

**Cookbook  
eHealthBox Consultation Web Service  
Version 3.0**

This document is provided to you free of charge by

**The eHealth platform**

**Willebroekkaai 38 - Quai de Willebroeck 38  
1000 BRUSSELS**

All are free to circulate this document with reference to the URL source.

# Table of contents

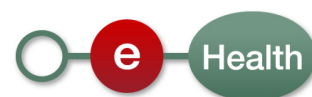
Table of contents .....	2
1 Document management .....	6
1.1 Document history .....	6
2 Introduction .....	7
2.1 Goal of the service .....	7
2.2 New in version 3.0 .....	7
2.3 Goal of the document .....	7
2.4 eHealth document references .....	8
2.5 Service history .....	8
3 Business and privacy requirements .....	9
3.1 Certificates .....	9
3.2 eHealth contact .....	9
4 Global overview .....	10
5 Step-by-step .....	11
5.1 Technical requirements .....	11
5.1.1 Use of the eHealth SSO solution .....	11
5.1.2 End-to-End Encryption for known recipient .....	11
5.2 Process overview .....	11
5.2.1 Web Service WSDL (Web Service Definition Language) .....	11
5.3 eHealthBox Consultation Web Service .....	13
5.3.1 Lifetime of a message .....	13
5.3.1 Out-of-Office system .....	13
5.3.1.1 Introduction .....	13
5.3.1.2 Out-of-Office restrictions .....	13
5.3.2 getBoxInfo Method .....	14
5.3.2.1 getBoxInfo Request .....	14
5.3.2.2 getBoxInfo Response .....	14
5.3.2.3 Example .....	16
5.3.3 getMessagesList Method .....	16
5.3.4 getMessagesList Request .....	17
5.3.4.1 getMessagesList Response .....	17
5.3.4.2 Example .....	19
5.3.5 GetAllEhboxesMessagesList Method .....	20
5.3.5.1 GetAllEhboxesMessagesList Request .....	20
5.3.5.2 GetAllEhboxesMessagesList Response .....	21
5.3.5.3 Example .....	23
5.3.6 getFullMessage Method .....	24
5.3.6.1 getFullMessage Request .....	24



5.3.6.2	getFullMessage Response .....	25
5.3.6.3	Example .....	26
5.3.7	MoveMessage Method .....	27
5.3.7.1	MoveMessage Request .....	28
5.3.7.2	MoveMessage Response .....	29
5.3.7.3	Example .....	29
5.3.8	DeleteMessage Method .....	30
5.3.8.1	DeleteMessage Request .....	30
5.3.8.2	DeleteMessage Response .....	31
5.3.8.3	Example .....	32
5.3.9	getHistory Method .....	33
5.3.9.1	getHistory Request .....	33
5.3.9.2	getHistory Response .....	33
5.3.9.3	Example .....	34
5.3.10	GetMessageAcknowledgmentsStatus Method .....	35
5.3.10.1	GetMessageAcknowledgmentsStatus Request .....	35
5.3.10.2	GetMessageAcknowledgmentsStatus Response .....	36
5.3.10.3	Example .....	37
5.3.11	InsertOoO Method .....	37
5.3.11.1	InsertOoO Request .....	38
5.3.11.2	InsertOoO Response .....	38
5.3.11.3	Example .....	40
5.3.12	DeleteOoO Method .....	40
5.3.12.1	DeleteOoO Request .....	41
5.3.12.2	DeleteOoO Response .....	41
5.3.12.3	Example .....	42
5.3.13	GetOoOList Method .....	42
5.3.13.1	GetOoOList Request .....	42
5.3.13.2	GetOoOList Response .....	43
5.3.13.3	Example .....	43
5.3.14	Used Types .....	44
5.3.14.1	Acknowledgment .....	44
5.3.14.2	Annex .....	45
5.3.14.3	BoxId .....	46
5.3.14.4	Content .....	46
5.3.14.5	ContentContext .....	47



5.3.14.6	ContentInfo .....	48
5.3.14.7	ContentSpecification .....	49
5.3.14.8	CustomMeta .....	49
5.3.14.9	DestinationContext .....	50
5.3.14.10	Document .....	51
5.3.14.11	Error .....	52
5.3.14.12	FreeInformations .....	52
5.3.14.13	Message .....	54
5.3.14.14	MessageInfo .....	55
5.3.14.15	Meta .....	55
5.3.14.16	Recipient .....	55
5.3.14.17	Sender .....	56
5.3.14.18	Substitute .....	57
6	Risks and security .....	59
6.1	SOAP with Attachments .....	59
6.2	Security .....	59
6.2.1	Business security .....	59
6.2.2	Web Service Security Policy .....	59
6.2.3	Security policies to apply .....	60
7	Test and release procedure .....	61
7.1	Procedure .....	61
7.1.1	Initiation .....	61
7.1.2	Development and test procedure .....	61
7.1.2.1	Create test cases .....	61
7.1.2.2	Request a certificate .....	61
7.1.2.3	Obtain SAML token .....	61
7.1.3	Release procedure .....	62
7.1.4	Operational follow-up .....	62
7.2	Test cases .....	62
8	Error and failure messages .....	63
8.1	Error Element Status Codes .....	63
8.2	Consultation Response Status Codes .....	63
8.3	Soap Fault Error Codes .....	65
8.3.1	Schema Validation Errors .....	67
8.3.2	Technical Errors .....	67
9	Annex 1 – SSO Access .....	69
9.1	Sender is an organization. ....	69
9.2	Sender is a professional .....	70
10	Annex 2 – List of Qualities .....	71
11	List of sources .....	75



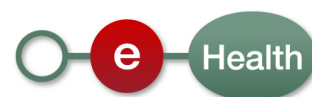
To the attention of: "IT expert" willing to integrate this web service.



# 1 Document management

## 1.1 Document history

Version	Date	Author	Description of changes / remarks
3.0	06/05/2013	eHealth	eHealthBox v3.0 Web Service.



## 2 Introduction

### 2.1 Goal of the service

The eHealthBox Consultation Web Service allows an authenticated user to consult information about the content associated with his eHealthBox.

A user can get general information on his eHealthBox, a list of messages for a specific folder and the content of a specific message. He can also move a message to his inbox and handle his out-of-offices.

Field 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 shouldn't be used by new partners. They don't provide any 'extra' feature.

The size of a message and of an eHealthBox is currently limited to 10MB, **on inbox and trash bin folder**. Note that an encrypted message weighs more due to the encryption overhead.

### 2.2 New in version 3.0

- A new "Out-of-Office" system was implemented. The will enable the sender to know if one of the recipients is absent and to send his message to a substitute or substitutes, so the sender's message can 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.
- Three new methods were added: InsertOoO used to add an Out-of-Office, DeleteOoO used to delete an inserted OoO, GetOoOList used to get the list of inserted OoO.
- eHealthBoxes are now limited in size on inbox and trash bin folder. A new method DeleteMessage was added in order to be able to delete message from the trash bin folder or directly from the inbox.
- SOAP with Attachments (SwA) standard is now being used in place of MTOM.
- News and Document types were merged. A news is now a special case of a document.
- ContentSpecification contains a ContentType element now, specifying the type of message. 2 new message types (error, acknowledgment) were added to the 2 existing (document, news).
- eHealthBoxes are now limited to 10MB on inbox and trash bin folder.

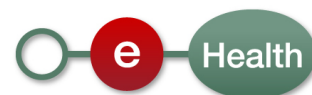
### 2.3 Goal of the document

This document provides functional and technical information about calling the eHealthBox Consultation Web Service, 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 web service. An example illustrates each of those messages. A list of possible errors can also be found in this document.

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

Some technical and legal requirements must be met in order for the eHealth web services 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; eHealth 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 partners must commit to comply with specifications, data format, and release processes described in this document.

In addition, our partners in the health 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.

## 2.4 eHealth document references

All the document references can be found in the technical library on the eHealth portal<sup>1</sup>. These versions or any following versions can be used for the eHealth service.

ID	Title	Version	Date	Author
1	Glossary.pdf	1.0	01/01/2010	eHealth
2	Cookbook STS	1.0	31/08/2010	eHealth
3	Cookbook voor end-to-end vercijfering: bekende bestemming / Cookbook destinataire connu	2.3	06/05/2011	eHealth
4	Cookbook eHealthBox Publication	3.0	07/01/2013	eHealth

## 2.5 Service history

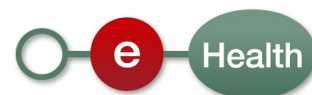
This chapter contains the list of changes applied to the service with respect to the previous version.

Previous version	Previous release date	changes
2.0	29/04/2012	Major changes: SOAP with Attachment, Out-of-Office system, DeleteMessage, MoveMessage, structured notification messages, performances.
1.0	03/02/2011	Major changes: Encryption, Multi-box, Publication WebApplication, Consultation WebApplication reviewed, general reliability ...

Remark: = "None" when the major version = 1

---

<sup>1</sup> [www.ehealth.fgov.be](http://www.ehealth.fgov.be)





## 3 Business and privacy requirements

### 3.1 Certificates

An eHealth certificate is used to identify the initiator of the request. If you don't have one, see:

Dutch version:

<https://www.ehealth.fgov.be/nl/support/basisdiensten/ehealth-certificaten>

French version:

<https://www.ehealth.fgov.be/fr/support/services-de-base/certificat-ehealth>

### 3.2 eHealth contact

For any business related question or new integration projects with the eHealth Basic Services, please contact:

eHealth Info: [info@ehealth.fgov](mailto:info@ehealth.fgov)

Available from 9:00 am till 16:30 pm

For problems with applications working in the acceptance / integration environment, please contact:

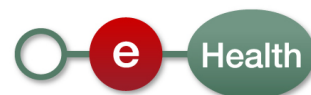
Integration Support eHealth: [integration-support@ehealth.fgov](mailto:integration-support@ehealth.fgov)

Available from 9:00 am till 16:30 pm

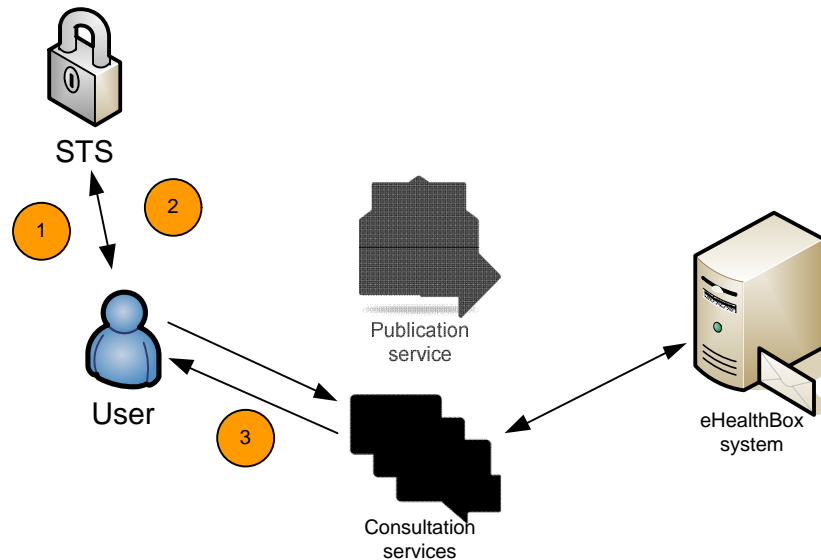
For problems with applications in the production environment, please contact:

Production Support eHealth: [support@ehealth.fgov](mailto:support@ehealth.fgov), or 02/788.51.55

Available from 7:00 am till 20:00 pm



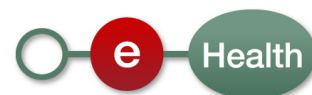
## 4 Global overview



This global overview aims to show the different steps needed to use the Consultation web services.

- Step 1. The first required step for the user to use the Consultation service is to contact the STS Service to acquire his proper token containing his data. (see Annex 1 – SSO Access and Cookbook STS)
- Step 2. Secondly, thanks to his token, the user can use the different “Consultation services” to retrieve his message(s), manage his eHealthBox or get information on it.
- Step 3. Depending on the request of the user, the “Consultation services” will provide the user with an answer concerning his eHealthBox or one of his eHealthBoxes.
- Step 4. Finally, if the message was encrypted for the recipient, the client deciphers his message with his private key and the Crypto Library (see **ETEE for known recipient Cookbook on the eHealth Portal**).

If the user wants to publish any message or reply, he is asked to please use the eHealthBox publication process.



## 5 Step-by-step

### 5.1 Technical requirements

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

#### 5.1.1 Use of the eHealth SSO solution

For more specific details, information and how the call to STS, see Annex 1 – SSO Access.

The complete overview of the profile and a step-by-step implementation to start protecting a new application with SSO @ eHealth is described in the eHealth SSO cookbook.

#### 5.1.2 End-to-End Encryption for known recipient

If an encrypted message was received, it has to be deciphered first. See Global overview.

In order to decipher the content of a message and the various fields, you have to use the local stored private key and the Crypto Libraries. Each field must be deciphered separately (one at a time).

All the information about the use of the encryption libraries is described in the ETEE for known recipient cookbook available on the eHealth technical library on the eHealth website.

**Encrypted message convention:** If an encrypted message is received, ALL "Encryptable" fields contain (one and all) encrypted content. You can detect if the message has been encrypted thanks to the element `IsEncrypted` in the responses from `GetMessagesList`, `GetAllEhboxesMessagesList` and `GetFullMessage`.

### 5.2 Process overview

#### 5.2.1 Web Service WSDL (Web Service Definition Language)

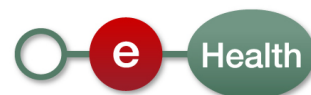
More technical information is found on the Registry website of eHealth,  
<https://services.ehealth.fgov.be/registry/uddi/bsc/web>

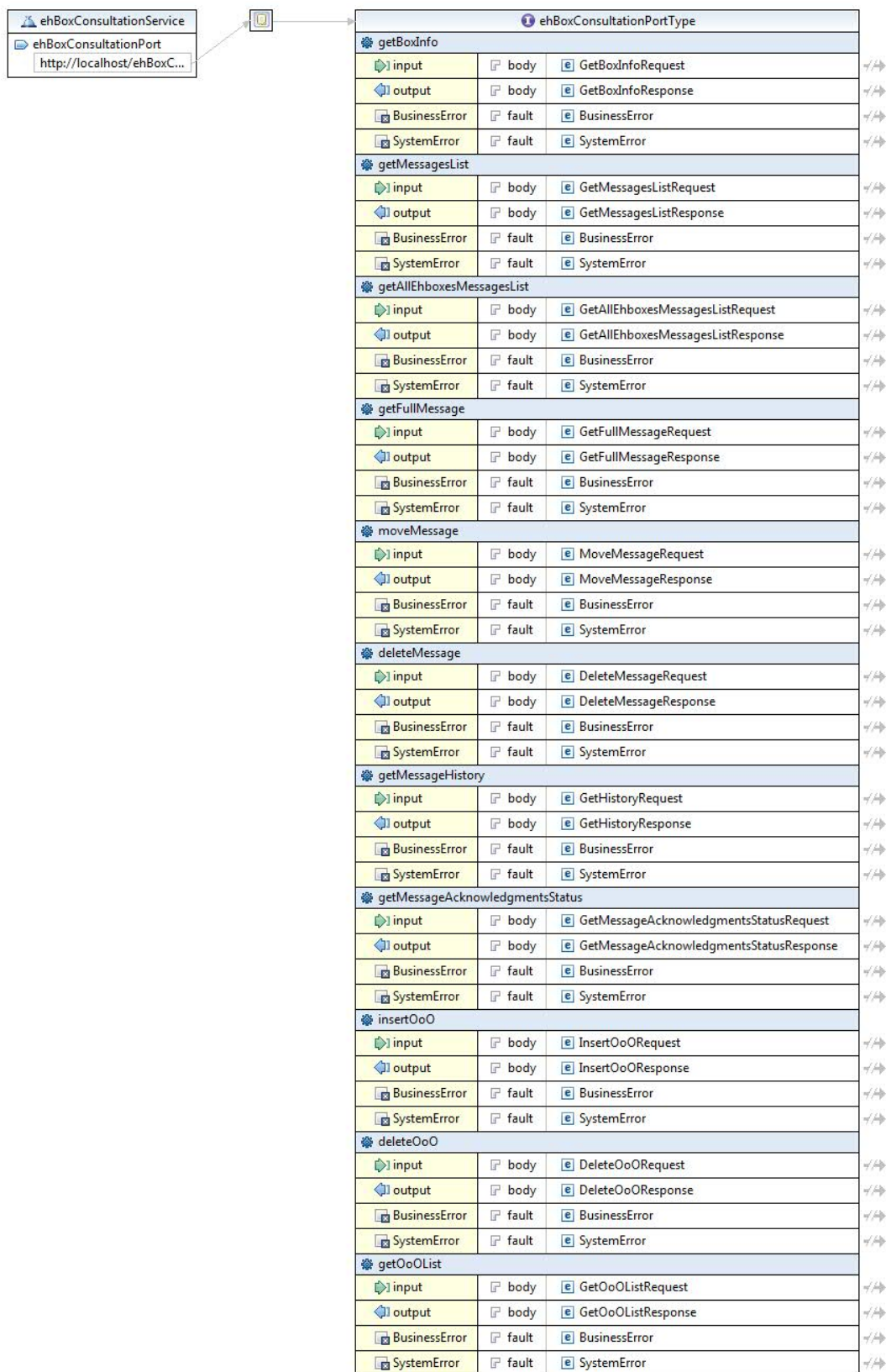
Dutch version:

[https://www.ehealth.fgov.be/nl/basic\\_service](https://www.ehealth.fgov.be/nl/basic_service)

French version:

[https://www.ehealth.fgov.be/fr/basic\\_service](https://www.ehealth.fgov.be/fr/basic_service)





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

- The applicable **Policies**, which cover the **security** aspects.
- The different methods: **getBoxInfo**, **getMessagesList**, **GetAllEhboxesMessagesList**, **getFullMessage**, **moveMessage**, **getMessageHistory**, **getMessageAcknowledgmentsStatus**, **InsertOoO**, **DeleteOoO**, **GetOoOList**.
- The types that are used by the methods.
- The fault messages are also defined for each method.

## 5.3 eHealthBox Consultation Web Service

### 5.3.1 Lifetime of a message

- When the expiration date of a message is reached AND it has already been placed in the recycle bin AND it has been read, the message is definitely removed from the application.
- When a message is older than 1 year (counted from publication date), it is definitely removed from the application, even if it hasn't been read.

### 5.3.1 Out-of-Office system

#### 5.3.1.1 Introduction

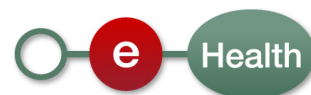
This system enables the sender to know if one of the recipients is absent and to send his message to a substitute or substitutes, so the sender's message can be treated. For example, physicians on holiday may want to ensure continuity of healthcare services for their patients. To do this, they can automatically transfer their messages to another colleague responsible during their holidays thanks to the "Out-of-Office" system.

#### 5.3.1.2 Out-of-Office restrictions

1. **A person cannot be defined as a substitute if he is already absent for this period (or part of the period).**
2. It is however possible to introduce an OoO while being a substitute for someone else.
3. A person may be substitute for several other persons.
4. The absent person and his substitutes are persons, not organizations.
5. A person is identified by its ID + quality.
6. There may be maximum 5 substitutes per OoO.
7. A maximum of 10 periods OoO may exist per eHealthBox.
8. **OoO periods may not overlap.**
9. A period may not end later than J + 1Year.
10. A period can last up to 1 Year.
11. The end date is mandatory.

A Webservice request contains one period and a maximum of 5 substitutes may be specified for that period.

The OoO activates automatically when the time period is reached.

