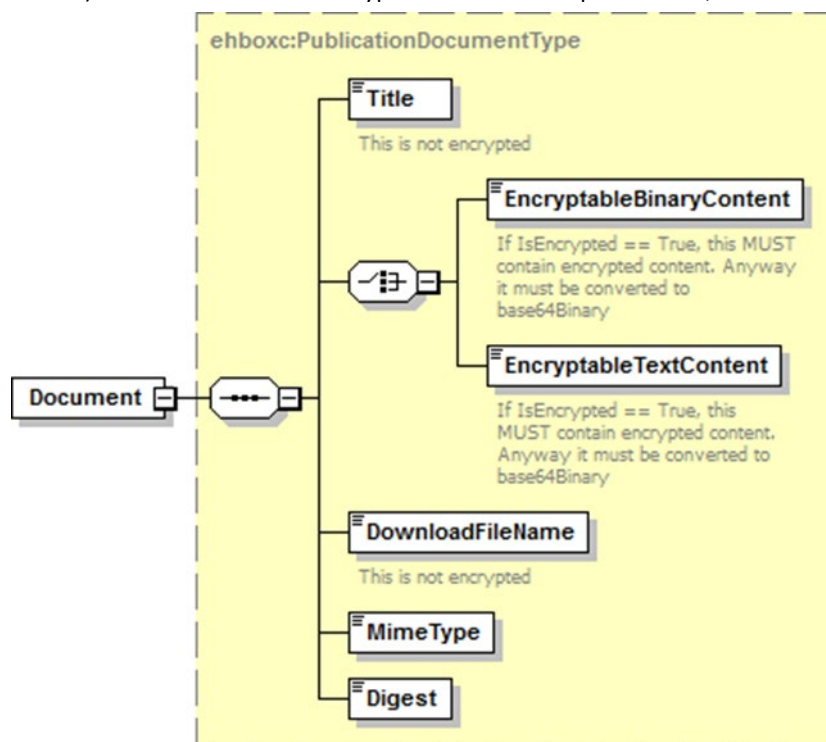
Field name	Description
ID	The recipient's identification number or "ALL" (publication to a "Quality"). This is a digital number representing an INSS, NIHII, FAMPH, or CBE. String.
Type	The recipient's ID type ("INSS", "NIHII", "FAMPH" or "CBE"). String.
Subtype (obsolete)	If the recipient is an organization, the <i>Subtype</i> allows (if necessary) further specification (such as "HOSPITAL" <i>SubType</i> for a Hospital <i>Quality</i> , or "GROUP" <i>SubType</i> for a Group <i>Quality</i>). String
Quality	A <i>Quality</i> defines the recipient's eHealthBox and is mandatory. String.
User	An optional <i>User</i> (<i>FirstName</i> and <i>LastName</i>) can be added in the destination context. In case of a publication to an organization, this field is used to specify a member of this organization (e.g. a doctor working in a hospital), (string minimum 1, maximum 100).
Mandate (obsolete)	Optional authority information will be added if the recipient has been granted an authority. The constituent's identification number (<i>Id</i>) and <i>Type</i> are requested. If the constituent is an organization, the <i>Subtype</i> allows (if necessary) further specification (such as "HOSPITAL" <i>SubType</i> for a Hospital <i>Quality</i> , or "GROUP" <i>SubType</i> for a Group <i>Quality</i>). The recipient's name may be specified.

OoOProcessed	Optional BOOLEAN, set to false by default. If during a previous call to SendMessageRequest an Out-of-Office notification was received and this notification has been processed, OoOProcessed should be set to true in order to avoid receiving the same notification again. Although, a new OoO notification could be received for one of the substitutes inserted in recipients.
--------------	---

5.3.6.8 Document

Please note that a message will contain either a News item OR a Document, not both.

New in V3: News items are now “updatable documents”. You can indicate that the document can be updated by specifying it (“NEWS”) in the element ContentType under ContentSpecification, see section 5.3.6.5.

