

<eXTra>

einheitliches XML-basiertes Transportverfahren



Schnittstellenspezifikation rvBEA

Version 1.00.05

Stand der Spezifikation: 06.10.2021
Version: 1.00.05
Redaktion: Deutsche Rentenversicherung Bund
Homepage: www.rvBEA.de
Fachlicher Ansprechpartner:

Herr Jürgen Hartmann
juergen.hartmann@drv-bund.de
Tel.: 0931/6002-73551
Technischer Ansprechpartner:
Frau Maria-Christina Bachmann
maria-christina.bachmann@drv-bund.de
Tel.: 0931/6002 -73284

0 Allgemeines

Das vorliegende Dokument dient als Grundlage für die Kommunikation zwischen den teilnehmenden Arbeitgebern und der Datenstelle der Rentenversicherung (DSRV) im Verfahren rvBEA.

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
1.00.02	28.06.2017	Alle	Registrierung des Dokuments	DSRV
1.00.03	18.07.2017	2	Unterstützung REST-Schnittstelle	Christina Bachmann
1.00.04	26.01.2018	2	Korrektur Dateiname bei Verarbeitungsprotokollen, Komprimierungsverfahren GZIP anstatt ZIP	Christina Bachmann
1.00.05	04.10.2021	2	Änderung Datatype „Bescheinigungen“ in „FORMS“	Christina Bachmann

1 Inhaltsverzeichnis

0	Allgemeines	2
1	Inhaltsverzeichnis	3
2	Teil A – Allgemeine Informationen	6
2.1	Grafischer Überblick Geschäftsprozesse	6
2.2	Übertragungsprotokoll	7
2.2.1	Unterschiede POST und SOAP	7
2.3	Server-Adressen.....	7
2.4	Authentifizierung.....	8
2.5	Verwendetes eXTra-Schema	8
2.5.1	Profile-Attribut des Root-Elements	8
2.5.2	Nutzdatenbeschreibung (DataSource).....	8
2.5.3	Verschlüsselung / Komprimierung (DataTransforms)	10
2.6	Beschreibung der einzelnen Geschäftsprozesse	10
2.6.1	Geschäftsprozesse Szenario 1.....	11
2.6.2	Geschäftsprozesse Szenario 2.....	13
2.7	Verwendung der ResponseID	15
2.7.1	Hotline	16
2.7.2	DataRequest	16
2.7.3	ConfirmationOfReceipt	16
3	Teil B – Beschreibung der Elemente.....	17
3.1	Verwendete Namensräume und Präfixe	17
3.2	Aufbau des Transport-Headers.....	17
3.2.1	Transport-Header des Request	17
3.2.2	Transport-Header der Response	21
3.2.3	Package-Header der Response	26
3.3	Aufbau der Plugins	27
3.3.1	Aufbau des DataTransforms-Plugins	27
3.3.2	Aufbau des DataSourcePlugins	29
3.3.3	Aufbau des Contact Plugins	30
3.4	Request Senden von rvBEA Daten.....	32
3.4.1	Element Transport.....	32
3.4.2	Element TransportHeader	32
3.4.3	Element TransportPlugins	32

3.4.4	Element TransportBody.....	33
3.4.5	Element Data	33
3.4.6	Element Base64CharSequence	33
3.5	Response Bestätigung über Erhalt der gesendeten rvBEA Daten.....	33
3.5.1	Element Transport.....	33
3.5.2	Element TransportHeader	34
3.5.3	Element TransportBody.....	34
3.6	Request Abholen von rvBEA Daten	34
3.6.1	Element Transport.....	34
3.6.2	Element TransportHeader	35
3.6.3	Element TransportBody.....	35
3.6.4	Element Data	35
3.6.5	Element ElementSequence	35
3.6.6	Element DataRequest	35
3.6.7	Element Query	36
3.6.8	Element Argument.....	36
3.6.9	Element XX	36
3.6.10	Element Control.....	37
3.6.11	Element MaximumPackages	37
3.7	Response Abholen der Anforderung von rvBEA Daten.....	37
3.7.1	Element Transport.....	38
3.7.2	Element TransportHeader	38
3.7.3	Element TransportBody.....	38
3.7.4	Element Package	38
3.7.5	Element PackageHeader.....	39
3.7.6	Element PackagePlugins.....	39
3.7.7	Element PackageBody	39
3.7.8	Element Data	39
3.7.9	Element Base64CharSequence	39
3.8	Request Bestätigung der abgeholten rvBEA Daten.....	40
3.8.1	Element Transport.....	40
3.8.2	Element TransportHeader	40
3.8.3	Element TransportBody.....	40
3.8.4	Element Data	40
3.8.5	Element ElementSequence	41
3.8.6	Element ConfirmationOfReceipt	41
3.8.7	Element PropertySet	41