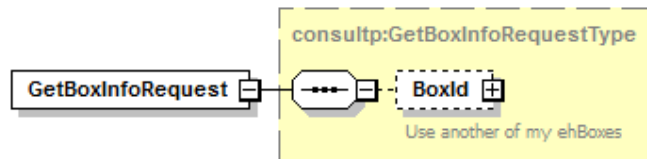


### 5.3.2 getBoxInfo Method

The getBoxInfo method allows an authenticated user to receive general information about his mailbox: the current used size of your mailbox, the maximum allowed size of your mailbox, and the number of message which couldn't be received because the mailbox was full. These messages are still waiting to be placed in your mailbox. You need to clean your mailbox until the current size is lower than the max size. The messages will then enter into your mailbox.

#### 5.3.2.1 getBoxInfo Request

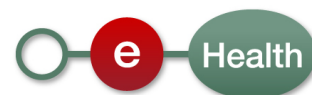
You can optionally request information of another of your mailboxes by specifying it via BoxId,



Field name	Description
BoxId	If the client wants to use another of his eHealthBoxes, he can specify it here (see section 5.3.14.3). This avoids the client having to re-authenticate himself each time.

#### 5.3.2.2 getBoxInfo Response

The response contains a success status code and general information on the eHealthBox as explained below. The BoxId element enables you to discover information on the current eHealthBox if you currently do not have any. Attention should be paid to `NbrMessagesInStandBy` larger than zero and if `CurrentSize > MaxSize`.





	<p>applicable, the Subtype is also returned.</p> <ul style="list-style-type: none"> <li>The Quality of the owner from the eHealthBox (see Annex 2 – List of Qualities).</li> </ul>
NbrMessagesInStandBy	The number of messages that are in standby because the eHealthBox is full. To consult these messages, the user has to delete others.
CurrentSize	The current size of the eHealthBox expressed in bytes.
MaxSize	The maximum size of the eHealthBox expressed in bytes.

### 5.3.2.3 Example

The following example does not contain the SAML assertion (see Annex 2 – List of Qualities).

Request:

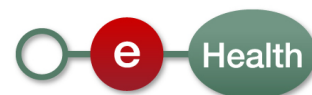
```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:urn="urn:be:fgov:health:ehbox:consultation:protocol:v3">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:GetBoxInfoRequest/>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns3:GetBoxInfoResponse
xmlns:ns2="urn:be:fgov:health:errors:service:v1"
xmlns:ns3="urn:be:fgov:health:ehbox:consultation:protocol:v3">
      <Status>
        <Code>100</Code>
        <Message Lang="EN">SUCCESS</Message>
      </Status>
      <BoxId>
        <Id>99999999964</Id>
        <Type>INSS</Type>
        <Quality>DOCTOR</Quality>
      </BoxId>
      <NbrMessagesInStandBy>0</NbrMessagesInStandBy>
      <CurrentSize>58</CurrentSize>
      <MaxSize>10485760</MaxSize>
    </ns3:GetBoxInfoResponse>
  </S:Body>
</S:Envelope>
```

### 5.3.3 getMessagesList Method

The getMessagesList method provides a list of messages for a specific folder of your eHealthBox listed in order by date (most recent first, index "1") A consequence could be that if a new message arrives between two consecutive queries, a message will be shown two times (message "1" becomes "2" , "2" becomes "3", etc.). Ex. if you requested the messages between "1" and "100" and then the messages between "101" and "200", then the message "100" would be the same as message "101".

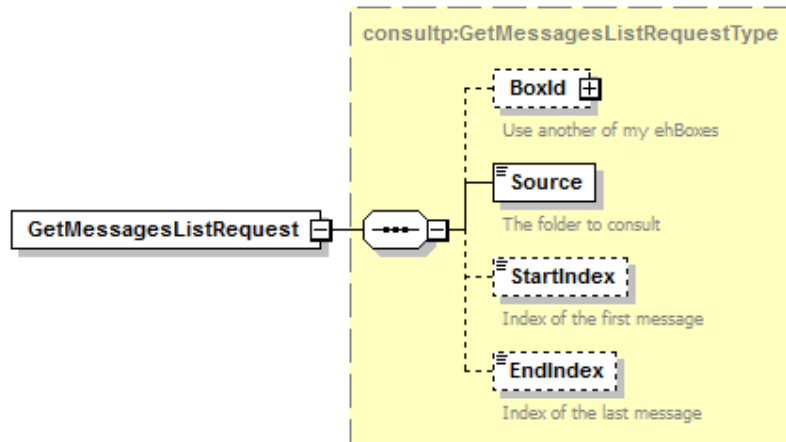




The content of the message is not yet returned by this method, but all the information needed is there in order to treat, filter, sort the messages. The sender, recipient, title message, publication date, message size, custom metas are all displayed for example.

### 5.3.4 getMessagesList Request

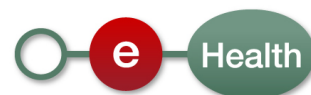
You can optionally request information of another of your mailboxes by specifying it via BoxId. This method can only return 100 messages at a time; consequently, you must use it multiple times if necessary.

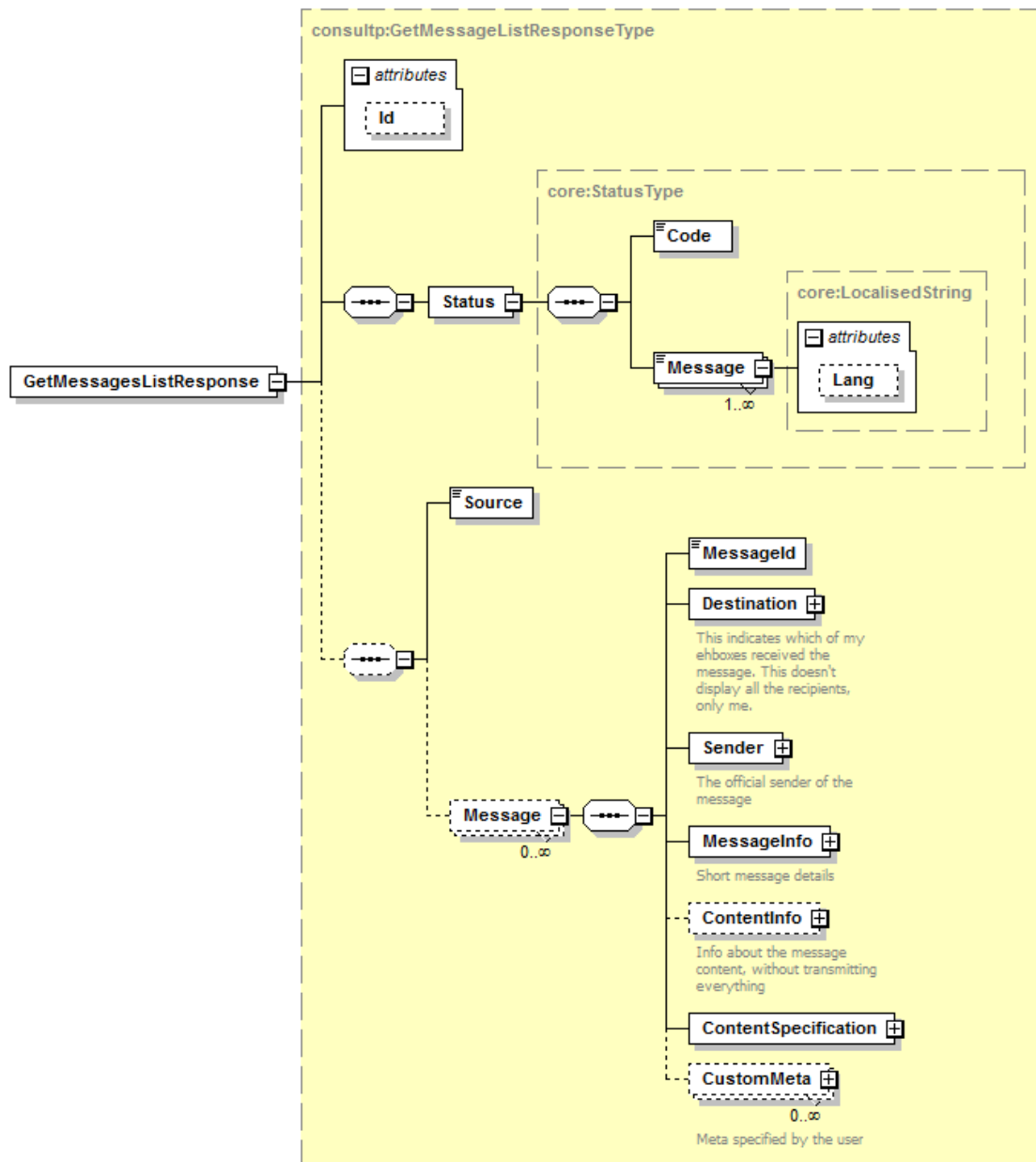


Field name	Description
BoxId	If the client wants to use another of his eHealthBoxes, he can specify it here (see section 5.3.14.3). This avoids the client having to re-authenticate himself each time.
Source	You can specify the folder specific to your request via the Source parameter.  The possible values are: <ul style="list-style-type: none"> <li>• "INBOX" for the inbox folder.</li> <li>• "SENTBOX" for the sentbox folder.</li> <li>• "BININBOX" for messages moved from the inbox folder.</li> <li>• "BINSENTBOX" for messages moved from the sentbox folder.</li> </ul>
StartIndex	Index of the first message (minimum 1).
EndIndex	Index of the last message (minimum 1). A maximum of 100 messages can be returned at once. $EndIndex < StartIndex + 100$

#### 5.3.4.1 getMessagesList Response

The response contains a success status code and as many Message elements as there are messages in the considered eHealthBox. Each element contains all necessary information to treat the message without downloading each individual message.





Field name	Description
Id	The ticket number (Id) is attributed to the exchange request/response by the eHealth platform. This is used to identify the eHealth session.
Status	<p>The Status block contains a code and a message. If no error has occurred during the transaction, the Code will be '100' and the Message 'SUCCESS'. Otherwise:</p> <p>In case of a business error:</p> <ul style="list-style-type: none"> <li>The Code is an error code that unequivocally identifies the problem (see chapter 7 for the possible values).</li> <li>The Message will be a description of the error. Each Message has a Lang (language) characteristic :</li> </ul>

	<ul style="list-style-type: none"> <li>○ "FR" : French</li> <li>○ "NL" : Dutch</li> <li>○ "EN" : English</li> <li>○ "DE" : German</li> <li>○ "NA" : Not applicable</li> </ul> <p>In case of technical errors, you will receive a Soap Fault message (see chapter 8).</p>
Source	<p>You can specify the folder specific to your request via the Source parameter.</p> <p>The possible values are:</p> <ul style="list-style-type: none"> <li>• "INBOX" for the inbox folder.</li> <li>• "SENTBOX" for the sentbox folder.</li> <li>• "BININBOX" for messages moved from the inbox folder.</li> <li>• "BINSENTBOX" for messages moved from the sentbox folder.</li> </ul>
Message	<p>0-to-more Message tag(s) describe(s) the eHealthBox Message(s). Each Message is defined by the following:</p> <ul style="list-style-type: none"> <li>• The MessageId that represents a unique message identification generated by the system and returned during publication and when calling the getMessagesList. String of 13 digits.</li> <li>• The Destination of the Message (see section 5.3.14.16).</li> <li>• The Sender of the Message (see section <b>Error! Reference source not found.</b>).</li> <li>• The MessageInfo that contains additional information about the Message such as publication date, size ... (see section 5.3.14.14).</li> <li>• The ContentInfo of the Message such as title, mime type ... (see section 5.3.14.6).</li> <li>• The ContentSpecification of the Message that contains information such as importance ... (see section 5.3.14.7).</li> <li>• The CustomMeta of the Message that contains free Meta data specified by the user (see section 5.3.14.8)</li> </ul>

### 5.3.4.2 Example

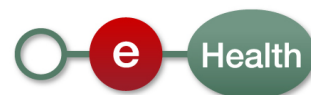
The following example does not contain the SAML assertion (see Annex 2 – List of Qualities).

#### Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:urn="urn:be:fgov:ehhealth:ehbox:consultation:protocol:v3">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:GetMessagesListRequest>
      <Source>INBOX</Source>
      <StartIndex>1</StartIndex>
      <EndIndex>100</EndIndex>
    </urn:GetMessagesListRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

#### Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
```



```

    <S:Body>
      <ns4:GetMessagesListResponse
xmlns:ns2="urn:be:fgov:ehhealth:errors:service:v1"
xmlns:ns4="urn:be:fgov:ehhealth:ehbox:consultation:protocol:v3">
        <Status>
          <Code>100</Code>
          <Message Lang="EN">SUCCESS</Message>
        </Status>
        <Source>INBOX</Source>
        <Message>
          <MessageId>9Y0002LKM100K</MessageId>
          <Destination>
            <Id>77012824158</Id>
            <Type>INSS</Type>
            <Quality>DOCTOR</Quality>
          </Destination>
          <Sender>
            <Id>71000139</Id>
            <Type>NIHII</Type>
            <Quality>HOSPITAL</Quality>
            <Name>Doe</Name>
            <FirstName>John</FirstName>
          </Sender>
          <MessageInfo>
            <PublicationDate>2011-06-28+02:00</PublicationDate>
            <ExpirationDate>2011-12-31+01:00</ExpirationDate>
            <Size>46</Size>
          </MessageInfo>
          <ContentInfo>
            <EncryptableINSSPatient>OTgwNTMwNDU3NDYyMQ0K</EncryptableINSSPatient>
            <ContentType>NEWS</ContentType>
            <Title>News in eHealthBox</Title>
            <MimeType>text/plain</MimeType>
            <HasFreeInformations>true</HasFreeInformations>
            <HasAnnex>>false</HasAnnex>
          </ContentInfo>
          <ContentSpecification>
            <IsImportant>>false</IsImportant>
            <IsEncrypted>>false</IsEncrypted>
          </ContentSpecification>
        </Message>
      </ns4:GetMessagesListResponse>
    </S:Body>
  </S:Envelope>

```

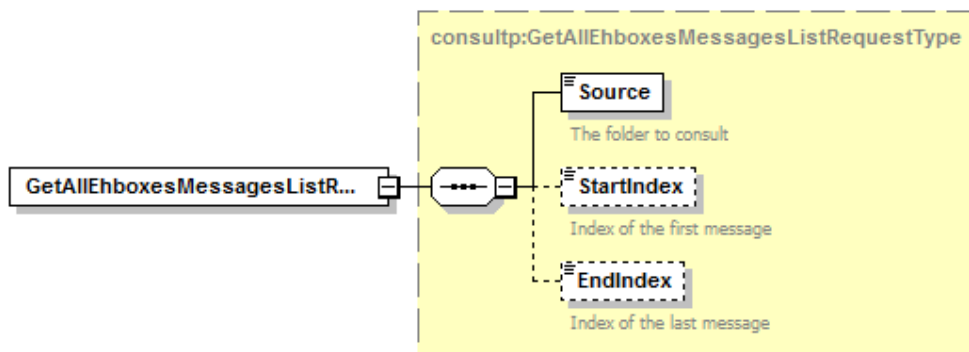
### 5.3.5 GetAllEhboxesMessagesList Method

The GetAllEhboxesMessagesList method provides a list of all the messages from all the eHealthBoxes of a user (personal and enterprise eHealthBoxes) for a specific folder. This method repeatedly calls upon the GetMessagesList method for all known eHealthBoxes of the currently connected user.

#### 5.3.5.1 GetAllEhboxesMessagesList Request

This method can only return 100 messages at a time; consequently you must call upon it multiple times if necessary.

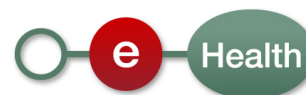


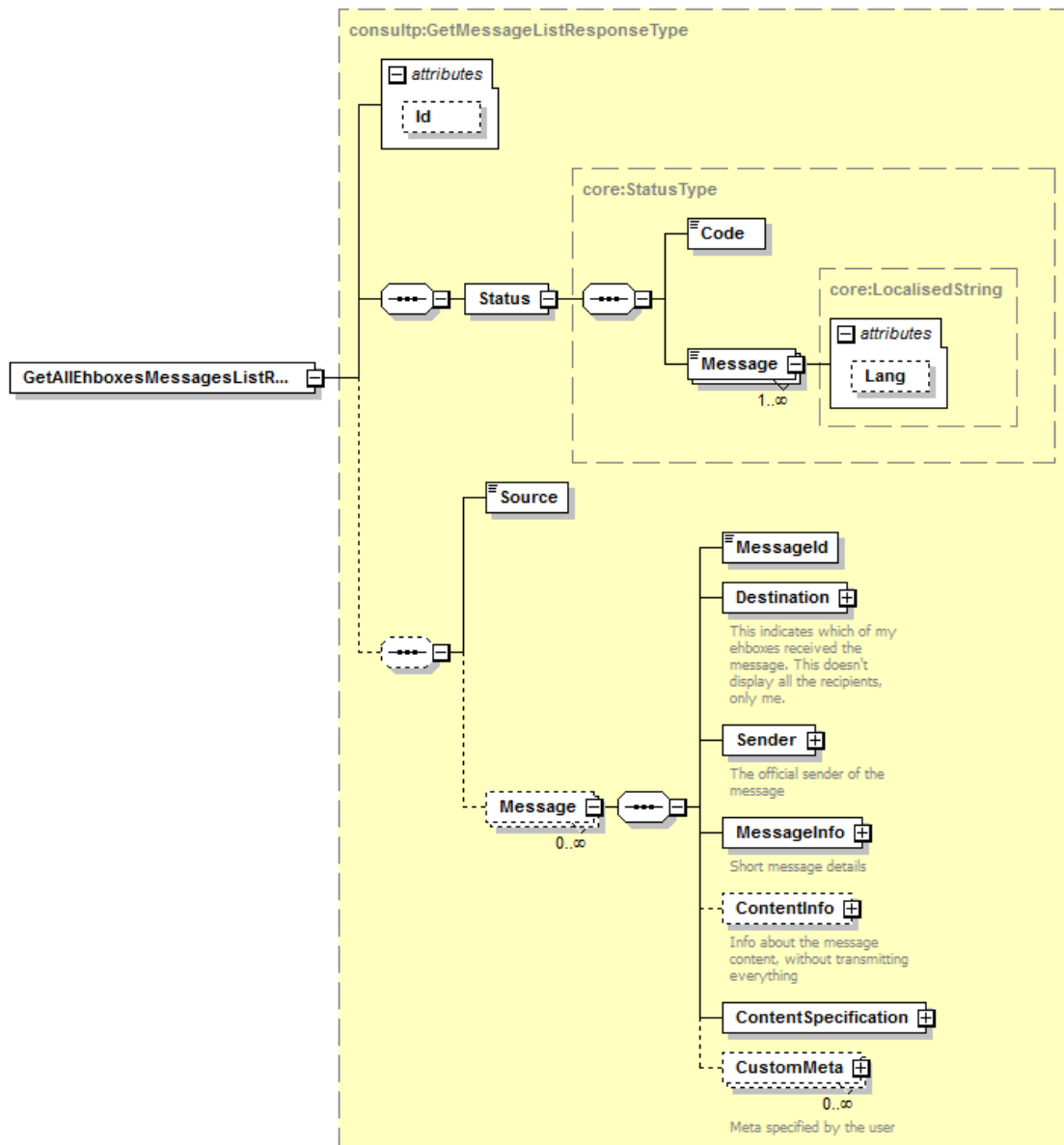


Field name	Description
Source	<p>You can specify the folder specific to your request via the Source parameter.</p> <p>The possible values are:</p> <ul style="list-style-type: none"> <li>• "INBOX" for the inbox folder.</li> <li>• "SENTBOX" for the sentbox folder.</li> <li>• "BININBOX" for messages moved from the inbox folder.</li> <li>• "BINSENTBOX" for messages moved from the sentbox folder.</li> </ul>
StartIndex	Index of the first message (minimum 1).
EndIndex	Index of the last message (minimum 1). A maximum of 100 messages can be returned at a time. $EndIndex < StartIndex + 100$

#### 5.3.5.2 GetAllEhboxesMessagesList Response

The response contains a success status code and as many Message elements as there are messages in all the eHealthBoxes of the currently connected user (personal and enterprise eHealthBoxes). Each element contains all necessary information to treat the message without downloading each individual message. You can identify which eHealthBox received the message via the Destination element.





Field name	Description
Id	The ticket number (Id) is attributed to the exchange request/response by the eHealth platform. This is used to identify the eHealth session.
Status	<p>The Status block contains a code and a message. If no error has occurred during the transaction, the Code will be '100' and the Message 'SUCCESS'. Otherwise:</p> <p>In case of a business error:</p> <ul style="list-style-type: none"> <li>The Code is an error code that unequivocally identifies the problem (see chapter 7 for the possible values).</li> <li>The Message will be a description of the error. Each Message has a Lang (language) characteristic : <ul style="list-style-type: none"> <li>"FR" : French</li> <li>"NL" : Dutch</li> </ul> </li> </ul>



	<ul style="list-style-type: none"> <li>o "EN" : English</li> <li>o "DE" : German</li> <li>o "NA" : Not applicable</li> </ul> <p>In case of technical errors, you will receive a Soap Fault message (see chapter 8).</p>
Source	<p>You can specify the folder which is concerned by your request via the Source parameter.</p> <p>The possible values are:</p> <ul style="list-style-type: none"> <li>• "INBOX" for the inbox folder.</li> <li>• "SENTBOX" for the sentbox folder.</li> <li>• "BININBOX" for messages moved from the inbox folder.</li> <li>• "BINSENTBOX" for messages moved from the sentbox folder.</li> </ul>
Message	<p>0-to-more Message tag(s) describe(s) the eHealthBox Message(s). Each Message is defined with the following:</p> <ul style="list-style-type: none"> <li>• The MessageId that represents a unique message identification generated by the system and returned during the publication and when calling upon the getMessagesList.</li> <li>• The Destination of the Message (see section 5.3.14.16).</li> <li>• The Sender of the Message (see section <b>Error! Reference source not found.</b>).</li> <li>• The MessageInfo that contains additional information about the Message such as publication date, size ... (see section 5.3.14.14).</li> <li>• The ContentInfo of the Message such as title, mime type ... (see section 5.3.14.6).</li> <li>• The ContentSpecification of the Message that contains information such as importance ... (see section 5.3.14.7).</li> <li>• The CustomMeta of the Message that contains free Meta data specified by the user (see section 5.3.14.8)</li> </ul>

### 5.3.5.3 Example

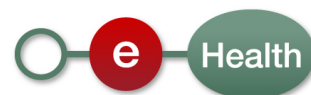
The following example does not contain the SAML assertion (see Annex 2 – List of Qualities).

#### Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:be:fgov:ehhealth:ehbox:consultation:protocol:v3">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:GetAllEhboxesMessagesListRequest>
      <Source>INBOX</Source>
      <StartIndex>1</StartIndex>
      <EndIndex>100</EndIndex>
    </urn:GetAllEhboxesMessagesListRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

#### Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
```



```

        <ns4:GetAllEhboxesMessagesListResponse
xmlns:ns2="urn:be:fgov:ehhealth:errors:service:v1"
xmlns:ns4="urn:be:fgov:ehhealth:ehbox:consultation:protocol:v3">
    <Status>
        <Code>100</Code>
        <Message Lang="EN">SUCCESS</Message>
    </Status>
    <Source>INBOX</Source>
    <Message>
        <MessageId>9Y0002LVHU003</MessageId>
        <Destination>
            <Id>999999999964</Id>
            <Type>INSS</Type>
            <Quality>DOCTOR</Quality>
        </Destination>
        <Sender>
            <Id>999999999964</Id>
            <Type>INSS</Type>
            <Quality>DOCTOR</Quality>
            <Name>Efstathios</Name>
            <FirstName>Frens</FirstName>
        </Sender>
        <MessageInfo>
            <PublicationDate>2011-10-12+02:00</PublicationDate>
            <ExpirationDate>2011-10-17+02:00</ExpirationDate>
            <Size>0</Size>
        </MessageInfo>
        <ContentInfo>
            <Encryptable>INSSPatient/>
            <ContentType>NEWS</ContentType>
            <Title>test title</Title>
            <MimeType>application/octet-stream</MimeType>
            <HasFreeInformations>false</HasFreeInformations>
            <HasAnnex>false</HasAnnex>
        </ContentInfo>
        <ContentSpecification>
            <IsImportant>false</IsImportant>
            <IsEncrypted>false</IsEncrypted>
        </ContentSpecification>
    </Message>
</ns4:GetAllEhboxesMessagesListResponse>
</S:Body>
</S:Envelope>

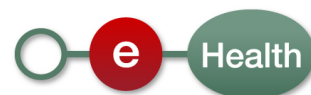
```

