

<eXTra>

einheitliches XML-basiertes Transportverfahren



Schnittstellenspezifikation zur elektronischen Anfrage von Rentenbescheid- daten

Version 2.05.00

Stand der Spezifikation: 20.05.2022
Version: 2.05.00
Redaktion: Deutsche Rentenversicherung Bund
Referat 0551
Berner Strasse 1
97084 Würzburg
Verfahrensverantwortlich: 0551-Behoerden-Gerichte@drv-bund.de

0 Allgemeines

Das vorliegende Dokument dient als Schnittstellenbeschreibung zur eXTra-Kommunikation, die bei der Datenstelle der Träger der Rentenversicherung (DSRV) für die elektronische Anfrage von Rentenbescheidaten mit den Versorgungsträgern eingesetzt wird

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	26.04.2017	Alle	Registrierung des Dokuments	DSRV
2.00.00	30.10.2018	Alle	Aktualisierung der Spezifikation	DSRV
2.01.00	14.03.2019	Alle	Aktualisierung der Spezifikation	DSRV
2.02.00	22.05.2020		Dokumentation (Kopfzeile)	DSRV
2.03.00	05.06.2020	3.4.5	Base64Sequence	DSRV
2.04.00	30.07.2021	3.11	StatusCodes – SendData E74 (doppelte UUID) wird als Fachfehler ins FDM aufgenommen	DSRV
2.05.00	09.05.2022	2.2	Änderung KommServer-URL	DSRV

1 Inhaltsverzeichnis

0 Allgemeines	2
1 Inhaltsverzeichnis	3
2 Teil A – Allgemeine Informationen	6
2.1 Grafischer Überblick Geschäftsprozesse	6
2.2 Server-Adressen.....	6
2.3 Authentifizierung.....	6
2.4 Verwendetes eXTra-Schema.....	7
2.4.1 Profile-Attribut des Root-Elements	7
2.4.2 Nutzdatenbeschreibung (DataSource).....	7
2.4.3 Verschlüsselung / Komprimierung (DataTransforms)	7
2.5 Beschreibung der einzelnen Geschäftsprozesse	8
2.5.1 Senden Anfrage von Rentenbescheidaten	8
2.5.2 Abholen der Rentenbescheidaten	8
2.5.3 Bestätigung der abgeholten Rentenbescheidaten	9
2.6 Verwendung der ResponseID.....	9
2.6.1 Hotline.....	9
2.6.2 DataRequest	9
2.6.3 ConfirmationOfReceipt	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
3.2.3 Package-Header der Response	19
3.3 Aufbau des TransportPlugins..... Fehler! Textmarke nicht definiert.	
3.3.1 Aufbau des DataTransforms-Plugins.....	20
3.4 Request Senden Anfrage von Rentenbescheidaten.....	22
3.4.1 Element Transport.....	22
3.4.2 Element TransportHeader	22
3.4.3 Element TransportPlugins	22
3.4.4 Element TransportBody.....	23
3.4.5 Element Data	23

3.4.6	Element Base64CharSequence	23
3.5	Response Senden Anfrage von Rentenbescheidaten.....	23
3.5.1	Element Transport.....	23
3.5.2	Element TransportHeader	23
3.5.3	Element TransportBody.....	24
3.6	Request Abholen der Rentenbescheidaten.....	24
3.6.1	Element Transport.....	24
3.6.2	Element TransportHeader	25
3.6.3	Element TransportBody.....	25
3.6.4	Element Data	25
3.6.5	Element ElementSequence	25
3.6.6	Element DataRequest	25
3.6.7	Element Query	26
3.6.8	Element Argument.....	26
3.6.9	Element Argument.....	26
3.6.10	Element Control.....	27
3.6.11	Element MaximumPackages	27
3.7	Response Abholen der Rentenbescheidaten	28
3.7.1	Element Transport.....	28
3.7.2	Element TransportHeader	28
3.7.3	Element TransportBody.....	29
3.7.4	Element Package	29
3.7.5	Element PackageHeader.....	29
3.7.6	Element PackagePlugins.....	29
3.7.7	Element PackageBody	30
3.7.8	Element Data	30
3.7.9	Element Base64CharSequence	30
3.8	Request Bestätigung der abgeholten Rentenbescheidaten.....	30
3.8.1	Element Transport.....	30
3.8.2	Element TransportHeader	31
3.8.3	Element TransportBody.....	31
3.8.4	Element Data	31
3.8.5	Element ElementSequence	31
3.8.6	Element ConfirmationOfReceipt	31
3.8.7	Element PropertySet	32
3.8.8	Element Value.....	32
3.9	Response Bestätigung der abgeholten Rentenbescheidaten.....	32

