Test protocol for Conversion service

The test protocol relates to the following standard:

|  |  |  |  |
| --- | --- | --- | --- |
| Standard’s name ENG | Standard’s name DK | **Version** | **Type** |
| Standard: Conversion service | Konverteringsløsning | 1.0.0 | HL7 FHIR / OIOXML |

|  |  |  |  |
| --- | --- | --- | --- |
| **Versioning** | | | |
| **Version** | **Initials** | **Date** | **Description** |
| 1.0 | TMS/KML | November 2024 | First release |
| 1.1 | KML/SKS/RCH | January 2025 | A note is added, stating that the files must be saved instead of sent at the end of test steps. |
| 1.2 | RCH/SKS | January 2025 | The test steps related to converting from Edifact to FHIR have been removed, as they are out of scope. Test files have been provided for the applicable test steps, and a clear distinction has been established between the test steps and the expected results. |
| 1.3 | SKS/KML | September 2025 | Test protocol is updated and simplified based on the changes in the use case document. |

Tabel of contents

[1 Introduction 4](#_Toc210115141)

[1.1 Purpose 4](#_Toc210115142)

[1.2 Prerequisites for live test 4](#_Toc210115143)

[1.3 Documentation of self-test 5](#_Toc210115144)

[1.4 Background materials 6](#_Toc210115145)

[1.5 Test examples and test persons 7](#_Toc210115146)

[1.6 Test tool 7](#_Toc210115147)

[1.7 Test Result 8](#_Toc210115148)

[2 Vendor, system under test (SUT) and test result information 9](#_Toc210115149)

[2.1 Information about the vendor 9](#_Toc210115150)

[2.2 Information about system under test (SUT) 9](#_Toc210115151)

[2.3 Information about the test result 9](#_Toc210115152)

[3 The test 10](#_Toc210115153)

[3.1 Documentation of the test 11](#_Toc210115154)

[3.2 Test of requirements for content and flow/workflows 12](#_Toc210115155)

[3.3 Test of general technical requirements 56](#_Toc210115156)

# Introduction

This is a test protocol for Conversion service.

All documentation concerning the conversion service will be the subject of testing, and the test protocol will be continuously updated to reflect the requirements in the best way possible.

The conversion service converts to/from FHIR CareCommunication v. 3.0 and XDIS91 as well as FHIR Acknowledgement and XCTL.

## Purpose

The test protocol forms the basis for the tests, which must ensure that system under test (SUT) complies with the established rules and requirements for the standard. The test protocol also forms the basis for the self-test that vendor carries out prior to a live test.

## Prerequisites for live test

The following prerequisites must be met prior to the live test:

1. The vendor has read the standard documentation, cf. the [background material](#_Baggrundsmaterialer_1).
2. The vendor has performed [self-test](#_Dokumentation_af_egentest), approved by MedCom.
3. The vendor has created [relevant test persons](#_Testeksempler_og_testpersoner) in SUT.
4. The vendor uses the same version of SUT during self-test and live test.

## Documentation of self-test

**Self-test**

The self-test is documented by the vendor completing this test protocol.

For self-tests, only the following column must be completed by the vendor:

* [Current result]: is filled in with the results of the self-test and relevant descriptions.

Other columns are reserved for MedCom.

**During the self-test the vendor must document the test results by saving relevant files and screen dumps, and subsequently send these in a combined ZIP file (together with the completed test protocol) to** [**fhir@medcom.dk**](mailto:fhir@medcom.dk)**.**

* Standard name
* The number of the relevant test step
* Consecutive letter
* File type

Eksempel: ConversionService\_3.4\_A.xml or ConversionService\_2.2\_B.png

## Background materials

| **Name** | **Version** | **Link/reference** | **Description** |
| --- | --- | --- | --- |
| Documentation site |  | <https://medcomdk.github.io/dk-medcom-carecommunication/#3-conversion-service> | The overall site for the conversion service |
| Use cases |  | <https://medcomdk.github.io/dk-medcom-carecommunication/#31-use-cases-and-rules> | Use cases and rules describing when to convert a message and which rules must be followed during the conversion. |
| Mapping of messages |  | <https://medcomdk.github.io/dk-medcom-carecommunication/#32-mapping-of-messages> | A mapping table of the messages |
| Governance for MedCom FHIR |  | <https://medcomdk.github.io/MedCom-FHIR-Communication/> | Governance for MedCom’s FHIR standards, which describes general rules for all MedCom standards and specific rules for this standard, as well as for sending the message |
| CareCommunication | 5.0 | <https://medcomdk.github.io/dk-medcom-carecommunication> | Documentation site with references to all relevant documentation, including:   * Clinical guidelines for application (Sundhedsfaglige retningslinjer for anvendelse) * Use cases * Technical specifications |
| XDIS91 |  | <https://svn.medcom.dk/svn/releases/Standarder/Den%20gode%20korrespondance/XML/> | Documentation site regarding XDIS91 |
| Acknowledgement | 2.0 | <https://medcomdk.github.io/dk-medcom-acknowledgement> | Documentation site with references to all relevant documentation, including:   * Use cases * Technical specifications |
| XCTL |  | <https://svn.medcom.dk/svn/releases/Standarder/Den%20gode%20CONTRL/XML/> | Documentation site regarding XCTL |
| VANSEnvelope | 1.0 | <https://svn.medcom.dk/svn/releases/Standarder/Den%20gode%20VANSEnvelope/Dokumentation/>  FHIR specific values in the VANSEnvelope: <https://medcomdk.github.io/MedCom-FHIR-Communication/assets/documents/FHIRMessages_NetworkEnvelopes_EN.html> | The overall site for the VANSEnvelope including relevant documentation |

## Test examples and test persons

|  |  |  |
| --- | --- | --- |
| **Name** | **Link/reference** | **Description** |
| Test examples /FHIR example files | <https://medcomdk.github.io/dk-medcom-carecommunication/#3-conversion-service> | Includes test examples used during the test and certification and an overview of all these. |
| Overview of the test persons | <https://www.medcom.dk/opslag/koder-tabeller-ydere/tabeller/nationale-test-cpr-numre> | Overview of national test personal identification number (DA: CPR-nummer), that can be used during test.  Note: During test and certification, the vendor must be able to use any of the test persons on the list. |

## Test tool

|  |  |  |
| --- | --- | --- |
| **Navne** | **Link/reference** | **Description** |
| FHIR-server with MedCom profiles | <https://fhir.medcom.dk/> | Public server that validates against MedCom's FHIR profiles. It is permitted to use the server for testing the upload/download of FHIR resources. |

## Test Result

The result for each test step is categorized based on the table below:

| **Marking** | **F1** | **F2** | **F3** | **F4** | **Ok** | **Not relevant** |
| --- | --- | --- | --- | --- | --- | --- |
| **Evaluation** | **Critical** | **Serious** | **Significant** | **Less significant** | **Approved** | **Not an error** |

To get the test and certification approved, the test protocol must consist exclusively of [F4] as well as [OK] results. All [F1], [F2] and [F3] must, therefore, be fixed prior to final approval.

For further information, please read [MedCom’s test og certification](#_Baggrundsmaterialer_2).

# Vendor, system under test (SUT) and test result information

## Information about the vendor

This table must be completed by **the vendor** prior to the test.

|  |  |
| --- | --- |
| Company | Completed by vendor |
| Address | Completed by vendor |
| Contact person | Completed by vendor |
| Telephone | Completed by vendor |
| E-mail | Completed by vendor |

## Information about system under test (SUT)

This table must be completed by **the vendor** prior to the test.

|  |  |
| --- | --- |
| System | Completed by vendor |
| Version | Completed by vendor |
| Description | Completed by vendor |
| Test type | Self-test  Final test/certification |

## Information about the test result

Note: This table must be completed by MedCom when the test has been completed.

|  |  |
| --- | --- |
| Test date | 2023-01-01 |
| Test location | Completed by MedCom |
| Approved | Yes  No |
| Remarks | Completed by MedCom |
| Carried out by | The name of the MedCom responsible (initials) for this test. |

# The test

This section describes the requirements which SUT must meet before final approval.

The test is divided into two sections:

1. Test of requirements for content and flow/workflows, including received receipts
2. Test of technical requirements

Test participants will be asked to complete tests as described in the tables.