### 5.3.6 getFullMessage Method

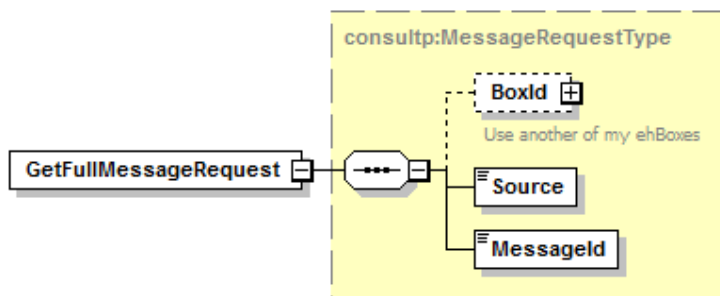
The getFullMessage method is used to get the corresponding complete message and its content to a provided MessageId.

#### 5.3.6.1 getFullMessage Request

You can optionally request a message from another of your mailboxes by specifying it via BoxId,



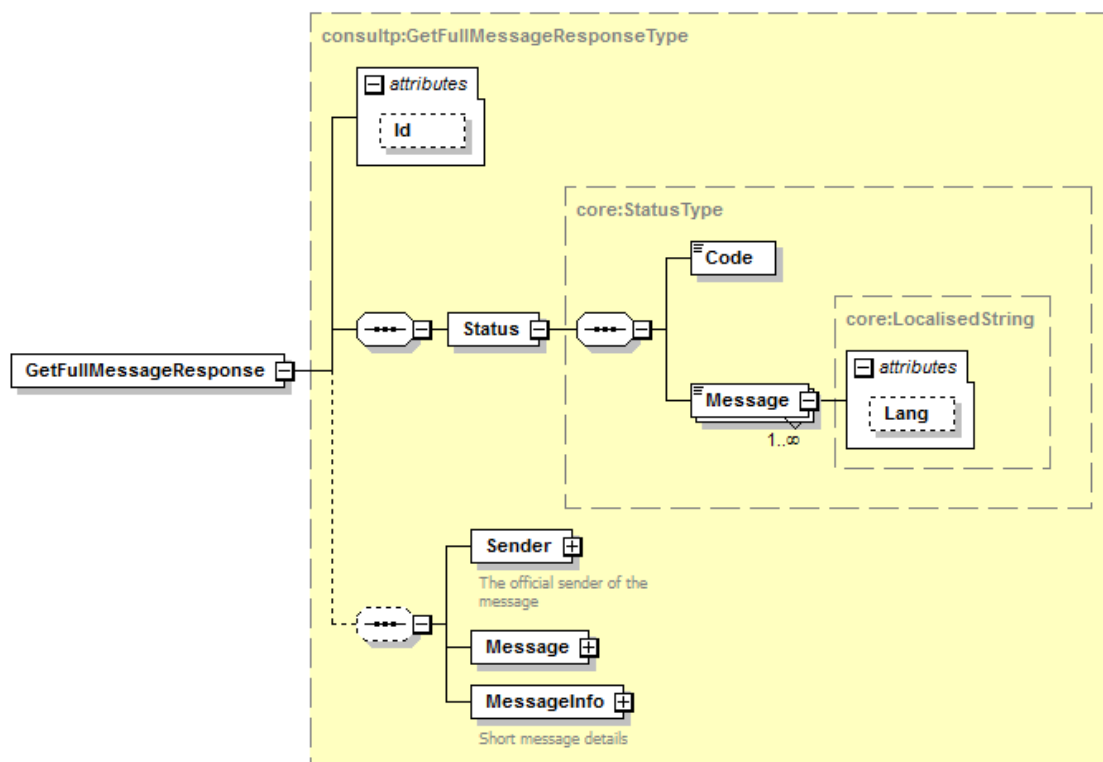




Field name	Description
BoxId	If the client wants to use another of his eHealthBoxes, he can specify it here (see section 5.3.14.3). This avoids the client having to re-authenticate himself each time.
Source	You can specify the folder specific to your request via the Source parameter.  The possible values are: <ul style="list-style-type: none"> <li>• "INBOX" for the inbox folder.</li> <li>• "SENTBOX" for the sentbox folder.</li> </ul>
MessageId	The MessageId is a unique message identification generated by the system and returned during the publication and when calling upon the <code>getMessagesList</code> . String of 13 digits.

### 5.3.6.2 getFullMessage Response

The response contains the same information as already returned by `GetMessagesList` plus the Message content in the element `Message`.



Field name	Description
Id	The ticket number (Id) is attributed to the exchange request/response by the eHealth platform. This is used to identify the eHealth session.
Status	<p>The Status block contains a code and a message. If no error has occurred during the transaction, the Code will be '100' and the Message 'SUCCESS'. Otherwise:</p> <p>In case of a business error:</p> <ul style="list-style-type: none"> <li>• The Code is an error code that unequivocally identifies the problem (see chapter 7 for the possible values).</li> <li>• The Message will be a description of the error. Each Message has a Lang (language) characteristic : <ul style="list-style-type: none"> <li>○ "FR" : French</li> <li>○ "NL" : Dutch</li> <li>○ "EN" : English</li> <li>○ "DE" : German</li> <li>○ "NA" : Not applicable</li> </ul> </li> </ul> <p>In case of technical errors, you will receive a Soap Fault message (see chapter 8).</p>
Sender	The Sender of the Message (see section <b>Error! Reference source not found.</b> ).
Message	The Message itself (see section 5.3.14.13).
MessageInfo	Additional information about the Message (see section 5.3.14.14).

### 5.3.6.3 Example

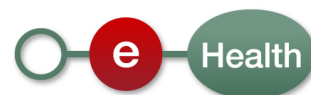
The following example does not contain the SAML assertion (see Annex 2 – List of Qualities).

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:urn="urn:be:fgov:hehealth:ehbox:consultation:protocol:v3">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:GetFullMessageRequest>
      <Source>INBOX</Source>
      <MessageId>9Y0002LKLP004</MessageId>
    </urn:GetFullMessageRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns4:GetFullMessageResponse
xmlns:ns2="urn:be:fgov:hehealth:errors:service:v1"
xmlns:ns4="urn:be:fgov:hehealth:ehbox:consultation:protocol:v3">
      <Status>
        <Code>100</Code>
        <Message Lang="EN">SUCCESS</Message>
      </Status>
      <Sender>
        <Id>71000139</Id>
        <Type>NIHII</Type>
        <Quality>HOSPITAL</Quality>
      </Sender>
    </ns4:GetFullMessageResponse>
  </S:Body>
</S:Envelope>
```



```

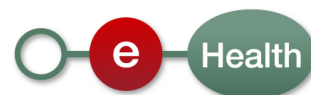
        <Name>Balduino</Name>
        <FirstName>Anil</FirstName>
    </Sender>
    <Message MessageId="9Y0002LKLP004">
        <PublicationId>InitialDoc</PublicationId>
        <DestinationContext>
            <Id>99999999964</Id>
            <Type>INSS</Type>
            <Quality>DOCTOR</Quality>
        </DestinationContext>
        <ContentContext>
            <Content>
                <Document>
                    <Title>Document in eHealthBox</Title>
                    <EncryptableBinaryContent/>
                    <DownloadFileName>test.txt</DownloadFileName>
                    <MimeType>text/plain</MimeType>
                </Document>
                <FreeInformations>
                    <EncryptableFreeText/>
                </FreeInformations>
            </Content>
            <ContentSpecification>
                <IsImportant>>false</IsImportant>
                <IsEncrypted>>false</IsEncrypted>
            </ContentSpecification>
            <CustomMeta>
                <Key>CategoryID</Key>
                <Value>2</Value>
            </CustomMeta>
            <CustomMeta>
                <Key>DocumentType</Key>
                <Value>Scan</Value>
            </CustomMeta>
        </ContentContext>
    </Message>
    <MessageInfo>
        <PublicationDate>2011-06-28+02:00</PublicationDate>
        <ExpirationDate>2011-12-31+01:00</ExpirationDate>
        <Size>12</Size>
    </MessageInfo>
</ns4:GetFullMessageResponse>
</S:Body>
</S:Envelope>

```

### 5.3.7 MoveMessage Method

The MoveMessage method enables the user to move a message from a Source ("INBOX", "SENTBOX", "BININBOX", "BINSENTBOX") to a Destination ("INBOX", "SENTBOX", "BININBOX", "BINSENTBOX"). Only some combinations are allowed as explained below. You need to indicate if a message was received or sent by the concerned eHealthBox. You can do this by looking in which folder the message is situated or thanks to a comparison between the currently connected user and the Sender and/ or Destination element.

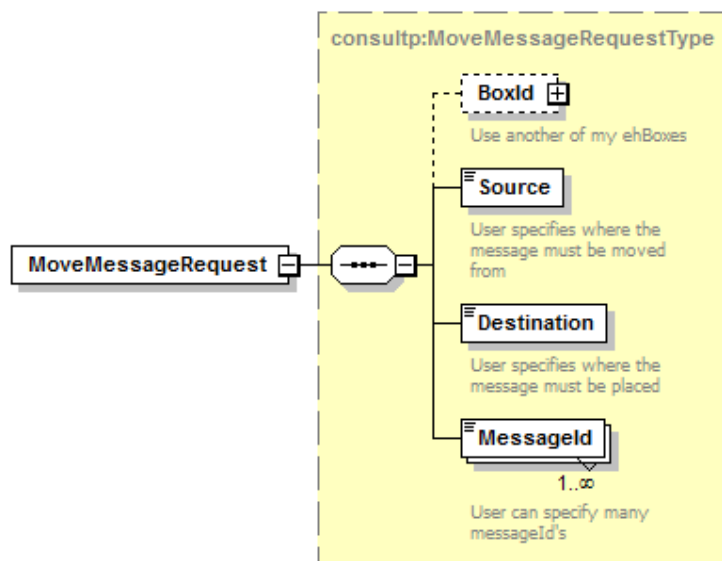
Allowed combinations:



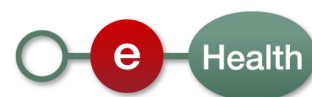
Source	Destination
INBOX	BININBOX
SENTBOX	BINSENTBOX
BININBOX	INBOX
BINSENTBOX	SENTBOX

### 5.3.7.1 MoveMessage Request

In Source specify where the message is currently situated and in Destination where the message must be moved to. In MessageId specify as many elements as there are messages to be moved.



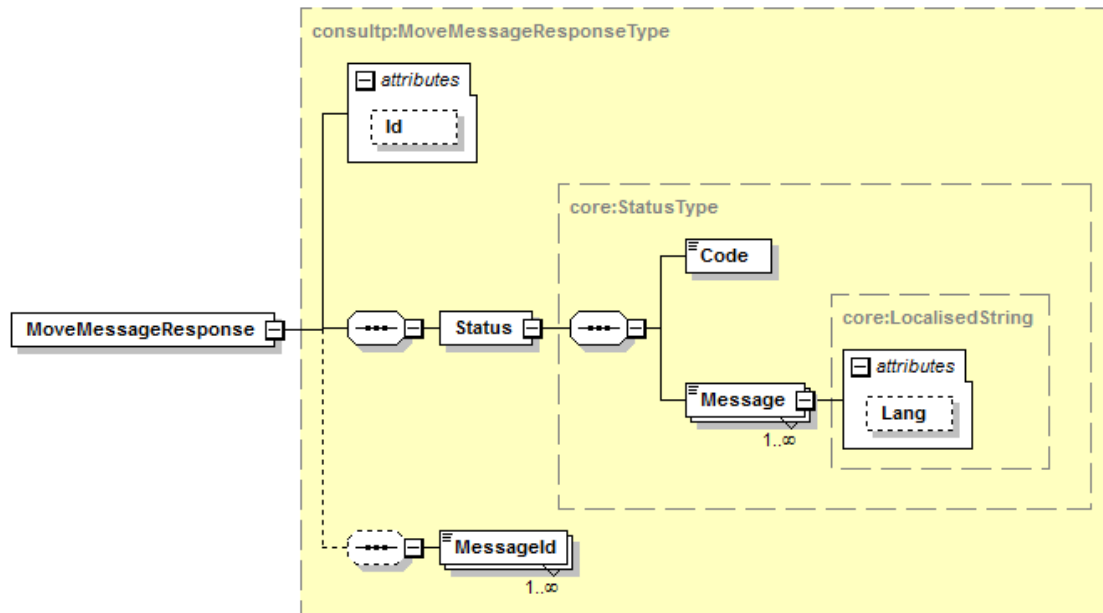
Field name	Description
BoxId	If the client wants to use another of his eHealthBoxes, he can specify it here (see section 5.3.14.3). This avoids the client having to re-authenticate himself each time.
Source	<p>You can specify the folder specific to your request via the Source parameter.</p> <p>The possible values are:</p> <ul style="list-style-type: none"> <li>• "INBOX" for the inbox folder.</li> <li>• "SENTBOX" for the sentbox folder.</li> <li>• "BININBOX" for messages moved from the inbox folder.</li> <li>• "BINSENTBOX" for messages moved from the sentbox folder.</li> </ul>
Destination	<p>You can specify the folder specific to your request via the Destination parameter.</p> <p>The possible values are:</p> <ul style="list-style-type: none"> <li>• "INBOX" for the inbox folder.</li> <li>• "SENTBOX" for the sentbox folder.</li> <li>• "BININBOX" for messages moved from the inbox folder.</li> <li>• "BINSENTBOX" for messages moved from the sentbox folder.</li> </ul>



MessageId	The MessageId's corresponding to the message(s) to move.
-----------	--

### 5.3.7.2 MoveMessage Response

The response contains a success status code or a Business Error as defined in chapter 8. The Business Error enables you to identify which messages were not successfully moved, even though all other have been moved successfully.

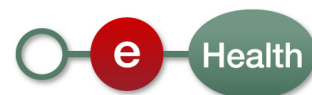


Field name	Description
Id	The ticket number (Id) is attributed to the exchange request/response by the eHealth platform. This is used to identify the eHealth session.
Status	<p>The Status block contains a code and a message. If no error has occurred during the transaction, the Code will be '100' and the Message 'SUCCESS'. Otherwise:</p> <p>In case of a business error:</p> <ul style="list-style-type: none"> <li>• The Code is an error code that unequivocally identifies the problem (see chapter 7 for the possible values).</li> <li>• The Message will be a description of the error. Each Message has a Lang (language) characteristic : <ul style="list-style-type: none"> <li>○ "FR" : French</li> <li>○ "NL" : Dutch</li> <li>○ "EN" : English</li> <li>○ "DE" : German</li> <li>○ "NA" : Not applicable</li> </ul> </li> </ul> <p>In case of technical errors, you will receive a Soap Fault message (see chapter 8).</p>
MessageId	List of MessageId which could not be moved. Though, the other messages were successfully moved.

### 5.3.7.3 Example

The following example does not contain the SAML assertion (see Annex 2 – List of Qualities).

Request:



```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:urn="urn:be:fgov:ehhealth:ehbox:consultation:protocol:v3">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:MoveMessageRequest>
      <Source>INBOX</Source>
      <Destination>BININBOX</Destination>
      <MessageId>9Y0002LKM3006</MessageId>
      <MessageId>9Y0002LKM3007</MessageId>
    </urn:MoveMessageRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

#### Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns4:MoveMessageResponse
xmlns:ns2="urn:be:fgov:ehhealth:errors:service:v1"
xmlns:ns4="urn:be:fgov:ehhealth:ehbox:consultation:protocol:v3">
      <Status>
        <Code>813</Code>
        <Message Lang="EN">Not all messages were moved
```

successfully.

Please verify for each message that the Source and the MessageID are correct.

Also pay attention that a message in the recycle bin which was moved from the Inbox cannot be restored back to the Sentbox and vice versa.</Message>

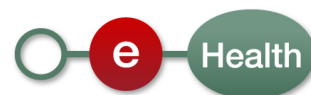
```
</Status>
      <MessageId>9Y0002LKM3006</MessageId>
    </ns4:MoveMessageResponse>
  </S:Body>
</S:Envelope>
```

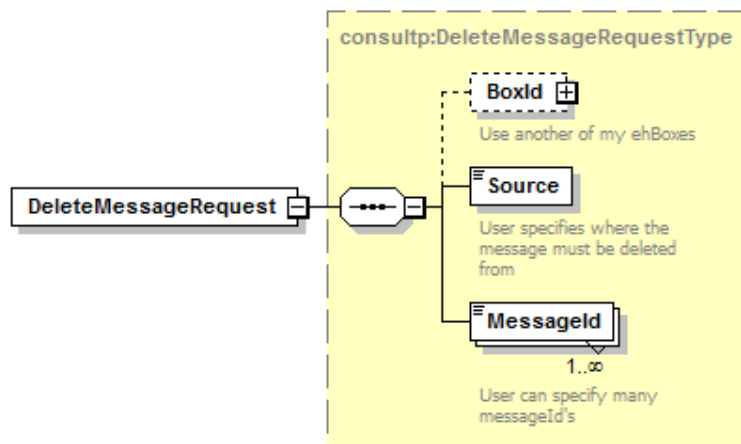
### 5.3.8 DeleteMessage Method

The DeleteMessage method enables the user to physically and definitely delete one or more messages from the inbox, sentbox or bin. This can be used to clean up the eHealthBox when reaching the size limit. Be cautious when using this method. Best is to show a warning message to the user before deleting the messages.

#### 5.3.8.1 DeleteMessage Request

In MessageId you specify where the messages to delete from the bin or directly from the inbox or sentbox.

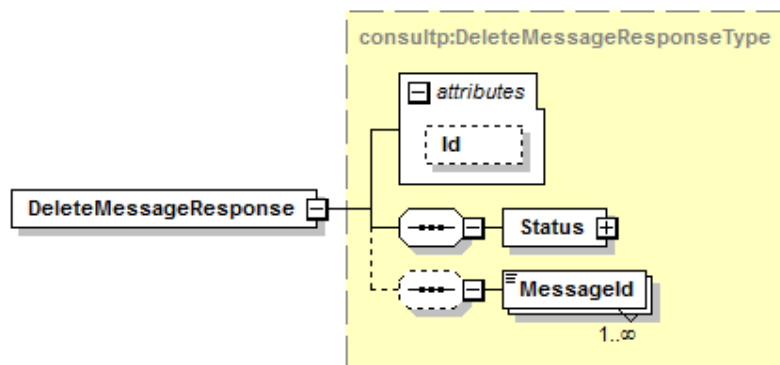




Field name	Description
BoxId	If the client wants to use another of his eHealthBoxes, he can specify it here (see section 5.3.14.3). This avoids the client having to re-authenticate himself each time.
Source	You can specify the folder specific to your request via the Source parameter.  The possible values are: <ul style="list-style-type: none"> <li>• "INBOX" for the inbox folder.</li> <li>• "SENTBOX" for the sentbox folder.</li> <li>• "BININBOX" for messages moved from the inbox folder.</li> </ul> "BINSENTBOX" for messages moved from the sentbox folder.
MessageId	The MessageId's corresponding to the message(s) to delete.

### 5.3.8.2 DeleteMessage Response

The response contains a success status code or a Business Error as defined in chapter 8. The Business Error enables you to identify which messages where not successfully deleted, even though all other have been deleted successfully.



Field name	Description
Id	The ticket number (Id) is attributed to the exchange request/response by the eHealth platform. This is used to identify the eHealth session.
Status	The Status block contains a code and a message. If no error has occurred during the transaction, the Code will be '100' and the Message 'SUCCESS'. Otherwise:

	<p>In case of a business error:</p> <ul style="list-style-type: none"> <li>• The Code is an error code that unequivocally identifies the problem (see chapter 7 for the possible values).</li> <li>• The Message will be a description of the error. Each Message has a Lang (language) characteristic : <ul style="list-style-type: none"> <li>○ "FR" : French</li> <li>○ "NL" : Dutch</li> <li>○ "EN" : English</li> <li>○ "DE" : German</li> <li>○ "NA" : Not applicable</li> </ul> </li> </ul> <p>In case of technical errors, you will receive a Soap Fault message (see chapter 8).</p>
MessageId	List of MessageId which could not be deleted. Though, the other messages were successfully deleted.

### 5.3.8.3 Example

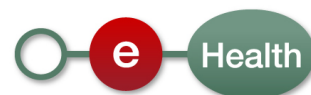
The following example does not contain the SAML assertion (see Annex 2 – List of Qualities).

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:urn="urn:be:fgov:ehhealth:ehbox:consultation:protocol:v3">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:DeleteMessageRequest>
      <MessageId>INBOX</MessageId>
      <MessageId>9Y0002LKM3006</MessageId>
      <MessageId>9Y0002LKM3007</MessageId>
    </urn:DeleteMessageRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns4:DeleteMessageResponse
xmlns:ns2="urn:be:fgov:ehhealth:errors:service:v1"
xmlns:ns4="urn:be:fgov:ehhealth:ehbox:consultation:protocol:v3">
      <Status>
        <Code>815</Code>
        <Message Lang="EN"> One or more messages couldn't be
deleted. All other messages were successfully deleted.
Please verify for each message that the MessageId is correct, and it is in
the Inbox or in the recycle bin.</Message>
      </Status>
      <MessageId>9Y0002LKM3006</MessageId>
    </ns4:DeleteMessageResponse>
  </S:Body>
</S:Envelope>
```





### 5.3.9 getHistory Method

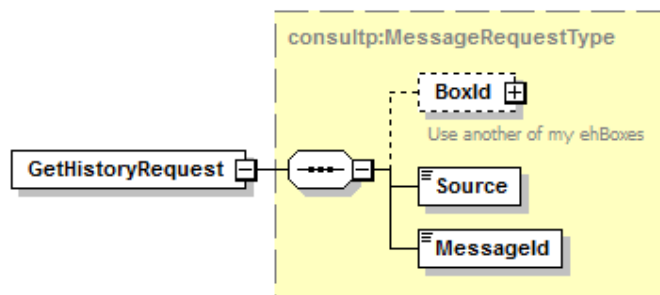
When a new message is sent and updates an old News item (by using the same PublicationId), the old news item is archived and the newer news item replaces the old one. This method enables you to request a list of the old version of that news item by using the MessageId attributed to the newer news item.

The getHistory method is used to get the older message versions of a “news” type message. The method returns a list of MessageId’s corresponding to the previous version of a “news item”, which enables the user to enter a getFullMessage on those MessageId’s.

The getHistory method can’t retrieve a history of a document.

#### 5.3.9.1 getHistory Request

You can request the list of MessageId of a news item from you Inbox or from your SentBox. You can optionally request information of another of your mailboxes by specifying it via BoxId. In MessageId, you can place the MessageId of the newer News item, or the MessageId of an old version of the same news item.

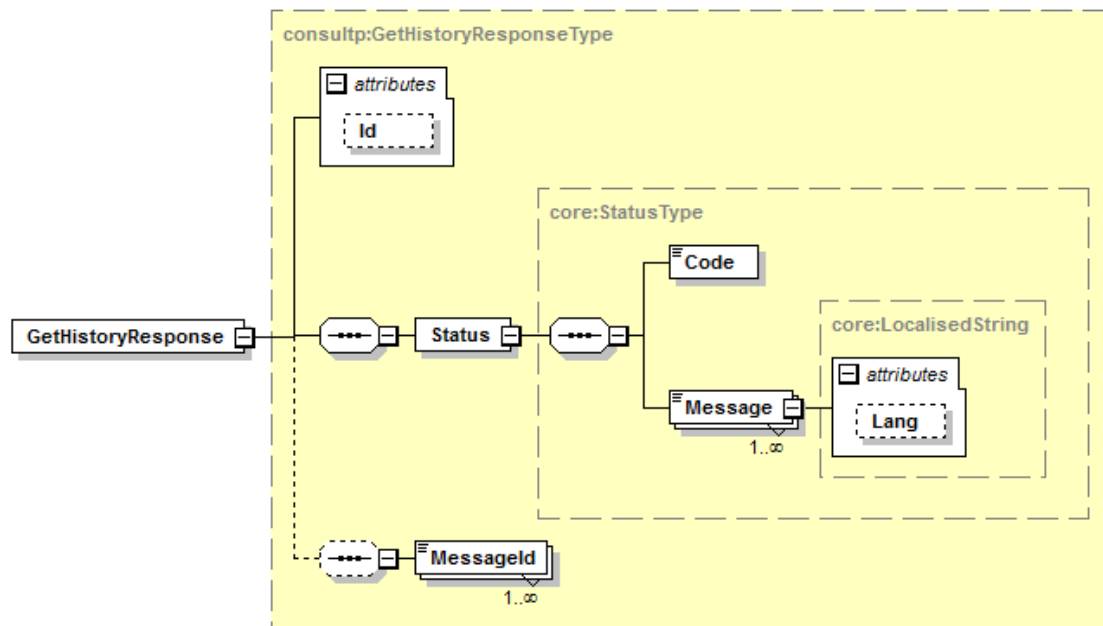


Field name	Description
BoxId	If the client wants to use another of his eHealthBoxes, he can specify it here (see section 5.3.14.3). This avoids the client having to re-authenticate himself each time.
Source	You can specify the folder specific to your request via the Source parameter.  The possible values are: <ul style="list-style-type: none"><li>• "INBOX" for the inbox folder.</li><li>• "SENTBOX" for the sentbox folder.</li></ul>
MessageId	The MessageId of the message to consult. The MessageId is a unique message identification generated by the system and returned during the publication and when calling upon <code>getMessagesList</code> . String of 13 digits.

#### 5.3.9.2 getHistory Response

The response gives you a group of MessageId’s which concern the same news item. You can then enter a `GetFullMessage` in order to retrieve the old news item if necessary.





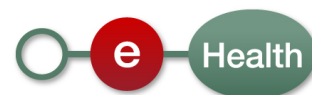
Field name	Description
Id	The ticket number (Id) is attributed to the exchange request/response by the eHealth platform. This is used to identify the eHealth session.
Status	<p>The Status block contains a code and a message. If no error has occurred during the transaction, the Code will be '100' and the Message 'SUCCESS'. Otherwise:</p> <p>In case of a business error:</p> <ul style="list-style-type: none"> <li>The Code is an error code that unequivocally identifies the problem (see chapter 7 for the possible values).</li> <li>The Message will be a description of the error. Each Message has a Lang (language) characteristic : <ul style="list-style-type: none"> <li>"FR" : French</li> <li>"NL" : Dutch</li> <li>"EN" : English</li> <li>"DE" : German</li> <li>"NA" : Not applicable</li> </ul> </li> </ul> <p>In case of technical errors, you will receive a Soap Fault message (see chapter 8).</p>
MessageId	List of MessageId's from the older message versions.

### 5.3.9.3 Example

The following example does not contain the SAML assertion see Annex 2 – List of Qualities).

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:be:fgov:ehealth:ehbox:consultation:protocol:v3">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:GetHistoryRequest>
      <Source>INBOX</Source>
      <MessageId>9Y0002LKLP004</MessageId>
    </urn:GetHistoryRequest>
  </soapenv:Body>
</soapenv:Envelope>
```



```

        </urn:GetHistoryRequest>
    </soapenv:Body>
</soapenv:Envelope>

```

Response:

```

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns4:GetHistoryResponse
xmlns:ns2="urn:be:fgov:ehhealth:errors:service:v1"
xmlns:ns4="urn:be:fgov:ehhealth:ehbox:consultation:protocol:v3">
      <Status>
        <Code>100</Code>
        <Message Lang="EN">SUCCESS</Message>
      </Status>
      <MessageId>9Y0002LKLN001</MessageId>
    </ns4:GetHistoryResponse>
  </S:Body>
</S:Envelope>

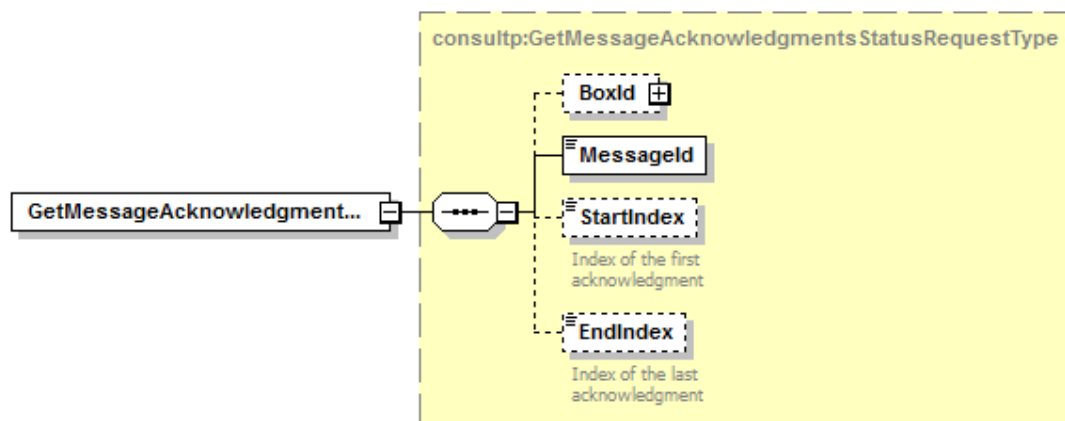
```

### 5.3.10 GetMessageAcknowledgmentsStatus Method

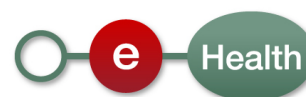
The GetMessageAcknowledgmentsStatus method is used to find out for a message that the user has sent which recipients have received, viewed or read the message and at what time.

#### 5.3.10.1 GetMessageAcknowledgmentsStatus Request

This method can only return 100 acknowledgements at a time; consequently, you must call upon it multiple times if necessary.



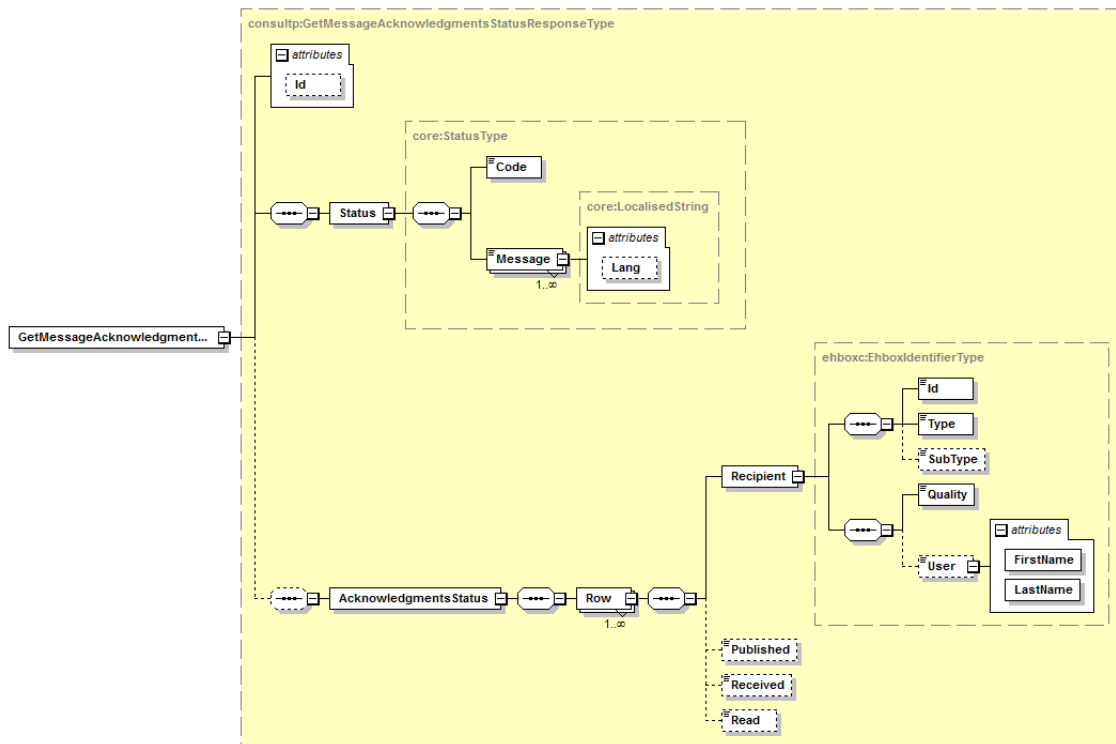
Field name	Description
BoxId	If the client wants to use another of his eHealthBoxes, he can specify it here (see section 5.3.14.3). This avoids the client having to re-authenticate himself each time.
MessageId	The MessageId's of the message to consult. The MessageId is a unique message identification generated by the system and returned during the publication and when calling upon the getMessagesList. String of 13 digits.
StartIndex	Index of the first acknowledgment (minimum 1).



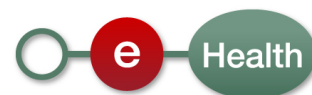
EndIndex	Index of the last acknowledgment (minimum 1). A maximum of 100 acknowledgments can be returned at once. EndIndex < StartIndex + 100
----------	---

### 5.3.10.2 GetMessageAcknowledgmentsStatus Response

The response gives you information about your sent message: who received and read your message and at what time.



Field name	Description
Id	The ticket number (Id) is attributed to the exchange request/response by the eHealth platform. This is used to identify the eHealth session.
Status	<p>The Status block contains a code and a message. If no error has occurred during the transaction, the Code will be '100' and the Message 'SUCCESS'. Otherwise:</p> <p>In case of a business error:</p> <ul style="list-style-type: none"> <li>• The Code is an error code that unequivocally identifies the problem (see chapter 7 for the possible values).</li> <li>• The Message will be a description of the error. Each Message has a Lang (language) characteristic : <ul style="list-style-type: none"> <li>○ "FR" : French</li> <li>○ "NL" : Dutch</li> <li>○ "EN" : English</li> <li>○ "DE" : German</li> <li>○ "NA" : Not applicable</li> </ul> </li> </ul> <p>In case of technical errors, you will receive a Soap Fault message (see chapter 8).</p>
AcknowledgmentsStatus	Contains a Row for each different Recipient of the message. Each Row contains the identification of the Recipient (Type EhboxIdentifier, see section 5.3.14.16), the time the



	message was published, the time the message was received (=viewed) by that Recipient, and the time the message was read by that Recipient.
--	--

### 5.3.10.3 Example

The following example does not contain the SAML assertion (see Annex 2 – List of Qualities).

#### Request:

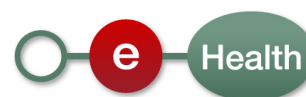
```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:urn="urn:be:fgov:ehhealth:ehbox:consultation:protocol:v3">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:GetMessageAcknowledgmentsStatusRequest>
      <MessageId>9Y0002LKH020J</MessageId>
      <StartIndex>1</StartIndex>
      <EndIndex>100</EndIndex>
    </urn:GetMessageAcknowledgmentsStatusRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

#### Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns4:GetMessageAcknowledgmentsStatusResponse
xmlns:ns2="urn:be:fgov:ehhealth:errors:service:v1"
xmlns:ns4="urn:be:fgov:ehhealth:ehbox:consultation:protocol:v3">
      <Status>
        <Code>100</Code>
        <Message Lang="EN">SUCCESS</Message>
      </Status>
      <AcknowledgmentsStatus>
        <Row>
          <Recipient>
            <Id>99999999964</Id>
            <Type>INSS</Type>
            <Quality>DOCTOR</Quality>
            <User LastName="Simon"
FirstName="Llew">99999999965</User>
          </Recipient>
          <Published>2011-12-17T09:30:47Z</Published>
          <Received>2011-12-17T10:31:17Z </Received>
          <Read>2011-12-17T09:30:47Z </Read>
        </Row>
      </AcknowledgmentsStatus>
    </ns4:GetMessageAcknowledgmentsStatusResponse>
  </S:Body>
</S:Envelope>
```

### 5.3.11 InsertOoO Method

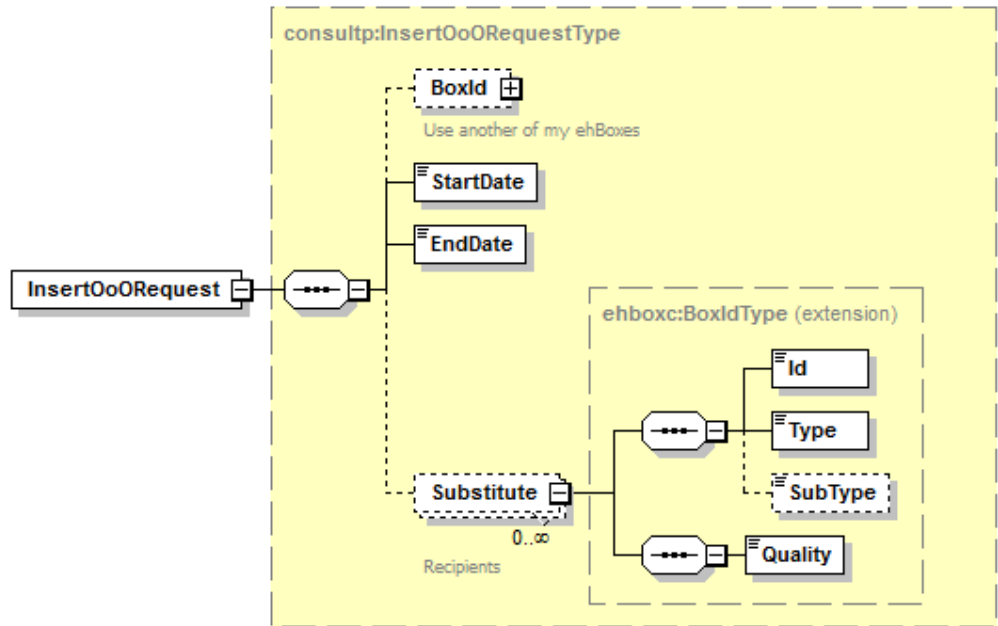
The InsertOoO method is used to insert an Out-of-Office for the considered eHealthBox. In order to insert an Out-of-Office for another eHealthBox, BoxId can be used. There are a number of rules and limitations to



respect as described in section 5.3.1.2. If these rules are not respected a Business error is returned as detailed in section 8.2.

### 5.3.11.1 InsertOoO Request

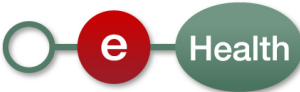
Only one OoO can be inserted by request, but multiple substitutes may be specified.

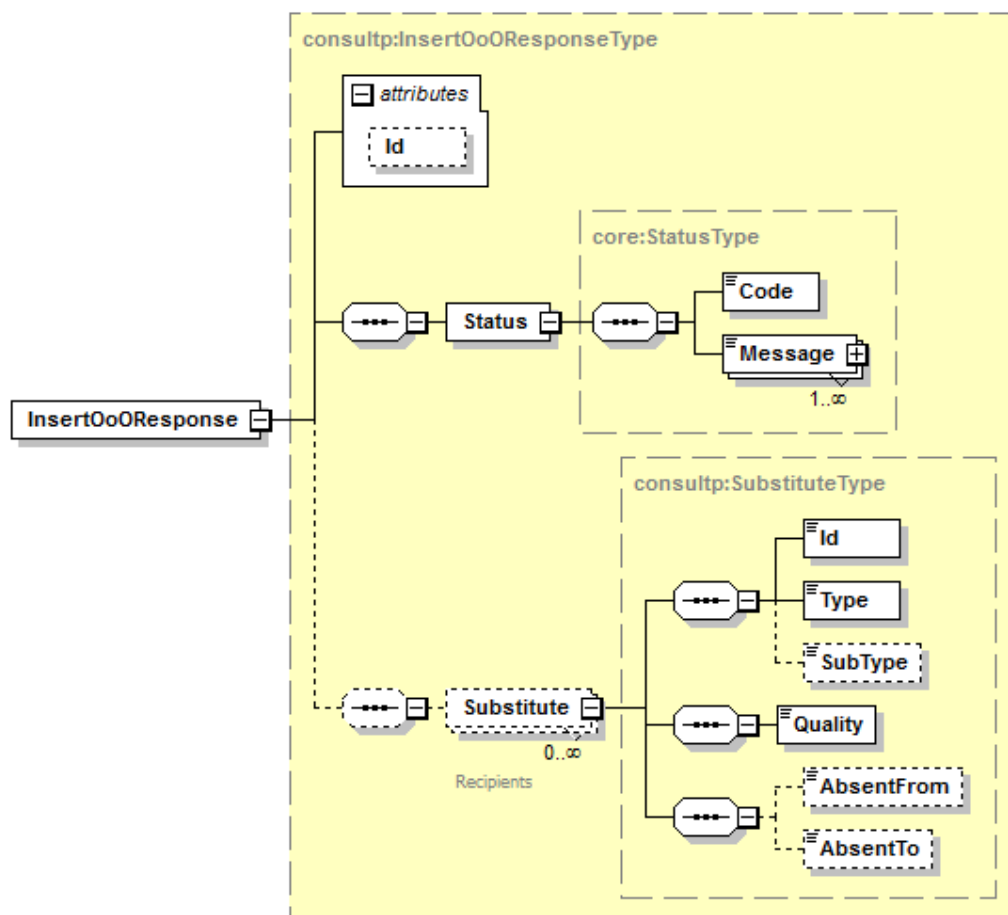


Field name	Description
BoxId	If the client wants to use another of his eHealthBoxes, he can specify it here (see section 5.3.14.3). This avoids the client having to re-authenticate himself each time.
StartDate	The StartDate of the Out-of-Office. This date is inclusive. Ex. 2013-07-03+02:00
EndDate	The EndDate of the Out-of-Office. This date is inclusive. Ex. 2013-07-17+02:00
Substitutue	Optional. You may want to specify someone which is replacing you during your absence. The structure is as usual, detailed in 5.3.14.3

### 5.3.11.2 InsertOoO Response

The response is success without further details if everything went right. Otherwise the wrong or absent substitutes will be returned in the response as detailed below.





Field name	Description
Id	The Id element contains the OoOld attributed to this OoO by the system, which can later be used in DeleteOoO for example.
Status	<p>The Status block contains a code and a message. If no error has occurred during the transaction, the Code will be '100' and the Message 'SUCCESS'. Otherwise:</p> <p>In case of a business error:</p> <ul style="list-style-type: none"> <li>The Code is an error code that unequivocally identifies the problem (see chapter 7 for the possible values).</li> <li>The Message will be a description of the error. Each Message has a Lang (language) characteristic :               <ul style="list-style-type: none"> <li>"FR" : French</li> <li>"NL" : Dutch</li> <li>"EN" : English</li> <li>"DE" : German</li> <li>"NA" : Not applicable</li> </ul> </li> </ul> <p>In case of technical errors, you will receive a Soap Fault message (see chapter 8).</p>
Substitute	Contains a Row for each different Substitute which is incorrect, unknown, or absent (absence period of this substitute is then specified).

### 5.3.11.3 Example

The following example does not contain the SAML assertion (see Annex 2 – List of Qualities).

Request:

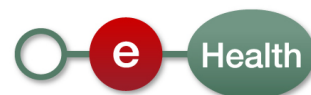
```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:be:fgov:ehhealth:ehbox:consultation:protocol:v3">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:InsertOoORequest>
      <BoxId>
        <Id>982374923273</Id>
        <Type>NIHII</Type>
        <Quality>DOCTOR</Quality>
      </BoxId>
      <StartDate>2013-07-03+02:00</StartDate>
      <EndDate>2013-07-17+02:00</EndDate>
      <Substitute>
        <Id>82351425106</Id>
        <Type>INSS</Type>
        <Quality>DOCTOR</Quality>
      </Substitute>
    </urn:InsertOoORequest>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns4:InsertOoOResponse
      xmlns:ns2="urn:be:fgov:ehhealth:errors:service:v1"
      xmlns:ns4="urn:be:fgov:ehhealth:ehbox:consultation:protocol:v3">
      <Status>
        <Code>824</Code>
        <Message Lang="EN">One or more substitutes cannot be chosen
because they are absent.</Message>
      </Status>
      <Substitute>
        <Id>82351425106</Id>
        <Type>INSS</Type>
        <Quality>DOCTOR</Quality>
      </Substitute>
    </ns4:InsertOoOResponse>
  </S:Body>
</S:Envelope>
```

### 5.3.12 DeleteOoO Method

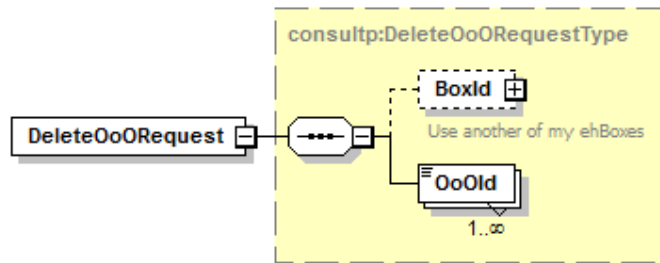
The DeleteOoO method is used to delete an Out-of-Office for the considered eHealthBox. In order to delete an Out-of-Office for another eHealthBox, BoxId can be used. The Out-of-Office Id must be specified. This can be retrieved from InsertOoO, in the Id element or from GetOoOList.





### 5.3.12.1 DeleteOoO Request

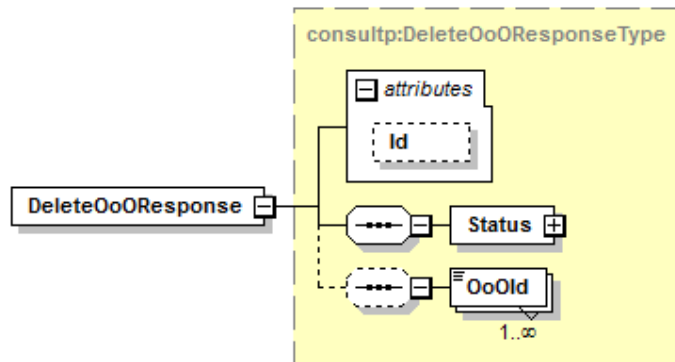
Only one OoO can be inserted by request, but multiple substitutes may be specified.



Field name	Description
BoxId	If the client wants to use another of his eHealthBoxes, he can specify it here (see section 5.3.14.3). This avoids the client having to re-authenticate himself each time.
OoOld	The OoOld from the Out-of-Offices that the user wants to delete. The OoOld's can be retrieved from InsertOoO, in the Id element or from GetOoOList.

### 5.3.12.2 DeleteOoO Response

The response is success without further details if everything went right. Otherwise the OoOld which could not be found will be returned in the response as detailed below. All other OoO which were found were deleted.



Field name	Description
Id	The ticket number (Id) is attributed to the exchange request/response by the eHealth platform. This is used to identify the eHealth session.
Status	<p>The Status block contains a code and a message. If no error has occurred during the transaction, the Code will be '100' and the Message 'SUCCESS'. Otherwise:</p> <p>In case of a business error:</p> <ul style="list-style-type: none"><li>• The Code is an error code that unequivocally identifies the problem (see chapter 7 for the possible values).</li><li>• The Message will be a description of the error. Each Message has a Lang (language) characteristic :<ul style="list-style-type: none"><li>○ "FR" : French</li><li>○ "NL" : Dutch</li><li>○ "EN" : English</li><li>○ "DE" : German</li><li>○ "NA" : Not applicable</li></ul></li></ul>

	In case of technical errors, you will receive a Soap Fault message (see chapter 8).
OoOld	Contains the OoOld's which couldn't be found and were not deleted. All others were deleted.

### 5.3.12.3 Example

The following example does not contain the SAML assertion (see Annex 2 – List of Qualities).

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:urn="urn:be:fgov:ehealth:ehbox:consultation:protocol:v3">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:DeleteOoORequest>
      <BoxId>
        <Id>99999999964</Id>
        <Type>INSS</Type>
        <Quality>DOCTOR</Quality>
      </BoxId>
      <OoOId>1005</OoOId>
    </urn:DeleteOoORequest>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

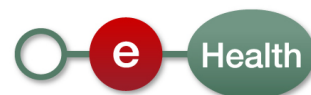
```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns4:DeleteOoOResponse
xmlns:ns2="urn:be:fgov:ehealth:errors:service:v1"
xmlns:ns4="urn:be:fgov:ehealth:ehbox:consultation:protocol:v3">
      <Status>
        <Code>840</Code>
        <Message Lang="EN">One or more OoOId are invalid.</Message>
      </Status>
      <OoOId>1005</OoOId>
    </ns4:DeleteOoOResponse>
  </S:Body>
</S:Envelope>
```

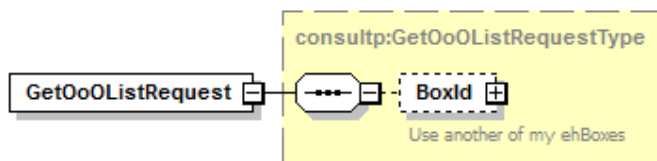
### 5.3.13 GetOoOList Method

The GetOoOList method is used to list all inserted Out-of-Office for the considered eHealthBox. The list contains all OoO, past, active and future OoO. The OoOld returned can be further used to delete an OoO.

#### 5.3.13.1 GetOoOList Request

Only one OoO can be inserted by request, but multiple substitutes may be specified.

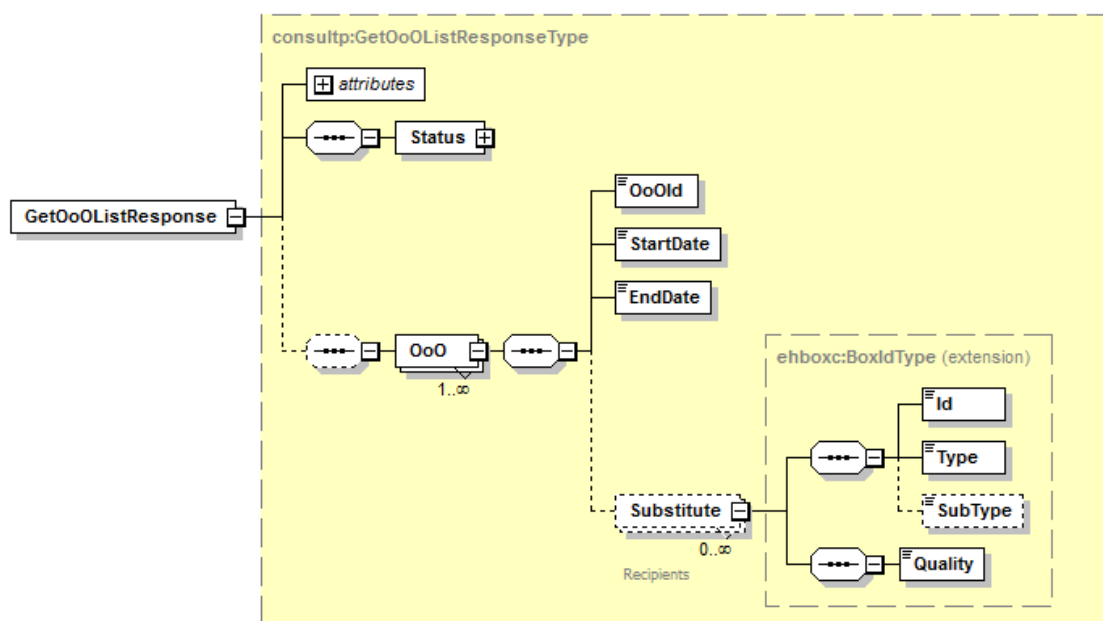




Field name	Description
BoxId	If the client wants to use another of his eHealthBoxes, he can specify it here (see section 5.3.14.3). This avoids the client having to re-authenticate himself each time.

### 5.3.13.2 GetOoList Response

The response contains a success status code and as many OoO elements as there are Out-of-Office in the considered eHealthBox. Each element contains all OoO data when it was inserted.



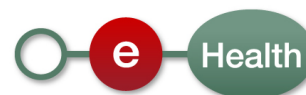
Field name	Description
OoOld	The OoOld assigned by the system to the Out-of-Office.
StartDate	The StartDate of the Out-of-Office. This date is inclusive. Ex. 2013-07-03+02:00
EndDate	The EndDate of the Out-of-Office. This date is inclusive. Ex. 2013-07-17+02:00
Substitutue	Optional. The person which is replacing the absent person. The structure is as usual, detailed in 5.3.14.3

### 5.3.13.3 Example

The following example does not contain the SAML assertion (see Annex 2 – List of Qualities).

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:be:fgov:ehhealth:ehbox:consultation:protocol:v3">
  <soapenv:Header/>
```



```

    <soapenv:Body>
      <urn:GetOoOListRequest/>
    </soapenv:Body>
  </soapenv:Envelope>

```

Response:

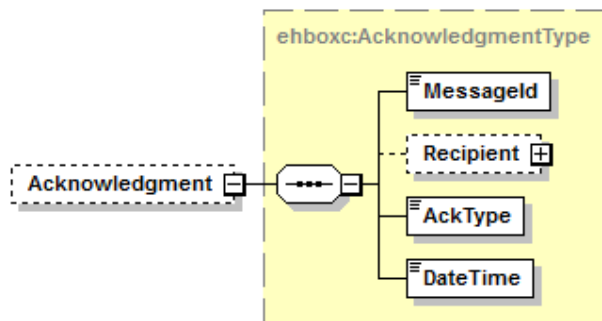
```

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns4:GetOoOListResponse
xmlns:ns2="urn:be:fgov:ehhealth:errors:service:v1"
xmlns:ns4="urn:be:fgov:ehhealth:ehbox:consultation:protocol:v3">
      <Status>
        <Code>100</Code>
        <Message Lang="EN">SUCCESS</Message>
      </Status>
      <OoO>
        <OoOId>1000</OoOId>
        <StartDate>2013-07-03+02:00</StartDate>
        <EndDate>2013-07-17+02:00</EndDate>
        <Substitute>
          <Id>82351425106</Id>
          <Type>INSS</Type>
          <Quality>DOCTOR</Quality>
        </Substitute>
      </OoO>
    </ns4:GetOoOListResponse>
  </S:Body>
</S:Envelope>

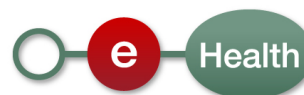
```

## 5.3.14 Used Types

### 5.3.14.1 Acknowledgment

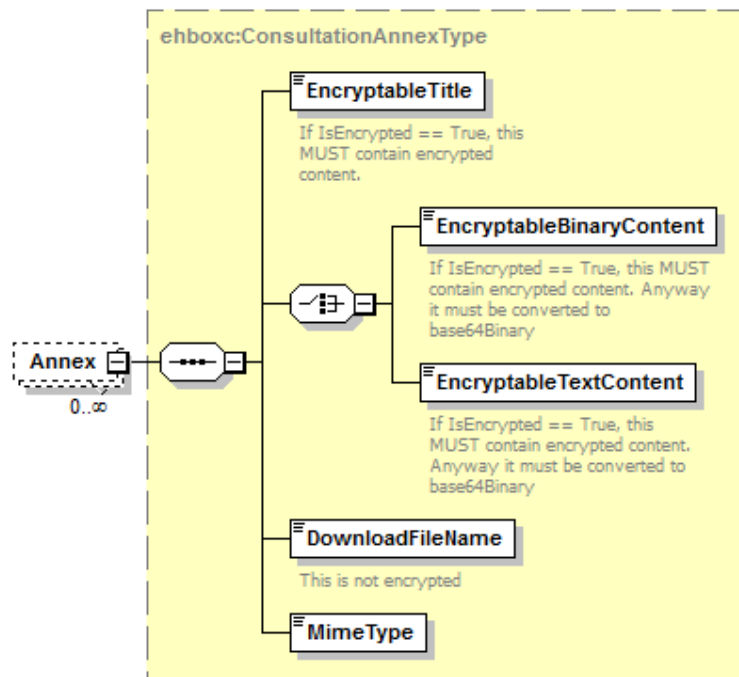


Field name	Descriptions
MessageId	The MessageId of the original message.
Recipient	The recipient receiving or reading the original message. This element is only present for an acknowledgment of type 'RECEIVED' or 'READ'.