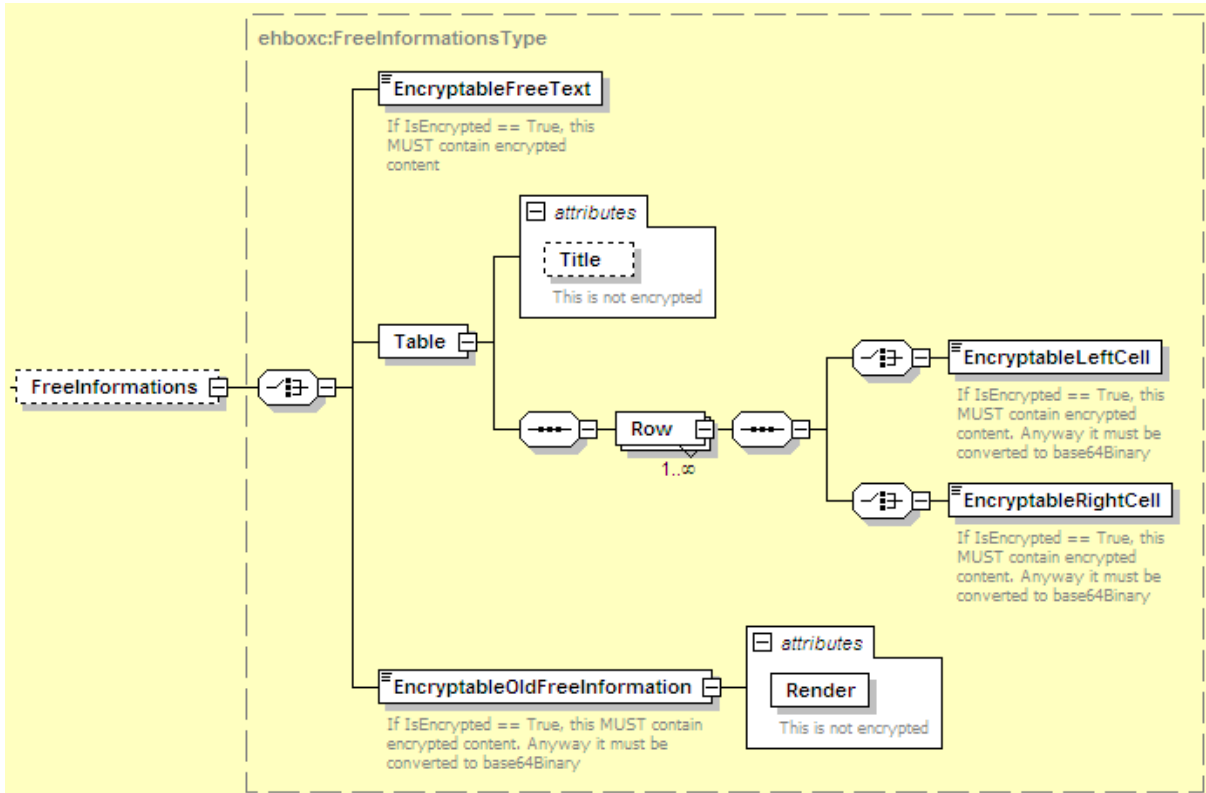


Field name	Description
Title	A Document has a <i>Title</i> , a human readable description of its content (string minimum 1, maximum 400).
EncryptableBinaryContent	Contains a SOAP reference to the attachment Eg. <code><EncryptableBinaryContent>cid:4906633928</EncryptableBinaryContent></code> . Type is <code>xsd:anyURI</code> . If <i>IsEncrypted</i> is true (see Section 5.3.6.5), the attachment must be encrypted (see section 5.1.2).
EncryptableTextContent	Text content, in binary form. If <i>IsEncrypted</i> is true (see Section 5.3.6.5), the text content must be encrypted (see section 5.1.2).
DownloadFileName	E.g. “principal.pdf” (string minimum 1, maximum 255).
MimeType	Represents the mime type of the content. E.g. “application/pdf”, “text/plain”, “application/octet-stream” (string minimum 1, maximum 255).