3.8.8	Element Value	41
3.9	Response Bestätigung der abgeholten rvBEA Daten	42
3.9.1	Element Transport	42
3.9.2	Element TransportHeader	42
3.9.3	Element TransportBody	42
4	Anhang	43
4.1	StatusCodes	43
4.2	Referenzierte Dokumente	44
4.3	Abbildungsverzeichnis	44

2 Teil A – Allgemeine Informationen

2.1 Grafischer Überblick Geschäftsprozesse

Folgende Grafiken zeigen den Ablauf der in diesem Dokument beschriebenen Geschäftsprozesse auf. Hierbei wird von zwei Szenarien ausgegangen. Bei beiden Szenarien ergreift immer der Arbeitgeber die Initiative zur Kommunikation, unabhängig davon wie der Fluss der fachlichen Daten tatsächlich verläuft. In Szenario 1 beginnt der Arbeitgeber die Prozesskette mit dem Senden fachlicher Daten an die DSRV. Beim Szenario 2 hingegen hat die DSRV fachliche Daten (nämlich Anforderungen) zunächst bereitgestellt, die der Arbeitgeber abholen, bestätigen und auswerten muss, um den Anforderungen der DSRV letztendlich nachkommen zu können.

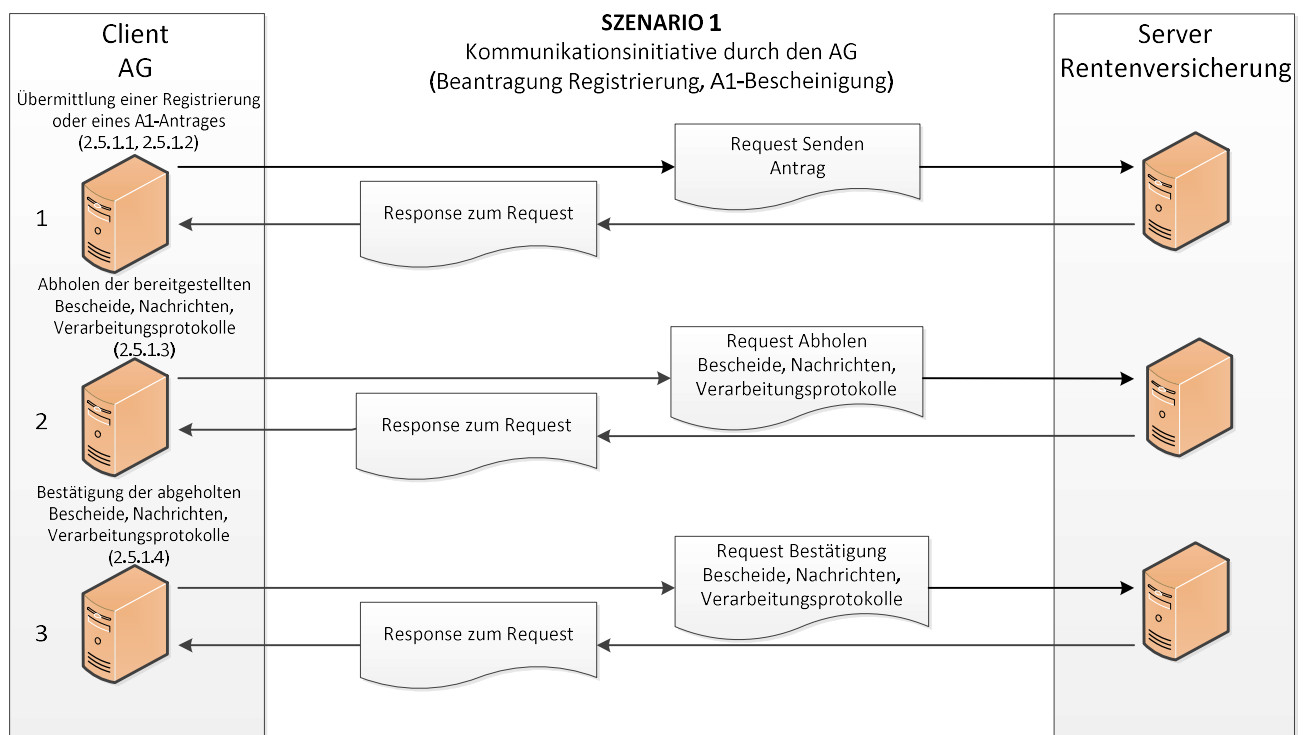


Abb. 1 Grafischer Ablauf der Geschäftsprozesse Szenario 1, Kommunikationsinitiative durch AG