3.9.1	Element Transport.....	32
3.9.2	Element TransportHeader	33
3.9.3	Element TransportBody.....	33
4	Anhang.....	34
4.1	StatusCodes	Fehler! Textmarke nicht definiert.
4.2	Referenzierte Dokumente	34
4.3	Abbildungsverzeichnis	36

2 Teil A – Allgemeine Informationen

2.1 Grafischer Überblick Geschäftsprozesse

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

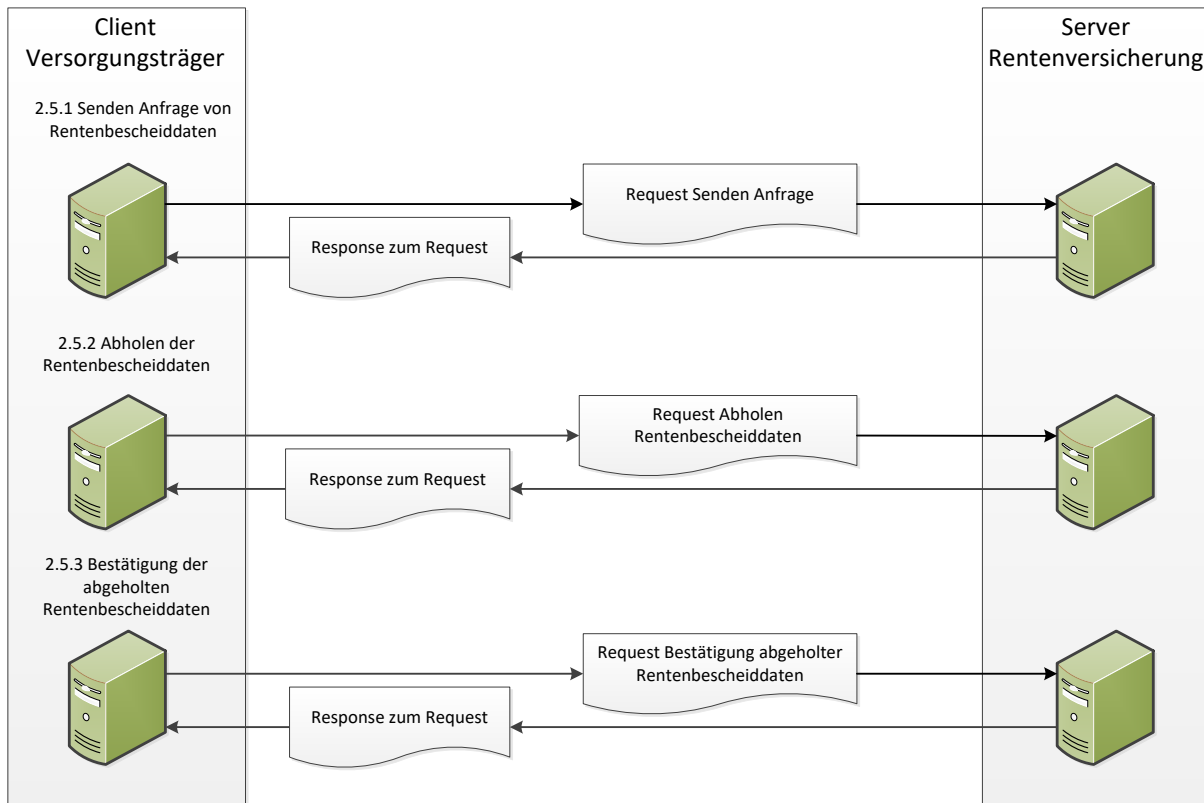


Abb. 1 Grafischer Ablauf der Kommunikation

2.2 Server-Adressen

Bei den Test- und Produktionsservern der DSRV handelt es sich um physikalisch getrennte Server-Systeme.

Aus diesem Grund werden unterschiedliche Adressen verwendet.