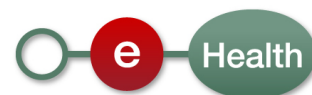


AckType	<p>3 acknowledgment types exist: 'PUBLISHED', 'RECEIVED' and 'READ':</p> <ol style="list-style-type: none"> <li>1. An ack. 'published' =&gt; when the system is treating the original message</li> <li>2. An ack. 'Received' =&gt; when the recipient has seen the original message (request getMessageList)</li> <li>3. An ack. 'Read' =&gt; when the recipient has opened the original message (request getFullMessage)</li> </ol>
DateTime	Date & Time when the acknowledgment was issued. Eg. <DateTime>2001-12-17T09:30:47Z</DateTime>

### 5.3.14.2 Annex



Field name	Descriptions
EncryptableTitle	<p>An Annex has an EncryptableTitle, a human readable description of its intent (minimum 1, maximum 400 characters in binary form).</p> <p>If IsEncrypted is true (see Section 5.3.14.7), the title must be encrypted (see section 5.1.2).</p>
EncryptableBinaryContent	<p>Contains a SOAP reference to the attachment Eg. &lt;EncryptableBinaryContent&gt; cid:4906633928&lt;/EncryptableBinaryContent&gt;. Type is xsd:anyURI.</p> <p>If IsEncrypted is true (see Section 5.3.14.7), the attachment must be encrypted (see section 5.1.2).</p>
EncryptableTextContent	<p>Text content, in binary form.</p> <p>If IsEncrypted is true (see Section 5.3.14.7), the title must be encrypted (see section 5.1.2).</p>



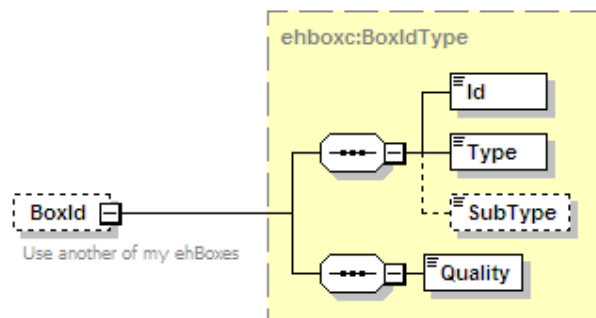
DownloadFileName	E.g. "principal.pdf" (string minimum 1, maximum 255).
MimeType	Represents the mime type of the content. E.g. "application/pdf"," text/plain", "application/octet-stream" (string minimum 1, maximum 255).

### 5.3.14.3 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.

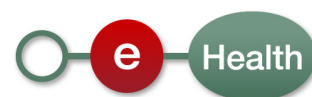
You will find all mandatory information about the allowed combinations Id-Type- Quality in Annex 2 – List of Qualities.

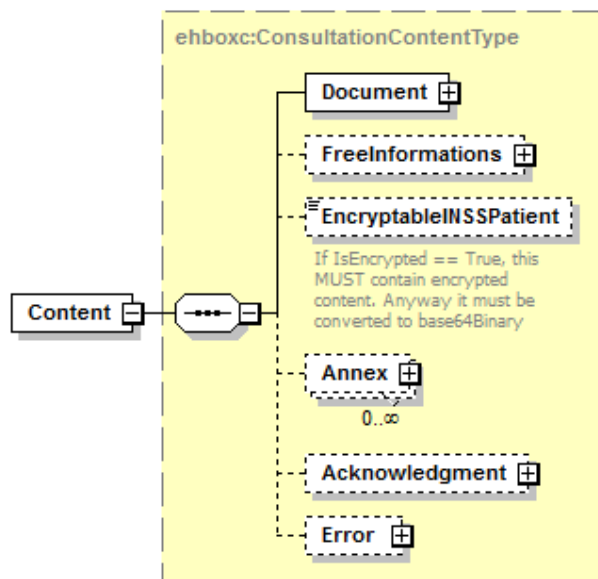


Field name	Descriptions
ID	Your eHealthBox's identification number. This is a digital number representing an INSS, NIHII, FAMPH, or CBE. String.
Type	Your eHealthBox's ID type ("INSS", "NIHII", "FAMPH" or "CBE"). String.
Subtype (obsolete)	If the recipient is an organization, the Subtype allows (if necessary) further specification (such as "HOSPITAL" SubType for a Hospital Quality, or "GROUP" SubType for a Group Quality). String.
Quality	Your eHealthBox's Quality. String (see Annex 2 – List of Qualities).

### 5.3.14.4 Content

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

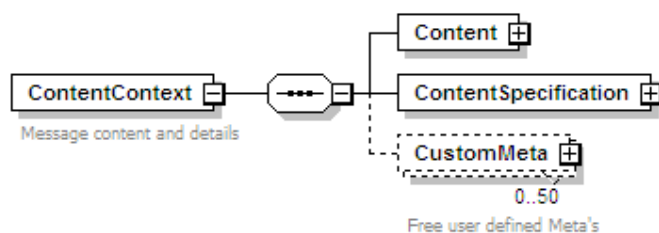




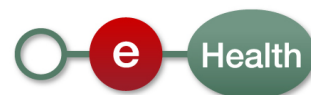
Field name	Descriptions
Document	A news item is now a special case of a Document. See section 5.3.14.10
FreeInformations	See section 5.3.14.12
EncryptableINSSPatient	This optional field allows to specify an INSS number of a patient concerned by the message content. If IsEncrypted is true (see Section 5.3.14.7), the INSS number must be encrypted (see section 5.1.2).
Annex	See section 5.3.14.2
Acknowledgment	See section 5.3.14.1
Error	See section 5.3.14.11

### 5.3.14.5 ContentContext

A ContentContext contains the message content and message details, as well as zero-or-more (50 maximum) free CustomMetas. These CustomMetas can be freely specified by the user for internal usage. You can define a Key and a value for each CustomMeta (see 5.3.14.8).

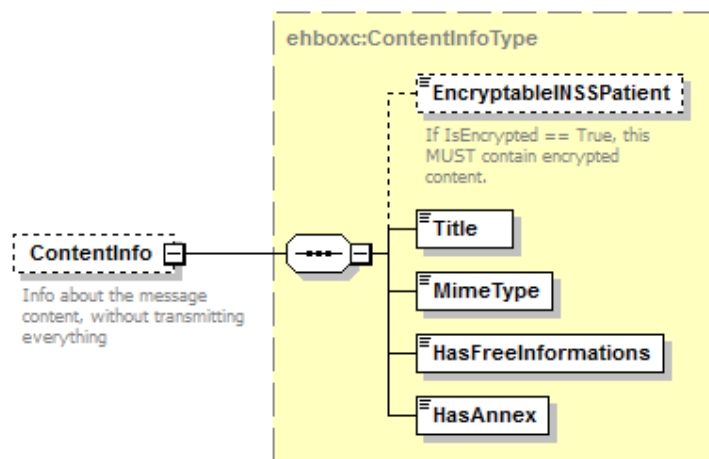


Field name	Descriptions
------------	--------------



Content	See section 5.3.14.4.
ContentSpecification	See section 5.3.14.7.
CustomMeta	See section 5.3.14.8

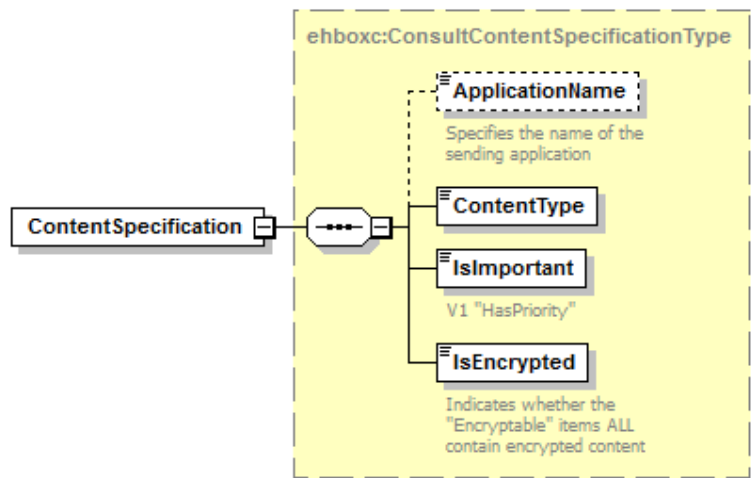
### 5.3.14.6 ContentInfo



Field name	Descriptions
EncryptableINSSPatient	This optional field allows to specify an INSS number of a patient concerned by the message content. If IsEncrypted is true (see Section 5.3.14.7), the content must be encrypted before being converted to xs:base64Binary (see section 5.1.2).
Title	The Title of the message, a human readable description of its purpose (string minimum 1, maximum 400).
MimeType	Represents the mime type of the content. E.g. "application/pdf"," text/plain", "application/octet-stream" (string minimum 1, maximum 255).
HasFreeInformation	A flag (true or false) that indicates if the message has any Free Information.
HasAnnex	A flag (true or false) that indicates if the message has an Annex.



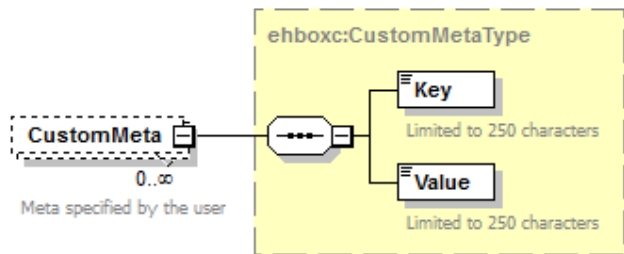
5.3.14.7 ContentSpecification



Field name	Descriptions
ApplicationName	The Application sending the message (optional, string minimum 1, maximum 25).
ContentType	The Content Type of the message ("DOCUMENT", "NEWS", "ACKNOWLEDGMENT", "ERROR").
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.

5.3.14.8 CustomMeta

CustomMeta was introduced 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 limited each to 250 characters. Those CustomMetas will be transported from the sender to the recipient. You can for example add a CustomMeta for internal usage as "CategoryId, 17", or "MessageContent, Blood analysis".

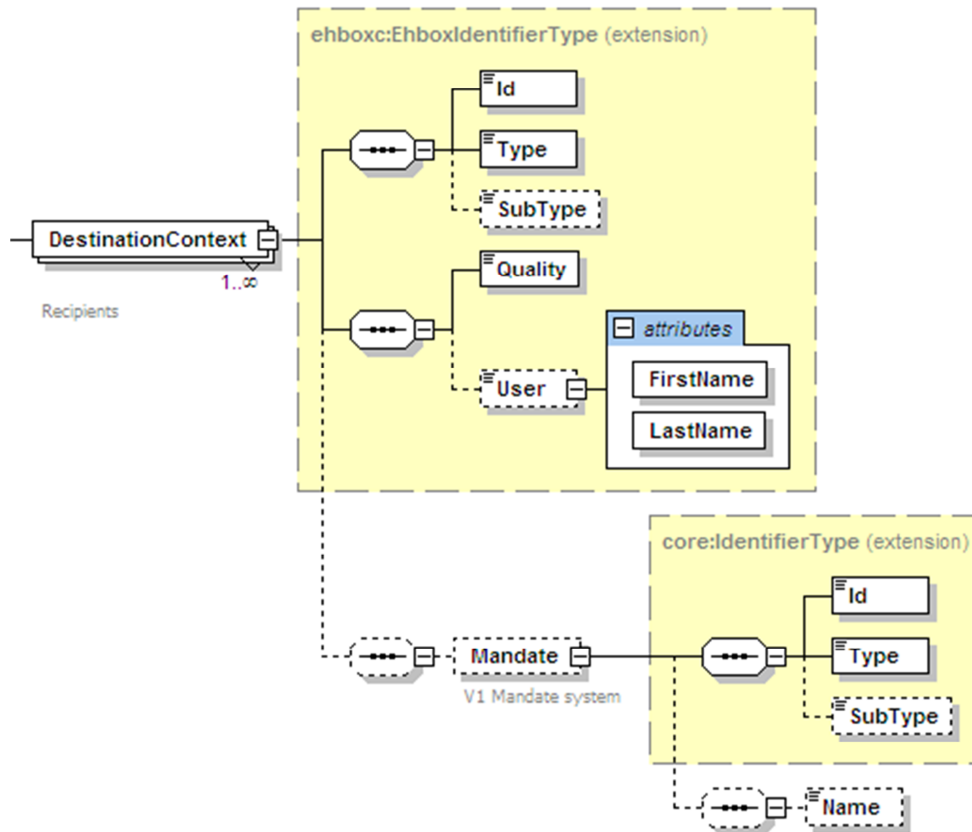


Field name	Descriptions
Key	Alphanumeric string used as a key (string minimum 1, maximum 250).
Value	Alphanumeric string value corresponding to the Key (string minimum 1, maximum 250).

### 5.3.14.9 DestinationContext

A DestinationContext contains all the information on the recipient.

You will find all mandatory information about the allowed combinations Id-Type- Quality in Annex 2 – List of Qualities.

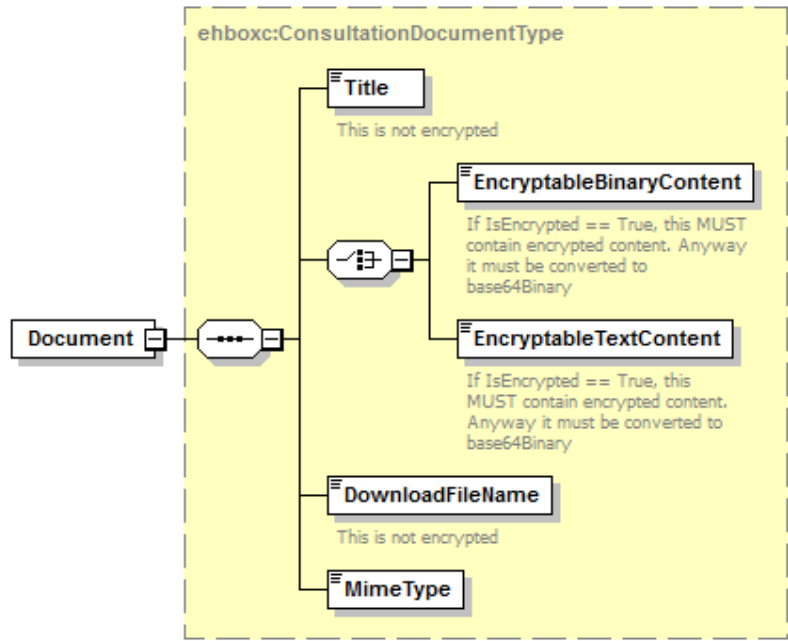


Field name	Descriptions
ID	The recipient's identification number. This is a digital number representing an INSS, NIHII, FAMPH, or CBE. String.
Type	The recipient's ID type ("INSS", "NIHII", "FAMPH" or "CBE"). String.
Subtype (obsolete)	If the recipient is an organization, the Subtype allows (if necessary) further specification (such as "HOSPITAL" SubType for a Hospital Quality, or "GROUP" SubType for a Group Quality). String.
Quality	A Quality defines the recipient's eHealthBox. String (see Annex 2 – List of Qualities)
User	An optional User (FirstName and LastName) 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). String.
Mandate (obsolete)	Optional authority information will be added if the recipient has been granted an authority. The constituent's identification number (Id) and Type are requested. If the constituent is an organization, the Subtype allows (if necessary) further specification

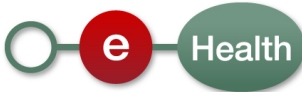
	(such as "HOSPITAL" SubType for a Hospital Quality, or "GROUP" SubType for a Group Quality). The recipient's name may be specified.
--	---

### 5.3.14.10 Document

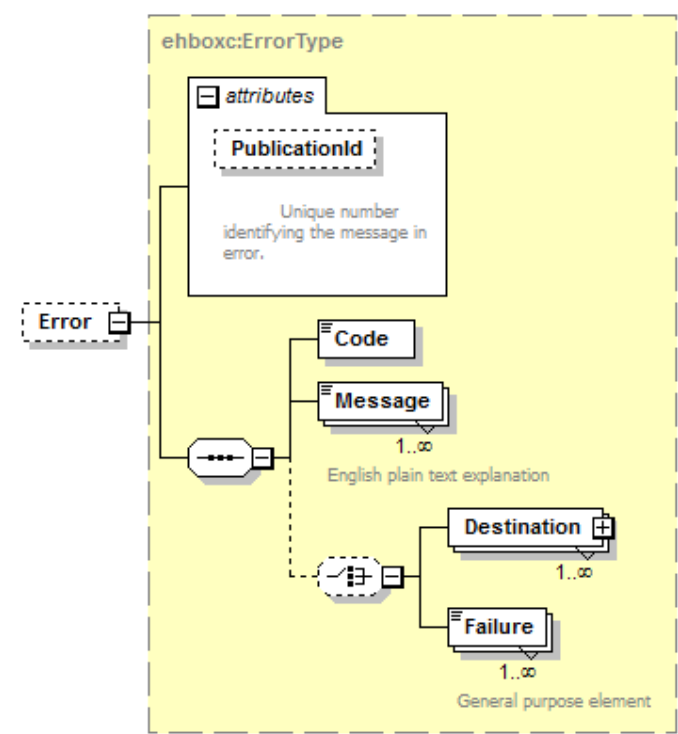
Please note that a message will either contain a News item or a Document, not both.



Field name	Descriptions
Title	A Document has a Title, a human readable description of its intent (string minimum 1, maximum 400).
EncryptableBinaryContent	Contains a SOAP reference to the attachment Eg. <EncryptableBinaryContent> cid:4906633928</EncryptableBinaryContent>. Type is xsd:anyURI. If IsEncrypted is true (see Section 5.3.14.7), the attachment must be encrypted (see section 5.1.2).
EncryptableTextContent	Text content, in binary form. If IsEncrypted is true (see Section 5.3.14.7), the title 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).