**OBS:**

* **The test EAN numbers (da: testlokationsnumre) used in the test as the sender and receiver of a CareCommunication must be manually created in the VANS system if an online lookup in SOR is not performed and a local copy is used instead. This is because local copies do not contain test location numbers.**
* **It is important that no files are sent to actual receivers in production during the test.**
* **In FHIR, systems need to be robust and capable of handling incoming bundles that may vary in structure while still being valid. This means that resources within a bundle can appear in different orders or formats, and the receiving system should be able to process them correctly without assuming a fixed sequence or structure.**
* **It is allowed to send CareCommunications in both XML and JSON format.**

## Documentation of the test

**Documentation of the test**

As valid documentation, the test participant or test manager must document completion by continuous screen dumps (.png/.jpeg) and/or files/log files (.xml/.json). **Before the test, it is agreed on who is responsible for this.**

The following applies:

* The files must be viewable in a standard tool and must not require further processing by MedCom
* All files and screen dumps must be named with:
  + Standard name
  + The number of the relevant test step
  + Consecutive letter
  + File type

Example: ConversionService\_3.4\_A.xml or ConversionService \_2.2\_B.png

If the vendor has documented the test themselves, the files must be sent in a ZIP file to [fhir@medcom.dk](mailto:fhir@medcom.dk).

## Test of requirements for content and flow/workflows

The purpose of these tests is to ensure that the standard is implemented with a satisfactory quality, i.e. that implementation meets the business requirements for flow and content as described in the [use case-material](#_Baggrundsmaterialer_2). These test steps are mainly for testing the system of the conversion service.

The table below reflects the use cases that are tested in relation to content and flow/workflows. The table also shows which [example fil](#_Testeksempler_og_testpersoner)es must be loaded in connection with each test step.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| [**Use case**](#_Baggrundsmaterialer_2) | **Description [ENG]** | **Description [DK]** | **Section** | **Example file** |
| R1 | Receive a CareCommunication or XDIS91 and check receiver capabilities (no conversion) | Modtag en CareCommunication eller XDIS91 og kontroller modtager (ingen konvertering) | 3.2.1 | ConSer\_CC\_01  ConSer\_XDIS91\_01 |
| S1 | Send a CareCommunication | Afsend en CareCommunication | 3.2.2 | ConSer\_XDIS91\_02 |
| S1.A1 + R1.A2 | Send a CareCommunication with attachments | Afsend en CareCommunication med bilag | 3.2.3 | Send a CareCommunication with attachment:  ConSer\_XDIS91\_03\_Xdis91  ConSer\_XDIS91\_03\_XBIN  Send a CareCommunication that includes multiple attachments:  ConSer\_XDIS91\_04\_XDIS91  ConSer\_XDIS91\_04\_XBIN01  Send a CareCommunication that includes more than ten attachments:  ConSer\_XDIS91\_09\_XDIS91  ConSer\_XDIS91\_09\_XBIN  CareCommunication exceeds 100 MB after conversion:  ConSer\_XDIS91\_05\_XDIS91  ConSer\_XDIS91\_05\_XBN01 |
| S1.A2 | XBIN01 is not received within 60 min | XBIN01 modtages ikke indenfor 60 min | 3.2.43.2.4 | ConSer\_XDIS91\_06 |
| S1.A3 | XDIS91 is not received within 60 min. | XDIS91 modtages ikke indenfor 60 min | 3.2.5 | ConSer\_XBIN\_07\_Standalone |
| S1.A4 | XBIN01 contains non-permitted filetypes | XBIN01 indeholder ikke-tilladt filtype | 3.2.6 | ConSer\_XDIS91\_08\_XDIS91  ConSer\_XDIS91\_08\_XBIN |
| S2 | Send an XDIS91 | Afsend en XDIS91 | 3.2.7 | ConSer\_CC\_02 |
| S2.A1 + R1.A1 | Send an XDIS91 and XBIN01 | Afsend en XDIS91 og XBIN01 | 3.2.8 | Send an XDIS91 and XBIN01  ConSer\_CC\_03\_attachment  Send an XDIS91 and XBIN01 with multiple files  ConSer\_CC\_04\_MultipleAttachments  Send an XDIS91 and XBIN01  ConSer\_CC\_05\_TooManyAttachments  Receiver cannot receive XBIN01  ConSer\_CC\_06\_attachment |
| S2.A2 | Send a reply or forwarding | Afsend en besvarelse eller videresendelse | 3.2.9 | ConSer\_CC\_07\_reply ConSer\_CC\_08\_reply\_attachment  ConSer\_CC\_09\_reply  ConSer\_CC\_10\_forward  ConSer\_CC\_11\_forward\_attachment |
| S3 | Send an Acknowledgement | Afsend en Acknowledgement | 3.2.10 | ConSer\_XCTL03\_01  ConSer\_XCTL02\_01 |
| S4 | Send an XCTL | Afsend en XCTL | 3.2.11 | ConSer\_ACK\_01  ConSer\_ACK\_02 |

*Tabel 1: Overview of use cases being tested.*

### R1: Receive a CareCommunication or XDIS91 and check receiver capabilities (no conversion)

| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
| **Receive a CareCommunication (receiver can receive a CareCommunication)** | | | | | |
|  | Receive a communication message of the type CareCommunication. | ConSer\_CC\_01 | A message is received. |  | Choose |
|  | Demonstrate that the SUT checks the VANSEnvelope to identify the message type. |  | Message type is a FHIR CareCommunication. |  | Choose |
|  | Demonstrate that the SUT checks the capabilities of the receiver of the message in SOR or a local copy of SOR.  Receiver can receive a CareCommunication message. |  | Receiver can receive a CareCommunication after look-up in SOR or a local copy. |  | Choose |
| **Receive a XDIS91 (receiver can receive an XDIS91)** | | | | | |
|  | Receive a communication message of the type XDIS91. | ConSer\_XDIS91\_01 | A message is received. |  | Choose |
|  | Check the VANSEnvelope to identify the message type. |  | Message type is a XDIS91. |  | Choose |
|  | Check the capabilities of the receiver of the message in SOR or a local copy of SOR.  Receiver can receive a XDIS91 message. |  | Receiver can receive a CareCommunication after look-up in SOR or a local copy. |  | Choose |

### S1: Send a CareCommunication (from XML to FHIR)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
| Choose | Receive a communication message of the type XDIS91.  Demonstrate that SUT checks the VANSEnvelope to identify the message type. | ConSer\_XDIS91\_02 | A message is received.  Message type is a XDIS91. |  | Choose |
|  | Demonstrate that SUT checks the capabilities of the receiver of the message in SOR or a local copy of SOR.  Receiver can receive a CareCommunication message. |  | Receiver can receive a CareCommunication after look-up in SOR or a local copy. |  | Choose |
|  | Demonstrate that the SUT maps the XDIS91 to a CareCommunication, cf. [mapping table](#_Baggrundsmaterialer_2). |  | Mapping is performed correctly as described in the mapping table. |  | Choose |
|  | Demonstrate that the EpisodeOfCare-identifier is transferred from XDIS91 to CareCommunication. |  | The episodeOfCare-identifier is transferred correctly to the CareCommunication. |  | Choose |
|  | Demonstrate that the statuses are transferred, cf. section 4.2 in use case document, see Background materials, from the XDIS91 to the CareCommunication. |  | Statuses are transferred correctly to the CareCommunication. |  | Choose |
|  | Demonstrate that the SUT saves relevant information from XDIS91:   * Emessage.Envelope.identifier * Emessage.Letter.identifier * Emessage.ClinicalEmail.Letter.VersionCode |  | Relevant information is saved. |  | Choose |
|  | Demonstrate that the SUT includes correct SOR-id on sender and receiver after look-up in SOR or a local copy. |  | Correct SOR-id is included after look-up in SOR or a local copy. |  | Choose |
|  | Demonstrate that the SUT includes telephone number of the sender in the author information after look-up in SOR or a local copy.  ***Note****: If the telephone number cannot be identified the value ‘0000 0000’ is inserted. This must not be common practise*. |  | Correct telephone number is included after look-up in SOR or a local copy. |  | Choose |
|  | Demonstrate that the SUT automatically transfers and inserts:   * Time of sending the message from XDIS91 * A unique communication identifier to the current message thread * Technical references between Provenance and message segments for the current message |  | The following is added:   * Time of sending is transferred to the CareCommunication. * A communication identifier is inserted in the Communication.identifier element as a UUID v4. * Correct technical references to the message segments (Communication.payload.extension:valueIdentifier) are inserted in the Provenance (Provenance.entity.role). |  | Choose |
|  | Demonstrate that the CareCommunication message is validated against the implementation guide. |  | Validation is performed and went well. |  | Choose |
|  | Demonstrate that the SUT saves relevant information from the CareCommunication:   * Bundle.id * MessageHeader.id |  | Relevant information is saved. |  | Choose |
|  | The CareCommunication is wrapped in a VANSEnvelope with correct content. |  | CareCommunication is wrapped in the VANSEnvelope. |  | Choose |
|  | Show that the message is ready to be sent to the correct receiver. |  | The message is mapped correctly to the correct receiver and is ready to be sent.  Save the file for documentation. |  | Choose |