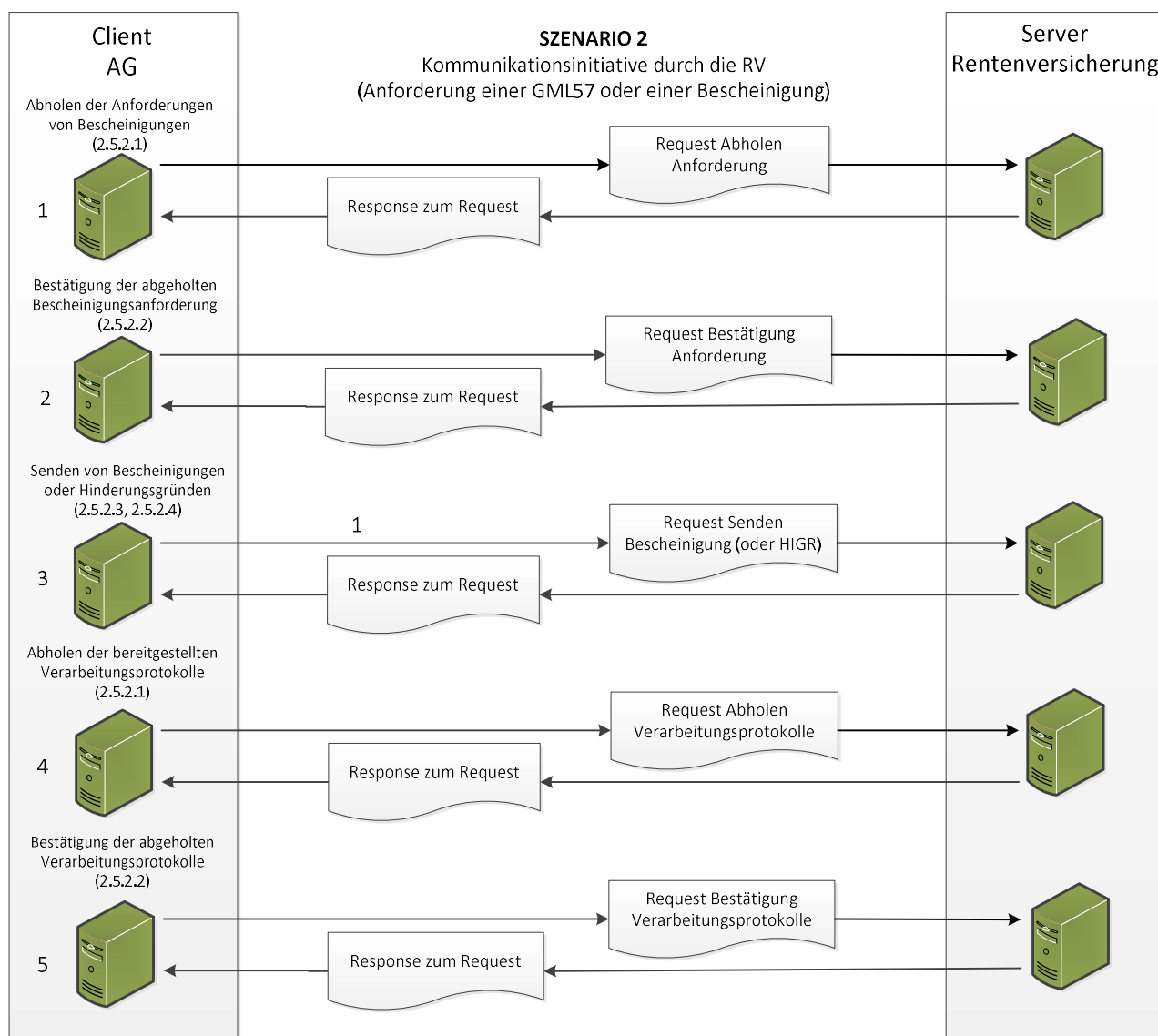


Abb. 2 Grafischer Ablauf der Geschäftsprozesse Szenario 2, Kommunikationsinitiative durch RV

2.2 Übertragungsprotokoll

Im Verfahren rvBEA findet die Kommunikation zwischen dem Arbeitgeber und der DSRV über http-POST-Requests und entsprechende Responses statt.

Zusätzlich wird von der DSRV auch eine Kommunikation über SOAP-WebServices angeboten.

2.2.1 Unterschiede POST und SOAP

Die Übertragung der Nutzdaten per MTOM wird nur beim Einsatz des SOAP-WebServices unterstützt. Bei Kommunikation über http-POST werden die Nutzdaten als Base64-codiertes Objekt im eXtra-XML transportiert.

2.3 Server-Adressen

Bei den Test- und Produktionsservern der DSRV handelt es sich um physikalisch getrennte Server. Aus diesem Grund werden unterschiedliche Adressen verwendet.

