

<eXTra>

einheitliches XML-basiertes Transportverfahren



Schnittstellenspezifikation zum XML-basierenden, elektronischen Datenaustausch mit der Rentenversicherung im Verfahren Reha§301

Version 1.02.00

Stand der Spezifikation: 09.03.2018

Version: 1.02.00

Redaktion: Deutsche Rentenversicherung Bund
Referat 0551
Berner Strasse 1
97084 Würzburg

Verfahrensverantwortlich: Andrea Bork
Telefon 0931/6002-73544
eMail: andrea.bork@drv-bund.de

Techn. Verantwortlich: Piotr Kolloch
Telefon 0931/6002-73339
eMail: piotr.kolloch@drv-bund.de

0 Allgemeines

Das vorliegende Dokument dient als Schnittstellenbeschreibung zur eXTra-Kommunikation zwischen den Kliniksystemen und der Datenstelle der Träger der Rentenversicherung (DSRV) im neuen XML-basierenden Reha§301-Verfahren.

Das Dokument unterteilt sich in zwei Abschnitte.

- Teil A – Allgemeine Informationen
- Teil B – Beschreibung der Elemente

In Teil A werden nur die allgemeinen Parameter und Voraussetzungen der Kommunikationsbeziehung erläutert.

In Teil B wird auf die einzelnen Elemente der verwendeten eXTra-Profilierung eingegangen.

Änderungsübersicht

Version	Datum	Kap.	Änderungsgrund	Bearbeiter
1.00.00	01.12.2015	Alle	Registrierung des Dokuments	DSRV
1.01.00	25.05.2016	Alle	Überarbeitung, Erweiterung um Plugins	DSRV
1.01.01	16.03.2017		Adressierung der DSRV von BBNR auf IK-Nummer geändert	DSRV
1.02.00	09.03.2018	Alle	Layout überarbeitet Diverse Korrekturen Originator-Attribut ins Profil aufgenommen Neue Beispieldatei: 02responseFehlenachricht.xml	DSRV

1 Inhaltsverzeichnis

0 Allgemeines	2
1 Inhaltsverzeichnis	3
2 Teil A – Allgemeine Informationen	5
2.1 Grafischer Überblick Geschäftsprozesse	5
2.2 Server-Adressen.....	5
2.3 Authentifizierung.....	6
2.4 Verwendetes eXTra-Schema.....	6
2.4.1 Profile-Attribut des Root-Elements	6
2.4.2 Nutzdatenbeschreibung (DataSource).....	6
2.4.3 Komprimierung/Verschlüsselung (DataTransforms)	6
2.4.4 Übertragung der Nutzdaten - MTOM	7
2.5 Beschreibung der einzelnen Geschäftsprozesse	7
2.5.1 Senden einer Reha-Nachricht an die Rentenversicherung	7
2.5.2 Abholen einer Reha-Nachricht von der Rentenversicherung	7
Keine weitere Daten vorhanden	9
2.6 Verwendung der ResponseID.....	9
2.6.1 Hotline.....	9
2.6.2 DataRequest	9
2.7 IK-Nummer der RV-Annahmestelle.....	9
3 Teil B – Beschreibung der Elemente.....	10
3.1 Verwendete Namensräume und Präfixe	10
3.2 Aufbau des Transport-Headers.....	10
3.2.1 Transport-Header des Request	10
3.2.2 Transport-Header der Response	14
3.3 Aufbau des TransportPlugins.....	18
3.3.1 Aufbau des DataTransforms-Plugins.....	18
3.4 Request: Senden einer Reha-Nachricht an die Rentenversicherung	21
3.4.1 Element Transport.....	21
3.4.2 Element TransportHeader	21
3.4.3 Element TransportPlugins	21
3.4.4 Element TransportBody.....	21
3.4.5 Element Data	22

3.4.6	Element Base64CharSequence	22
3.5	Response: Senden einer Reha-Nachricht an die Rentenversicherung.....	22
3.5.1	Element Transport.....	22
3.5.2	Element TransportHeader	23
3.5.3	Element TransportBody.....	23
3.6	Request: Abholen einer Reha-Nachricht von der Rentenversicherung.....	23
3.6.1	Element Transport.....	23
3.6.2	Element TransportHeader	24
3.6.3	Element TransportBody.....	24
3.6.4	Element Data	24
3.6.5	Element ElementSequence	24
3.6.6	Element DataRequest	24
3.6.7	Element Query	25
3.6.8	Element Argument.....	25
3.7	Response: Abholen einer Reha-Nachricht von der Rentenversicherung.....	26
3.7.1	Element Transport.....	26
3.7.2	Element TransportHeader	26
3.7.3	Element TransportPlugins	26
3.7.4	Element TransportBody.....	27
3.7.5	Element Data	28
3.7.6	Element Base64CharSequence	28
4	Anhang.....	28
4.1	StatusCodes	28
4.2	Referenzierte Dokumente	29
4.3	Abbildungsverzeichnis	29

2 Teil A – Allgemeine Informationen

2.1 Grafischer Überblick Geschäftsprozesse

Die Kommunikation mit dem Server der Rentenversicherung (DSRV) ist in zwei Geschäftsprozesse unterteilt:

- Senden einer Reha-Nachricht an die Rentenversicherung
- Abholen einer Reha-Nachricht von der Rentenversicherung

In beiden Geschäftsprozessen erfolgt die Kommunikation synchron, d.h. zu dem jeweiligen Request wird vom Server der Rentenversicherung eine adäquate Response generiert.