### S1.A1: Send a CareCommunication with attachments

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
|  | Receive two communication messages.  Demonstrate that the SUT checks the VANSEnvelope to identify the message types. | ConSer\_XDIS91\_03\_XDIS91  ConSer\_XDIS91\_03\_XBIN | Two messages are received.  Message type is a XDIS91 and XBIN01. |  | Choose |
|  | Demonstrate that the SUT checks the capabilities of the receiver of the message in SOR or a local copy of SOR. |  | Receiver can receive a CareCommunication after look-up in SOR or a local copy. |  | Choose |
|  | The SUT has identified a need for conversion of the XDIS91 and the XBIN01. |  | The XDIS91 and XBIN01 must be converted to a CareCommunication before the message is ready to be sent to receiver. |  | Choose |
|  | Map the XDIS91 and XBIN01 to a CareCommunication, cf. [mapping table](#_Baggrundsmaterialer_2). |  | Mapping is performed as described in the mapping table. |  | Choose |
|  | Saves relevant information from XDIS91 and XBIN01:  XDIS91:   * Emessage.Envelope.identifier * Emessage.Letter.identifier * Emessage.ClinicalEmail.Letter.VersionCode   XBIN01:   * Emessage.Envelope.identifier * Emessage.BinaryLetter.Letter.identifier * Emessage.BinaryLetter.Letter.VersionCode |  | Relevant information is saved. |  | Choose |
|  | Demonstrate that the SUT includes correct SOR-id on sender and receiver after look-up in SOR or a local copy. |  | Correct SOR-id is included after look-up in SOR or a local copy. |  | Choose |
|  | Demonstrate that the SUT includes telephone number of the sender in the author information after look-up in SOR or a local copy.  ***Note****: If the telephone number cannot be identified the value ‘0000 0000’ is inserted. This must not be common practise*. |  | Correct telephone number is included after look-up in SOR or a local copy. |  | Choose |
|  | Demonstrate that the SUT automatically transfers and inserts:   * Time of sending the message and attached files from XDIS91. * A unique communication identifier to the current message thread. * Ensures the title of the attachment is transferred correctly. * Technical references between Provenance and message segments for the current message. |  | The following is added:   * Time of sending is transferred to the CareCommunication and the attached files. * A communication identifier is inserted in the Communication.identifier element as a UUID v4. * The title of the attachment is transferred. * Correct technical references to the message segments (Communication.payload.extension:valueIdentifier) are inserted in the Provenance (Provenance.entity.role). |  | Choose |
|  | CareCommunication is validated against the implementation guide. |  | Validation is performed and went well. |  | Choose |
|  | Saves relevant information from the CareCommunication:   * Bundle.id * MessageHeader.id |  | Relevant information is saved. |  | Choose |
|  | The CareCommunication is wrapped in a [VANSEnvelope](#_Baggrundsmaterialer_2) with correct content. |  | CareCommunication is wrapped in the VANSEnvelope. |  | Choose |
|  | Show that the CareCommunication including attachment is ready to be sent to the correct receiver. |  | The message including attachment is mapped correctly and is ready to be sent to the correct receiver.  Save the file for documentation. |  | Choose |
|  | **Send a CareCommunication that includes multiple attachments**  Demonstrate that SUT converts XDIS91 and the associated XBIN01 (including 3 attachments) to a CareCommunication, cf. [mapping table](#_Baggrundsmaterialer_2). | ConSer\_XDIS91\_04\_XDIS91  ConSer\_XDIS91\_04\_XBIN01 | Mapping is performed as described in the mapping table.  CareCommunication includes several attachments. |  | Choose |
|  | Demonstrate that SUT can perform test step 3.2.3.5-3.2.3.12 when multiple attachments are included in CareCommunication. |  | CareCommunication including multiple attachments is correctly converted and is ready to be sent to correct receiver. |  | Choose |
|  | **Send a CareCommunication that includes more than ten attachments**  The XDIS91 and XBIN01 must be converted to CareCommunication.  Evaluation states that more than ten attachments are included.  Demonstrate that SUT maps XDIS91 and all the associated XBIN01 to a CareCommunication, cf. [mapping table](#_Baggrundsmaterialer_2).  ***Note****: As described in “*[*Precondition 5”*](#_Baggrundsmaterialer_2)*, cf. section 1*.*4.2.5 in use case document,* *if more than ten files are sent to an XDIS91 in one/more XBIN01, the conversion service must convert and send all files in a CareCommunication to the receiver system.* | ConSer\_XDIS91\_09\_XDIS91  ConSer\_XDIS91\_09\_XBIN |  |  | Choose |
|  | Demonstrate that the CareCommunication including all attachments is ready to be sent to the correct receiver. |  | The CareCommunication including attachments is mapped correctly and is ready to be sent to the correct receiver. |  | Choose |
|  | **CareCommunication exceeds 100 MB after conversion**  Demonstrate that SUT converts XDIS91 + XBIN01 to a CareCommunication that exceeds 100 MB in total.  The CareCommunication is ready to be sent to the correct receiver. | ConSer\_XDIS91\_05\_XDIS91  ConSer\_XDIS91\_05\_XBN01 | CareCommunication is converted and exceeds 100 MB.  CareCommunication is ready to be sent to correct receiver. |  | Choose |

### S1.A2: XBIN01 is not received within 60 min.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
|  | Describe and, if possible, show how SUT handles the following situation: Receive an XDIS91 with a reference to an XBIN01.  **The referenced XBIN01 does not arrive within 60 minutes**. | ConSer\_XDIS91\_06 | An XDIS91 is received but not the referenced XBIN01. |  | Choose |
|  | Demonstrate that SUT maps XDIS91 to a CareCommunication including a payload to the associated XBIN01.  At a minimum, SUT must not fill out the elements ‘data’ and ‘title’ in the payload. |  | Mapping of an incomplete CareCommunication is performed correctly. |  | Choose |
| Choose | Saves relevant information from XDIS91:  XDIS91:   * Emessage.Envelope.identifier * Emessage.Letter.identifier * Emessage.ClinicalEmail.Letter.VersionCode |  | Relevant information is saved. |  | Choose |
|  | Demonstrate that the SUT includes correct SOR-id on sender and receiver after look-up in SOR or a local copy. |  | Correct SOR-id is included after look-up in SOR or a local copy. |  | Choose |
|  | Demonstrate that the SUT includes telephone number of the sender in the author information after look-up in SOR or a local copy.  ***Note****: If the telephone number cannot be identified the value ‘0000 0000’ is inserted. This must not be common practise*. |  | Correct telephone number is included after look-up in SOR or a local copy. |  | Choose |
|  | CareCommunication is validated against the implementation guide. |  | Validation is performed, and VANS has ignored specific error codes related to the incomplete payload as described in “[Precondition 7”](#_Baggrundsmaterialer_2), cf. section 1.4.2.7 in use case document. |  | Choose |
|  | Saves relevant information from the CareCommunication:   * Bundle.id * MessageHeader.id |  | Relevant information is saved. |  | Choose |
|  | The CareCommunication is wrapped in a [VANSEnvelope](#_Baggrundsmaterialer_2) with correct content. |  | Communication is wrapped in the VANSEnvelope. |  | Choose |
|  | Show that the CareCommunication including payload to the attachment is ready to be sent to the correct receiver. |  | The message including attachment is mapped correctly and is ready to be sent to the correct receiver.  Save the file for documentation. |  | Choose |

### S1.A3: XDIS91 is not received within 60 min (XBIN01 is received)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
|  | Receive a communication message of the type XBIN01. | ConSer\_XBIN\_07\_Standalone | A XBIN01 is received. |  | Choose |
|  | **The XDIS91 with a reference to the XBIN01 does not arrive within 60 minutes.**  The XBIN01 is wrapped in a VANSEnvelope. |  | The XBIN01 has not been converted and is correctly wrapped. |  | Choose |
|  | Forward the XBIN01 to the correct receiver. |  | XBIN01 is ready to be forwarded to correct receiver. |  | Choose |

