

**CoBRHA Publication
Common Base Registry for Healthcare Actors
Cookbook
Version 3.4**

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

eHealth platform

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

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

Table of contents

Table of contents	2
1 Document management	4
1.1 Document history	4
2 Introduction	6
2.1 Goal of the service	6
2.2 Goal of the document	6
2.3 eHealth platform document references	7
2.4 External document references	7
3 Support	8
3.1 Helpdesk eHealth platform	8
3.1.1 Certificates	8
3.1.2 For issues in production	8
3.1.3 For issues in acceptance	8
3.1.4 For business issues	8
3.2 Status	8
4 Global overview	9
5 Step-by-step	10
5.1 Global technical requirements	10
5.2 Technical requirements for file exchange	10
5.2.1 Creation SFTP account procedure	10
5.2.2 Creation publication file	11
5.3 Technical requirements for web services	11
5.3.1 Synchronous Web Service WSDL (Web Service Definition Language)	11
5.3.2 Security policies to apply (Web Service)	11
5.3.3 WS-I Basic Profile 1.1	12
5.3.4 Tracing	12
5.4 Process overview for file exchange by SFTP	12
5.5 Process overview for web services	13
5.6 Publication Input Data (ehealth-cobrha-schema-core.xsd)	13
5.7 Request Report (ehealth-cobrha-schema-report.xsd)	13
5.8 Synchronous Publish Operation Web Service	13
6 Risks and security	14
6.1 Security	14
6.1.1 Business security	14
6.1.2 File Exchange	14
6.1.3 Web service	14
6.1.4 MTOM Policy	14
6.1.5 The use of username, password and token	14
7 Test and release procedure	15



7.1	Procedure.....	15
7.1.1	Procedure for file exchange.....	15
7.1.2	Initiation	15
7.1.3	Development and test procedure	15
7.1.4	Release procedure.....	15
7.1.5	Operational follow-up	15
7.1.6	Test cases.....	15
7.1.7	Procedure for web services	16
8	Status codes and error messages	18
8.1	CoBRHA Business Status Codes.....	18
8.2	Soap Fault Error Codes.....	21
8.2.1	Schema Validation Errors	23
8.2.2	Technical Errors	23
9	Example	25



1 Document management

1.1 Document history

Version	Date	Author	Description of changes / remarks
1.0	01/04/2011	eHealth platform	initial version
1.2	10/05/2011	eHealth platform	removed CRAB AS for Municipality in AddressType
1.4	26/05/2011	eHealth platform	added web service technical requirements + xsd structure
1.5	30/05/2011	eHealth platform	added SOAP Technical error message structure
1.6	30/05/2011	eHealth platform	corrected strange characters in text
2.0	31/05/2011	eHealth platform	Include internal eHealth reviewers' comments
2.1	14/06/2011	eHealth platform	added the nillable functionality and explanation for the InvalidElement attribute.
2.2	17/06/2011	eHealth platform	Changed National Registry as authentic source to NR
2.3	11/08/2011	eHealth platform	Added type 'municipality' for the NIScode for municipality in the explanation of the Municipality tag in Address
2.4	14/09/2011	eHealth platform	Removed address element from NIHIIProfessionalType , corrected NIHIIProfessionalD to NIHIIProfessionalID, Changed The Language and origin elements to CodeWithDescriptionType.
2.5	26/09/2011	eHealth platform	Corrected the SFTP account login procedure (changed username-password to username-private key file)
2.6	26/09/2011	eHealth platform	Better explanation of the nillable functionality
2.7	27/09/2011	eHealth platform	removed InvalidElement from CodeWithDescriptionType + changed Abbreviation multiplicity + removed other languages than English in labeltype + added missing statuses + added LabelType
2.8	03/11/2011	eHealth platform	Changed namespace cdm:v1 to cobrha:core
2.9	04/11/2011	eHealth platform	Changed cobrha-cdm.xsd to cobrha-core.xsd Changed IndividualID multiplicity + attribute current
3.0	16/12/2011	eHealth platform	Added the five rules explanation for the publication
3.1	22/05/2014	eHealth platform	Remove deprecated content
3.2	19/06/2017	eHealth platform	Updates related to new xsd version 1.7
3.3	04/07/2018	eHealth platform	Update
3.4	05/08/2022	eHealth platform	§ 2.3 eHealth document references (updated) § 2.4 External document references (added) § 3 Support (updated)

			§ 5.3.3 WS-I Basic Profile (added) § 5.3.4 Tracing (added)
--	--	--	---



2 Introduction

2.1 Goal of the service

The goal is to publish data concerning healthcare providers into a common database (known as CoBRHA for Common Base Registry for Healthcare Actors). This common DB will be essentially used by partners to control access to specific healthcare applications or web services (WS) via the user access management (service provided by the eHealth platform).