Digest	Mandatory element. The SHA-256 hash of the content after encryption (if encrypted) in binary form. E.g. <Digest>w1+mJdkfliLwJeo4ySUHDu/2nWuXJwBpj9atfTDZZig=</Digest>
--------	--

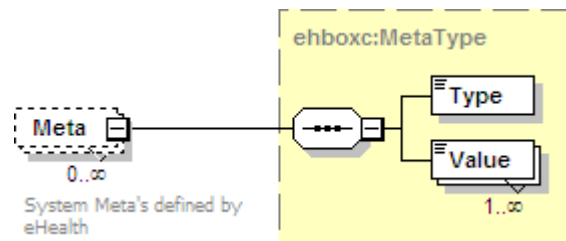
5.3.6.9 FreeInformations

The sender is free to add more information via the *FreeInformations* field. These *FreeInformations* will be provided transparently to the recipient(s).



Field name	Description
EncryptableFreeText	Text content, in binary form. If <i>IsEncrypted</i> is true (see Section 5.3.6.5), the text must be encrypted (see section 5.1.2).
Table	<i>Title</i> (the title of the table) and 1 or more row(s) (each row has a <i>EncryptableLeftCell</i> and a <i>EncryptableRightCell</i> as string). If <i>IsEncrypted</i> is True (see Section 5.3.6.5). The content must be encrypted (see section 5.1.2).
EncryptableOldFreeInformation (obsolete)	Text content, in binary form and a render attribute (as a string). If <i>IsEncrypted</i> is true (see Section 5.3.6.5), the content must be encrypted (see section 5.1.2).

5.3.6.10 Meta



Field name	Description
Type	The type of the meta information (string minimum 1, maximum 250).
Value	A list of <i>Values</i> for this <i>Type</i> (string minimum 1, maximum 250).

5.3.7 Extensions and Encoding (BODY vs ANNEX)

For compatibility with the legacy interface, and for service performance, the interpretation of the CONTENT of the `SendMessageRequest` request (in a BODY or an ANNEX) **depends on the type of content** specified in the request (`EncryptableTextContent` VS `EncryptableBinaryContent`, `MimeType` and extensions of file)

If the encoding and extensions in the requests are not as expected, the eHealthBox backend and the WebApp will "transform" the BODY content into an ANNEX for the recipient.

5.3.7.1 Send correct BODY

Here are the (mandatory) conditions for the sending of a request and a document to be interpreted as a BODY in good shape, and visible as such in a web application :

- 1) Use `<Document>` element in the interface (NOT `<Annex>` element)
- 2) Use `EncryptableTextContent` in `<Document>` (NOT `EncryptableBinaryContent`)
- 3) The `MimeType` must be `text/plain` or `text/html`
- 4) The extension of the file must be `.txt` or `.html`

In other cases, even when using the `<Document>` element in the interface, the content will be interpreted as an appendix which will be renamed "BODY" (displayed as an annex in the webapp)

Example:

```
<Document>
  <EncryptableTitle>U2Now6ltYSBQTEFTTUeGLSAwNjA1MjAyMSAtIHYYLnBuZw==</EncryptableTitle>
  <EncryptableTextContent> cid:7c6786ce-4b5c-4713-a6f8-930b409d7238@apache.orgEncryptableTextContent>
  <DownloadFileName>Schema image.png</DownloadFileName>
  <MimeType>image/png</MimeType>
  <Digest>n4bQgYhMfWWaL+qgxVrQFaO/TxsrC4ls0V1sFbDwCgg=</Digest>
</Document>
```

5.3.7.2 Send correct ANNEX

To correctly send an annex, you must use the `<Annex>` element of the interface and `EncryptableBinaryContent`. `MimeType` and file extensions depend on annex type.