### S1.A4: XBIN01 includes non-permitted filetypes

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
|  | Perform test step 3.2.3.1-3.2.3.3. | ConSer\_XDIS91\_08\_XDIS91  ConSer\_XDIS91\_08\_XBIN | The XDIS91 and XBIN01 is received and must be converted to CareCommunication. |  | Choose |
| Choose | **The XBIN01 does include non-permitted file type**.  *Permitted fil types can be found* [*here*](https://medcomfhir.dk/ig/terminology/ValueSet-medcom-core-attachmentMimeTypes.html). |  | Evaluation states that the file type in the XBIN01 is not allowed. |  | Choose |
|  | Demonstrate that SUT maps the XDIS91 and XBIN01 containing a non-permitted file type to a CareCommunication, cf. [mapping table](#_Baggrundsmaterialer_2). |  | Mapping is performed correctly, cf. “[Precondition 6”](#_Baggrundsmaterialer_2), section 1.4.2.6 in use case document. |  | Choose |
|  | Saves relevant information from XDIS91 and XBIN01:  XDIS91:   * Emessage.Envelope.identifier * Emessage.Letter.identifier * Emessage.ClinicalEmail.Letter.VersionCode   XBIN01:   * Emessage.Envelope.identifier * Emessage.BinaryLetter.Letter.identifier * Emessage.BinaryLetter.Letter.VersionCode |  | Relevant information is saved. |  | Choose |
|  | Demonstrate that the SUT includes correct SOR-id on sender and receiver after look-up in SOR or a local copy. |  | Correct SOR-id is included after look-up in SOR or a local copy. |  | Choose |
|  | Demonstrate that the SUT includes telephone number of the sender in the author information after look-up in SOR or a local copy.  ***Note****: If the telephone number cannot be identified the value ‘0000 0000’ is inserted. This must not be common practise*. |  | Correct telephone number is included after look-up in SOR or a local copy. |  | Choose |
|  | Demonstrate that the SUT automatically transfers/inserts:   * Time of sending the message and attached files from XDIS91. * A unique communication identifier to the current message thread. * Ensures the title of the attachment is transferred correctly. * Technical references between Provenance and message segments for the current message. |  | The following is added:   * Time of sending is transferred to the CareCommunication and the attached files. * A communication identifier is inserted in the Communication.identifier element as a UUID v4. * The title of the attachment is transferred. * Correct technical references to the message segments (Communication.payload.extension:valueIdentifier) are inserted in the Provenance (Provenance.entity.role). |  | Choose |
|  | CareCommunication is validated against the implementation guide. |  | Validation is performed, and VANS has ignored specific error codes related to the incomplete payload, cf. “[Precondition 7”](#_Baggrundsmaterialer_2), section 1.4.2.7 in use case document. |  | Choose |
|  | Saves relevant information from the CareCommunication:   * Bundle.id * MessageHeader.id |  | Relevant information is saved. |  | Choose |
|  | The CareCommunication is wrapped in a [VANSEnvelope](#_Baggrundsmaterialer_2) with correct content. |  | CareCommunication is wrapped in the VANSEnvelope. |  | Choose |
|  | Show that the CareCommunication including attachment is ready to be sent to the correct receiver. |  | The message including attachment is mapped correctly and is ready to be sent to the correct receiver.  Save the file for documentation. |  | Choose |

### S2: Send an XDIS91 (from FHIR to XML)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
|  | Receive a communication message of the type CareCommunication.  Demonstrate that the SUT checks the VANSEnvelope to identify the message type. | ConSer\_CC\_02 | A message is received.  Message type is a FHIR CareCommunication. |  | Choose |
|  | Demonstrate that the SUT checks the capabilities of the receiver of the message in SOR or a local copy of SOR.  Receiver can receive a XDIS91 message. |  | Receiver can receive a XDIS91 after look-up in SOR or a local copy. |  | Choose |
|  | Demonstrate that the SUT maps the CareCommunication to a XDIS91, cf. [mapping table](#_Baggrundsmaterialer_2). |  | Mapping is performed correctly as described in the mapping table. |  | Choose |
| Choose | Demonstrate that the EpisodeOfCare-identifier is transferred correctly from CareCommunication to XDIS91.  ***Note****: it is allowed to include more episodeOfCare-identifiers, but there is only room for one in the XDIS91. In case more episodeOfCare-identifiers are included, the locally defined one is expected to be included cf.* [*Governance for EpisodeofCare-identifier*](https://medcomdk.github.io/MedCom-FHIR-Communication/assets/documents/080_Governance-for-episode-of-care-identifiers.html) |  | The locally defined EpisodeOfCare-identifier is included the XDIS91. |  | Choose |
|  | Demonstrate that the Encounter.status and class are transferred, cf. section 4.2 in use case document, see Background materials, from the CareCommunication to the XDIS91. |  | Encounter.status and Encounter.class are transferred correctly to the XDIS91 |  | Choose |
|  | Demonstrate that the SUT saves relevant information from the CareCommunication:   * Bundle.id * MessageHeader.id |  | Relevant information is saved. |  | Choose |
|  | Demonstrate that the SUT saves relevant information from the XDIS91:  XDIS91:   * Emessage.Envelope.identifier * Emessage.Letter.identifier |  | Relevant information is saved. |  | Choose |
|  | The XDIS91 is wrapped in a VANSEnvelope with correct content. |  | XDIS91 is wrapped in the VANSEnvelope. |  | Choose |
|  | Show that the message is ready to be sent to the correct receiver. |  | The message is mapped correctly and is ready to be sent to the correct receiver.  Save the file for documentation. |  | Choose |

### S2.A1: Send an XDIS91 and XBIN01

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
|  | Receive a communication message including attachment.  Demonstrate that the SUT checks the VANSEnvelope to identify the message type. | ConSer\_CC\_03\_attachment | A message is received.  Message type is a FHIR CareCommunication. |  | Choose |
|  | Demonstrate that the SUT checks the capabilities of the receiver *of the message* in SOR or a local copy of SOR. |  | Receiver can receive a XDIS91 after look-up in SOR or a local copy. |  | Choose |
|  | Demonstrate that the SUT checks the capabilities of the receiver *of the attachment* in SOR or a local copy of SOR. |  | Receiver can receive a XBIN01 (or MedBin) after look-up in SOR or a local copy. |  | Choose |
|  | The SUT has identified a need for conversion of the CareCommunication with attachment. |  | The CareCommunication with attachment must be converted before the message is ready to be sent to receiver. |  | Choose |
|  | Demonstrate that the SUT saves relevant information from the CareCommunication:   * Bundle.id * MessageHeader.id |  | Relevant information is saved. |  | Choose |
|  | Demonstrate that the SUT maps the CareCommunication to a XDIS91 and a XBIN01, cf. [mapping table](#_Baggrundsmaterialer_2). |  | Mapping is performed correctly as described in the mapping table. |  | Choose |
|  | Demonstrate that the title of the attachment is transferred. |  | The title of the attachment is transferred to the XBIN01. |  | Choose |
|  | Demonstrate that the SUT includes correct SOR-id on sender and receiver after look-up in SOR or a local copy. |  | Correct SOR-id is included after look-up in SOR or a local copy. |  | Choose |
|  | Demonstrate that the SUT saves relevant information from the XDIS91 and XBIN01:  XDIS91:   * Emessage.Envelope.identifier * Emessage.Letter.identifier   XBIN01:   * Emessage.Envelope.identifier * Emessage.BinaryLetter.Letter.identifier |  | Relevant information is saved. |  | Choose |
|  | The XDIS91 and XBIN01 is wrapped in individual VANSEnvelope with correct content. |  | XDIS91 and XBIN01 is wrapped in individual VANSEnvelopes. |  | Choose |
|  | Show that the messages are ready to be sent to the correct receiver. |  | The messages are mapped correctly and are ready to be sent to the correct receiver.  Save the files for documentation. |  | Choose |
|  | **Send an XDIS91 and XBIN01 with multiple files**  Perform test step 3.2.8.1-3.2.8.5 with the new test data.  Demonstrate that the SUT maps the CareCommunication to one XDIS91 and one XBIN01 with multiple attachments, cf. [mapping table](#_Baggrundsmaterialer_2). | ConSer\_CC\_04\_MultipleAttachments | Mapping is performed correctly as described in the mapping table. |  | Choose |
|  | Demonstrate that the SUT includes correct SOR-id on sender and receiver after look-up in SOR or a local copy. |  | Correct SOR-id is included after look-up in SOR or a local copy. |  | Choose |
|  | Demonstrate that the titles of the attachments are transferred. |  | The title of the attachments is transferred to the XBIN01. |  | Choose |
|  | Demonstrate that the SUT saves relevant information from the XDIS91 and XBIN01:  XDIS91:   * Emessage.Envelope.identifier * Emessage.Letter.identifier   XBIN01:   * Emessage.Envelope.identifier * Emessage.BinaryLetter.Letter.identifier |  | Relevant information is saved. |  | Choose |
|  | The XDIS91 and XBIN01 is wrapped in individual VANSEnvelope with correct content. |  | XDIS91 and XBIN01 is wrapped in the VANSEnvelope. |  | Choose |
|  | Show that the messages are ready to be sent to the correct receiver. |  | The messages are mapped correctly and are ready to be sent to the correct receiver.  Save the files for documentation. |  | Choose |
|  | **Send an XDIS91 and XBIN01 that includes more than ten files**  The CareCommunication must be converted to XDIS91 and XBIN01.  Evaluation states that more than ten files are included.  Demonstrate that SUT maps CareCommunication to XDIS91 and XBIN01 with all the associated files, cf. [mapping table](#_Baggrundsmaterialer_2). | ConSer\_CC\_05\_TooManyAttachments |  |  | Choose |
|  | Demonstrate that the XDIS91 and XBIN01 including more than ten files are ready to be sent to the correct receiver. |  | The XDIS91 and XBIN01 are mapped correctly and is ready to be sent to the correct receiver. |  | Choose |
|  | **Receiver cannot receive XBIN01**  Receive a CareCommunication including attachment. | ConSer\_CC\_06\_attachment | A CareCommunication is received. |  | Choose |
|  | Demonstrate that the SUT checks the capabilities of the receiver of the message in SOR or a local copy of SOR. |  | Receiver can receive an XDIS91 after look-up in SOR or a local copy.  The receiver **cannot** receive a XBIN01 (or MedBin) message. |  | Choose |
|  | Demonstrate that SUT maps CareCommunication to XDIS91 and XBIN01, cf. [mapping table](#_Baggrundsmaterialer_2). |  | The XDIS91 and XBIN01 are mapped correctly and is ready to be sent to the correct receiver. |  | Choose |