The data scope of the **CoBRHA DB** is limited to core data that are exchanged between partners and that concern Belgian recognized healthcare actors. Hence, this scope is limited to inter-organisational rather than intra-organisational data exchanges. We essentially attempt to answer three questions over a healthcare actor:

Who is he? Healthcare actor identification as an *individual* (Physician, Dentist, etc.) or as an *organisation* (Hospital, Pharmacy, Nurse Group, etc.). It includes attributes such as: Identification Number (INSS, KBO, INAMI, EHP), Address(es), contact information, etc.

What is he allowed to do? Activities realized by healthcare organisations (ex: General Hospital, Intensive care, SMUR, etc.) or professions, specialities obtained by individuals (Cardiologist, radiologist, etc.).

What are his responsibilities? Roles played by healthcare actors (ex: Chief Physician) eventually for another healthcare actor.

This service gives the user the possibility to publish data in the CoBRHA DB by file exchange (SFTP) or WS.

This service **allows**:

- New publications and updates.
- Logical deletes of published data if they do not result into broken relationships with other data.

This service **checks** publication authorization rules and data integrity business rules to validate the input data. This input data is also consolidated if needed with additional authentic values before inserting them in the CoBRHA DB. For example, the authentic source for general info about individuals is the NR. So if an authentic source like the FPS_PH-PH creates a new healthcare professional, general information about this individual is pulled from the NR to consolidate the FPS_PH-PH data.

All publication requests are tracked in details in a follow-up service. Hence, a detailed report can be requested at all times to the follow-up.

The publication service will receive as **input** an XML structure that will be similar both for file exchange and WS publications.

The publication service will provide as **output**, an acknowledgment or error/warning message(s) describing why the data has not been published or potential issues (See section 5.4 for more details).

2.2 Goal of the document

This document is not a development or programming guide for internal applications. Instead it provides functional and technical information and allows an organization to integrate and use this service of the eHealth platform.

However, in order to interact in a smooth, homogeneous and risk controlled way with a maximum of partners, eHealth partners must commit to comply with the requirements of specifications, data format and release processes described in this document.

Technical and business requirements must be met in order to allow the integration and validation of the eHealth service in the client application.



2.3 eHealth platform document references

All the document references can be found on the eHealth platform portal¹. These versions or any following versions can be used for the service of the eHealth platform.

<i>ID</i>	<i>Title</i>	<i>Version</i>	<i>Date</i>	<i>Author</i>
1	Glossary	1.0		eHealth platform
2	STS HolderofKey – Cookbook	1.5	13/07/2022	eHealth platform
3	Call to the eHealth SSO web service to obtain a SAML Token	1.0	21/10/2010	eHealth platform
4	Cookbook CoBRHA Consultation	1.2	04/07/2018	eHealth platform

2.4 External document references

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

<i>ID</i>	<i>Title</i>	<i>Source</i>	<i>Date</i>	<i>Author</i>
1	MTOM Policy	http://www.w3.org/TR/soap12-mtom/ for the technical specification		
2	Basic Profile Version 1.1	http://www.ws-i.org/Profiles/BasicProfile-1.1-2004-08-24.html	24/08/2004	Web Services Interoperability Organization

¹ <https://www.ehealth.fgov.be/ehealthplatform>

3 Support

3.1 Helpdesk eHealth platform

3.1.1 Certificates

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

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

For technical issues regarding eHealth platform certificates

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

3.1.2 For issues in production

eHealth platform contact centre:

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

3.1.3 For issues in acceptance

Integration-support@ehealth.fgov.be

3.1.4 For business issues

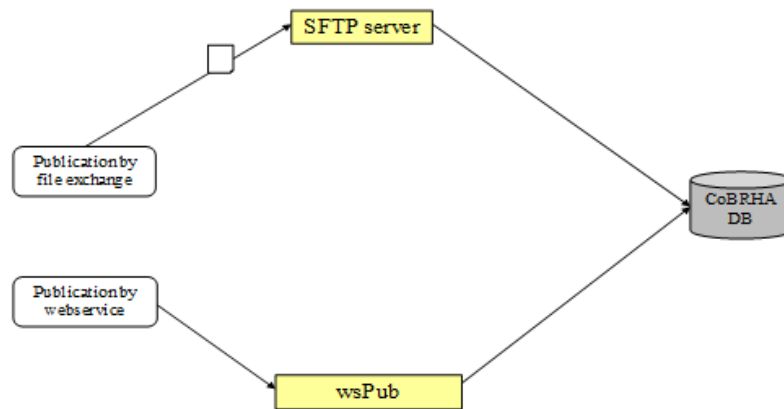
- regarding an existing project: the project manager in charge of the application or service
- regarding a new project or other business issues: info@ehealth.fgov.be

3.2 Status

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



4 Global overview



Asynchronous Publication by file exchange

The publication file is a zip-file that can contain multiple files. One file is the metadata **utf-8** xml file and the other files are operation **utf-8** xml files, containing multiple operations (max 20000 per file). You can directly drop a file on the system using SFTP. The file is published by batch.

Synchronous Publication by web service

You can make a WS 'publish' operation (**one operation**) to the COBRHAPublicationService WS with one publication operation in **utf-8** inside the call. This is done by calling the WS publication (wsPub). An attribute can be set in the PublishRequest element named IssueLevel. This IssueLevel can have as values: *debug*, *info*, *warning*, *error* or *critical*. The IssueLevel is used to filter out all operations in the report that has a status that has a lower severity level than his value. For example: IssueLevel is 'warning' -> only operations with current status warning, error and critical are allowed in the report. By default, the IssueLevel is on debug for the reporting.

5 Step-by-step

5.1 Global technical requirements

- Publication xml containing operations must follow the ehealth-cobrha-schema-core.xsd structure described in 5.6.
- All xml sent to the eHealth platform must be encoded in **utf-8**.
- These are the five rules that the publication algorithm tries not to break:
 - Our system will try to publish every element or attribute that is available in the XML.
 - It will never publish an element or attribute inside an operation that is not in the xml. You are not authorized to guess the value that should be given.
 - You cannot set attributes to null.
 - Only SimpleType elements can be set to null by giving them the attribute `xsi:nil = 'true'`.
 - You cannot set ComplexTypes to null (because they are records in our DB), but you can delete them logically by setting `InvalidElement="true"`.

5.2 Technical requirements for file exchange

5.2.1 Creation SFTP account procedure

To receive a SFTP account on our system, you should contact the eHealth team (info@ehealth.fgov.be). They will deliver a username+ private key file and the ftp URL.

They are three folders on the SFTP:

One folder is called 'PublicationReport'. This folder contains the request reports linked to a publication request.

Second folder is called 'ToBePublished'. This folder contains the publication files that have to be published. The filename structure has the same structure that files in the publication report.

Files in these folders have the following filename:

PUBLISHERNAME_YYYYMMDD_SEQ_requestid_type_ENV.zip where:

- PUBLISHERNAME = authentic source which has published the data. Ex: WVG_VAZG, FPS_PH,...
- YYYY_MM_DD_SEQ = date + sequence number given by the publisher
- request_id = technical id given by our system to the publication request (only for files in PublicationReport).
- type = file type : report (publicationReport) or pub (ToBePublished')
- ENV = environment where the files will be (publication) or have been (report) published : acc = acceptance / prod = production).

Third folder is named snapshot, which contains a filter view of data published in CoBRHA database by an authentic source. A snapshot is not an automatic process. Therefore, the authentic source should contact the eHealth platform to configure and receive a snap host. The file in this folder has the following filename structure: **PUBLISHERNAME_YYYYMMDD_REQUESTID_Snapshot_TYPE_FILTER_ENV.zip** where

- PUBLISHERNAME = Authentic Source that will receive the snapshot (in the specific folder on the SFTP. Ex: WVG_VAZG, FPS_PH, ...)
- YYYYMMDD: execution date of the snapshot
- REQUESTID = Technical ID of the request
- TYPE = Explicit name that describes the selection of data contained in the snapshot
- FILTER = filter(s) used to generate the snapshot
- ENV: environment where the snapshot has been created: acc = acceptance / prod = production)



5.2.2 Creation publication file

To publish in CoBRHA database, you need to create a zip file, which contains one or several publications xml file and a zip metadata xml file (zipmetadata.xml). The zip metadata is required because used to define the publication xml file to publish in CoBRHA database.