Example :

```
<Annex>
  <EncryptableTitle>NTAwIGtiGFubmV4</EncryptableTitle>
  <EncryptableBinaryContent>cid:465657487</EncryptableBinaryContent>
  <DownloadFileName>annex1.txt</DownloadFileName>
  <MimeType>text/plain</MimeType>
  <Digest>R8s1tOUlf85OH7stne7f3maueY1GhgPKLeJaKE52gXU=</Digest>
</Annex>
```

5.3.7.3 Types and extensions files

Non-exhaustive list of examples, the eHealthBox service normally allows the sending of annexes of all types.

MimeType	Extension File
text/plain	.txt
text/html	.html
image.jpg	.jpg
image.png	.png
application/pdf	.pdf
application/document	.docx
Etc...	Etc...

5.3.8 Txt Size in XML tags

In addition to the total message size limit (30mb), there is another existing technical limit that limits the txt content for **each XML tag to 10mb**

If an XML tag contains more than 10mb, an **“SOA-03001: Malformed message”** error will occur.



6. Risks and security

6.1 SOAP with attachments

The standard is used to transfer binary data via SOAP WS. SOAP with attachments is an older alternative to MTOM. SwA is used in combination with WS-Security.

The SwA specification describes how to send binary data separated from the SOAP body using multipart MIME messages.

SOAP Messages with Attachments is a link to the W3C specification of the standard:

<http://www.w3.org/TR/SOAP-attachments>

Attachments Profile Version 1.0 is a link to the WS-I profile with clarifications and amendments to the SwA specification:

<http://www.ws-i.org/Profiles/AttachmentsProfile-1.0.html>

To keep a reference to the attachment from within the message payload, WS-I defined the simpleType SwA:

<http://ws-i.org/profiles/basic/1.1/swaref.xsd>

The attachment data is no part of the message payload. Therefore, when you sign the SOAP body for secure transfer, the attachment data itself is not signed.

6.2 SecurityBusiness security

In case the development adds a use case based on an existing integration, the eHealth platform must be informed at least one month in advance. A detailed estimate of the expected load is necessary to be able to ensure an effective capacity management.

When technical issues occur on the WS, the partner can obtain support from the contact centre (see Chap 3)

If the eHealth platform should find a bug or vulnerability in its software, the partner must update his application with the latest version of the software, within ten (10) business days.

If the partner finds a bug or vulnerability in the software or web service made available by the eHealth platform, he is obliged to contact and inform us immediately. He is not allowed, under any circumstances, to publish this bug or vulnerability.

6.2.1 Web service

WS security used in this manner is in accordance with the common standards. Your call will provide:

- SSL one way
- "Time-to-live" of the message: one minute.
- Signature of the timestamp, body and binary security token. This allows the eHealth platform to verify the integrity of the message and the identity of its author.
- No encryption on the message.

6.2.2 The use of username, password and token

The username, password, and token are strictly personal.

Every user takes care of his username, password and token, and he is forced to confidentiality of it. It is prohibited to transfer them to partners and clients. Until inactivation, every user is responsible for every use, including the use by a third party.

7. Test and release procedure

7.1 Procedure

This chapter explains the procedures for testing and releasing an application in acceptance or production.

7.1.1 Initiation

If you intend to use the eHealth platform service, please contact info@ehealth.fgov.be. The project department will provide you with the necessary information and mandatory documents.

7.1.2 Development and test procedure

You have to develop a client in order to connect to our WS. Most of the information needed to integrate is published on the portal of the eHealth platform.

Upon request and depending on the case, the eHealth platform provides you with a **test case** in order for you to test your client before releasing it in the acceptance environment.

7.1.3 Create test cases

Rules to access the Publication services are the same in test and in production. Access rules:

- to use the Publication services, the user must have one of the following profiles: **hospital, nurse, group, institution, doctor, laboratory ...**
- authentication with a certificate

All test cases are configured by the eHealth platform development team;

7.1.3.1 Request a certificate

Prior to requesting the certificate, you need to have installed the latest version of Java 1.6 and the Belgium eID middleware. You also need a smart-card reader and a Belgian eID.