Folgende Grafik zeigt den groben Ablauf der in diesem Dokument beschriebenen Geschäftsprozesse auf

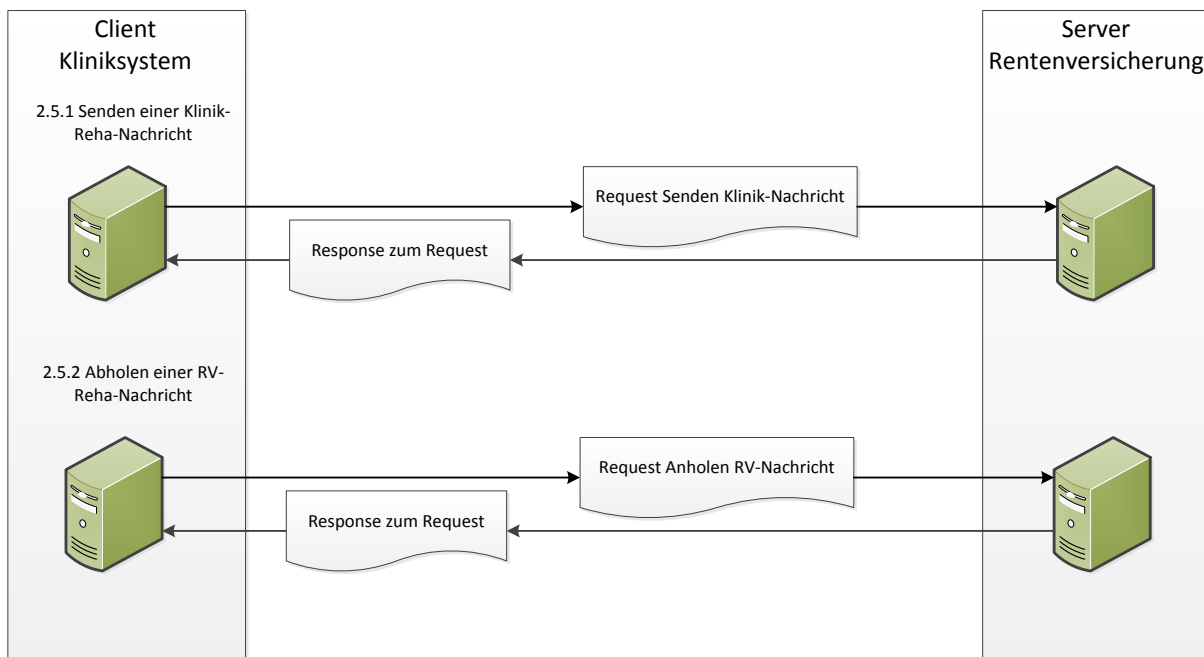


Abbildung 1 – Geschäftsprozesse

2.2 Server-Adressen

Bei den Test- und Produktionsservern der DSRV handelt es sich um physikalisch getrennte Server-Systeme, die über folgende URL's erreichbar sind:

Testsystem:

<https://itsg.eservicet-drv.de/SPoC/ExtraService>

Produktionssystem:

<https://itsg.eservice-drv.de/SPoC/ExtraService>

2.3 Authentifizierung

Zum Aufbau der SSL-Verbindung wird clientseitig das Leistungserbringerzertifikat der ITSG benötigt. Dieses muss im Verlauf des SSL-Handshakes vom Client an den Server übergeben werden.

Für die Kommunikationsverschlüsselung wird TLS 1.2 verwendet.

2.4 Verwendetes eXTra-Schema

Zum Einsatz kommt das eXTra-Basischema in der Version 1.3.0.

2.4.1 Profile-Attribut des Root-Elements

Folgendes Profile Attribut müssen alle Geschäftsprozesse im Root – Element einbinden:

<http://www.extra-standard.de/profile/Reha301/1.0>

2.4.2 Nutzdatenbeschreibung (DataSource)

Die Nutzdaten werden immer im Zeichensatz ISO 8859-1 codiert. Deshalb entfällt das DataSource-Plugin.

2.4.3 Komprimierung/Verschlüsselung (DataTransforms)

Vor der Verschlüsselung können Sie die Nutzdaten mit GZIP-Verfahren komprimieren.

Zulässige Werte für eXTra-Element Compression

Daten wurden nicht komprimiert	http://www.extra-standard.de/transforms/compression/NONE
Daten komprimiert	http://www.extra-standard.de/transforms/compression/GZIP

Gemäß [KKS](#) müssen die Daten vom Leistungserbringer (Absender) signiert und mit dem öffentlichen Schlüssel der DSRV (Empfänger) verschlüsselt (PKCS#7) werden.

Für die Signatur und Verschlüsselung sind Leistungserbringerzertifikate der ITSG zu verwenden.

Zulässige Werte für eXTra Element Algorithm

Verschlüsselung	http://www.extra-standard.de/transforms/encryption/PKCS7
Keine Verschlüsselung	http://www.extra-standard.de/transforms/encryption/NONE

	XML-basierender Datenaustausch mit der Rentenversicherung im Verfahren Reha§301 Schnittstellenbeschreibung	Seite: 7 Version: 1.02.00 Stand: 09.03.18
--	--	--

Reihenfolge der Komprimierung und der Verschlüsselung

Die im DataTransforms beschriebene Komprimierung und Verschlüsselung hat folgende Sortierung. Die Reihenfolge wird über das Attribut „order“ festgelegt.

1. Komprimierung der Nutzdaten
2. Verschlüsselung/Signatur der Nutzdaten mit PKCS#7

Anmerkung zur Signatur:

Das eXtra-Schema sieht das Plugin-Element „Signature“ vor, das in diesem Verfahren nicht benutzt wird, da die Signatur gemäß KKS implizit im PKCS#7-Objekt integriert ist.