Some remarks:

- The content of the two xml has to be encoded in utf-8
- The filename of zip file ought to reflect the following regex:
PUBLISHERNAME_YYYY_MM_DD_SEQ_pub_ENV
- A publication request by file exchange has to be sent between 9 am and 22 pm to be treated correctly by the batch during the night.

5.3 Technical requirements for web services

5.3.1 Synchronous Web Service WSDL (Web Service Definition Language)



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

- The types (**PublishRequest** and **PublishResponse**) that are used by the publish operation. The fault message is also defined (see section 5.8).
- The **publish** operation.
- The SOAP technical error message (see section 8.2).
- The applicable **security** aspects (see section 6.1).

Remark: only encoding with UTF-8 is allowed.

5.3.2 Security policies to apply (Web Service)

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

As WS security policy (OASIS standard), we expect:

- A timestamp (the date of the request), with a Time to live of one minute (if the message does not 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 (see section **Error! Reference source not found.**)

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 from the eHealth platform.



5.3.3 WS-I Basic Profile 1.1

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

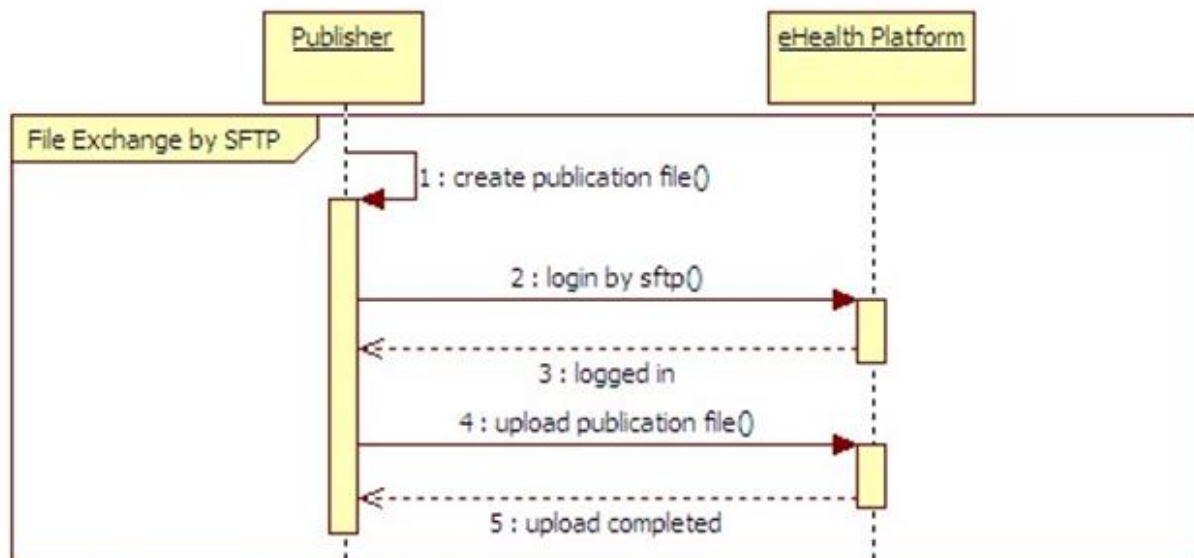
5.3.4 Tracing

To use this service, the request SHOULD contain the following two http header values (see RFC <https://datatracker.ietf.org/doc/html/rfc7231#section-5.5.3>):