The process is described in the “How to request an eHealth platform test certificate”¹.

Depending on the user, you will need NIHII, INSS or CBE identification numbers in order to obtain the certificate.

7.1.3.2 Obtain SAML token

The usage of the STS and the structure of the exchanged xml-messages are described in the eHealth STS cookbook¹.

In the case of eHealthBox Publication WS, see eHealthBox – SSO v1.2.

7.1.4 Release procedure

When development tests are successful, you can request to access the acceptance environment of the eHealth platform.

From the moment you start the integration and acceptance tests, the eHealth platform suggests testing during at least one month.

After the acceptance tests have been successfully completed, the partner sends his test results and performance results with a sample of the “eHealth request” and “eHealth answer” to the eHealth platform point of contact by email.

Then the eHealth platform and the partner agree on a release date. The eHealth platform prepares the connection to the production environment and provides the partner with the necessary information. During the release day, the partner provides the eHealth platform with feedback on the test and on the performance tests.

For further information and instructions, please contact: info@ehealth.fgov.be.



7.1.5 Operational follow-up

Once in production, the partner using the eHealth platform service for one of his applications will always perform tests first in the acceptance environment before releasing any adaptations of his application in production. In addition, he will inform the eHealth platform on the progress and the test period.

7.2 Test cases

This section describes a systematic process to test the Publication WS.

The eHealth platform recommends performing tests for all of the following cases:

- Publishing a document message to the sender's eHealthBox and get a successful response.
- Publishing a news item message to the sender's eHealthBox and get a successful response. In addition, the organization should also run negative test cases:
- Publishing a wrong xml input and get an error response message: XSD compliance failure (see section 8).

8. Error and failure messages

8.1 Send Message Response Status Codes

Error codes originating from the eHealth platform:

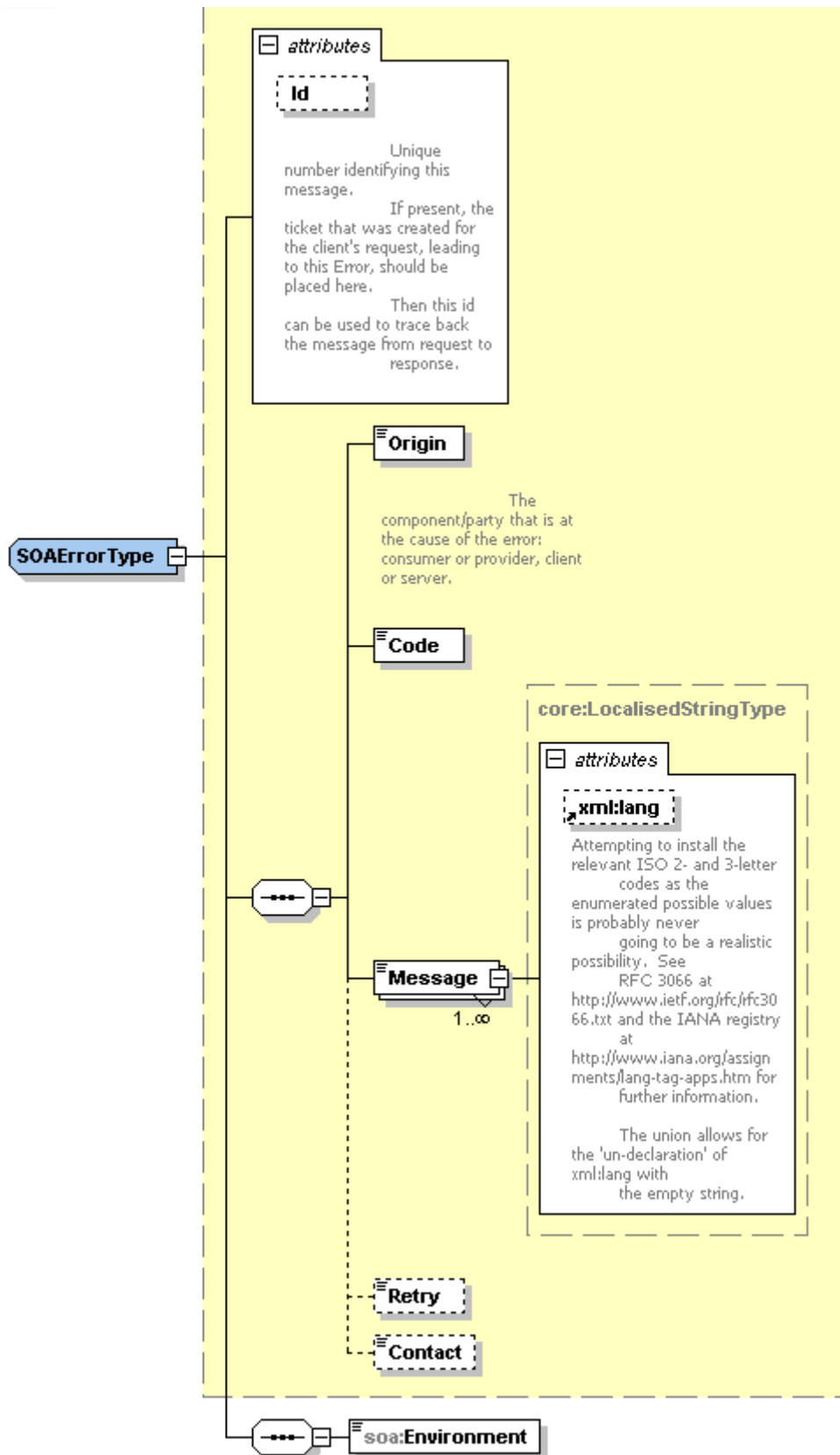
These error codes first indicate a problem in the sent arguments. This table can evolve.