2.4.4 Übertragung der Nutzdaten - MTOM

Enthält eine eXtra-Nachricht Nutzdaten (Reha-XML-Nachricht), so werden diese als MTOM-Objekt mit der SOAP-Nachricht gesendet. Die Referenz auf das MTOM-Objekt wird im Element Base64CharSequence abgelegt, wobei diese Referenzangabe nicht in base64-kodiert werden muss.

2.5 Beschreibung der einzelnen Geschäftsprozesse

An dieser Stelle werden die im [Abschnitt 2.1](#) dargestellten Geschäftsprozesse beschrieben.

2.5.1 Senden einer Reha-Nachricht an die Rentenversicherung

Dieser Geschäftsprozess dient zum Senden einer Reha-Nachricht durch den Leistungserbringer (Klinik) an die Rentenversicherung. Es kann immer nur eine XML-Nachricht pro Request gesendet werden.

Procedure: <http://www.extra-standard.de/procedures/REHA301>

Data Type: <http://www.extra-standard.de/datatypes/SendeDaten>

Scenario: <http://www.extra-standard.de/scenario/request-with-acknowledgement>

Beim Senden ist die eXtra-Paketebene nicht vorgesehen. Die Nutzdaten werden als MTOM-Objekt gesendet.

Ein Beispiel eines Request kann der Datei 01request.xml entnommen werden.

Als Antwort auf den Request bekommen sie von der Rentenversicherung synchron eine Response, die als Annahmequittung fungiert.

Ein Beispiel einer Antwort kann der Datei 01response.xml entnommen werden.

2.5.2 Abholen einer Reha-Nachricht von der Rentenversicherung

In diesem Prozess werden die von der Rentenversicherung erstellten Reha-Nachrichten durch den Leistungserbringer abgeholt. Darüber hinaus können hier auch Fehlernachrichten, die aus Fehlern der DSRV-Verarbeitung resultieren, bereitgestellt werden. Im Request wird mittels der eXtra-Query die Sendungsnummer angegeben, die bereitgestellt werden soll.

Die DSRV erzeugt für jede von der Rentenversicherung bereitgestellte XML-Nachricht eine fortlaufende Sendungsnummer pro IK-Empfänger. Diese Sendungsnummer wird über das eXtra-Attribut ResponselD im Query angegeben.

Im Response wird immer nur eine XML-Nachricht übertragen.

Wenn mehrere Nachrichten abgeholt werden sollen, muss der Request entsprechend wiederholt gesendet werden mit angepasster Sendungsnummer.

Falls zur angegebenen Query keine Reha-Nachricht gefunden wird, wird im Response eine entsprechende Statusmeldung geliefert.

Procedure: <http://www.extra-standard.de/procedures/REHA301>

Data Type: <http://www.extra-standard.de/datatypes/DataRequest>

Scenario: <http://www.extra-standard.de/scenario/request-with-response>

Ein Beispiel eines Query-Request kann der Datei 02request.xml entnommen werden.

Als Antwort auf den Request bekommen sie von der Rentenversicherung eine Response, die entweder die Nutzdaten mit einer XML-Nachricht oder eine Statusmeldung beinhaltet.

Bei der Response ist die eXtra-Paketebene nicht vorgesehen. Die Nutzdaten werden als MTOM-Objekt gesendet.

Beispiele einer Antwort können den Dateien

- 02responseDaten.xml
- 02responseKeineDaten.xml
- 02responseFehlernachricht.xml

entnommen werden.

Beispiel:

Der Absender mit der SenderID 27 ruft das erste Mal Daten bei der DSRV ab. Für den initialen Abruf fragt er nach allen Sendungen die größer der ResponselD 0 sind.

Als erste Rückmeldung bekommt er eine Sendung mit der ResponselD 1. Diese kann als Grundlage für die weiteren Abfragen in nachfolgenden DataRequests genutzt werden.

Die folgenden Abfragen und Ergebnisse könnten dann wie folgt aussehen:

ResponselD >	ResponselD der gelieferten Sendung
0	1
1	2
2	3

usw.

Dieser Vorgang sollte dann so lange wiederholt werden, bis vom Server gemeldet wird, dass keine Daten mehr zur Abholung zur Verfügung stehen (Siehe auch StatusCode I51).

	XML-basierender Datenaustausch mit der Rentenversicherung im Verfahren Reha§301 Schnittstellenbeschreibung	Seite: 9 Version: 1.02.00 Stand: 09.03.18
--	--	--

Keine weitere Daten vorhanden

Im Fall, dass keine oder keine weiteren Daten mehr zum Abruf vorhanden sind, kann von der DSRV auch keine ResponseID in Form einer eindeutigen Kennung einer Sendung zurückgeliefert werden.

Da aber auch in diesem Fall vom eXTra-Schema eine ResponseID zwingend gefordert wird, wird die ResponseID dann mit einem Timestamp in Millisekunden befüllt.

2.6 Verwendung der ResponseID

Über die ResponseID lässt sich die Sendung im System DSRV eindeutig identifizieren. Dadurch wird zum Beispiel eine Sendungsverfolgung möglich und vereinfacht die Bearbeitung im Fehlerfall.

2.6.1 Hotline

Bei einer Anfrage bei der Hotline der DSRV durch den Versorgungsträger muss die ResponseID angegeben werden, um die betroffene Sendung zu identifizieren. Dadurch ist es möglich, in einem Fehlerfall die Ursache zu ermitteln und zu beheben. Die Kontaktdaten der Hotline können Sie dem Internetauftritt der DSRV unter www.dsrv.info entnehmen.

2.6.2 DataRequest

Die ResponseID wird im Geschäftsprozess [2.5.2 Abholen einer Reha-Nachricht von der Rentenversicherung](#) als Eingrenzungskriterium genutzt.