### S2.A2: Send a reply or forward

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
| Receive a reply and send an XDIS91 | | | | | |
|  | Receive a communication message.  Demonstrate that the SUT checks the VANSEnvelope to identify the message type. | ConSer\_CC\_07\_reply | A message is received.  Message type is a FHIR CareCommunication.  The CareCommunication is a reply. |  | Choose |
|  | Demonstrate that SUT checks the capabilities of the receiver of the message in SOR or a local copy of SOR. |  | Receiver can receive a XDIS91 after look-up in SOR or a local copy. |  | Choose |
|  | Demonstrate that the SUT saves relevant information from the CareCommunication:   * Bundle.id * MessageHeader.id |  | Relevant information is saved. |  | Choose |
|  | Demonstrate that the SUT maps the CareCommunication to an XDIS91, cf. [mapping table](#_Baggrundsmaterialer_2). |  | Mapping is performed correctly as described in the mapping table. |  | Choose |
|  | Demonstrate that the SUT includes a note in the text informing that the message is a reply. |  | A XDIS91 is created with information about being a reply. |  | Choose |
|  | Demonstrate that the SUT saves relevant information from the XDIS91:   * Emessage.Envelope.identifier * Emessage.Letter.identifier |  | Relevant information is saved. |  | Choose |
|  | The XDIS91 is wrapped in individual VANSEnvelope with correct content. |  | XDIS91 is wrapped in the VANSEnvelope. |  | Choose |
|  | Show that the message is ready to be sent to the correct receiver. |  | The message is mapped correctly and is ready to be sent to the correct receiver with information about being a reply.  Save the file for documentation. |  | Choose |
|  | **Receive a reply with attachment and send an XDIS91**  Receive a communication message of the type CareCommunication with an attachment.  Demonstrate that the SUT registers that the CareCommunication is a reply and includes an attachment. | ConSer\_CC\_08\_reply\_attachment | The CareCommunication is a reply and includes an attachment.  The CareCommunication must be converted to XDIS91 and XBIN01 |  | Choose |
|  | Perform test step 3.2.9.33.2.9.3-3.2.9.83.2.9.8, but map the CareCommunication to an XDIS91 and XBIN01 cf. mapping table, and save relevant information for both XDIS91 and XBIN01. |  | XDIS91 and XBIN01 are mapped correctly and wrapped in individual VANSEnvelopes.  The messages are and are ready to be sent to the correct receiver with information about being a reply.  Save the files for documentation. |  | Choose |
|  | **Receive a reply (to a message with an attachment) and send an XDIS91**  Receive a communication message of the type CareCommunication.  Demonstrate that the SUT registers that the CareCommunication is a reply to a previously sent message with an attachment.  ***Note****: The base64-encoded content of the attachment is not included, when the CareCommunication is a reply.* | ConSer\_CC\_09\_reply | The CareCommunication is a reply.  The CareCommunication must be converted to XDIS91 with a reference to the attachment. |  | Choose |
|  | Perform test step 3.2.9.3-3.2.9.8, with the difference that SUT maps the CareCommunication to an XDIS91 with a reference in the text to the attachment in the original reply, cf. [mapping table](#_Baggrundsmaterialer_2). |  | Mapping is performed correctly and XDIS91 is wrapped in the VANSEnvelope.  XDIS91 is ready to be sent to the correct receiver with information about being a reply with a reference to the attachment.  Save the files for documentation. |  | Choose |
| Receive a forward and send a XDIS91 | | | | | |
|  | Receive a communication message.  Demonstrate that the SUT checks the VANSEnvelope to identify the message type. | ConSer\_CC\_10\_forward | The CareCommunication is a forward. |  | Choose |
|  | Demonstrate that SUT checks the capabilities of the receiver of the message in SOR or a local copy of SOR. |  | Receiver can receive a XDIS91 after look-up in SOR or a local copy. |  | Choose |
|  | Demonstrate that the SUT saves relevant information from the CareCommunication:   * Bundle.id * MessageHeader.id |  | Relevant information is saved |  | Choose |
|  | Demonstrate that the SUT maps the CareCommunication to an XDIS91, cf. [mapping table](#_Baggrundsmaterialer_2). |  | Mapping is performed correctly as described in the mapping table. |  | Choose |
|  | Demonstrate that the SUT includes a note in the text informing that the message is a forward. |  | A XDIS91 is created with information about being a forward. |  | Choose |
|  | Demonstrate that the SUT saves relevant information from the XDIS91:   * Emessage.Envelope.identifier * Emessage.Letter.identifier |  | Relevant information is saved. |  | Choose |
|  | The XDIS91 is wrapped in individual VANSEnvelope with correct content. |  | XDIS91 is wrapped in the VANSEnvelope. |  | Choose |
|  | Show that the message is ready to be sent to the correct receiver. |  | The message is mapped correctly and is ready to be sent to the correct receiver with information about being a forward.  Save the file for documentation. |  | Choose |
|  | **Receive a forward with attachment and send a XDIS91**  Receive a communication message of the type CareCommunication.  Demonstrate that the SUT registers that the CareCommunication is a forward and includes an attachment. | ConSer\_CC\_11\_forward\_attachment | The CareCommunication is a forward with an attachment.  The CareCommunication must be converted to XDIS91 and XBIN01. |  | Choose |
|  | Perform test step 3.2.9.15-3.2.9.20, but map the CareCommunication to an XDIS91 and XBIN01 cf. mapping table, and save relevant information for both XDIS91 and XBIN01. |  | XDIS91 and XBIN01 are mapped correctly and wrapped in individual VANSEnvelopes.  XDIS91 and XBIN01 are ready to be sent to the correct receiver with information about being a forward.  Save the files for documentation. |  | Choose |