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

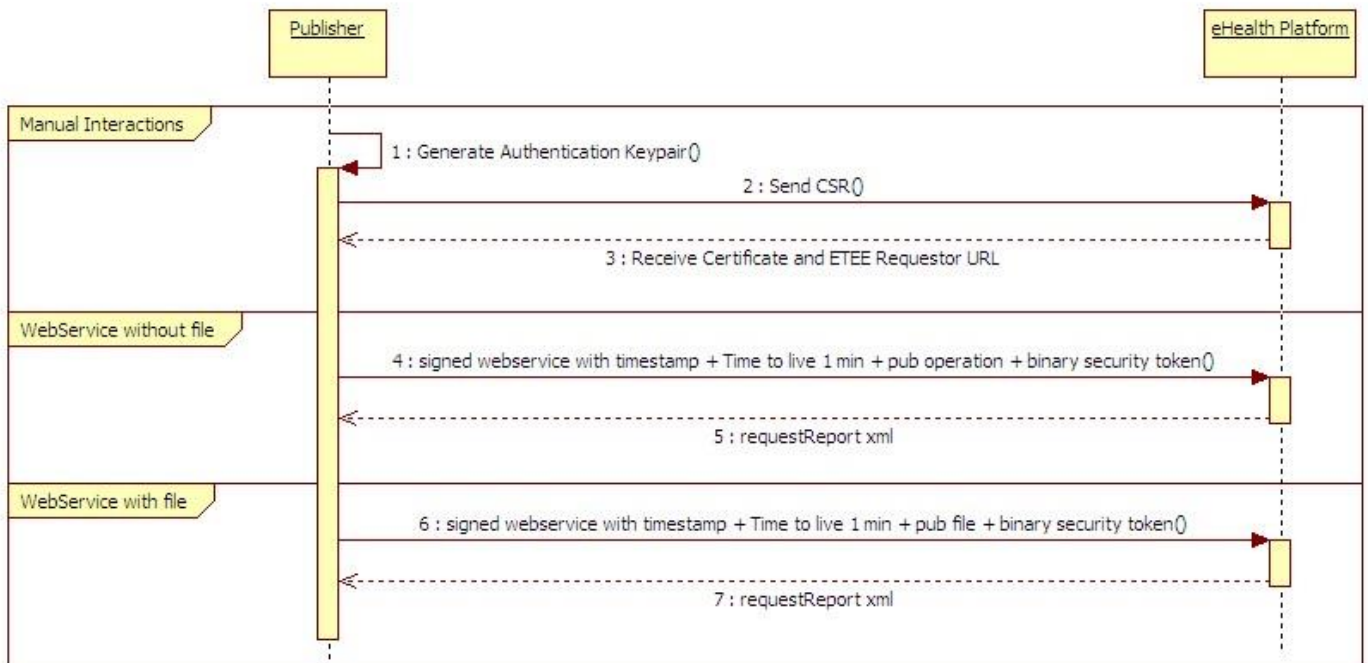
5.4 Process overview for file exchange by SFTP

Overview to drop a file on the system by file exchange using SFTP.



5.5 Process overview for web services

Overview to get a certificate to be able to use the eHealth WS and then to use the WS with or without file.



5.6 Publication Input Data (ehealth-cobrha-schema-core.xsd)

See cookbook xsd cobrha

5.7 Request Report (ehealth-cobrha-schema-report.xsd)

See cookbook xsd cobrha

5.8 Synchronous Publish Operation Web Service

This WS allows authorized data furnishers to feed in real time the common registry by providing in input, a correct XML element corresponding to either one Individual or one HCAssociation or one HCInstitution or one Codification or one CBEcompany with all the data relative to it. In output, the WS will return a report containing eventual publication warnings or errors.

6 Risks and security

6.1 Security

6.1.1 Business security

In case the development adds an additional use case based on an existing integration, the eHealth platform 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 WS, the partner may obtain support from the contact center (Section 3).

In case the eHealth platform 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 the eHealth platform delivered, he is obliged to contact and inform the eHealth platform immediately and he is not allowed to publish this bug or vulnerability in any case.

6.1.2 File Exchange

Security is provided by the secure file transfer protocol when dropping files on the SFTP server.

6.1.3 Web service

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

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

6.1.4 MTOM Policy

For binary content sending, the “Message Transmission Optimization Mechanism” (MTOM/XOP) should be used.

See <http://www.w3.org/TR/soap12-mtom/> for the technical specification.

6.1.5 The use of username, password and token

The username, password and token are strictly personal and are not allowed to transfer.

Every user takes care of his username, password and token and is forced to confidentiality of it. Every user is also responsible of every use, which includes the use by a third party, until the inactivation.



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 Procedure for file exchange

7.1.2 Initiation

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

7.1.3 Development and test procedure

You should develop a tool that creates correct publication files for full extract and delta of your healthcare provider data.