Testsystem:

agse.eservicet-drv.de

Produktionssystem:

agse.eservice-drv.de

2.3 Authentifizierung

Auf beiden Systemen wird zur Client-Authentifizierung das Versorgungsträgerzertifikat verwendet. 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-Basisschema in der Version 1.4.

2.4.1 Profile-Attribut des Root-Elements

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

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

2.4.2 Nutzdatenbeschreibung (DataSource)

Die Nutzdaten werden immer als UTF-8 im Zeichenvorrat ISO 8859-1 codiert. Deshalb entfällt das DataSource-Plugin.

2.4.3 Verschlüsselung / Komprimierung (DataTransforms)

Gemäß [KKS](#) müssen die verschlüsselten Daten vom Versorgungsträger (Absender) signiert werden.

Die fachlichen Nutzdaten in den Geschäftsprozessen werden mit dem PKCS7 verschlüsselt. Hierbei werden die Nutzdaten mit dem öffentlichen Zertifikat des Empfängers verschlüsselt.

Zulässige Werte für eXTra Element Algorithm sind:

Wert
http://www.extra-standard.de/transforms/encryption/PKCS7

	Elektronische Anfrage von Rentenbescheidaten Schnittstellenbeschreibung	Seite: 8 Version: 2.05.00 Stand: 20.05.2022
--	---	--

2.4.3.1.1 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 Daten. Die jeweilige, zugelassene Komprimierung wird in den Geschäftsprozessen beschrieben
2. Verschlüsselung der Daten mit PCKS7

2.5 Beschreibung der einzelnen Geschäftsprozesse

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

2.5.1 Senden Anfrage von Rentenbescheidaten

Dieser Geschäftsprozess dient zum Senden von „Anfrage von Rentenbescheidaten“ durch den Versorgungsträger an die Rentenversicherung.

Procedure: SendeAV

DataType: <http://www.extra-standard.de/datatypes/SendData>

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

Komprimierung: <http://www.extra-standard.de/transforms/compression/GZIP>

Beim Senden von „Anfrage von Rentenbescheidaten“ ist die Paketebene nicht vorgesehen. Die Anfrage von Rentenbescheidaten werden in dem Element Base64CharSequence des TransportBodys übermittelt.

Ein Beispiel XML kann der Datei 01-request.xml entnommen werden.

2.5.2 Abholen der Rentenbescheidaten

In diesem Prozess werden die Rentenbescheidaten, berechnet durch die Rentenversicherung, durch den Versorgungsträger abgeholt.

Procedure: HolenAV

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

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

Die Rentenbescheidaten sind in der Paketebene enthalten.

Für jedes Paket in der Paketebene wird eine eigene ResponseID vergeben. Diese ResponseID muss im Bestätigungsprozess verwendet werden.

Das XML-Element Scenario dient der Beschreibung der Kommunikationsart (request-with-response).

Die Anzahl der Rentenbescheidaten kann mit dem Element Control gesteuert werden. Ist dieses Element nicht mit angegeben, werden maximal die 10 ältesten Berechnungen von Rentenbescheidaten an den anfragenden Versorgungsträger ausgeliefert.

Ein Beispiel XML kann der Datei 02-request.xml entnommen werden.

	Elektronische Anfrage von Rentenbescheididaten Schnittstellenbeschreibung	Seite: 9 Version: 2.05.00 Stand: 20.05.2022
--	---	--

2.5.3 Bestätigung der abgeholten Rentenbescheididaten

Procedure: HolenAV

DataType: <http://www.extra-standard.de/datatypes/ConfirmationOfReceipt>

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

Das XML-Element Scenario dient der Beschreibung Kommunikationsart (request-with-acknowledgement).

Ein Beispiel XML kann der Datei 03-request.xml entnommen werden.

Solange die abgerufenen Daten nicht bestätigt worden sind, können sie beliebig oft abgerufen werden.

2.6 Verwendung der ResponseID

Bei der ResponseID handelt es sich um eine eindeutige, fortlaufende und numerische Ticketnummer der Datenstelle der Rentenversicherung (DSRV).

2.6.1 Hotline

Bei einer Anfrage bei der Hotline der DSRV durch den Versorgungsträger muss die ResponseID angegeben werden, um den betroffenen Übermittlungsvorgang zu identifizieren. Dadurch ist es möglich, in einem durch das eXTra Empfangssystem festgestellten Fehlerfall die Ursache zu ermitteln und zu beheben.