Error code	Component	Description	Solution
100	SendMessage	SUCCESS	
801	SendMessage	The message exceeds the maximum authorized size.	Reduce appendixes size. Split the message and send multiple messages.
802	SendMessage/ DestinationContext	The specified Identifier is invalid; please verify the ID.	Is the NISS, NIHII, CBE valid and known by the eHealth platform?
803	SendMessage/ DestinationContext	The specified quality is invalid; please verify that quality is a quality recognized by the system.	Verify allowed combination in eHBox_Quality v1.01 20180227
804	SendMessage/ DestinationContext	The specified type is invalid; please verify that the type is a type recognized by the system.	Verify allowed combination in eHBox_Quality v1.01 20180227
805	SendMessage/ DestinationContext	The specified subtype is invalid; please verify that subtype is recognized by the system.	Verify allowed combination in eHBox_Quality v1.01 20180227
810	SendMessage/ BoxId	The specified BoxId is invalid; please verify the data and that you can access it.	Can you normally access that eHealthBox?
814	SendMessage/ DestinationContext	You are not authorized to publish to this Quality.	Contact the eHealth platform to get authorization.
816	SendMessage / Document or Annex	The digest of the streamed document does not match the supplied one.	Recalculate the SHA-256 hash on the content.
826	SendMessage	One or more recipients have an Out-Of- Office active.	Check the returned substitutes for the absent recipient.
907	SendMessage/Annex	The message exceeds the limit of total annexes count.	Reduce the number to 25 maximum.

8.2 Soap Fault Error Codes

They contain the following attributes:

Field name	Descriptions
Id	Unique number identifying this message. If present, the ticket that was created for the client's request, which lead to this error. Then this Id can be used to trace back the message from the request.
Origin	The component/party causing the error: consumer or provider, client or server.
Code	The error code
Message	A human readable message
Retry	An optional BOOLEAN that indicates if it is worth resending the same request.
Contact	An optional field specifying a contact description.
Environment	The eHealth platform environment in which the error occurs: integration, acceptance, or production.