In some cases, eHealth provides you with test cases in order for you to test your client before releasing it in the acceptance environment.

7.1.4 Release procedure

When development tests are successful, you can request to access the acceptance environment of the eHealth platform by sending your publication files on our acceptance SFTP server by the acceptance portal or by WS in our acceptance environment.

From this moment, you start integration and acceptance tests. The eHealth platform suggests testing during minimum one month.

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

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

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

7.1.5 Operational follow-up

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

7.1.6 Test cases

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

- Submit a full extract publication request;
- Submit publication request with only updates;
- Submit a large amount of publication requests;

In addition, the organization should also run negative test cases:

- Submit a publication zip file with the metadata file not supplied or with wrong sequence numbering;
- Submit a publication zip file with more than 50 erroneous operations;
- Submit a publication zip file containing a not authorized operation;
- Submit a publication zip file containing an operation with missing fields;
- Submit a publication zip file containing an operation with parent element missing.



7.1.7 Procedure for web services

7.1.7.1 Initiation

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

7.1.7.2 Development and test procedure

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

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

7.1.7.3 Release procedure

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

From this moment on, you start integration and acceptance tests. The eHealth platform suggests testing during minimum one month.

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

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

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

7.1.7.4 Request a certificate

Prior to requesting the certificate, you need the latest version of *Java* and the *Belgium eID middleware*. You also need a smart-card reader and a Belgian eID. You can request the test certificate at one of the following URL:

- Dutch version: http://wwwacc.ehealth.fgov.be/JWS/ETEE/etee-requestor_nl.jnlp
- French version: http://wwwacc.ehealth.fgov.be/JWS/ETEE/etee-requestor_fr.jnlp

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.

7.1.7.5 Operational follow-up

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

7.1.7.6 Test cases

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

- Submit a valid operation that creates a new element;
- Submit a very large operation;
- Submit an operation that makes a delta (update an existing element in the CoBRHA).

In addition, the organization should also run negative test cases:

- Submit an operation with missing fields;
- Submit an operation where parent elements are missing;



- Submit a not authorized operation.



8 Status codes and error messages

8.1 CoBRHA Business Status Codes

Status codes originating from the eHealth platform for the request, operation and action (part of operation): These status codes are linked to the publication request, operation or a single action of the operation. Status codes with severity 'notice' do not block anything. Status codes with severity 'warning' linked to an action blocks the execution of the action, but not the operation or request. A status code with severity 'error' linked to an operation blocks the execution of the operation, but not the execution of the request. However, after 50 operations failed, the execution of the request is blocked. A status code with severity 'error' linked to the request blocks the execution of the request. A status code with severity 'critical' blocks everything immediately. The input xml of the operation is present in the report for all status codes with severity 'error' and 'critical' linked to the operation. The xml linked to the action is present in the report for all status codes linked to the action with severity 'warning'. We can decide to filter out the warning statuses from the report for performance reasons.

<i>Status code</i>	<i>Severity</i>	<i>Level</i>	<i>Description</i>	<i>Solution</i>
CRE	notice	request	"created": A new request has entered the system, but is not ready to be treated.	NA
PEN	notice	request	"pending": The request is ready to be treated.	NA
EXE	notice	request	"executing": The request is in treatment.	NA
FIN	notice	request	"finished": The request is completely executed.	NA
FIW	warning	request	"finished with warning/s": The request is completely executed but warnings have occurred during the execution.	Check the issue list of the report for more info about the warnings.
FIE	error	request	"finished with error/s": The request is completely executed but errors have occurred during the execution. It's possible that also warnings were raised during the execution.	Check the issue list of the report for more info about the errors.
FAT	error	request	"failed technically": A technical problem has occurred and stopped the execution of the request. The request will be retreated when the technical problem is solved.	The eHealth platform has encountered a technical issue and will try to solve it as soon as possible.
TPF	error	request	"too many publication operations failed": More than 50 operations of the request have raised an error.	Check the issue list of the report for more info about the errors.
RBF	error	request	"report build failed": Building the publication report has failed.	The eHealth platform has encountered a technical issue and will try to solve it as soon as possible.