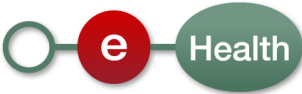
5.3.14.11 Error

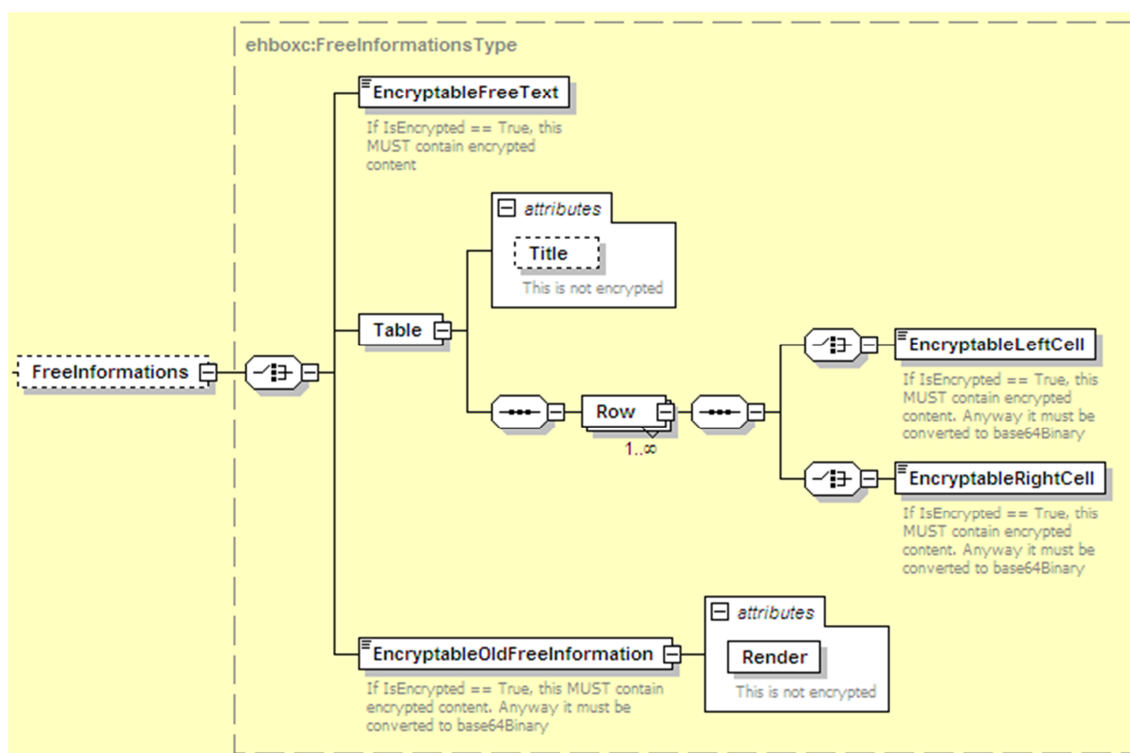


Field name	Descriptions
PublicationId	The PublicationId of the sent message (in error).
Code	The Error Code as defined in table 8.1. The number of possible error codes can evolve.
Message	A human readable English message.
Destination	The recipient causing an error because he could not be found for example. Check if the identifier and identifier type is correct.
Failure	All-purpose field containing technical Ids as PublicationId or MessageId. Not currently used, but for future needs.

5.3.14.12 FreeInformations

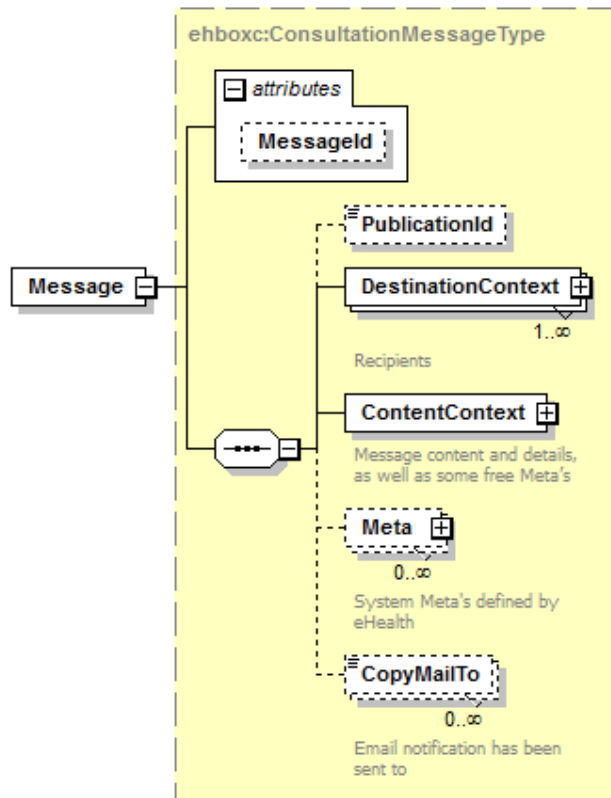
The sender is free to add more information via the FreeInformations field. These FreeInformations will be transparently provided to the recipient(s).



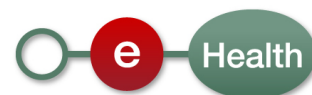


Field name	Descriptions
EncryptableFreeText	Text content, in binary form. If IsEncrypted is true (see Section 5.3.14.7), the text must be encrypted (see section 5.1.2).
Table	Title (the title of the table) and 1 or more Row(s) (each Row has a EncryptableLeftCell and a EncryptableRightCell as string). If IsEncrypted is True (see Section 5.3.14.7). 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 IsEncrypted is true (see Section 5.3.14.7), the content must be encrypted (see section 5.1.2).

### 5.3.14.13 Message



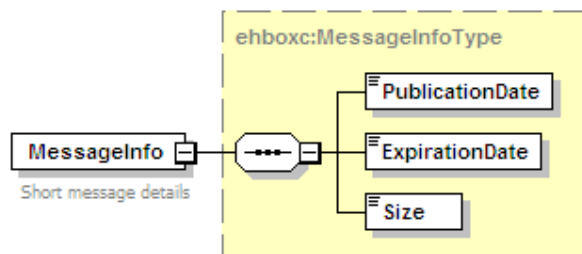
Field name	Description
MessageId	The MessageId is a unique message identification generated by the system and returned during the publication and when calling upon the getMessagesList. String of 13 digits.
PublicationId	The Id that the sender has used to publish the message. String, minimum 1, maximum 13.
DestinationContext	The DestinationContext is a complex type that contains information about the recipients. This type is further detailed in section 5.3.14.9. A Message can have numerous DestinationContext (numerous recipients).
ContentContext	The ContentContext is a complex type that contains the message content. This type is detailed in section 5.3.14.5.
Meta	<b>Currently, no meta information is defined.</b> Additional system meta information can be defined by eHealth and used in convention 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 5.3.14.15).
CopyMailTo	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 CopyMailTo



	<p>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 eHealthUpdateInfo (<a href="https://www.ehealth.fgov.be/webehealthupdateinfo/ehealthupdateinfo/">https://www.ehealth.fgov.be/webehealthupdateinfo/ehealthupdateinfo/</a>).</p>
--	--

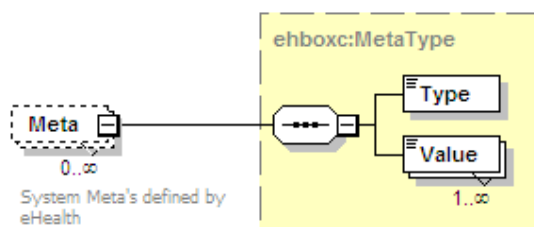
#### 5.3.14.14 MessageInfo

A MessageInfo contains short details about the message.



Field name	Descriptions
PublicationDate	The Date on which the message has been published (format: yyyy-mm-dd+hh:mm).
ExpirationDate	The Expiration Date of the message (format: yyyy-mm-dd+hh:mm). Please find more information about the Expiration Date in Lifetime of a message.
Size	The Size of the message in bytes.

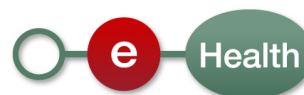
#### 5.3.14.15 Meta



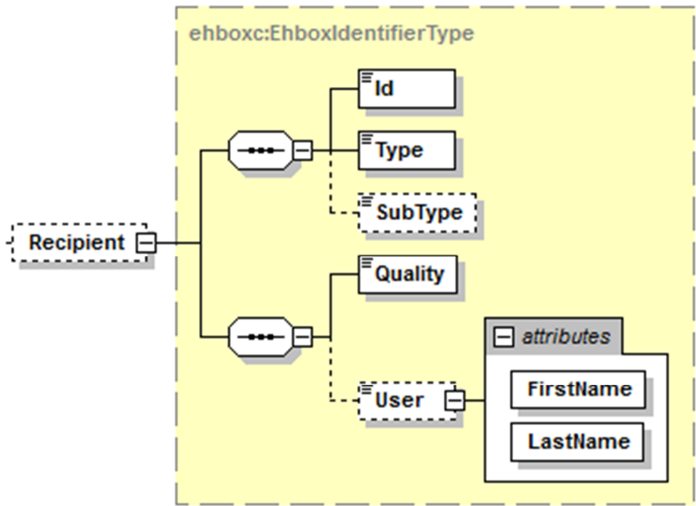
Field name	Descriptions
Type	The type of the meta information (string minimum 1, maximum 250).
Value	A list of Values for this Type (string minimum 1, maximum 250).

#### 5.3.14.16 Recipient

A Recipient contains all the information relative to your eHealthBox that received the message. This is very useful when you call upon the GetAllEhboxesMessagesList to identify which message was received by which



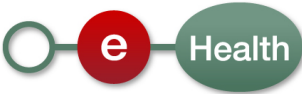
eHealthBox. After all, two different eHealthBoxes could receive the same message. The message would then appear two times in GetAllEhboxesMessagesList.



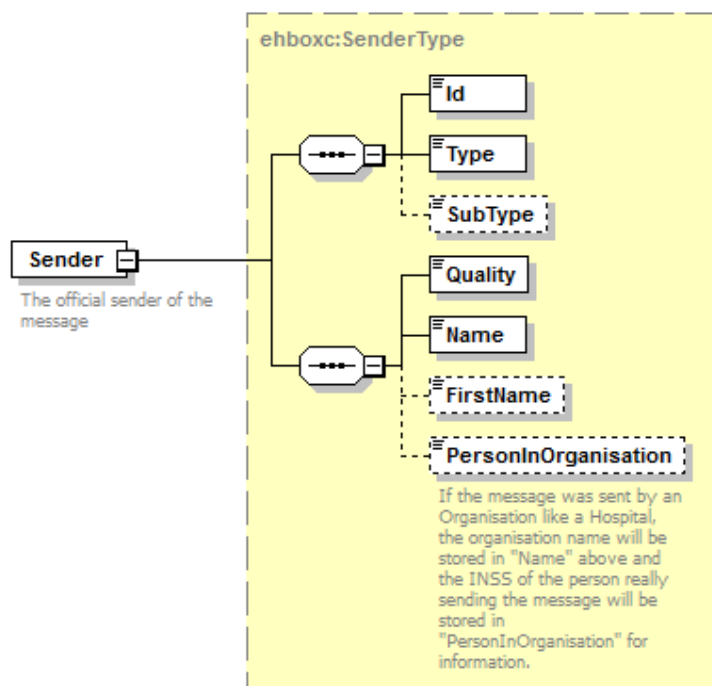
Field name	Descriptions
ID	The recipient's identification number. This is a digital number representing an INSS, NIHII, FAMPH, or CBE. String.
Type	The recipient's ID type ("INSS", "NIHII", "FAMPH" or "CBE"). String.
Subtype (obsolete)	If the recipient is an organization, the Subtype allows (if necessary) further specification (such as "HOSPITAL" SubType for a Hospital Quality, or "GROUP" SubType for a Group Quality). String.
Quality	A Quality defines the recipient's eHealthBox. String (see Annex 2 – List of Qualities).
User	An optional User (FirstName and LastName) 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).

5.3.14.17 Sender

A Sender element contains all the information relative to the Sender of the message.



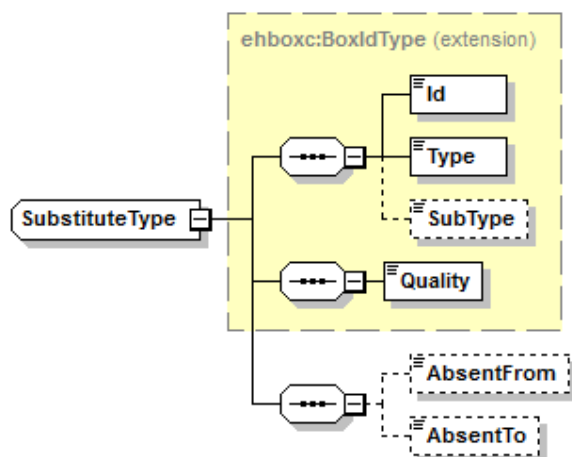




Field name	Descriptions
ID	This is a digital number representing an INSS, NIHII, FAMPH, or CBE. String.
Type	The Sender ID type ("INSS", "NIHII", "FAMPH" or "CBE"). String.
Subtype (obsolete)	If the sender is an organization, the Subtype allows (if necessary) further specification (such as "HOSPITAL" SubType for a Hospital Quality, or "GROUP" SubType for a Group Quality). String.
Quality	A Quality defines the Sender eHealthBox. String (see Annex 2 – List of Qualities).
Name	Name of the Sender. String.
FirstName	FirstName of the Sender (optional). String.
PersonInOrganisation	If the message was sent by an Organization like a Hospital, the organization name will be stored in "Name" above and the INSS of the person really sending the message will be stored in "PersonInOrganisation" for information. String.

#### 5.3.14.18 Substitute

A Substitute element contains all the information relative to the Substitute designated for a person.



Field name	Descriptions
ID	This is a digital number representing an INSS, NIHII, FAMPH, or CBE. String.
Type	The Sender ID type ("INSS", "NIHII", "FAMPH" or "CBE"). String.
Subtype (obsolete)	If the sender is an organization, the Subtype allows (if necessary) further specification (such as "HOSPITAL" SubType for a Hospital Quality, or "GROUP" SubType for a Group Quality). String.
Quality	A Quality defines the Sender eHealthBox. String (see Annex 2 – List of Qualities).
AbsentFrom	The AbsentFrom of the Out-of-Office. This date is inclusive. Ex. 2013-07-03+02:00
AbsentTo	The AbsentTo of the Out-of-Office. This date is inclusive. Ex. 2013-07-17+02:00

## 6 Risks and security

### 6.1 SOAP with Attachments

The standard is used to transfer binary data via SOAP web services. 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 swaRef:  
<http://ws-i.org/profiles/basic/1.1/swaref.xsd>

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

### 6.2 Security

#### 6.2.1 Business security

In case the development adds an additional use case based on an existing integration, eHealth (info@eHealth.fgov.be) must be informed at least one month in advance with a detailed estimate of the expected load. This will ensure an effective capacity management.

In case of technical issues on the web service, the partner may obtain support from the contact center (see eHealth contact page 8) that is responsible for this service.

In case eHealth finds a bug or vulnerability in its software, the partner is advised to update his application with the newest version of the software within 10 business days.

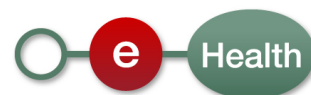
In case the partner finds a bug or vulnerability in the software or web service that eHealth has delivered, he is obliged to contact and inform eHealth immediately and he is prohibited to publish this bug or vulnerability in any case.

#### 6.2.2 Web Service Security Policy

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

- that the request is authenticated with the SAML security profile policy.

See the internet link <http://docs.oasis-open.org/ws-sx/ws-securitypolicy/200702/> for the specifications.



See also Annex 1 – SSO Access Annex 2 – List of Qualities for a more detailed description of the SSO Access in the case of eHealth.

- SSL one way.
- an X.509 certificate. This certificate will contain the identifiers of the caller: INSS or NIHL number or CBE enterprise number. More information on how to obtain a certificate:  
French version:  
<https://www.ehealth.fgov.be/fr/support/services-de-base/certificat-ehealth>  
Dutch version:  
<https://www.ehealth.fgov.be/nl/support/basisdiensten/ehealth-certificaten>
- the time-to-live of the message: one minute.
- the signature of the timestamp, body and binary security token. This will allow eHealth to verify the integrity of the message and the identity of the message author.

In order to use the web services, an authorization from eHealth is required.

### 6.2.3 Security policies to apply

We expect that you use SSL one way for the transport layer.

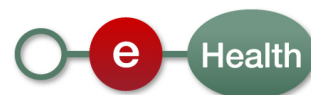
As a web service security policy, we expect:

- a timestamp (the date of the request), with a Time to live of one minute. (If the message doesn't arrive during this minute, it shall not be treated).
- the signature with the certificate of
  - the timestamp, (the one mentioned above)
  - the body (the message itself)
  - and the binary security token: an eHealth certificate or a SAML token issued by STS

This will allow eHealth to verify the integrity of the message and the identity of the message author.

A document explaining how to implement this security policy can be obtained on the eHealth portal website, support.

The STS cookbook can be found on the eHealth portal, Support.



## 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 service, please contact [info@ehealth.fgov.be](mailto: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 (service consumer) in order to connect to our web service. Most of the required information to integrate is published in the technical library on the eHealth portal site.

In some cases eHealth provides you with a mock-up service or test cases in order for you to test your client (service consumer) before releasing it in the acceptance environment.

##### 7.1.2.1 Create test cases

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

Access rules:

- To use the Consultation services, the user must be part of one of the following profiles: **hospital, nurse, group, institution, doctor, and laboratory ...**
- authentication with a certificate

All test cases have to be configured by the eHealth development team.

##### 7.1.2.2 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. You can request the test certificate through one of the following URLs:

- Dutch version: <https://www.ehealth.fgov.be/nl/support/basisdiensten/ehealth-certificaten>
- French version: <https://www.ehealth.fgov.be/fr/support/services-de-base/certificat-ehealth>

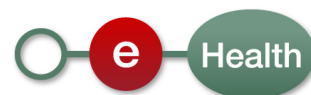
The process is described in the “How to request an eHealth test certificate”.

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

##### 7.1.2.3 Obtain SAML token

The usage of the Secure Token Service (STS) and the structure of the exchanged xml-messages are described in the eHealth STS cookbook.

In the case of eHealthBox Publication web service, see Annex 2 – List of Qualities.



### 7.1.3 Release procedure

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

From the moment you start the integration and acceptance tests, eHealth 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 point of contact by email.

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

For further information and instructions, please contact: [info@ehealth.fgov.be](mailto:info@ehealth.fgov.be).

### 7.1.4 Operational follow-up

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

## 7.2 Test cases

This section describes a step-by-step process to test the Consultation service.

eHealth recommends performing tests for all of the following cases:

1. Consult your eHealthbox information with the method "getBoxInfo".
2. Based on your tests cases defined previously for the Publication service, get the list of messages contained in your eHealthbox with the method "getMessagesList()". Execute this request on your inbox, sentbox and on your bin.
3. Pick one of the "MessageIds" returned from your messages list and use it to get the full message with the method "getFullMessage()". Execute this request on a message from your inbox and sentbox.
4. Use again your "MessageID" to move your received message from your inbox to your recycle bin with the method "moveMessage()".
5. Always consult the previous history of your message through a "MessageID" with the method "getMessageHistory()".
6. Use a "MessageID" from your sentbox to consult the state of that message with "getMessageAcknowledgmentsStatus()".
7. Consult all your messages from all your eHealthboxes with "GetAllEhboxesMessagesList()". Execute this request on your inbox, sentbox and on your bin.



## 8 Error and failure messages

### 8.1 Error Element Status Codes

Mail delivery system error codes originating from the application:

These error codes first indicate a problem with the message and or its recipients. This table can evolve.

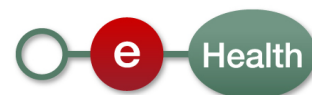
Error code	Component	Description	Solution
700	Error	Unknown technical error.	Call the Contact Center
701	Error	Business validation error.	Check all fields, especially the recipients. The publication id can be found under Publication id.
702	Error	Duplicate publication id.	Chose a new publication id, and send again. The duplicate publication id can be found under PublicationId.
703	Error	One or more recipients are invalid.	Invalid recipients can be found under Destination.

### 8.2 Consultation Response Status Codes

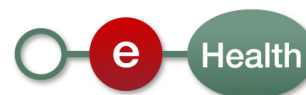
Error codes originating from the eHealth platform:

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

Error code	Component	Description	Solution
100	*	SUCCESS	
806	* / MessageId	The specified MessageId is invalid; please verify that the Source and the MessageId are correct and that you can access it.	Is the MessageId correct? Is the message really present in "Source"?
807	* / Start & EndIndex	EndIndex must be larger or equal to StartIndex; please correct StartIndex and EndIndex.	Is "EndIndex > StartIndex"?
808	* / Start & EndIndex	A maximum of 100 messages can be returned by request; please correct StartIndex and EndIndex.	Is "EndIndex - StartIndex + 1 <= 100"?