### S3: Send an Acknowledgement (from XML to FHIR)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
| Send a positive Acknowledgement (a positive XCTL is received) | | | | | |
|  | VANS has converted ConSer\_CC\_02 to an XDIS91. Now, VANS receives the XCTL from the receiver of XDIS91, which must be converted to an ACK.  The test file provided, is generic, VANS must fill in the needed information to connect it to the converted XDIS91 by VANS.  Demonstrate that the SUT registers that the XCTL is positive.  ***Note:*** *This covers scenarios where all XCTL are positive.*  ***Note:*** *To complete this test step, VANS tester must transfer relevant information manually from the XDIS91 that is generated in test step 3.2.7.1 to the correct elements in the given test data file for this test step. This is to ensure that the XCTL matches* ConSer\_CC\_02 *test file*. | ConSer\_XCTL03\_01  ***Note:*** *The file provided is a general example that needs to be updated with correct information by VANS.* | The message is a positive XCTL and must be converted to a positive Acknowledgement before the message is ready to be sent to receiver. |  | Choose |
|  | Demonstrate that the SUT maps the positive XCTL to a positive Acknowledgement, cf. [mapping table](#_Baggrundsmaterialer_2) including using the saved data from the initial CareCommunication converted in 3.2.7. |  | Mapping is performed correctly as described in the mapping table. |  | Choose |
|  | Demonstrate that the SUT uses relevant information, saved by VANS, to ensure that the Acknowledgement is paired to the correct CareCommunication. |  | Acknowledgement is paired to the correct CareCommunication. |  | Choose |
|  | Acknowledgement is validated against the implementation guide. |  | Validation is performed and went well. |  | Choose |
|  | The Acknowledgement is wrapped in a [VANSEnvelope](#_Baggrundsmaterialer_2) with correct content. |  | Acknowledgement is wrapped in the VANSEnvelope. |  | Choose |
|  | Show that the positive Acknowledgement is ready to be sent to the correct receiver. |  | The message is mapped correctly and is ready to be sent to the correct receiver.  Save the file for documentation. |  | Choose |
| Send a negative Acknowledgement (a negative XCTL is received) | | | | | |
|  | VANS has converted ConSer\_CC\_09\_reply to an XDIS91. Now, VANS receives the XCTL from the receiver of XDIS91, which must be converted to an ACK.  The test file provided, is generic, VANS must fill in the needed information to connect it to the converted XDIS91 by VANS.  Demonstrate that the SUT registers that the XCTL is negative.  ***Note:*** *This covers scenarios where one or more XCTL are negative.*  ***Note:*** *To complete this test step, VANS tester must transfer relevant information manually from the XDIS91 that is generated in test step 3.2.9.11 to the correct elements in the given test data file for this test step. This is to ensure that the XCTL matches* ConSer\_CC\_09\_reply *test file*. | ConSer\_XCTL02\_01  ***Note:*** *The file provided is a general example file that needs to be updated with correct information by VANS.* | The message is a negative XCTL and must be converted to a negative Acknowledgement before the message is sent to receiver. |  | Choose |
|  | Demonstrate that the SUT maps the negative XCTL to a negative Acknowledgement, cf. [mapping table](#_Baggrundsmaterialer_2), including using the saved data from the initial CareCommunication converted in 3.2.7. |  | Mapping is performed correctly as described in the mapping table. |  | Choose |
|  | Demonstrate that the SUT uses relevant information, saved by VANS, to ensure that the Acknowledgement is paired to the correct CareCommunication. |  | Acknowledgement is paired to the correct CareCommunication. |  | Choose |
|  | Acknowledgement is validated against the implementation guide. |  | Validation is performed and went well. |  | Choose |
|  | The Acknowledgement is wrapped in a [VANSEnvelope](#_Baggrundsmaterialer_2) with correct content. |  | Acknowledgement is wrapped in the VANSEnvelope. |  | Choose |
|  | Show that the negative Acknowledgement is ready to be sent to the correct receiver. |  | The message is mapped correctly and is ready to be sent to the correct receiver.  Save the file for documentation. |  | Choose |
| Send a positive Acknowledgement (two positive XCTL is received) | | | | | |
|  | VANS has converted ConSer\_CC\_03\_attachment to an XDIS91 and XBIN01. Now, VANS receives two XCTL from the receiver of XDIS91 and XBIN01, which must be converted to an ACK.  The test file provided, is generic, VANS must fill in the needed information to connect it to the converted XDIS91 by VANS.  Demonstrate that the SUT registers that the XCTL is positive.  ***Note:*** *This covers scenarios where all XCTL are positive.*  ***Note:*** *To complete this test step, VANS tester must transfer relevant information manually from the XDIS91 that is generated in test step 3.2.8.1 to the correct elements in the given test data file for this test step. This is to ensure that the XCTL matches* ConSer\_CC\_03\_attachment *test file*. | ConSer\_XCTL03\_01  ***Note:*** *The file provided is a general example that needs to be updated with correct information by VANS.* | The two messages are positive XCTLs and must be converted to one positive Acknowledgement before the message is ready to be sent to receiver. |  | Choose |
|  | Demonstrate that the SUT maps the positive XCTL to a positive Acknowledgement, cf. [mapping table](#_Baggrundsmaterialer_2) including using the saved data from the initial CareCommunication converted in 3.2.8. |  | Mapping is performed correctly as described in the mapping table. |  | Choose |
|  | Demonstrate that the SUT uses relevant information, saved by VANS, to ensure that the Acknowledgement is paired to the correct CareCommunication. |  | Acknowledgement is paired to the correct CareCommunication. |  | Choose |
|  | Acknowledgement is validated against the implementation guide. |  | Validation is performed and went well. |  | Choose |
|  | The Acknowledgement is wrapped in a [VANSEnvelope](#_Baggrundsmaterialer_2) with correct content. |  | Acknowledgement is wrapped in the VANSEnvelope. |  | Choose |
|  | Show that the positive Acknowledgement is ready to be sent to the correct receiver. |  | The message is mapped correctly and is ready to be sent to the correct receiver.  Save the file for documentation. |  | Choose |
| Send a negative Acknowledgement (two negative XCTL is received) | | | | | |
|  | VANS has converted ConSer\_CC\_04\_MultipleAttachments to an XDIS91 and XBIN01. Now, VANS receives two XCTLs from the receiver of XDIS91 and XBIN01, which must be converted to an ACK.  The test file provided, is generic, VANS must fill in the needed information to connect it to the converted XDIS91 by VANS.  Demonstrate that the SUT registers that the XCTL is negative.  ***Note:*** *This covers scenarios where one or more XCTL are negative.*  ***Note:*** *To complete this test step, VANS tester must transfer relevant information manually from the XDIS91 that is generated in test step3.2.8.12 to the correct elements in the given test data file for this test step. This is to ensure that the XCTL matches* ConSer\_CC\_04\_MultipleAttachments *test file*. | ConSer\_XCTL02\_01  ***Note:*** *The file provided is a general example file that needs to be updated with correct information by VANS.* | The message is a negative XCTL and must be converted to a negative Acknowledgement before the message is sent to receiver. |  | Choose |
|  | Demonstrate that the SUT maps the negative XCTLs to a negative Acknowledgement, cf. [mapping table](#_Baggrundsmaterialer_2), including using the saved data from the initial CareCommunication converted in 3.2.8.12*.* |  | Mapping is performed correctly as described in the mapping table. |  | Choose |
|  | Demonstrate that the SUT uses relevant information, saved by VANS, to ensure that the Acknowledgement is paired to the correct CareCommunication. |  | Acknowledgement is paired to the correct CareCommunication. |  | Choose |
|  | Acknowledgement is validated against the implementation guide. |  | Validation is performed and went well. |  | Choose |
|  | The Acknowledgement is wrapped in a [VANSEnvelope](#_Baggrundsmaterialer_2) with correct content. |  | Acknowledgement is wrapped in the VANSEnvelope. |  | Choose |
|  | Show that the negative Acknowledgement is ready to be sent to the correct receiver. |  | The message is mapped correctly and is ready to be sent to the correct receiver.  Save the file for documentation. |  | Choose |
|  | VANS has converted ConSer\_CC\_08\_reply\_attachment to an XDIS91 and XBIN01. Now, VANS receives one positive and one XCTL from the receiver of XDIS91 and XBIN01, which must be converted to an ACK.  The test file provided, is generic, VANS must fill in the needed information to connect it to the converted XDIS91 by VANS.  Demonstrate that the SUT registers that the XCTL is negative.  ***Note:*** *This covers scenarios where one or more XCTL are negative.*  ***Note:*** *To complete this test step, VANS tester must transfer relevant information manually from the XDIS91 that is generated in test step 3.2.9.10 to the correct elements in the given test data file for this test step. This is to ensure that the XCTL matches* ConSer\_CC\_08\_reply\_attachment *test file*. | ConSer\_XCTL02\_01  ***Note:*** *The file provided is a general example file that needs to be updated with correct information by VANS.* | The messages received is a negative XCTL and a positive XCTL and must be converted to a negative Acknowledgement before the message is sent to receiver. |  |  |
|  | Demonstrate that the SUT maps the negative and positive XCTL to a negative Acknowledgement, cf. [mapping table](#_Baggrundsmaterialer_2), including using the saved data from the initial CareCommunication converted in 3.2.9.9. |  | Mapping is performed correctly as described in the mapping table. |  |  |
|  | Demonstrate that the SUT uses relevant information, saved by VANS, to ensure that the Acknowledgement is paired to the correct CareCommunication. |  | Acknowledgement is paired to the correct CareCommunication. |  |  |
|  | Acknowledgement is validated against the implementation guide. |  | Validation is performed and went well. |  |  |
|  | The Acknowledgement is wrapped in a [VANSEnvelope](#_Baggrundsmaterialer_2) with correct content. |  | Acknowledgement is wrapped in the VANSEnvelope. |  |  |
|  | Show that the negative Acknowledgement is ready to be sent to the correct receiver. |  | The message is mapped correctly and is ready to be sent to the correct receiver.  Save the file for documentation. |  |  |