TMF	error	request	“too many publication operation failure”: More than 1 operation has been done in the synchronous web service call	Reduce the amount of operations in the ‘publish operation of the web service call to one.’
MIN	error	request	“missing input”: Not enough input data available to create new publication request.	Do not forget to put the metadata xml file in the zip file as explained in section 5.8.
XCU	error	request	“xml corrupted”: Corrupted xml. The xml in the request doesn’t follow the xsd schema explained above.	Restructure your publication request xml according to the ehealth-cobrha-schema-core.xsd.
ZCU	error	request	“zip file corrupted”: Uncompressing the zip file failed.	The zip file has not been compressed in one of the following formats: rar – 7z – gzip – zip
FNI	error	request	‘file package integrity failed’: The files contained in the zip file don’t correspond to the files described in the meta data file of the zip file.	Check if all publication files referenced by filename in the meta data file are present in the zip file and that all publication files present in the zip file are referenced by filename in the meta data file.
ZNA	error	request	“zip file not available”: The zip-file is not available on our file system.	Try to publish the zip file again. If it does not work at the second attempt, contact the eHealth team (info@ehealth.fgov.be).
OPE	notice	operation	“operation executed”: The operation has successfully been executed.	NA
ONE	warning	operation	“operation not executed”: All values inside the operation were already published by previous operations.	Filter this duplicate operation from future publication requests.
ONP	warning	operation	“operation not permitted”: All actions inside the operation were not permitted.	Filter the operation from future publication request. If you should be able to publish this data, contact the team of the eHealth platform to request access.
ONW	warning	operation	“operation not executed with warnings”: Some of the actions inside the operations were duplicated and/or some of the actions were not permitted. It results in non-execution of the operation.	Check the warning list linked to the operation in the report