2.6.2 DataRequest

Die ResponseID wird im Geschäftsprozess [2.5.2 Abholen der Rentenbescheididaten](#) als Eingrenzungskriterium genutzt. Sollte es sich hierbei um den initialen Request handeln, so ist die ID als 0 anzugeben. Ansonsten kann die ResponseID des letzten erfolgreich übermittelten Paketes als Kriterium genutzt werden.

2.6.3 ConfirmationOfReceipt

Die ResponseID eines Paketes eines erfolgreichen Geschäftsprozesses [2.5.2](#) wird im Geschäftsprozess [2.5.3 Bestätigung der abgeholten Rentenbescheididaten](#) als Bestätigungskriterium verwendet. Durch die Übermittlung der ResponseID in diesem Prozess werden die Anforderungen im Fachdienst als „abgeholt“ gekennzeichnet, so dass bei einer erneuten Abholung durch den Versorgungsträger die bestätigten Anforderungen nicht erneut ausgeliefert werden.

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/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. 2 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>	Betriebsnummer des Absender Diese muss mit der Betriebsnummer 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>	Betriebsnummer Empfänger; muss <i>immer</i> Betriebsnummer 66667777 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. ‚2018-10-30T15:09:00‘ zum Beginn der Übertragung des Senders

Datentyp xs:dateTime

Benutzt von xcpt:RequestDetails

3.2.1.9 Element Application

xcpt:Application

Inhalt Enthält Product und Manufacturer

Datentyp ##elements

Benutzt von xcpt:RequestDetails

3.2.1.10 Element Product

xcpt:Product

Inhalt Bezeichnung eines (Software-) Produkts des Senders

Datentyp xs:string

Benutzt von xcpt:Application

3.2.1.11 Element Manufacturer

xcpt:Manufacturer

Inhalt Herstellerbezeichnung des Software-Produktes des Senders

Datentyp xs:string

Benutzt von xcpt:Application

3.2.1.12 Element Procedure

xcpt:Procedure

<i>Inhalt</i>	Der zulässige Inhalt ist in Teil A unter 2.5 Beschreibung der einzelnen Geschäftsprozesse beschrieben.
<i>Datentyp</i>	xs:anyURI
<i>Benutzt von</i>	xcpt:RequestDetails

3.2.1.13 Element DataType

xcpt:DataType

<i>Inhalt</i>	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.
<i>Datentyp</i>	xs:anyURI
<i>Benutzt von</i>	xcpt:RequestDetails

3.2.1.14 Element Scenario

xcpt:Scenario

<i>Inhalt</i>	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 Geschäftsprozesse beschrieben.
<i>Datentyp</i>	xs:anyURI
<i>Benutzt von</i>	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. 3 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 Application

xcpt:Application

Das Element wurde unter [3.2.1.9 Element Application](#) des Request-Headers beschrieben. Original aus Request.

3.2.2.10 Element Product

xcpt:Product

Das Element wurde unter [3.2.1.10 Element Product](#) des Request-Headers beschrieben. Original aus Request.

3.2.2.11 Element Manufacturer

xcpt:Manufacturer

Das Element wurde unter [3.2.1.11 Element Manufacturer](#) des Request-Headers beschrieben. Original aus Request.

3.2.2.12 Element Procedure

xcpt:Procedure

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

3.2.2.13 Element DataType

xcpt:DataType

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

3.2.2.14 Element Scenario

xcpt:Scenario

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

3.2.2.15 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

3.2.2.16 Element ResponseID

xcpt:ResponseID

<i>Inhalt</i>	Eindeutige fortlaufende numerische Meldungsnummer aus der DSRV Monitordatenbank, die den Vorgang beim Empfänger eindeutig identifiziert. Diese ID erleichtert die Suche und Nachvollziehbarkeit der Sendung während der Verarbeitung.
---------------	---

Datentyp `xs:string`
Benutzt von `xcpt:ResponseDetails`

3.2.2.17 Element TimeStamp

xcpt:TimeStamp

Inhalt Ein Zeitstempel z.B. ‚2018-10-30T15:09:00‘, der das Eingangsdatum beim Empfänger repräsentiert.