POST-URLs

Testsystem: <https://itsg.eservicet-drv.de/rvbea/rest>

Produktionssystem: <https://itsg.eservice-drv.de/rvbea/rest>

SOAP-WebService

Testsystem: https://itsg.eservicet-drv.de/SPoC/ExtraService_v1.4

Produktionssystem: https://itsg.eservice-drv.de/SPoC/ExtraService_v1.4

2.4 Authentifizierung

Die Authentifizierung des Absenders findet beim Aufbau der https-Verbindung über ein Client-Zertifikat statt. Zum Einsatz kommt in diesem Verfahren ein Zertifikat des Trust-Centers der ITSG (Informationstechnische Servicestelle der Gesetzlichen Krankenversicherung GmbH).

Für die Kommunikationsverschlüsselung wird TLS 1.2 verwendet.

2.5 Verwendetes eXTra-Schema

Zum Einsatz kommen das eXTra-Basisschema in der Version 1.4 sowie die eXTra-Standardnachrichten der Version 1.4.

Sendet der Arbeitgeber über SOAP-WebService und ist die Kommunikation mittels MTOM beim Webservice-Client des Anfragers aktiviert, so werden die Daten als Anhang außerhalb der eXTra-XML dem Request beigefügt.

2.5.1 Profile-Attribut des Root-Elements

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

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

2.5.2 Nutzdatenbeschreibung (DataSource)

Nachfolgend wird der Aufbau der Nutzdaten beschrieben.

2.5.2.1 Encoding

Die Nutzdaten sind als ISO 8859-1 codiert. Zulässige Werte sind:

Wert	Beschreibung
I1	ISO8859-1
I8	ISO 8-Bit, Code gemäß DIN 66303:2000-06

2.5.2.2 Dateiname

Hier wird der Dateiname der Sendung angegeben.

Folgende Struktur des Dateinamens ist im System der Datenstelle der Rentenversicherungsträger (DSRV) im Verfahren rvBEA zulässig:

Stelle 1 System- Umgebung	Stelle 2-4 Verfah- renskürzel	Stelle 5 Versions- nummer	Stelle 6-11 Dateifolge- nummer	Erläuterung
T	ARV	0	nnnnnn	<p>Datei vom Arbeitgeber zur Rentenversicherung bei Datatype Registrierung, GML57 und FORMS im Testsystem (betrifft Nutzdaten AGTOSV mit Verfahren DXEB)</p> <p>Verarbeitungsprotokolle zu o.g. Dateien von der Rentenversicherung an den Arbeitgeber (betrifft Nutzdaten SVTOAG mit Verarbeitungsergebnis; Hinweis: Dateifolgenummer hat eigenen Nummernkreis und entspricht nicht der Dateinummer der Eingangssendung!)</p> <p>TARV0nnnnnn</p>
T	RVA	0	nnnnnn	<p>Datei von Rentenversicherungsträger zum Arbeitgeber bei Datatype GML57 und FORMS im Testsystem (betrifft Nutzdaten SVTOAG mit Verfahren DXAR)</p> <p>TRVA0nnnnnn</p>
T	A1A	0	nnnnnn	<p>Datei vom Arbeitgeber zur Rentenversicherung bei Datatype A1 im Testsystem (betrifft Nutzdaten AGTOSV mit Verfahren A1_Antrag_Entsendung)</p> <p>Verarbeitungsprotokoll zu o.g. Datei von der Rentenversicherung an den Arbeitgeber (betrifft Nutzdaten SVTOAG mit Verarbeitungsergebnis; Hinweis: Dateifolgenummer hat eigenen Nummernkreis und entspricht nicht der Dateinummer der Eingangssendung!)</p> <p>TA1A0nnnnnn</p>
T	A1S	0	nnnnnn	<p>Datei von Rentenversicherungsträger zum Arbeitgeber bei Datatype A1 im Testsystem (betrifft Nutzdaten SVTOAG mit Verfahren A1_Bewilligung und A1_Ablehnung)</p> <p>TA1S0nnnnnn</p>
E	ARV	0	nnnnnn	<p>Analog TARV, aber Produktivsystem</p> <p>EARV0nnnnnn</p>

E	RVA	0	nnnnnn	Analog TRVA, aber Produktivsystem ERVA0nnnnnn
E	A1A	0	nnnnnn	Analog TA1A, aber Produktivsystem EA1A0nnnnnn
E	A1S	0	nnnnnn	Analog TA1S, aber Produktivsystem EA1S0nnnnnn

Bei Dateien vom Arbeitgeber wird die Dateifolgenummer für jedes Verfahrenskürzel in lückenloser aufsteigender Reihenfolge bei 1 beginnend erwartet. Bei der DSRV werden die Dateien nach dieser Reihenfolge abgearbeitet. Sendungen, die zu der vorangegangenen Datei eine Lücke aufweisen, werden im nachgelagerten Fachverfahren abgewiesen (Verarbeitungsprotokoll).