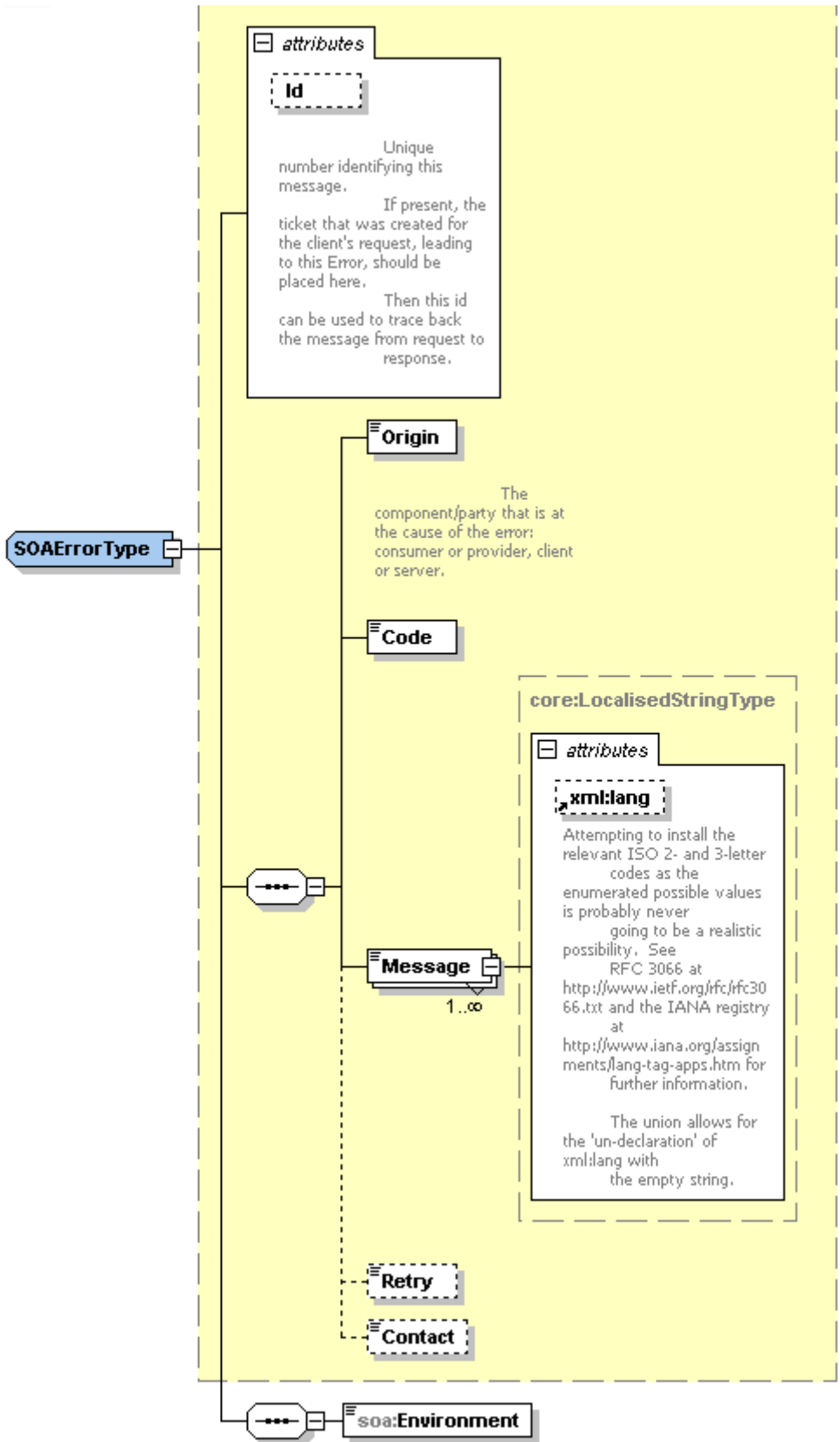
OEW	warning	operation	“operation executed with warnings”: The operation has been executed but some warnings occurred linked to the action inside the operation.	Check the warning list of the operation and read the solutions linked to the warning statuses
ONI	error	operation	“A BU rule check for data integrity of an operation failed”. Query does not follow the business rules described in this cookbook. For example, the NISS should be formatted right. Activity type should be of code activity. The IndividualID should be of type NISS or NISSbis. These are all checks that are not standard xsd checks, but we have business rules to enforce them.	Analyse the xml of the operation with the ehealth-cobrha-schema-core.xsd explanation inside this cookbook. If you do not find the error, contact the team of the eHealth platform (info@ehealth.fgov.be).
ONT	error	operation	“operation not translated”: An action of the operation could not be translated for querying.	
ONM	error	operation	“operation not executed because missing fields”: Query extracted from an action of the operation gave a missing field error when inserting in DB.	A mandatory field of the operation was not present in the xml. If you do not have the required data in your DB, contact eHealth platform team (info@ehealth.fgov.be).
OND	error	operation	“operation not executed because of wrong data type”: Query extracted from an action of the operation gave a wrong data type error.	A value inside the operation is in the wrong type. For more details read the explanation of the ehealth-cobrha-schema-core.xsd inside this cookbook (section 5.6.1).
ONF	error	operation	“operation not executed because of foreign key error”: Query extracted from an action of the operation gave a FK constraint error.	Analyse the xml of the operation with the ehealth-cobrha-schema-core.xsd explanation inside this cookbook (section 5.6.1). If you do not find the error, contact eHealth platform team (info@ehealth.fgov.be).
ANP	error	action	“action not permitted”: The furnisher is not authorized to publish this action.	Filter this action from future publication request. If you should be able to publish this data, contact the eHealth platform team to request access (info@ehealth.fgov.be).

ANE	error	action	“action not executed”: An action of the operation is not executed because already present in DB.	Filter this action from future publication requests.
AIG	notice	action	“action ignored”: An action of the operation is ignored because only used for referencing by his business key to a known record.	NA
AEX	notice	action	“action executed”: The action has successfully been executed	NA

8.2 Soap Fault Error Codes

They contain the following attributes:

Field name	Descriptions
Id	Unique number identifying this message. If present, the ticket that was created for the client's request, leading to this error, should be placed here. Then this id can be used to trace back the message from request to
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.
Environment	The environment in which the error occurs: development, test, integration, simulation, acceptance or production.



8.2.1 Schema Validation Errors

When invoking the WS, a valid XML must be provided.

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:ehealth:errors:soa:v1" Id="5bbd8a2a-bb21-4cf8-99bc-8d52c18e2801">
<Origin>Consumer</Origin>
<Code>SOA-03006</Code>
<Message xml:lang="en">XSD compliance failure.</Message>
<soa:Environment>Production</soa:Environment>
</soa:SystemError>
</detail>
</soapenv:Fault>
</soapenv:Body>
</soapenv:Envelope>
```

8.2.2 Technical Errors

Technical errors are errors inherent to the internal working of the eHealth WS. It 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 service desk.

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 Example

See example.zip