Datentyp `xs:dateTime`

Benutzt von `xcpt:ResponseDetails`

3.2.2.18 Element Report

xcpt:Report

Inhalt Report zum Empfangsvorgang des abgesendeten Requests

Datentyp `##elements`

Benutzt von `xcpt:ResponseDetails`

@highestWeight

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 konnte

<http://www.extra-standard.de/weight/ERROR> wenn es bei der Verarbeitung des Requests zu einem Fehler gekommen ist

3.2.2.19 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

3.2.2.20 Element Code

xcpt:Code

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

3.2.2.21 Element Text

xcpt:Text

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

3.2.3 Package-Header der Response

Der Aufbau des PackageHeaders der Response unterscheidet sich nicht vom Aufbau des Transport-Headers der Response.

Die Inhalte der Elemente beziehen sich innerhalb dieses Headers auf die bereitgestellte Antwort des Fachverfahrens, die im Package ausgeliefert wird und nicht auf die Response als Ganzes.

Es werden aber dieselben Werte verwendet wie beim Transport-Header.

```

<xs:complexType name="PackageResponseHeaderType">
  <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. 4 Aufbau Package-Header

Folgende Anmerkungen sollten jedoch beachtet werden.

3.2.3.1 ResponseID

Bei der ResponseID handelt es sich um die ID der bereitgestellten fachlichen Antwort, die im jeweiligen Package ausgeliefert wird und nicht um die ResponseID für den gesamten Request.

3.2.3.2 Timestamp

Beim Timestamp handelt es sich um den Verarbeitungs-Zeitstempel der bereitgestellten fachlichen Antwort.

3.2.3.3 Report

Im Report der Package Ebene werden nur Fehler angezeigt, die der eXtra Server beim Bereitstellen der fachlichen Antwort festgestellt hat und nicht der Status des Abrufs von Anfragen (dies würde auf der Transport Ebene gemeldet werden).

3.3 Plugins auf Transport- und Paket Ebene

3.3.1 Aufbau des DataTransforms-Plugins

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

Hinweis: das Plugin DataTransforms wird beim Sendeprozess 2.5.1 auf Transport Ebene verwendet und bei der Response des Holprozesses 2.5.2 auf Paket Ebene.

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

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 Anfrage von Rentenbescheidaten

```

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

```

Abb. 5 Aufbau Request Anfrage Versorgungsträger

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 2.4.3 Verschlüsselung / Komprimierung (DataTransforms) beschrieben behandelt und dann Base64-codiert versendet.
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xreq:TransportBody

Inhalt des Elementes Base64Sequence sind die verschlüsselten und base64-codierten Rentenbescheidaten, welche bei der DSRV zur Abholung durch den Versorgungsträger bereitgestellt wird.

3.4.6 Element Base64CharSequence

xcpt:Base64CharSequence

<i>Inhalt</i>	Base64-Zeichenfolge der verschlüsselten Nutzdaten
<i>Datentyp</i>	xs:base64Binary
<i>Benutzt von</i>	xcpt:Data

3.5 Response Senden Anfrage von Rentenbescheidaten

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 der Rentenbescheidaten