Die Dateifolgenummer 1 wird immer angenommen und verarbeitet. Daher kann auch, sofern eine lückenlose Nummerierung der Dateien beim Arbeitgeber organisatorisch nicht von Bedeutung ist, jede Sendung mit Dateifolgenummer 1 geschickt werden.

Die von der DSRV zur Abholung bereit gestellten Dateien werden ebenfalls pro Arbeitgeber und Verfahrenskürzel bei 1 beginnend lückenlos nummeriert. Aufgrund von beispielsweise einem Versionswechsel ist auch hier eine Rücksetzung auf 1 möglich.

Es ist zu beachten, dass Verarbeitungsprotokolle das gleiche Verfahrenskürzel aufweisen, wie die zugehörige Eingangsdatei, d.h. in diesem Fall wird das Arbeitgeberkürzel übernommen. Der Nummernkreis der Dateifolgenummer ist jedoch ein eigener, d.h. die Nummer des Verarbeitungsprotokolls ist nicht identisch mit der Nummer der Eingangssendung! Alle anderen zur Abholung bereitgestellten Daten verwenden das DSRV-spezifische Kürzel im Dateinamen.

2.5.2.3 Erstellungsdatum

Erstellungsdatum der Sendung.

2.5.3 Verschlüsselung / Komprimierung (DataTransforms)

Die fachlichen Nutzdaten in den Geschäftsprozessen werden mit dem Verfahren PKCS7 verschlüsselt und signiert. Hierbei werden die Nutzdaten mit dem öffentlichen Zertifikat des Empfängers verschlüsselt und mit dem geheimen Schlüssel des Senders implizit signiert.

Verwendete Verschlüsselung: <http://www.extra-standard.de/transforms/encryption/PKCS7>

2.5.3.1 Reihenfolge der Komprimierung und der Verschlüsselung

Die im DataTransforms beschriebene Komprimierung und Verschlüsselung hat folgende Sortierung, welche über das Attribut ‚order‘ gekennzeichnet wird.

1. Komprimierung der Daten mit GZIP oder unkomprimiert, beschrieben in den einzelnen Geschäftsprozessen
2. Verschlüsselung der Daten mit PCKS7

2.6 Beschreibung der einzelnen Geschäftsprozesse

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

	rvBEA Schnittstellenbeschreibung	Seite: 11 Version: 1.00.05 Stand: 06.10.21
--	--	---

2.6.1 Geschäftsprozesse Szenario 1

In diesem Szenario ist der aus anderen Verfahren bekannte Verfahrensablauf beschrieben. Hier geht die Kommunikationsinitiative vom Arbeitgeber aus. Der Arbeitgeber sendet Daten an die DSRV. Das Feedback zu den Daten, wie z.B. Verarbeitungsprotokolle, werden von der DSRV zur Abholung bereitgestellt. Der Arbeitgeber holt aktiv ab und bestätigt den Erhalt der abgeholten Daten.

2.6.1.1 Geschäftsprozess 1 - Senden von Registrierungen bzw. A1 Anträgen

Dieser Prozess dient zur Registrierung des Arbeitgebers im Verfahren „rvBEA“ bei der DSRV und zum Übermitteln von A1-Anträgen

Procedure: rvBEA

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

DataType A1: <http://www.extra-standard.de/datatypes/A1>

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

Mögliche Komprimierungen:

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

<http://www.extra-standard.de/transforms/compression/NONE>

Es darf als fachliche Nachricht immer nur ein Exemplar des jeweiligen Datentyps gesendet werden. Sollen mehrere Exemplare übermittelt werden, sind auch mehrere Sendeprozesse erforderlich.

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

XML-Beispiele können folgenden Dateien entnommen werden:

- S1-G1-request-Registrierung.xml
- S1-G1-request-A1.xml
- S1-G1-response-Registrierung.xml
- S1-G1-response-Registrierung-Fehler.xml
- S1-G1-response-A1.xml
- S1-G1-response-A1-Fehler.xml

2.6.1.2 Geschäftsprozess 2 – Abholen von rvBEA Daten

Dieser Geschäftsprozess dient zur Abholung der rvBEA Daten.

Procedure: rvBEA

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

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

Mögliche Komprimierungen:

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

<http://www.extra-standard.de/transforms/compression/NONE>

Die angeforderten fachlichen Nachrichten, die rvBEA Daten, sind in der eXtra Response in der Paket Ebene enthalten. Für jedes ausgelieferte Paket vergibt die Rentenversicherung eine eigene, eindeutige ResponseID. Diese ResponseID muss im Bestätigungsprozess verwendet werden.

Die Komprimierung und Verschlüsselung der fachlichen Nachricht richtet sich danach, wie die zuletzt vom Arbeitgeber gesendeten Daten aufbereitet waren. Bei der ersten Abholung wird sich auf die Registrierung des Arbeitgebers bezogen. Werden die Daten komprimiert und verschlüsselt geschickt, so werden auch die rvBEA Daten komprimiert und verschlüsselt zur Abholung bereitgestellt. Werden die Daten nur verschlüsselt geschickt, entfällt auch bei der DSRV die Komprimierung der rvBEA Daten.