809	GetMessage-AcknowledgmentsStatus/MessageId	The specified MessageID is invalid; please verify that the MessageID is correct and that you are the sender.	Are you the sender of the message? Is the MessageId correct?
810	* / BoxId	The specified BoxId is invalid; please verify the data and that you can access it.	Can you normally access that eHealthBox?
812	MoveMessage/ Source & Destination	You can not move a message from your Inbox to your Sentbox (even via recycle bin) and vice versa.	Is the message really present in "Source"?
813	MoveMessage/ Source & Destination	Not all messages were moved successfully. Please verify for each message that the Source and the MessageID are correct. Also pay attention that a message in the recycle bin which was moved from the Inbox cannot be restored back to the Sentbox and vice versa.	Some messages were not found in the folder specified in "Source". Some messages can not be moved to "Destination".
815	DeleteMessage	Not all messages were deleted successfully. Please verify for each message that the Source and MessageId are correct.	Not deleted Messages are listed in MessageId in the response.
820	InsertOoO	The period 01/07/2012 to 12/07/2012 is invalid because it overlaps another period.	Correct start or end date.
821	InsertOoO	The end of the period cannot be further than a year in the future.	Correct end date.
822	InsertOoO	The start date can't be after the end date.	Correct start date.
823	InsertOoO	The start date can't be in the past.	Correct start date.
824	InsertOoO	One or more substitutes cannot be chosen because they are absent.	Change one or more substitutes.
825	InsertOoO	The number of substitutes may not exceed 5.	Remove a substitute.
826	InsertOoO	The number of out of office for one eHealthBox may not exceed 10.	Delete and out of office.
827	InsertOoO	One or more substitutes are unknown or not correct, please correct them.	Correct one or more substitutes.
828	InsertOoO	The user is unknown or not correct, please correct him.	Re-authenticate.
829	InsertOoO	A valid substitute is a person, not an organization.	Change one or more substitutes.
830	InsertOoO	A person cannot be substitute for himself.	Change one or more substitutes.
840	DeleteOoO	One or more OoOld are invalid.	Verify the OoOld's.

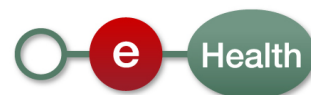


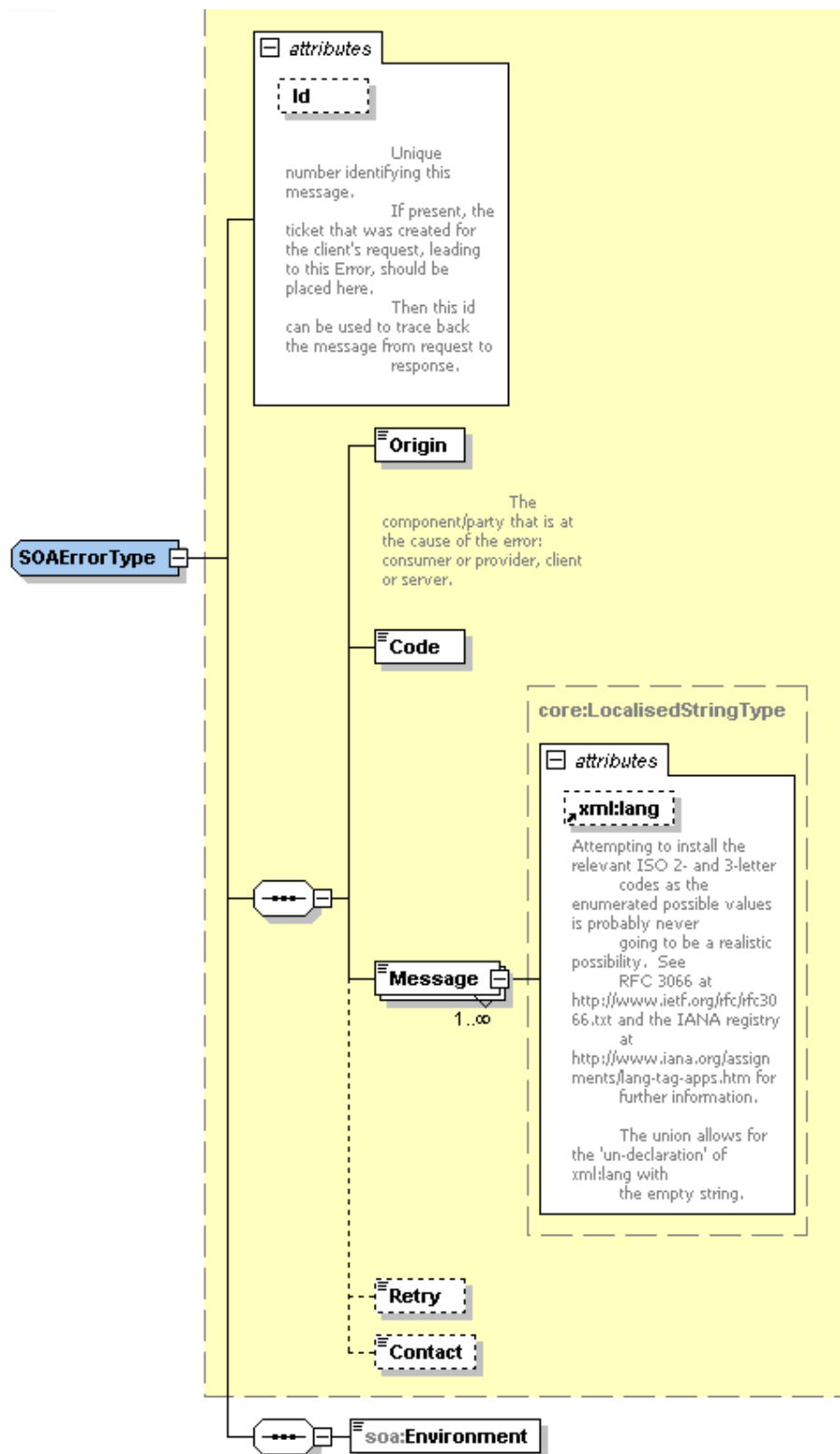


## 8.3 Soap Fault Error Codes

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, leading to this error. Should be placed here. 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 environment in which the error occurs: integration, acceptance or production.





### 8.3.1 Schema Validation Errors

When invoking the Web Service, 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:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-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>
```

### 8.3.2 Technical Errors

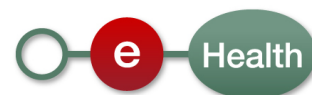
Technical errors are errors inherent to the internal working of the eHealth Web Service. These errors can also occur if the token used to call the web service is not valid.

They contain the standard SOAP Fault attributes.

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

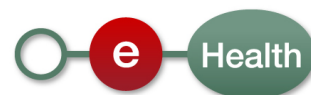
Code	Message
SOA-00001	An internal error has occurred. Please contact the Contact Center.

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.ehealth.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>
```



## 9 Annex 1 – SSO Access

To access the eHealthBox web service, a previous call must be made to the STS Web Service to obtain a valid Token. This Token will be used to identify the sender.

The usage of the Secure Token Service and the structure of the exchanged xml-messages are described in the eHealth STS cookbook.

You will find a step-by-step description of how to call the STS with JAXWS and Axis in the “How to call a webservice” cookbook. Every identification attribute that is required by a particular service must be provided in the request. In the case of eHealthBox these attributes depend on the sender quality.

NIHII Numbers can be of 8 or 11 characters, CBE numbers are 8 characters long and INSS are 11 characters long.

### 9.1 Sender is an organization.

The identification attributes are the following:

- `AttributeNameSpace="urn:be:fgov:identification-namespace "`  
`AttributeName=certificate attribute`
- `AttributeNameSpace="urn:be:fgov:identification-namespace "`  
`AttributeName=identification attribute`
- `AttributeNameSpace="urn:be:fgov:identification-namespace "`  
`AttributeName="urn:be:fgov:health:person:ssin"`

You will find the `certificate attribute` and `identification attribute` for an organization in annex (Annex 2 – List of Qualities).

We ask you to add the national registry number (ssin) from the person who is responsible for the access to the web service inside the hospital. This national number is used in the security loggings from the web service.

The certification attribute is:

- `AttributeNameSpace="urn:be:fgov:certified-namespace:health"`  
`AttributeName=certification attribute`

You will find the `certification attribute` for an organization in 10Annex 2 – List of Qualities.

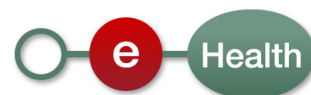
#### Remark:

Some organizations have not certification attribute. For example, a healthcare organization (quality: INSTITUTION) where the `certification attribute` is useless. However the eHealthBox system will check in the “Usermanagement” system that the cbe number from the organization is a known “healthcare” organization.

In addition, as described on page 5 of the “How to call a webservice” cookbook, you need to specify every requested attribute in a SAMLAttributeDesignator list.

Example: for a hospital, we expect the following list:

- the 3 identification attributes:



- `AttributeNamespace="urn:be:fgov:identification-namespace"`  
`AttributeName="urn:be:fgov:health:1.0:certificateholder:hospital:nihii-number"`
- `AttributeNamespace="urn:be:fgov:identification-namespace"`  
`AttributeName="urn:be:fgov:health:1.0:hospital:nihii-number"`
- `AttributeNamespace="urn:be:fgov:identification-namespace"`  
`AttributeName="urn:be:fgov:health:person:ssin"`

- the certification attribute is:

- `AttributeNamespace="urn:be:fgov:certified-namespace:health"`  
`AttributeName="urn:be:fgov:health:1.0:hospital:nihii-number:recognisedhospital:boolean"`

To access the eHealthBox web service, the response token must contain "true" for the certification attribute.

## 9.2 Sender is a professional

Use the certificate from the person to call the STS.

- The identification attributes are the following:

- `AttributeNamespace="urn:be:fgov:identification-namespace"`  
`AttributeName="urn:be:fgov:health:1.0:certificateholder:person:ssin"`
- `AttributeNamespace="urn:be:fgov:identification-namespace"`  
`AttributeName="urn:be:fgov:health:person:ssin"`

- The certification attribute is:

- `AttributeNamespace="urn:be:fgov:certified-namespace:health"`  
`AttributeName= certification urn`

You will find the [certification attribute](#) for a professional in **Error! Reference source not found.**

Remark:

A non-professional (citizen) has not identification and certification attributes.

In addition, as described on page 5 of the "How to call a webservice" cookbook, you need to specify every requested attribute in a SAMLAttributeDesignator list.

Example: for a doctor, we expect the following list:

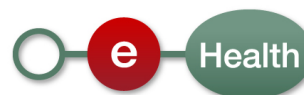
- The identification attributes are the following:

- `AttributeNamespace="urn:be:fgov:identification-namespace"`  
`AttributeName="urn:be:fgov:health:1.0:certificateholder:person:ssin"`
- `AttributeNamespace="urn:be:fgov:identification-namespace"`  
`AttributeName="urn:be:fgov:health:person:ssin"`

- The certification attribute is:

- `AttributeNamespace="urn:be:fgov:certified-namespace:health"`  
`AttributeName="urn:be:fgov:person:ssin:doctor:Boolean"`

To access the eHealthBox web service, the response token must contain "true" for the certification attribute.



## 10 Annex 2 – List of Qualities

Currently, the allowed combinations Type – Quality for the owner of an eHealthBox are:

**Special Note:** It is not possible anymore to send a message to a CITIZEN.

Quality name	Quality eHbox	Type	Certificate attribute	Identification attribute	Certification attribute
Daycare center	DAY_CARE_CENTER	NIHII	urn:be:fgov:health:1.0:certificateholder:daycarecenter:nihi-number	urn:be:fgov:health:1.0:daycarecenter:nihi-number	urn:be:fgov:health:1.0:daycarecenter:nihi-number:recogniseddaycarecenter:boolean
Group of nurse	GROUP	NIHII	urn:be:fgov:health:1.0:certificateholder:groupofnurses:nihi-number	urn:be:fgov:health:1.0:groupofnurses:nihi-number	urn:be:fgov:health:1.0:groupofnurses:nihi-number:recognisedgroupofnurses:boolean
Group of doctors	GROUP_DOCTORS	NIHII	urn:be:fgov:health:1.0:certificateholder:groupdoctors:nihi-number	urn:be:fgov:health:1.0:groupofdoctors:nihi-number	urn:be:fgov:health:1.0:groupofdoctors:nihi-number:recognisedgroupofdoctors:boolean
Guard post	GUARD_POST	NIHII	urn:be:fgov:health:1.0:certificateholder:guardpost:nihi-number	urn:be:fgov:health:1.0:guardpost:nihi-number	urn:be:fgov:health:1.0:guardpost:nihi-number:recognisedguardpost:boolean
Homecare service	HOME_SERVICES	NIHII	urn:be:fgov:health:1.0:certificateholder:homecareservices:nihi-number	urn:be:fgov:health:1.0:homecareservices:nihi-number	urn:be:fgov:health:1.0:homecareservices:nihi-number:recognisedhomecareservices:boolean
Hospital	HOSPITAL	NIHII	urn:be:fgov:health:1.0:certificateholder:hospital:nihi-number	urn:be:fgov:health:1.0:hospital:nihi-number	urn:be:fgov:health:1.0:hospital:nihi-number:recognisedhospital:boolean
Institution	INSTITUTION	CBE	urn:be:fgov:health:1.0:certificateholder:enterprise:cbe-number	urn:be:fgov:kbo-bce:organization:cbe-number	
Institution EHP	INSTITUTION_EHP	EHP	urn:be:fgov:health:1.0:certificateholder:or	urn:be:fgov:health:1.0:organization:ehp-	urn:be:fgov:health:1.0:organization:ehp-







Psychiatric house	PSYCH_HOUSE	NIHII	urn:be:fgov:health:1.0:certificateholder:psychiatrichouse:nihi-number	urn:be:fgov:health:1.0:psychiatrichouse:nihi-number	urn:be:fgov:health:1.0:psychiatrichouse:nihi-number:recognisedpsychiatrichouse:boolean
Rest home	RETIREMENT	NIHII	urn:be:fgov:health:1.0:certificateholder:retirement:nihi-number	urn:be:fgov:health:1.0:retirement:nihi-number	urn:be:fgov:health:1.0:retirement:nihi-number:recognisedretirement:boolean
Treatment center	TREATMENT_CENTER	CBE	urn:be:fgov:health:1.0:certificateholder:enterprise:cbe-number:treatmentcenter	urn:be:fgov:kbo-bce:organization:cbe-number:health:1.0:treatmentcenter	

This list is able to evolve.

Quality name	Quality eHbox	Type	Certification attribute
Ambulance	AMBULANCE_RESCUER	INSS	urn:be:fgov:person:ssin:health:1.0:professional:ambulancerescuer:boolean
Audician	AUDICIEN	INSS	urn:be:fgov:person:ssin:health:1.0:professional:audician:Boolean
Audiologist	AUDIOLOGIST	INSS	urn:be:fgov:person:ssin:health:1.0:professional:audiologist:boolean
Bachelor family science	FAMILY_SCIENCE_BACHELOR	INSS	urn:be:fgov:person:ssin:health:1.0:professional:bachelorfamilyscience:boolean
Bachelor readaptation	READAPTATION_BACHELOR	INSS	urn:be:fgov:person:ssin:health:1.0:professional:readaptationbachelor:boolean
Bandagist	TRUSS_MAKER	INSS	urn:be:fgov:person:ssin:health:1.0:professional:trussmaker:boolean
Citizen	CITIZEN	INSS	
Dentist	DENTIST	INSS	urn:be:fgov:person:ssin:health:1.0:professional:dentist:Boolean
Dietist	DIETICIAN	INSS	urn:be:fgov:person:ssin:health:1.0:professional:dietician:Boolean
Doctor	DOCTOR	INSS	urn:be:fgov:person:ssin:doctor:boolean
Implant provider	IMPLANTPROVIDER	INSS	urn:be:fgov:person:ssin:health:1.0:professional:implantprovider:boolean
Labo technologist	LAB_TECHNOLOGIST	INSS	urn:be:fgov:person:ssin:health:1.0:professional:labtechnologist:boolean
Logopedist	LOGOPEDIST	INSS	urn:be:fgov:person:ssin:health:1.0:professional:logopedist:boolean



Master applied psychology	APPLIED_PSYCH_BACHELOR	INSS	urn:be:fgov:person:ssin:ehealth:1.0:professional:appliedpsychbachelor:boolean
Mastergerontology	GERONTOLOGY_MASTER	INSS	urn:be:fgov:person:ssin:ehealth:1.0:professional:gerontologymaster:boolean
Masterpsychomotorthrapy	PSYCHOMOTOR_THERAPY	INSS	urn:be:fgov:person:ssin:ehealth:1.0:professional:psychomotorthrapy:boolean
Medicalimagingtechnologist	IMAGING_TECHNOLOGIST	INSS	urn:be:fgov:person:ssin:ehealth:1.0:professional:imagingtechnologist:boolean
Midwife	MIDWIFE	INSS	urn:be:fgov:person:ssin:ehealth:1.0:professional:midwife:Boolean
Nurse	NURSE	INSS	urn:be:fgov:person:ssin:nurse:boolean
Occupational therapist	OCCUPATIONAL_THERAPIST	INSS	urn:be:fgov:person:ssin:ehealth:1.0:professional:occupationaltherapist:boolean
Optician	OPTICIEN	INSS	urn:be:fgov:person:ssin:ehealth:1.0:professional:optician:boolean
Orthopedagogist master	ORTHOPEDEGOGIST_MASTER	INSS	urn:be:fgov:person:ssin:ehealth:1.0:professional:orthopedagogistmaster:boolean
Orthopedist	ORTHOPEDIST	INSS	urn:be:fgov:person:ssin:ehealth:1.0:professional:orthopedist:boolean
Orthoptist	ORTHOPTIST	INSS	urn:be:fgov:person:ssin:ehealth:1.0:professional:orthoptist:boolean
Pediatricnurse	PEDIATRIC_NURSE	INSS	urn:be:fgov:person:ssin:ehealth:1.0:professional:pediatricnurse:boolean
Pharmacist	PHARMACIST	INSS	urn:be:fgov:person:ssin:ehealth:1.0:professional:pharmacist:boolean
Pharmacy assistant	PHARMACIST_ASSISTANT	INSS	urn:be:fgov:person:ssin:ehealth:1.0:professional:pharmacistassistant:boolean
Psychologist	PSYCHOLOGIST	INSS	urn:be:fgov:person:ssin:ehealth:1.0:professional:psychologist:boolean
Physiotherapist	PHYSIOTHERAPIST	INSS	urn:be:fgov:person:ssin:ehealth:1.0:professional:physiotherapist:boolean
Podiatrist	PODOLOGIST	INSS	urn:be:fgov:person:ssin:ehealth:1.0:professional:podologist:boolean
Practical nurse	PRACTICALNURSE	INSS	urn:be:fgov:person:ssin:ehealth:1.0:professional:practicalnurse:boolean
Social worker	SOCIAL_WORKER	INSS	urn:be:fgov:person:ssin:ehealth:1.0:professional:socialworker:boolean
Specialized educator	SPECIALIZED_EDUCATOR	INSS	urn:be:fgov:person:ssin:ehealth:1.0:professional:specializededucator:boolean

This list is able to evolve.



## 11 List of sources

This list is a summary of all links used in the document referring to internet resources. eHealth is not responsible for these links.

	Date
<a href="http://www.w3.org/standards/xml/">http://www.w3.org/standards/xml/</a>	24/03/11
<a href="http://www.oasis-open.org/specs/">http://www.oasis-open.org/specs/</a>	24/03/11
<a href="http://www.w3.org/TR/soap12-mtom/">http://www.w3.org/TR/soap12-mtom/</a>	30/09/11