Dabei kann das Attribut GT für „greater then“ oder EQ für „equals“ zur Steuerung verwendet werden.

Sollte es sich hierbei um den initialen Request handeln, so ist die ResponseID „0“ und das Attribute GT anzugeben. Ansonsten kann die letzte übermittelte ResponseID als Kriterium bei GT genutzt werden.

Bei Verwendung von EQ kann eine bestimmte Nachricht mit ResponseID > 0 wiederholt abgeholt werden.

2.7 IK-Nummer der RV-Annahmestelle

Die DSRV übernimmt die Funktion der zentralen Annahmestelle für die Rentenversicherung. Im Header-Attribut ReceiverID ist daher beim Senden immer die IK-Nummer 110999007 der DSRV zu verwenden.

3 Teil B – Beschreibung der Elemente

3.1 Verwendete Namensräume und Präfixe

Innerhalb der eXTra-Kommunikation werden folgende Namensräume verwendet:

Namensraum	Präfix
http://www.extra-standard.de/namespace/webservice	xws
http://www.extra-standard.de/namespace/request/1	xreq
http://www.extra-standard.de/namespace/response/1	xres
http://www.extra-standard.de/namespace/components/1	xcpt
http://www.extra-standard.de/namespace/plugins/1	xplg
http://www.extra-standard.de/namespace/message/1	xmsg
http://www.w3.org/2001/XMLSchema	xs

3.2 Aufbau des Transport-Headers

Da der generelle Aufbau des Transport-Headers bei allen Anfragen identisch ist, wird er an dieser Stelle zentral beschrieben.

3.2.1 Transport-Header des Request

Auszug aus der Schema-Datei:

```

<xs:complexType name="TransportRequestHeaderType">
  <xs:complexContent>
    <xs:extension base="xcpt:ElementWithOptionalVersionType">
      <xs:sequence minOccurs="1" maxOccurs="1">
        <xs:element ref="xcpt:Sender" minOccurs="1" maxOccurs="1"/>
        <xs:element ref="xcpt:Receiver" minOccurs="1" maxOccurs="1"/>
        <xs:element ref="xcpt:RequestDetails" minOccurs="1" maxOccurs="1"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

Abb. 1 Aufbau Request-Header

3.2.1.1 Element TransportHeader

xreq:TransportHeader

<i>Inhalt</i>	Enthält die relevanten Steuerungsinformationen, die zwischen Sender und der DSRV als Empfänger auszutauschen sind
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xreq:Transport

3.2.1.2 Element Sender

xcpt:Sender

<i>Inhalt</i>	Enthält SenderID des Absenders
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xreq:TransportHeader

3.2.1.3 Element SenderID

xcpt:SenderID

<i>Inhalt</i>	IK-Nummer des Absender Diese muss mit der IK-Nummer des verwendeten Zertifikats übereinstimmen
<i>Datentyp</i>	xs:string

3.2.1.4 Element Receiver

xcpt:Receiver

<i>Inhalt</i>	Enthält ReceiverID des Empfängers
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xreq:TransportHeader

3.2.1.5 Element ReceiverID

xcpt:ReceiverID

<i>Inhalt</i>	IK-Nummer des Empfänger; muss immer IK-Nummer 110999007 der DSRV sein
<i>Datentyp</i>	xs:string
<i>Benutzt von</i>	xcpt:Receiver

3.2.1.6 Element RequestDetails

xcpt:RequestDetails

<i>Inhalt</i>	Diverse Request-spezifische Informationen
<i>Datentyp</i>	##elements

Benutzt von xreq:TransportHeader

3.2.1.7 Element RequestID

xcpt:RequestID

Inhalt ID des Requests, eindeutiger Begriff aus der Begriffswelt des Senders zur genauen Identifikation des Sendevorgangs
z.B. Auftragsnummer aus dem System des Senders

Datentyp xs:string

Benutzt von xcpt:RequestDetails

3.2.1.8 Element TimeStamp

xcpt:TimeStamp

Inhalt Ein Zeitstempel z.B. ‚2008-10-30T15:09:00‘ zum Beginn der Übertragung des Senders

Datentyp xs:dateTime

Benutzt von xcpt:RequestDetails

3.2.1.9 Element Procedure

xcpt:Procedure

Inhalt Der zulässige Inhalt ist in Teil A unter [2.5 Beschreibung der einzelnen Geschäftsprozesse](#) beschrieben.

Datentyp xs:anyURI

Benutzt von xcpt:RequestDetails

3.2.1.10 Element DataType

xcpt:DataType

Inhalt Mit dem DataType wird der jeweilige Geschäftsprozess bei der DSRV adressiert.

Der zulässige Inhalt ist in Teil A unter [2.5 Beschreibung der einzelnen Geschäftsprozesse](#) beschrieben.

Datentyp xs:anyURI

Benutzt von xcpt:RequestDetails

3.2.1.11 Element Scenario

xcpt:Scenario

Inhalt Mit dem optionalen Element Scenario wird die Art des Datenaustauschs zwischen Client und Server definiert.

Der zulässige Inhalt ist in Teil A unter [2.5 Beschreibung der einzelnen](#)

	XML-basierender Datenaustausch mit der Rentenversicherung im Verfahren Reha§301 Schnittstellenbeschreibung	Seite: 13 Version: 1.02.00 Stand: 09.03.18
--	--	---

[Geschäftsprozesse](#) beschrieben.

Datentyp xs:anyURI

Benutzt von xcpt:RequestDetails

3.2.2 Transport-Header der Response

Beim Transport Header der Response handelt es sich um eine Kopie des Request-Headers, die um die Informationen des Empfängers ergänzt wird.