8.2.1 Schema Validation Errors

When invoking the WS, you must provide a valid XML.

Before executing any action, the eHealthBox system verifies if the XML is valid by running a validation check towards the SendMessageRequest XSD.

If the validation fails, a SOAP Fault is returned with the following code and message:

Code	Message
SOA-03006	XSD compliance failure

Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
<soapenv:Body xmlns:wssu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wsssecurity-utility-1.0.xsd" wsu:id="id-6">
<soapenv:Fault>
<faultcode>soapenv:Client</faultcode>
<faultstring>SOA-03006</faultstring>
<detail>
<soa:SystemError xmlns:soa="urn:be:fgov:health:errors:soa:v1" Id="5bbd8a2a-
bb21-4cf8-99bc-8d52c18e2801">
<Origin>Consumer</Origin>
<Code>SOA-03006</Code>
<Message xml:lang="en">XSD compliance failure.</Message>
<soa:Environment>Production</soa:Environment>
</soa:SystemError>
</detail>
</soapenv:Fault>
</soapenv:Body>
</soapenv:Envelope>
```

8.2.2 Technical Errors

Technical errors are errors inherent to the internal working of the eHealth WS. These errors can also occur when the token used to call the WS is not valid.

They contain the standard SOAP Fault attributes.

The table provides the different codes and messages returned in a SOAP fault message:

Code	Message
SOA-00001	An internal error has occurred. Please contact the Contact Center.

This list can evolve.

Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
<env:Body xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
<env:Fault>
<faultcode>soapenv:Server</faultcode>
<faultstring>SOA-00001</faultstring>
<detail>
<soa:SystemError Id="ec582704-d623-4b05-ab7f-
98d5c9706dd1"
xmlns:soa="urn:be:fgov:health:errors:soa:v1">
<Origin>Server</Origin>
<Code>SOA-00001</Code>
```



```

<Message xml:lang="en">An internal error has occurred. Please contact service desk.</Message>
<soa:Environment>Production</soa:Environment>
</soa:SystemError>
</detail>
</env:Fault>
</env:Body>
</soapenv:Envelope>

```

This list is able to evolve.

8.3 Error codes originating from the eHealth platform:

These error codes first indicate a problem in the arguments sent, or a technical error.

Error code	Component	Description	Solution
SOA-03001	<i>Consumer</i>	<i>This is the default error for content related errors in case no more details are known.</i>	Malformed message
SOA-03002	<i>Consumer</i>	<i>Message does not respect the SOAP standard.</i>	Message must be SOAP
SOA-03003	<i>Consumer</i>	<i>Message respects the SOAP standard, but body is missing.</i>	Message must contain SOAP body

9. Annex 1 – Publish a message to a list of professionals

It is now possible to publish a message to all doctors, dentists, nurses, practical nurses (this list is able to evolve) at once. This list of professionals is kept up to date on a daily basis.

However, there are some limitations of use:

- Prior permission of the eHealth platform is needed (see chapter 5.1).
- When you send a message to a list of professionals, you may only address the message to one list (e.g. to “all nurses” and not “all nurses and all dentists”). In other words, one “SendMessageRequest” may contain only one “DestinationContext” addressed to a list of professionals (Id = “ALL”).
- If needed, you can add some recipients (not containing Id = “ALL”) to the same request, like in the example below.
- Please be aware that this type of request requires a lot of system resources and time to complete (up to 1 hour). Please use it with caution and moderation as it could negatively affect other messages.

Example (partial) request:

```
<DestinationContext>
  <Id>ALL</Id>
  <Type>INSS</Type>
  <Quality>NURSE</Quality>
</DestinationContext>
<DestinationContext>
  <Id>333333333333</Id>
  <Type>INSS</Type>
  <Quality>NURSE</Quality>
</DestinationContext>
<DestinationContext>
  <Id>222222222222</Id>
  <Type>INSS</Type>
  <Quality>DOCTOR</Quality>
</DestinationContext>
```