### S4: Send an XCTL (from FHIR to XML)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
| Send a positive XCTL (a positive Acknowledgement is received) | | | | | |
|  | VANS has converted ConSer\_XDIS91\_02 to a CareCommunication. Now, VANS receives the ACK from the receiver of CareCommunication, this must be converted to an XCTL.  The test file provided, is generic, VANS must fill in the needed information to connect it to the converted CareCommunication by VANS.  ***Note:*** *To complete this test step, VANS tester must transfer relevant information manually from the CareCommunication that is generated in test step 3.2.2.3 to the correct elements in the given test data file for this test step. This is to ensure that the XCTL matches* ConSer\_XDIS91\_02 *test file*. | ConSer\_ACK\_01  *Note: This is a general example. VANS must enrich the ACK with the necessary information so that it matches the converted CareCommunication, e.g. the MessageHeader ID must be inserted into the ACK.* | The message is a positive Acknowledgement and must be converted to XCTL before the message is ready to be sent to receiver. |  | Choose |
|  | Demonstrate that the SUT maps the Acknowledgement to an XCTL, cf. [mapping table](#_Baggrundsmaterialer_2).  ***Note****: The mapping table includes XCTL01, 02 and 03.* |  | Mapping is performed correctly as described in the mapping table. |  | Choose |
|  | Demonstrate that the SUT uses relevant information, saved by VANS, to ensure that the XCTL is paired to the correct XDIS91. |  | XCTL is paired to the correct XDIS91. |  | Choose |
|  | The XCTL is wrapped in a [VANSEnvelope](#_Baggrundsmaterialer_2) with correct content. |  | XCTL is wrapped in the VANSEnvelope. |  | Choose |
|  | Show that the XCTL is ready to be sent to the correct receiver. |  | The message is mapped correctly and is ready to be sent to the correct receiver.  Save the file for documentation. |  | Choose |
| Send two positive XCTL (a positive Acknowledgement is received) | | | | | |
|  | VANS has converted ConSer\_XDIS91\_03\_XDIS91 and ConSer\_XDIS91\_03\_XBIN01 to a CareCommunication. Now, VANS receives the ACK from the receiver of CareCommunication, this must be converted to two XCTL.  The test file provided, is generic, VANS must fill in the needed information to connect it to the converted CareCommunication by VANS.  ***Note:*** *To complete this test step, VANS tester must transfer relevant information manually from the CareCommunication that is generated in test step 3.2.3.4 to the correct elements in the given test data file for this test step. This is to ensure that the XCTL matches* ConSer\_XDIS91\_03\_XDIS91 and ConSer\_XDIS91\_03\_XBIN01 *test file*. | ConSer\_ACK\_01  *Note: This is a general example. VANS must enrich the ACK with the necessary information so that it matches the converted CareCommunication, e.g. the MessageHeader ID must be inserted into the ACK.* | The message is a positive Acknowledgement and must be converted to two XCTL. |  | Choose |
|  | Demonstrate that the SUT maps the Acknowledgement to two XCTL (one for XDIS91 and one for XBIN01), cf. [mapping table](#_Baggrundsmaterialer_2).  ***Note****: The mapping table includes XCTL01, 02 and 03.* |  | Mapping is performed correctly as described in the mapping table. |  | Choose |
|  | Demonstrate that the SUT uses relevant information, saved by VANS, to ensure that the XCTL is paired to the correct XDIS91 and XBIN01. |  | XCTL is paired to the correct XDIS91 and XBIN01. |  | Choose |
|  | Both XCTL is wrapped in individual [VANSEnvelope](#_Baggrundsmaterialer_2) with correct content. |  | Both XCTL are wrapped in individual VANSEnvelope. |  | Choose |
|  | Show that both XCTL are ready to be sent to the correct receiver. |  | The messages are mapped correctly and are ready to be sent to the correct receiver.  Save the file for documentation. |  | Choose |
| Send a negative XCTL (a negative Acknowledgement is received) | | | | | |
|  | VANS has converted ConSer\_XDIS91\_10\_replacementCPR and to a CareCommunication. Now, VANS receives the ACK from the receiver of CareCommunication, this must be converted to two XCTL.  The test file provided, is generic, VANS must fill in the needed information to connect it to the converted CareCommunication by VANS.  ***Note:*** *To complete this test step, VANS tester must transfer relevant information manually from the CareCommunication that is generated in test step 3.3.1.4 to the correct elements in the given test data file for this test step. This is to ensure that the XCTL matches* ConSer\_XDIS91\_10\_ReplacementCPR *test file*. | ConSer\_ACK\_02  *Note: This is a general example. VANS must enrich the ACK with the necessary information so that it matches the converted CareCommunication, e.g. the MessageHeader ID must be inserted into the ACK.* | The message is a negative Acknowledgement and must be converted to XCTL before the message is ready to be sent to receiver. |  | Choose |
|  | Demonstrate that the SUT maps the negative Acknowledgement to a negative XCTL, cf. [mapping table](#_Baggrundsmaterialer_2).  ***Note****: The mapping table includes XCTL01, 02 and 03.* |  | Mapping is performed correctly as described in the mapping table. |  | Choose |
|  | Demonstrate that the SUT uses relevant information, saved by VANS, to ensure that the XCTL is paired to the correct XDIS91. |  | XCTL is paired to the correct XDIS91. |  | Choose |
|  | Demonstrate that the SUT transfers the error message from the Acknowledgment to XCTL. |  | XCTL contains correct error message. |  | Choose |
|  | The XCTL is wrapped in a [VANSEnvelope](#_Baggrundsmaterialer_2) with correct content. |  | XCTL is wrapped in the VANSEnvelope. |  | Choose |
|  | Show that the XCTL is ready to be sent to the correct receiver. |  | The message is mapped correctly and is ready to be sent to the correct receiver.  Save the file for documentation. |  | Choose |
| Send two negative XCTL (a negative Acknowledgement is received) | | | | | |
|  | VANS has converted ConSer\_XDIS91\_09\_XDIS91 and ConSer\_XDIS91\_09\_XBIN01 to a CareCommunication. Now, VANS receives the ACK from the receiver of CareCommunication, this must be converted to two XCTL.  The test file provided, is generic, VANS must fill in the needed information to connect it to the converted CareCommunication by VANS.  ***Note:*** *To complete this test step, VANS tester must transfer relevant information manually from the CareCommunication that is generated in test step 3.2.3.4 to the correct elements in the given test data file for this test step. This is to ensure that the XCTL matches* ConSer\_XDIS91\_09\_XDIS91 and ConSer\_XDIS91\_09\_XBIN01 *test file*. | ConSer\_ACK\_02  *Note: This is a general example. VANS must enrich the ACK with the necessary information so that it matches the converted CareCommunication, e.g. the MessageHeader ID must be inserted into the ACK.* |  |  | Choose |
|  | Demonstrate that the SUT maps the negative Acknowledgement to two negative XCTL cf. [mapping table](#_Baggrundsmaterialer_2). |  | Mapping is performed correctly as described in the mapping table. |  | Choose |
|  | Demonstrate that the SUT uses relevant information, saved by VANS, to ensure that the XCTL’s is paired to the correct XDIS91 and XBIN01. |  | XCTL is paired to the correct XDIS91 and XBIN01. |  | Choose |
|  | Demonstrate that the SUT transfers the error message from the Acknowledgment to both XCTL. |  | Both XCTL contain correct error message. |  | Choose |
|  | Demonstrate that both XCTL are wrapped in individual VANSEnvelope with correct content and are ready to be sent to the correct receiver |  | XCTLs are wrapped in individual VANSEnvelope.  The messages are ready to be sent to the correct receiver.  Save the file for documentation |  | Choose |
| Send a negative XCTL and supply with correct error message | | | | | |
|  | VANS has converted ConSer\_XDIS91\_06 to an invalid CareCommunication. Now, VANS receives a negative ACK from the receiver of the CareCommunication, this must be converted to a negative XCTL.  The test file provided, is generic, VANS must fill in the needed information to connect it to the converted CareCommunication by VANS.  ***Note:*** *To complete this test step, VANS tester must transfer relevant information manually from the CareCommunication that is generated in test step 3.2.4.2 to the correct elements in the given test data file for this test step. This is to ensure that the XCTL matches* ConSer\_XDIS91\_06 *test file*. | ConSer\_ACK\_02  *Note: This is a general example. VANS must enrich the ACK with the necessary information so that it matches the converted CareCommunication, e.g. the MessageHeader ID must be inserted into the ACK.* | The message is a negative Acknowledgement and must be converted to XCTL before the message is ready to be sent to receiver. Furthermore, the SUT must insert a text explaining why, VANS sent an invalid CareCommunication |  |  |
|  | Demonstrate that the SUT maps the negative Acknowledgement to a negative XCTL cf. [mapping table](#_Baggrundsmaterialer_2). |  | Mapping is performed correctly as described in the mapping table. |  |  |
|  | Demonstrate that the SUT uses relevant information, saved by VANS, to ensure that the XCTL is paired to the correct XDIS91. |  | XCTL is paired to the correct XDIS91. |  |  |
|  | Demonstrate that the SUT transfers the error message from the Acknowledgement to the XCTL. |  | The XCTL contains correct error message. |  |  |
| D | Demonstrate that the SUT inserts the error message describing why the SUT sent an invalid CareCommunication.  See error messages in [use case document, section 4.4](#_Baggrundsmaterialer_2). |  | SUT inserts the error message describing why CareCommunication is invalid. |  |  |
|  | Demonstrate that the XCTL is wrapped in a VANSEnvelope with correct content and is ready to be sent to the correct receiver |  | XCTL is wrapped in aVANSEnvelope.  The message is ready to be sent to the correct receiver.  Save the file for documentation |  |  |