```

<xs:complexType name="TransportResponseHeaderType">
  <xs:complexContent>
    <xs:extension base="xcpt:ElementWithOptionalVersionType">
      <xs:sequence minOccurs="1" maxOccurs="1">
        <xs:element ref="xcpt:Sender" minOccurs="1" maxOccurs="1"/>
        <xs:element ref="xcpt:Receiver" minOccurs="1" maxOccurs="1"/>
        <xs:element ref="xcpt:RequestDetails" minOccurs="1" maxOccurs="1"/>
        <xs:element ref="xcpt:ResponseDetails" minOccurs="1" maxOccurs="1"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

Abb. 2 Aufbau Transport-Header Response

3.2.2.1 Element TransportHeader

xres:TransportHeader

<i>Inhalt</i>	Enthält die relevanten Steuerungsinformationen, die zwischen Sender und der DSRV als Empfänger auszutauschen sind
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xres:Transport

Der Response Header ist nach eXtra-Philosophie eine Kopie des Request-Headers, den der Empfänger lediglich um die ResponseDetails ergänzt. Damit ist sichergestellt, dass beide Seiten alle Informationen in einer Datenstruktur finden, die ein Vorgang beim Sender und beim Empfänger auslöst.

3.2.2.2 Element Sender

xcpt:Sender

<i>Inhalt</i>	Enthält SenderID des Absenders, Original aus Request
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xres:TransportHeader

3.2.2.3 Element SenderID

xcpt:SenderID

Das Element wurde unter [3.2.1.3 Element SenderID](#) des Request-Headers beschrieben. Original aus Request.

3.2.2.4 Element Receiver

xcpt:Receiver

<i>Inhalt</i>	Enthält ReceiverID des Empfängers, Original aus Request
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xres:TransportHeader

3.2.2.5 Element ReceiverID

xcpt:ReceiverID

Das Element wurde unter [3.2.1.5 Element ReceiverID](#) des Request-Headers beschrieben. Original aus Request.

3.2.2.6 Element RequestDetails

xcpt:RequestDetails

<i>Inhalt</i>	Diverse Request-spezifische Informationen, Original aus Request
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xres:TransportHeader

3.2.2.7 Element RequestID

xcpt:RequestID

Das Element wurde unter [3.2.1.7 Element RequestID](#) des Request-Headers beschrieben. Original aus Request.

3.2.2.8 Element TimeStamp

xcpt:TimeStamp

Das Element wurde unter [3.2.1.8 Element TimeStamp](#) des Request-Headers beschrieben. Original aus Request.

3.2.2.9 Element Procedure

xcpt:Procedure

Das Element wurde unter [3.2.1.12 Element Procedure](#) des Request-Headers beschrieben. Original aus Request.

3.2.2.10 Element DataType

xcpt:DataType

Das Element wurde unter [3.2.1.13 Element DataType](#) des Request-Headers beschrieben. Original aus Request.

3.2.2.11 Element Scenario

xcpt:Scenario

Das Element wurde unter [3.2.1.14 Element Scenario](#) des Request-Headers beschrieben. Original aus Request.

3.2.2.12 Element ResponseDetails

xcpt:ResponseDetails

<i>Inhalt</i>	Diverse Response-spezifische Informationen, die die DSRV als Empfänger dem ursprünglichen Sender zur Verfügung stellt
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xres:TransportHeader

3.2.2.13 Element ResponseID

xcpt:ResponseID

<i>Inhalt</i>	<p>Beim Response des Sendevorgangs wird hier eine eindeutige Ticket-Nummer der DSRV geliefert.</p> <p>Beim Response des Abholvorgangs mit Daten wird hier eine eindeutige fortlaufende numerische Sendungsnummer geliefert, die von der DSRV für jeden IK-Empfänger fortgezählt wird. Diese Sendungsnummer wird im Query-Element „ResponseID“ bei der Abholung angegeben.</p>
<i>Datentyp</i>	xs:string
<i>Benutzt von</i>	xcpt:ResponseDetails

3.2.2.14 Element TimeStamp

xcpt:TimeStamp

<i>Inhalt</i>	Ein Zeitstempel z.B. ‚2015-10-30T15:09:00‘ der Erzeugung der Response durch die DSRV.
<i>Datentyp</i>	xs:dateTime
<i>Benutzt von</i>	xcpt:ResponseDetails

3.2.2.15 Element Report

xcpt:Report

<i>Inhalt</i>	Report zum Empfangsvorgang des abgeschickten Requests
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xcpt:ResponseDetails

@highestWeight

	XML-basierender Datenaustausch mit der Rentenversicherung im Verfahren Reha§301 Schnittstellenbeschreibung	Seite: 17 Version: 1.02.00 Stand: 09.03.18
--	--	---

Inhalt Höchste Gewichtung der Art des Reports
Datentyp xs:anyURI

Im unprofilieren eXTra-Schema ist es möglich im Report mehrere Flag-Elemente anzuführen, die jeweils einen eigenen Report beinhalten. Innerhalb dieses Verfahrens wird immer nur ein Report zurückgeliefert, weshalb die höchste Gewichtung immer der Gewichtung des Reports entspricht.

Mögliche Inhalte für highestWeight sind:

- <http://www.extra-standard.de/weight/INFO> wenn der Request angenommen oder verarbeitet werden konnten
- <http://www.extra-standard.de/weight/ERROR> wenn es bei der Verarbeitung des Requests zu einem Fehler gekommen ist.
- <http://www.extra-standard.de/weight/FATAL> wenn es bei der Verarbeitung des Requests zu einem schwerwiegendem technischen Fehler bei der DSRV gekommen ist. Der Request sollte zu späterem Zeitpunkt wiederholt werden.