```

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

```

Abb. 6 Aufbau Request Datenabholung

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

Datentyp `xs:anyURI`

3.6.2 Element **TransportHeader**

xreq:TransportHeader

Inhalt 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.

Datentyp `##elements`

Benutzt von `xreq:Transport`

3.6.3 Element **TransportBody**

xreq:TransportBody

Inhalt Enthält den Body der Transportebene eines Requests

Datentyp `##elements`

Benutzt von `xreq:Transport`

Der **TransportBody** enthält beim Request keine fachlichen Daten sondern die eXTra Standardnachricht **DataRequest** mit einer Query zur Anforderung der Anfragen.

3.6.4 Element **Data**

xcpt>Data

Inhalt Enthält die Query

Datentyp `##elements`

Benutzt von `xreq:TransportBody`

3.6.5 Element **ElementSequence**

xcpt:ElementSequence

Inhalt Enthält das Element **DataRequest**

Datentyp `xs:any`

Benutzt von `xcpt>Data`

3.6.6 Element **DataRequest**

xmsg>DataRequest

Inhalt Enthält die Elemente der Query

Datentyp `xs:string`

Benutzt von `xcpt:ElementSequence`

@version

Inhalt 1.3
Datentyp xs:string

3.6.7 Element Query

xmsg:Query

Inhalt Kann eine Reihe von Parametern zur Einschränkung der Arbeitsergebnisse enthalten.
Datentyp xs:string
Benutzt von xmsg:DataRequest

Innerhalb des Query-Elements sind 1-n Argument-Elemente zulässig

3.6.8 Element Argument

xmsg:Argument

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

@property

Inhalt Enthält eine URI die definiert welche Art von Suchbegriff definiert wird
Datentyp xs:anyURI

Zulässig sind hierbei:

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

xmsg:EQ

Inhalt Elemente zum Einschränken der Auswahl
Datentyp xs:string
Benutzt von xmsg:Argument

Die genaue Einschränkung wird in dem Geschäftsprozess für die Abholung der Daten beschrieben.

3.6.9 Element Argument

xmsg:Argument

Inhalt Enthält die Parameter nach denen die Query aufgebaut werden soll
Datentyp xs:string
Benutzt von xmsg:Query

@property

Inhalt Enthält eine URI die definiert welche Art von Suchbegriff definiert wird

Datentyp `xs:anyURI`

Zulässig sind hierbei:

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

xmsg:GT

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

Zulässig sind folgende Inhalte

Wert	Beschreibung
ResponseID	Siehe Teil A 2.6 Verwendung der ResponseID

3.6.10 Element Control

xmsg:Control

<i>Inhalt</i>	Optionales Element dass die maximale Anzahl der auszuliefernden Pakete enthält
<i>Datentyp</i>	<code>xs:string</code>
<i>Benutzt von</i>	<code>xmsg:DataRequest</code>

3.6.11 Element MaximumPackages

xmsg:MaximumPackages

<i>Inhalt</i>	Anzahl der maximal zu liefernden Pakete innerhalb der Rückmeldung
<i>Datentyp</i>	<code>xs:positiveInteger</code>
<i>Benutzt von</i>	<code>xmsg:Control</code>

Sollte das Element Control mit MaximumPackages fehlen, so werden maximal 10 Pakete pro Anfrage an die anfordernde Stelle zurückgeliefert.

3.7 Response Abholen der Rentenbescheidaten

```

<xs:complexType name="TransportResponseBodyType">
  <xs:complexContent>
    <xs:extension base="xcpt:ElementWithOptionalVersionType">
      <xs:choice minOccurs="0">
        <xs:element ref="Package" maxOccurs="unbounded"/>
      </xs:choice>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="PackageResponseType">
  <xs:complexContent>
    <xs:extension base="xcpt:ElementWithOptionalVersionType">
      <xs:sequence>
        <xs:element ref="PackageHeader"/>
        <xs:element ref="PackagePlugins" minOccurs="0"/>
        <xs:element ref="PackageBody"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

Abb. 7 Aufbau Response Datenabholung

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 TransportBody

xres:TransportBody

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

Die Anzahl der Packages im TransportBody können über das im [Kapitel 3.6.10](#) beschriebenen Element Control gesteuert werden. Ist das Element nicht angegeben, so werden maximal 10 Packages ausgeliefert. Der TransportBody bleibt leer, wenn keine Daten vorhanden sind

3.7.4 Element Package

xres:Package

<i>Inhalt</i>	Ein Package enthält eine komplette Berechnung der Rentenbescheidaten (fachliche Daten) des Fachverfahrens als XML.
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xres:TransportBody

3.7.5 Element PackageHeader

xres:PackageHeader

<i>Inhalt</i>	Enthält den Header der Paketebene. Der Aufbau des PackageHeader entspricht dem Aufbau des TransportHeader, der Inhalt des PackageHeader ist unter „Package-Header der Response“ beschrieben.
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xres:Package

3.7.6 Element PackagePlugins

xres:PackagePlugins

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

3.7.7 Element PackageBody

xres:PackageBody

<i>Inhalt</i>	Enthält die fachlichen Daten, welche das Fachverfahren bei der Verarbeitung der ursprünglichen Sendung erzeugt.
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xres:Package

3.7.8 Element Data

xcpt:Data

<i>Inhalt</i>	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) “ verschlüsselt und unkomprimiert an den Versorgungsträger ausgeliefert
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xreq:PackageBody

3.7.9 Element Base64CharSequence

xcpt:Base64CharSequence

<i>Inhalt</i>	Base64-Zeichenfolge
<i>Datentyp</i>	xs:base64Binary
<i>Benutzt von</i>	xcpt:Data

Inhalt des Elementes Base64Sequence ist die verschlüsselte und base64-codierte Berechnung der Rentenbescheidaten, welche bei der DSRV zur Abholung durch den Versorgungsträger bereitgestellt wird.