## Test of general technical requirements

The purpose of these test steps is to ensure that the technical aspects of messages generated by the Conversion service is implemented with satisfactory quality.

### General technical requirements

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test step #** | **Action** | **Test data** | **Expected results** | **Actual result** | **MedCom assessment** |
| Use of SOR | | | | | |
|  | If SUT does not have sufficient information to complete the mapping of a message, SUT must perform a lookup in SOR or local copy. Describe the use of look-up in SOR or a local copy.  Describe how SOR is used to gain the needed information to complete the conversion. |  | E.g.” *When a XDIS91, XBIN01, or XCTL is received, only including an EAN-number, this is used to lookup the corresponding code in SOR. When a CareCommunication and Acknowledgement is received EAN-number and SOR-code are mapped to the corresponding elements in the XDIS91, XBIN01, or XCTL*” |  | Choose |
|  | Describe in which format you create CareCommunications. |  | E.g. FHIR-XML or FHIR-JSON. |  | Choose |
|  | **Narrative text:**  All instances in CareCommunication include a narrative text.  The narrative text must include the element marked with “O” in the CareCommunication IG. |  | Elements marked with “O” shall be included in the narrative text. See MedCom governance for narratives [here](https://medcomdk.github.io/MedComLandingPage/assets/documents/GeneralGovernanceFHIRStandards.html). |  | Choose |
| Replacement CPR or more than one regular CPR/replacement CPR | | | | | |
|  | Convert the provided XDIS91 to CareCommunication and make sure to include the replacement CPR correctly . | ConSer\_XDIS91\_10\_ReplacementCPR | The replacement CPR is transferred correctly. |  | Choose |
|  | Convert the provided CareCommunication to XDIS91 and make sure to include the replacement CPR correctly | ConSer\_CC\_12\_ReplacementCPR | The replacement CPR is transferred correctly. |  | Choose |
|  | The provided CareCommunication has two or more identifiers in the patient resource, a regular CPR-number and a replacement CPR-number.  VANS must, in the conversion of the CareCommunication to XDIS91, only include the regular CPR-number in the patient segment. The replacement CPR-number must be transferred to the text string in the XDIS91 with an additional text stating: “Der er modtaget flere end et CPR-nummer I korrespondancemeddelelsen. Derfor er de resterende medsendte CPR-numre indsat her”. | ConSer\_CC\_13\_DoubleCPR | The regular CPR-number is transferred correctly to XDIS91, and the replacement CPR-number is transferred to the text string in the XDIS91 with an additional text. |  | Choose |
|  | The provided CareCommunication has one regular CPR-number and two replacement CPR-numbers of different types included in the patient resource.  VANS must convert the CareCommunication to XDIS91 only including the regular CPR-number in the patient segment. The two replacement CPR-numbers must be transferred to the text string in the XDIS91 with an additional text stating: “Der er modtaget flere end et CPR-nummer i korrespondancemeddelelsen. Derfor er de resterende medsendte CPR-numre indsat her”. | ConSer\_CC\_14\_DoubleCPR | The regular CPR-number is transferred correctly to the patient to XDIS91, and the two replacement CPR-numbers are transferred to the text string in the XDIS91 with additional text. |  | Choose |
|  | The provided CareCommunication has two or more replacement CPR-numbers included in the patient resource.  VANS must convert the CareCommunication to XDIS91 only including one of the replacement CPR-numbers in the patient segment. The other regular CPR-numbers must be transferred to the text string in the XDIS91 with an additional text stating: “Der er modtaget flere end et CPR-nummer i korrespondancemeddelelsen. Derfor er de resterende medsendte CPR-numre indsat her”.  *\** *SUT must always select the national replacement CPR number as the identifier to be included in the patient segment. The national replacement CPR number is identified by the system value: urn:oid:1.2.208.176.1.6.1.1* | ConSer\_CC\_15\_DoubleCPR | A replacement CPR-number is transferred correctly to XDIS91, and the other replacement CPR-number(s) is transferred to the text string in the XDIS91 with an additional text. |  |  |
| Attachment size | | | | | |
|  | Describe and demonstrate how the size of the attachment is calculated before sending the XBIN01.  ***Note:*** *In a CareCommunication with an attachment, the size of the attachment is not mandatory to include*. |  | E.g. “*The size of the base64encoded content is calculated before including it into the XBIN01*” |  | Choose |
| Timestamps | | | | | |
|  | Demonstrate the timestamps and time zones are transferred correctly from the CareCommunication to the XDIS91.  ***Note****: Time zone is included in CareCommunication and can be zulutime or another timezone.* | Demonstrate on the file from test step 3.2.1.5: ConSer\_CC\_02 | Timestamps are transferred correctly.  The timestamps in the XDIS91 and XBIN01 has the format: HH:MM = "00:00", where seconds and timezone is included in the CareCommunication. |  | Choose |

### Embedment into VANSEnvelope

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test step #** |  | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
|  | Demonstrate that the XDIS91 is correctly wrapped in VANSEnvelope.  ***Note****: Link to VANSEnvelope specifications can be found in Background materials.*  ***Note****: Test data performed earlier in the test protocol can be referenced in this and the following test steps.* |  | The XDIS91 is wrapped correctly in the VANSEnvelope. |  | Choose |
|  | Demonstrate that the XBIN01 is correctly wrapped in VANSEnvelope. |  | The XBIN01 is wrapped correctly in the VANSEnvelope. |  | Choose |
|  | Demonstrate that the CareCommunication is correctly wrapped in VANSEnvelope. |  | The CareCommunication is wrapped correctly in the VANSEnvelope. |  | Choose |
|  | Demonstrate that the XCTL is correctly wrapped in VANSEnvelope. |  | The XCTL is wrapped correctly in the VANSEnvelope. |  | Choose |
|  | Demonstrate that the Acknowledgement is correctly wrapped in VANSEnvelope. |  | The Acknowledgement is wrapped correctly in the VANSEnvelope. |  | Choose |

### Attachments from DNHF

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test step #** |  | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
|  | Demonstrate that BIN01 and XBIN01 which are sent from DNHF, are passed on without delay.  ***Note****: Both test data files must be tested.* | ConSer\_BIN\_DNHF  ConSer\_XBIN\_DNHF | The BIN01 and XBIN01 with DNHF as the sender is passed on without delay. |  | Choose |