3.2.2.16 Element Flag

xcpt:Flag

<i>Inhalt</i>	Attribut weight
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xcpt:Report

@weight

<i>Inhalt</i>	Gewichtung des Reports
<i>Datentyp</i>	xs:anyURI

Mögliche Inhalte siehe Attribut highestWeight

3.2.2.17 Element Code

xcpt:Code

<i>Inhalt</i>	Alphanummerischer Statuscode (Siehe Anhang StatusCodes)
<i>Datentyp</i>	xs:string
<i>Benutzt von</i>	xcpt:Report

3.2.2.18 Element Text

xcpt:Text

<i>Inhalt</i>	Text zum Statuscode
<i>Datentyp</i>	xs:string
<i>Benutzt von</i>	xcpt:Report

3.2.2.19 Element Originator

xcpt:Text

<i>Inhalt</i>	RequestID der ursprünglich gesendeten Klinik-Nachricht die bei der DSRV-Verarbeitung einen Fehler verursacht hat.
<i>Datentyp</i>	xs:string
<i>Benutzt von</i>	xcpt:Report

Anmerkung: Originator ist ein optionales Attribut und wird nur im Response beim Prozess: [Abholen einer Reha-Nachricht von der Rentenversicherung](#) im Fehlerfall geliefert.

Beispiel: Eine verschlüsselte Nachricht kann von der DSRV nicht entschlüsselt werden.

3.3 Aufbau des TransportPlugins

3.3.1 Aufbau des DataTransforms-Plugins

Im DataTransforms-Plugin wird die Verschlüsselung und Komprimierung der Nutzdaten beschrieben.

3.3.1.1 Element DataTransforms

xplg:DataTransforms

<i>Inhalt</i>	Enthält die Informationen, wie die Nutzdaten für den Transport aufbereitet wurden
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xreq:TransportPlugins
@version	
<i>Inhalt</i>	1.2
<i>Datentyp</i>	xs:string

3.3.1.2 Element Compression

xplg:Compression

<i>Inhalt</i>	Enthält die Informationen zur Komprimierung der Daten
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xplg:DataTransforms
@order	
<i>Inhalt</i>	1
<i>Datentyp</i>	xs:positiveInteger

3.3.1.3 Element Algorithm

xplg:Algorithm

<i>Inhalt</i>	Verwendete Kompression
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xplg:Compression
@id	
<i>Inhalt</i>	Bezeichnung der Komprimierung. Der zulässige Inhalt wird in Teil A unter 2.5 Beschreibung der einzelnen Geschäftsprozesse beschrieben.
<i>Datentyp</i>	xs:anyURI

3.3.1.4 Element InputData

xplg:InputData

<i>Inhalt</i>	Enthält die Größe der Nutzdaten vor der Komprimierung
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xplg:Encryption

@bytes

<i>Inhalt</i>	Dateigröße in Bytes
<i>Datentyp</i>	xs:nonNegativeInteger

3.3.1.5 Element Encryption

xplg:Encryption

<i>Inhalt</i>	Enthält die Verschlüsselungsinformation
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xplg:DataTransforms

@order

<i>Inhalt</i>	2
<i>Datentyp</i>	xs:positiveInteger

3.3.1.6 Element Algorithm

xplg:Algorithm

<i>Inhalt</i>	Verwendeter Verschlüsselungs-Algorithmus
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xplg:Encryption

@id

<i>Inhalt</i>	Bezeichnung der Verschlüsselung Der zulässige Inhalt wird in Teil A unter 2.4.3 Verschlüsselung / Komprimierung (DataTransforms) beschrieben.
<i>Datentyp</i>	xs:anyURI

3.3.1.7 Element OutputData

xplg:OutputData

<i>Inhalt</i>	Enthält die Größe der Nutzdaten nach der Verschlüsselung
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xplg:Encryption

@bytes

<i>Inhalt</i>	Dateigröße in Bytes
<i>Datentyp</i>	xs:nonNegativeInteger

3.4 Request: Senden einer Reha-Nachricht an die Rentenversicherung

```

<xs:complexType name="TransportRequestType">
  <xs:complexContent>
    <xs:extension base="xcpt:RootElementType">
      <xs:sequence minOccurs="1" maxOccurs="1">
        <xs:element ref="TransportHeader" minOccurs="1" maxOccurs="1"/>
        <xs:element ref="TransportPlugIns" minOccurs="1" maxOccurs="1"/>
        <xs:element ref="TransportBody" minOccurs="1" maxOccurs="1"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

Abbildung 2 – Aufbau Request: Senden einer Reha-Nachricht

3.4.1 Element Transport

@version

<i>Inhalt</i>	1.4
<i>Datentyp</i>	xs:string

@profile

<i>Inhalt</i>	siehe 2.4.1 Profile-Attribut des Root-Elements
<i>Datentyp</i>	xs:anyURI

3.4.2 Element TransportHeader

xreq:TransportHeader

<i>Inhalt</i>	Die Elemente und der Aufbau des Transport-Headers werden im Abschnitt „Transport-Header des Request“ beschrieben
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xreq:Transport

3.4.3 Element TransportPlugIns

xreq:TransportPlugIns

<i>Inhalt</i>	Enthält die unter 3.3 Aufbau des TransportPlugIns beschriebenen PlugIns.
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xreq:Transport

3.4.4 Element TransportBody

xreq:TransportBody

<i>Inhalt</i>	Enthält den Body der Transportebene des Requests
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xreq:Transport

3.4.5 Element Data

xcpt:Data

<i>Inhalt</i>	Enthält die fachlichen Daten gemäß den Angaben in den RequestDetails procedure und datatype. Die Nutzdaten werden wie im Abschnitt 2.4.2 Nutzdatenbeschreibung (DataSource) und 2.4.3 Verschlüsselung / Komprimierung (DataTransforms) beschrieben behandelt und dann Base64-kodiert versendet.
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xreq:TransportBody

3.4.6 Element Base64CharSequence

xcpt:Base64CharSequence

<i>Inhalt</i>	Referenz auf die CID des MTOM-Anhangs, in dem die Nutzdaten übertragen werden. Die Referenzangaben müssen nicht in Base64-Kodiert werden.
<i>Datentyp</i>	xs:base64Binary
<i>Benutzt von</i>	xcpt:Data

3.5 Response: Senden einer Reha-Nachricht an die Rentenversicherung