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

Das Element Query im Transport-Body gibt dem Abholer die Möglichkeit eine Vorauswahl über die Ergebnismenge zu treffen. Über Argumente kann so gesteuert werden, von welchem Datentyp die abzuholenden Dateien sein sollen. Eine Auswahl kann in Kombination der Response_ID mit dem Datatype getroffen werden. Fehlt die Angabe der ResponseID werden immer alle Dateien zurückgeliefert, die noch nicht bestätigt wurden. Bei dem Datatype sind Mehrfachangaben möglich. Fehlt die Angabe des Datatypes, so werden die Dateien zu allen Datatypes innerhalb des rvBEA-Verfahrens abgeholt (Szenario1 und Szenario 2)

Übersicht über die Argumente mit den Auswahlparametern:

ResponseID	
GT (greater than)	größer als die angegebene Response-ID; es werden nur Dateien berücksichtigt, die noch nicht als abgeholt gekennzeichnet sind. Bei Nichtangabe wird daher standardmäßig GT 0, also größer 0 angenommen. Bei Verwendung des Parameters ist dies auch die empfohlene Einstellung. Diese setzt zwingend den Geschäftsprozess 3 (Bestätigung der abgeholtten rvBEA Daten) voraus!

DataType	
EQ (equal)	gleich dem angegebenen Datentyp; bei Nichtangabe werden alle Datentypen innerhalb des rvBEA-Verfahrens berücksichtigt; gültige Angaben für Szenario 1: <div style="display: flex; justify-content: space-between;"> Registrierung Verarbeitungsprotokolle Registrierung </div> <div style="display: flex; justify-content: space-between;"> A1 Verarbeitungsprotokolle, Bewilligung, Ablehnung A1 </div>

Die Anzahl der auszuliefernden rvBEA Daten kann in der Standardnachricht DataRequest mit dem Element Control gesteuert werden. Ist dieses Element nicht mit angegeben, werden maximal die - gemäß Query - 10 ältesten (und noch nicht bestätigten) ausgelieferten rvBEA Daten an den anfragenden Arbeitgeber ausgeliefert.

XML-Beispiele können folgenden Dateien entnommen werden:

- S1-G2-request-queryexample ohne DataType.xml
- S1-G2-request-queryexample mit DataType Registrierung.xml
- S1-G2-request-queryexample mit DataType Registrierung und A1.xml

- S1-G2-response-queryexample ohne DataType.xml
- S1-G2-response-queryexample mit DataType Registrierung.xml
- S1-G2-response-queryexample mit DataType Registrierung und A1.xml
- S1-G2-response-leer.xml

2.6.1.3 Geschäftsprozess 3 - Bestätigung der abgeholten rvBEA Daten

In diesem Prozess wird die erfolgreiche Abholung der rvBEA Daten bestätigt.

Procedure: rvBEA

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 der Kommunikationsart (request-with-acknowledgement).

WICHTIG: Solange die abgerufenen Daten nicht bestätigt worden sind, werden diese immer wieder geliefert. Dies kann bei einer limitierten Anzahl der abzuholenden Daten zu einer „Endlosschleife“ führen!!

XML-Beispiele können folgenden Dateien entnommen werden:

- S1-G3-request-confirmation.xml
- S1-G3-response-confirmation.xml
- S1-G3-response-confirmation-Fehler.xml

2.6.2 Geschäftsprozesse Szenario 2

Bei Szenario 2 geht nun die Kommunikationsinitiative von der DSRV aus. Das bedeutet, dass die DSRV für Arbeitgeber, die sich im rvBEA-Verfahren über einen Registrierungsdatensatz angemeldet haben (siehe 2.6.1.1 [Senden von Registrierungen](#)), Anforderungen für eine gesonderte Meldung oder andere Bescheinigungen zur Abholung bereit stellen kann. Der Arbeitgeber wird sozusagen dazu aufgefordert, Daten zu schicken. Dieser holt sich diese Daten ab, bestätigt den Erhalt der Daten und sendet die angeforderte Bescheinigung bzw. einen Hinderungsgrund an die DSRV. Für diese Sendung wird dann – wie schon unter Szenario 1 beschrieben, ein Verarbeitungsprotokoll zur Sendung zur Abholung bereit gestellt, vom Arbeitgeber abgeholt und bestätigt.

2.6.2.1 Geschäftsprozess 1 - Abholen von rvBEA Daten

Dieser Geschäftsprozess dient zur Abholung der rvBEA Daten und wurde bereits im Kapitel 2.5.1.3 [Abholen von rvBEA Daten](#) beschrieben.