3.8 Request Bestätigung der abgeholten Rentenbescheidaten

3.8.1 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

3.8.2 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

3.8.3 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 Informationen über erfolgreich abgeholte Anfragen.

3.8.4 Element Data

xcpt:Data

<i>Inhalt</i>	Enthält die Bestätigungsmeldung erfolgreich abgeholter Rückmeldungen
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xreq:TransportBody

3.8.5 Element ElementSequence

xcpt:ElementSequence

<i>Inhalt</i>	Enthält die Standardnachricht ConfirmationOfReceipt
<i>Datentyp</i>	xs:any
<i>Benutzt von</i>	xcpt:Data

3.8.6 Element ConfirmationOfReceipt

xmsg:ConfirmationOfReceipt

<i>Inhalt</i>	Enthält die Informationen, welche ResponseIDs erfolgreich empfangen wurden
<i>Datentyp</i>	xs:string
<i>Benutzt von</i>	xmsg:ConfirmationOfReceipt

@version

Inhalt 1.3
Datentyp xs:string

3.8.7 Element PropertySet

xmsg:PropertySet

Inhalt Enthält die Informationen, nach welchem Kriterium die folgenden Values bestätigt werden
Datentyp xs:string
Benutzt von xmsg:ConfirmationOfReceipt

@name

Inhalt Enthält das Kriterium, nach dem die Daten bestätigt werden.
Zulässige Inhalte:

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

Datentyp xs:anyURI

@type

Inhalt Art des Kriteriums. Hierbei handelt es sich um ein optionales Attribut, das nicht aufgeführt werden muss.
Datentyp xs:string

3.8.8 Element Value

xmsg:Value

Inhalt Enthält die Information zur Empfangsbestätigung.
Datentyp xs:string
Benutzt von xmsg:PropertySet

Das Element Value kann innerhalb des Elements PropertySet 1-n mal vorkommen. Unbestätigte Pakete können solange abgeholt werden, bis diese bestätigt werden.

3.9 Response Bestätigung der abgeholten Rentenbescheidaten

3.9.1 Element Transport

@version

Inhalt 1.4
Datentyp xs:string

@profile

Inhalt siehe [2.4.1 Profile-Attribut des Root-Elements](#)
Datentyp xs:anyURI

3.9.2 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.

3.9.3 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 im Sendevorgang zur Bestätigung enthält nur eine technische Bestätigung des eXtra Empfangssystems, jedoch keine fachlichen Daten des Fachverfahrens. Deshalb ist das Element TransportBody leer.

Anhang

3.10 StatusCodes - Allgemein

Die hier aufgeführten Codes können in jedem Geschäftsprozess auftreten

Statuscode	Text	Erläuterung	
E84	Variabel	Beim Fehler E84 handelt es sich um den generischen Fehler des eESB. In diesem Fall muss der Fehlertext für eine genauere Analyse der Fehlerursache beachtet werden	E
E85	Variabel	Der gewünschte Fachdienst befindet sich zurzeit im Wartungsmodus. Bitte den Fehlertext beachten	E
E80	Es kam zu einem Fehler. Bitte wenden Sie sich an die Hotline.	Allgemeine Fehlermeldung	E

3.11 StatusCodes - SendData

Die hier aufgeführten Codes treten beim Geschäftsprozess „SendData“ auf.