```

<xs:complexType name="TransportResponseType">
  <xs:complexContent>
    <xs:extension base="xcpt:RootElementType">
      <xs:sequence minOccurs="1" maxOccurs="1">
        <xs:element ref="TransportHeader" minOccurs="1" maxOccurs="1"/>
        <xs:element ref="TransportBody" minOccurs="1" maxOccurs="1"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

Abbildung 3 - Aufbau Response zu 3.4

3.5.1 Element Transport

@version

<i>Inhalt</i>	1.4
<i>Datentyp</i>	xs:string

@profile

<i>Inhalt</i>	siehe 2.4.1 Profile-Attribut des Root-Elements
<i>Datentyp</i>	xs:anyURI

3.5.2 Element TransportHeader

xres:TransportHeader

<i>Inhalt</i>	Enthält die relevanten Steuerungsinformationen, die zwischen Sender und der DSRV als Empfänger auszutauschen sind. Der Aufbau ist unter „Transport-Header der Response“ beschrieben.
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xres:Transport

Der Response Header ist nach eXtra-Philosophie eine Kopie des RequestHeaders, den der Empfänger lediglich um die ResponseDetails ergänzt. Damit ist sichergestellt, dass beide Seiten alle Informationen in einer Datenstruktur finden, die ein Vorgang beim Sender und beim Empfänger auslöst.

3.5.3 Element TransportBody

xres:TransportBody

<i>Inhalt</i>	Enthält einen leeren Body der Transportebene
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xres:Transport

Die Response des Empfängers auf einen Sendevorgang, enthält nur eine technische Bestätigung, jedoch keine fachlichen Daten des Fachverfahrens. Deshalb ist das Element TransportBody leer.

3.6 Request: Abholen einer Reha-Nachricht von der Rentenversicherung

```

<xs:complexType name="TransportRequestType">
  <xs:complexContent>
    <xs:extension base="xcpt:RootElementType">
      <xs:sequence minOccurs="1" maxOccurs="1">
        <xs:element ref="TransportHeader" minOccurs="1" maxOccurs="1"/>
        <xs:element ref="TransportBody" minOccurs="1" maxOccurs="1"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

Abbildung 4 - Aufbau Request: Abholen einer Reha-Nachricht

3.6.1 Element Transport

@version	
<i>Inhalt</i>	1.4
<i>Datentyp</i>	xs:string

@profile	
<i>Inhalt</i>	siehe 2.4.1 Profile-Attribut des Root-Elements
<i>Datentyp</i>	xs:anyURI

3.6.2 Element TransportHeader

xreq:TransportHeader

<i>Inhalt</i>	Enthält die relevanten Steuerungsinformationen, die zwischen Sender und der DSRV als Empfänger auszutauschen sind. Der Aufbau des Transport-Header ist unter „Transport-Header des Request“ beschrieben.
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xreq:Transport

3.6.3 Element TransportBody

xreq:TransportBody

<i>Inhalt</i>	Enthält den Body der Transportebene eines Requests
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xreq:Transport

Der TransportBody enthält beim Request keine fachlichen Daten sondern lediglich eine Query zur Identifikation der abzuholenden Nachrichten.

3.6.4 Element Data

xcpt:Data

<i>Inhalt</i>	Enthält die Query
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xreq:TransportBody

3.6.5 Element ElementSequence

xcpt:ElementSequence

<i>Inhalt</i>	Enthält das Element DataRequest
<i>Datentyp</i>	xs:any
<i>Benutzt von</i>	xcpt:Data

3.6.6 Element DataRequest

xmsg:DataRequest

<i>Inhalt</i>	Enthält die Elemente der Query
<i>Datentyp</i>	xs:string
<i>Benutzt von</i>	xcpt:ElementSequence

@version

<i>Inhalt</i>	1.3
<i>Datentyp</i>	xs:string

3.6.7 Element Query

xmsg:Query

<i>Inhalt</i>	Enthält Parameter zur Einschränkung des Abholergebnisses.
<i>Datentyp</i>	xs:string
<i>Benutzt von</i>	xmsg:DataRequest

3.6.8 Element Argument

xmsg:Argument

<i>Inhalt</i>	Enthält die Parameter, nach denen die Query aufgebaut werden soll
<i>Datentyp</i>	xs:string
<i>Benutzt von</i>	xmsg:Query

@property

<i>Inhalt</i>	Enthält eine URI die definiert welche Art von Suchbegriff definiert wird
<i>Datentyp</i>	xs:anyURI

Zulässiger Wert:

<http://www.extra-standard.de/property/ResponseID>

@type

<i>Inhalt</i>	xs:string
<i>Datentyp</i>	xs:string

Die Auswahl des Arguments wird wie folgt eingeschränkt:

xmsg:EQ bzw. xmsg:GT

<i>Inhalt</i>	Elemente zum Einschränken der Auswahl
<i>Datentyp</i>	xs:string
<i>Benutzt von</i>	xmsg:Argument

Die genaue Einschränkung wird in dem [Geschäftsprozess für die Abholung der Daten beschrieben](#).

3.7 Response: Abholen einer Reha-Nachricht von der Rentenversicherung