Übersicht über die Argumente mit den Auswahlparametern:

ResponseID	
GT (greater than)	größer als die angegebene Response-ID (Default); es werden nur Dateien berücksichtigt, die noch nicht als abgeholt gekennzeichnet sind. Bei Nichtangabe wird daher standardmäßig GT 0, also größer 0 angenommen. Bei Verwendung des Parameters ist dies auch die

	empfohlene Einstellung. Diese setzt zwingend den Geschäftsprozess 2 (Bestätigung der abgeholten rvBEA Daten) voraus!
--	--

DataType					
EQ (equal)	gleich dem angegebenen Datentyp; bei Nichtangabe werden alle Datentypen innerhalb des rvBEA-Verfahrens berücksichtigt; gültige Angaben für Szenario 2: <table style="width: 100%; border: none;"> <tr> <td style="width: 25%;">GML57</td> <td>Anforderung gesonderte Meldung, Verarbeitungsprotokolle zu Antworten zur gesonderten Meldung</td> </tr> <tr> <td>FORMS</td> <td>Anforderung Bescheinigung, Verarbeitungsprotokolle zur Bescheinigungsübersendung</td> </tr> </table>	GML57	Anforderung gesonderte Meldung, Verarbeitungsprotokolle zu Antworten zur gesonderten Meldung	FORMS	Anforderung Bescheinigung, Verarbeitungsprotokolle zur Bescheinigungsübersendung
GML57	Anforderung gesonderte Meldung, Verarbeitungsprotokolle zu Antworten zur gesonderten Meldung				
FORMS	Anforderung Bescheinigung, Verarbeitungsprotokolle zur Bescheinigungsübersendung				

Die Komprimierung und Verschlüsselung der fachlichen Nachricht richtet sich danach, wie die zuletzt vom Arbeitgeber gesendeten Daten aufbereitet waren. Bei der ersten Abholung wird sich auch in Szenario 2 auf die Registrierung des Arbeitgebers bezogen.

HINWEIS: Auch wenn hier Szenario 2 beschrieben wird, also das Abholen von Anforderungen, so werden bei diesem Abholprozess natürlich alle Daten zu dem angegebenen Datentyp abgeholt, also z.B. auch Verarbeitungsprotokolle zu vorangegangenen Sendungen.

XML-Beispiele können folgenden Dateien entnommen werden:

- S2-G1-G4-request-queryexample ohne DataType.xml
- S2-G1-G4-request-queryexample mit DataType GML57.xml
- S2-G1-G4-request-queryexample mit DataType FORMS.xml
- S2-G1-G4-response-queryexample ohne DataType.xml
- S2-G1-G4-response-queryexample mit DataType GML57.xml
- S2-G1-G4-response-queryexample mit DataType FORMS.xml
- S2-G1-G4-response-leer.xml

2.6.2.2 Geschäftsprozess 2 - Bestätigung der abgeholten rvBEA Daten

In diesem Prozess wird die erfolgreiche Abholung der rvBEA Daten bestätigt.

Beschreibung siehe Kapitel 2.5.1.4 [Bestätigung der abgeholten rvBEA Daten](#)

XML-Beispiele können folgenden Dateien entnommen werden:

- S2-G2-G5-request-confirmation.xml
- S2-G2-G5-response-confirmation.xml
- S2-G2-G5-response-confirmation-Fehler.xml

2.6.2.3 Geschäftsprozess 3 - Senden von gesonderten Meldungen GML57 oder von Bescheinigungen

In diesem Prozess werden die gesonderten Meldungen GML57 durch den Arbeitgeber an die DSRV gesendet.

Procedure: rvBEA

DataType GML57: <http://www.extra-standard.de/datatypes/GML57>

	rvBEA Schnittstellenbeschreibung	Seite: 15 Version: 1.00.05 Stand: 06.10.21
--	--	---

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

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

Mögliche Komprimierungen:

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

<http://www.extra-standard.de/transforms/compression/NONE>

Es darf als fachliche Nachricht immer nur ein Exemplar des jeweiligen Datentyps gesendet werden. Sollen mehrere Exemplare übermittelt werden, sind auch mehrere Sendeprozesse erforderlich.

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

XML-Beispiele können folgenden Dateien entnommen werden:

- S2-G3-request-GML57.xml
- S2-G3-request-FORMS.xml
- S2-G3-response-GML57.xml
- S2-G3-response-GML57-Fehler.xml
- S2-G3-response-FORMS.xml
- S2-G3-response-FORMS-Fehler.xml

2.6.2.1 Geschäftsprozess 4 - Abholen von rvBEA Daten

Entspricht inhaltlich dem Geschäftsprozess 1. Abgeholt werden sowohl Verarbeitungsprotokolle, als auch neue Anforderungen.