Statuscode	Text	Erläuterung	
I00	Daten wurden erfolgreich verarbeitet	Daten wurden erfolgreich verarbeitet	I
E90	Die Nachricht konnte wegen technischer Probleme nicht gespeichert werden. Bitte versuchen Sie es später noch einmal oder wenden Sie sich an die Hotline	Während der Anfrage ist es zu einem Fehler in der Verarbeitung gekommen. Dies kann an einem Datenbankproblem bei der DSRV liegen.	E
E92	Fehler bei Entschlüsselung	Die Daten konnten bei der DSRV nicht entschlüsselt werden.	E
E93	Fehler bei der Signaturprüfung	Die Signatur der verschlüsselten Daten stimmt nicht mit den Absenderdaten überein	E
E94	Fehler beim Entpacken	Daten konnten nicht entpackt werden	E
E99	Die Nachricht kann nicht verarbeitet werden	Die eingegangene Nachricht kann nicht verarbeitet werden, da nicht alle notwendigen Daten mitgeliefert wurden.	E
E70	Die Betriebsnummer nimmt nicht am Verfahren Teil	Fehler bei der Überprüfung der Betriebsnummer des Absenders	E
E71	Die BetriebsnummerAbsender stimmt nicht mit BetriebsnummerAbsenders des Einzeldatensatzes	Fehler bei der Überprüfung BetriebsnummerAbsender des Einzeldatensatzes	E
E72	Fehler bei der Überprüfung des Zertifikats	Zertifikat passt nicht mit BBNR des Absenders überein, BBNRAbsender nicht in der Datenbanktabelle vorhanden	E
E73	Paket ist nicht sortenrein	Fehler beim Überprüfen des Pakets	E

E75	Die MeldelD des empfangenen AntwortFehlers ist nicht vorhanden	Die MeldelD des empfangenen AntwortFehlers wird nicht verwendet / kann nicht gefunden werden	E
E76	Fehlende MeldelD in den Paketeinzeldatensätzen	Nicht alle Einzeldatensätze im Paket haben eine MeldelD	E
E77	Der Steuerungsblock ist im Einzeldatensatz nicht vorhanden	Fehler bei der Überprüfung des Steuerungsblocks	E
E78	Sachverhalt fehlt	Fehler bei der Überprüfung des Sachverhaltsblock	E

3.12 StatusCodes - DataRequest

Die hier aufgeführten Codes treten beim Geschäftsprozess „DataRequest“ auf.

Statuscode	Text	Erläuterung	
I01	Daten wurden erfolgreich verarbeitet	Daten wurden erfolgreich verarbeitet	I
E90	Die Nachricht konnte wegen technischer Probleme nicht gespeichert werden. Bitte versuchen Sie es später noch einmal oder wenden Sie sich an die Hotline	Während der Anfrage ist es zu einem Fehler in der Verarbeitung gekommen. Dies kann an einem Datenbankproblem bei der DSRV liegen.	E
I97	Anfrage erfolgreich – Keine Datensätze vorhanden	Die Anfrage nach Anforderungen vom Rentenversicherungsträger auf dem Kommunikationsserver der DSRV war erfolgreich. Es liegen für die abrufende Betriebsnummer aber keine Datensendungen zur Auslieferung vor	I
E99	Die Nachricht kann nicht verarbeitet werden	Die eingegangene Nachricht kann nicht verarbeitet werden, da nicht alle notwendigen Daten mitgeliefert wurden.	E

3.13 StatusCodes - ConfirmationOfReceipt

Die hier aufgeführten Codes treten beim Geschäftsprozess „ConfirmationOfReceipt“ auf.

Statuscode	Text	Erläuterung	
E86	Empfang der Nachricht konnte nicht bestätigt werden.	Der Empfang der Datensendungen konnte nicht korrekt quittiert werden.	E
I02	Die Nachricht entspricht dem geforderten Aufbau und kann im Fachverfahren verwendet werden	Sendung wurde erfolgreich bei der DSRV gespeichert	I
E99	Die Nachricht kann nicht verarbeitet werden	Die eingegangene Nachricht kann nicht verarbeitet werden, da nicht alle notwendigen Daten mitgeliefert wurden.	E

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

3.14 Referenzierte Dokumente

Name des Dokuments	Beschreibung
Beispiele.zip	Zip-Datei mit Beispielen für die eXtra – Kommunikation
SpocWSDL1.4.zip	Zip-Datei mit der Webservice WSDL
Schema.zip	Zip-Datei mit den Schematas für die eXtra - Kommunikation

3.15 Abbildungsverzeichnis

Abb. 1 Grafischer Ablauf der Kommunikation

6

Abb. 2 Aufbau Request-Header

10

Abb. 3 Aufbau Transport-Header Response

14

Abb. 4 Aufbau Package-Header

19

Abb. 5 Aufbau Request Anfrage Versorgungsträger

22

Abb. 6 Aufbau Request Datenabholung

24

Abb. 7 Aufbau Response Datenabholung

28