```

<xs:complexType name="TransportResponseType">
  <xs:complexContent>
    <xs:extension base="xcpt:RootElementType">
      <xs:sequence minOccurs="1" maxOccurs="1">
        <xs:element ref="TransportHeader" minOccurs="1" maxOccurs="1"/>
        <xs:element ref="TransportPlugIns" minOccurs="0" maxOccurs="1"/>
        <xs:element ref="TransportBody" minOccurs="1" maxOccurs="1"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

Abbildung 5 - Aufbau Response zu 3.6

3.7.1 Element Transport

	@version
<i>Inhalt</i>	1.4
<i>Datentyp</i>	xs:string
	@profile
<i>Inhalt</i>	siehe 2.4.1 Profile-Attribut des Root-Elements
<i>Datentyp</i>	xs:anyURI

3.7.2 Element TransportHeader

	xres:TransportHeader
<i>Inhalt</i>	Enthält die relevanten Steuerungsinformationen, die zwischen Sender und der DSRV als Empfänger auszutauschen sind. Der Aufbau ist unter „Transport-Header der Response“ beschrieben.
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xres:Transport

Der Response Header ist nach eXtra-Philosophie eine Kopie des RequestHeaders, den der Empfänger lediglich um die ResponseDetails ergänzt. Damit ist sichergestellt, dass beide Seiten alle Informationen in einer Datenstruktur finden, die ein Vorgang beim Sender und beim Empfänger auslöst.

3.7.3 Element TransportPlugins

	xreq:TransportPlugins
<i>Inhalt</i>	Enthält die unter 3.3 Aufbau des TransportPlugins beschriebenen Plugins.
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xreq:Transport

3.7.4 Element **TransportBody**

xres:TransportBody

Inhalt Enthält Body der Transportebene

Datentyp ##elements

Benutzt von xres:Transport

3.7.5 Element Data

xcpt:Data

Inhalt Enthält die fachlichen Daten, welche das Fachverfahren bei der Verarbeitung der ursprünglichen Sendung erzeugt.
 Die Daten werden wie unter „[2.4.3 Verschlüsselung / Komprimierung \(DataTransofrms\)](#)“ beschrieben ausgeliefert.

Datentyp ##elements

Benutzt von xreq:PackageBody

3.7.6 Element Base64CharSequence

xcpt:Base64CharSequence

Inhalt Base64-Zeichenfolge

Datentyp xs:base64Binary

Benutzt von xcpt:Data

Referenz auf die CID des MTOM-Anhangs, in dem die Nutzdaten übertragen werden.

4 Anhang

4.1 StatusCodes

Statuscode	Text	Erläuterung	
E01	Steuerdaten unvollständig	Die übermittelten Steuerdaten des eXTra-Requests waren falsch oder unvollständig. Die Sendung wurde nicht gespeichert	E
E02	Nutzdaten unvollständig	Es wurden keine Nutzdaten übermittelt	E
E03	Annahme fehlgeschlagen	Die Annahme der Daten ist aus technischen Gründen fehlgeschlagen. Bitte probieren Sie es später noch einmal oder wenden sich an die Hotline Die Sendung wurde nicht gespeichert	E
E04	Fehler beim Entschlüsseln der Daten	Das Entschlüsseln der Daten war nicht erfolgreich. Diese Fehlermeldung wird beim Abholprozess mitgeteilt, da die Entschlüsselung asynchron zum Sendeprozess erfolgt. Im Attribut „Originator“ wird Bezug auf die RequestId der Ursprungsnachricht hergestellt.	E

E05	Fehler beim Entkomprimieren der Daten	Das Entkomprimieren der Daten war nicht erfolgreich. Diese Fehlermeldung wird beim Abholprozess mitgeteilt, da die Entkomprimierung asynchron zum Sendeprozess erfolgt. Im Attribut „Originator“ wird Bezug auf die RequestId der Ursprungsnachricht hergestellt.	E
E50	Anfrage unvollständig	Die Anfrage-Informationen waren unvollständig oder falsch. Eine Abfrage auf die Datenbank konnte nicht durchgeführt werden.	E
E84	Variabel	Die Annahme der Daten ist fehlgeschlagen. Dies kann verschiedene Gründe haben. Bitte den entsprechenden Fehlertext beachten.	F
I01	Daten erfolgreich gespeichert	Die Daten und Steuerinformationen entsprechen den Anforderungen und wurden erfolgreich bei der DSRV gespeichert	I
I50	Anfrage erfolgreich	Die Anfrage bei der DSRV mit den übermittelten Parametern war erfolgreich. Der Datensatz wird in der Response ausgeliefert	I
I51	Anfrage erfolgreich – Keine Daten vorhanden	Die Anfrage bei der DSRV mit den übermittelten Parametern war erfolgreich. Es konnte aber kein Datensatz zu diesen Parametern gefunden werden	I

Legende: I und E stehen für die Gewichtung im Report: I(nfo) und E(rror)

4.2 Referenzierte Dokumente

Name des Dokuments	Beschreibung
Beispiele.zip	Zip-Datei mit Beispielen für die eXTra – Kommunikation
WSDL.zip	Zip-Datei mit der Webservice WSDL und dem eXTra-Schema V1.3
Schema.zip	Zip-Datei mit dem profiliertem eXTra-Schema für Reha§301

4.3 Abbildungsverzeichnis

Abbildung 1 – Geschäftsprozesse	5
Abbildung 2 – Aufbau Request: Senden einer Reha-Nachricht	21
Abbildung 3 - Aufbau Response zu 3.4	22
Abbildung 4 - Aufbau Request: Abholen einer Reha-Nachricht	23
Abbildung 5 - Aufbau Response zu 3.6	26