XML-Beispiele können folgenden Dateien entnommen werden:

- S2-G1-G4-request-queryexample mit DataType All.xml
- S2-G1-G4-request-queryexample mit DataType GML57.xml
- S2-G1-G4-request-queryexample mit DataType FORMS.xml
- S2-G1-G4-response-queryexample mit DataType All.xml
- S2-G1-G4-response-queryexample mit DataType GML57.xml
- S2-G1-G4-response-queryexample mit DataType FORMS.xml
- S2-G1-G4-response-leer.xml

2.6.2.1 Geschäftsprozess 5 - Bestätigung der abgeholten rvBEA Daten

Entspricht inhaltlich dem Geschäftsprozess 2. Der Erhalt der abgeholten Daten, egal ob Verarbeitungsprotokolle oder Anforderungen, wird bestätigt.

XML-Beispiele können folgenden Dateien entnommen werden:

- S2-G2-G5-request-confirmation.xml
- S2-G2-G5-response-confirmation.xml
- S2-G2-G5-response-confirmation-Fehler.xml

2.7 Verwendung der ResponseID

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

	rvBEA Schnittstellenbeschreibung	Seite: 16 Version: 1.00.05 Stand: 06.10.21
--	--	---

Die ResponseID eines ausgelieferten Paketes spielt in zweifacher Hinsicht eine entscheidende Rolle: einerseits im Holprozess als Eingrenzungskriterium in der Query der Standardnachricht DataRequest und andererseits im Bestätigungsprozess als Bestätigungskriterium in der Standardnachricht ConfirmationOfReceipt.

2.7.1 Hotline

Bei einer Anfrage bei der Hotline der DSRV durch den Arbeitgeber muss die ResponseID angegeben werden, um die betroffene Sendung zu identifizieren. Dadurch ist es möglich, in einem Fehlerfall effektiv und schnell die Ursache zu ermitteln und zu beheben.

2.7.2 DataRequest

Die ResponseID wird im Szenario 1 [Geschäftsprozess 2 – Abholen von rvBEA Daten](#), Szenario 2 [Geschäftsprozess 1 - Abholen von rvBEA Daten](#) sowie Szenario 2 [Geschäftsprozess 4 - Abholen von rvBEA Daten](#) als Eingrenzungskriterium genutzt. Sollte es sich hierbei um den initialen Request handeln, so ist die ID als 0 anzugeben. Darüber hinaus kann die ResponseID auf Paket Ebene des letzten erfolgreichen Holprozesses als Kriterium verwendet werden.

Die DSRV empfiehlt, die ResponseID im DataRequest immer mit „GT = 0“ anzugeben. Dadurch wird gewährleistet, dass jedes noch offene Fachdatenpaket an den anfragenden Arbeitgeber übermittelt wird.

2.7.3 ConfirmationOfReceipt

Die ResponseID wird im Szenario 1 [Geschäftsprozess 3 - Bestätigung der abgeholten rvBEA Daten](#), Szenario 2 [Geschäftsprozess 2 - Bestätigung der abgeholten rvBEA Daten](#) sowie Szenario 2 [Geschäftsprozess 5 - Bestätigung der abgeholten rvBEA Daten](#) 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 Arbeitgeber 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. 3 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

Inhalt Enthält SenderID des Absenders
Datentyp ##elements
Benutzt von xreq:TransportHeader

3.2.1.3 Element SenderID

xcpt:SenderID

Inhalt Betriebsnummer des Absender
Diese muss mit der Betriebsnummer des verwendeten Zertifikats übereinstimmen
Datentyp xs:string

3.2.1.4 Element Receiver

xcpt:Receiver

Inhalt Enthält ReceiverID des Empfängers
Datentyp ##elements
Benutzt von xreq:TransportHeader

3.2.1.5 Element ReceiverID

xcpt:ReceiverID

Inhalt Betriebsnummer Empfänger; muss immer Betriebsnummer 66667777 der DSRV sein
Datentyp xs:string
Benutzt von xcpt:Receiver

3.2.1.6 Element RequestDetails

xcpt:RequestDetails

Inhalt Diverse Request-spezifische Informationen
Datentyp ##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
Datentyp xs:string
Benutzt von xcpt:RequestDetails

3.2.1.8 Element TimeStamp

xcpt:TimeStamp

<i>Inhalt</i>	Ein Zeitstempel z.B. ‚2008-10-30T15:09:00‘ zum Beginn der Übertragung des Senders
<i>Datentyp</i>	xs:dateTime
<i>Benutzt von</i>	xcpt:RequestDetails

3.2.1.9 Element Application

xcpt:Application

<i>Inhalt</i>	Enthält Product und Manufacturer
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xcpt:RequestDetails

3.2.1.10 Element Product

xcpt:Product

<i>Inhalt</i>	Bezeichnung eines (Software-) Produkts des Senders
<i>Datentyp</i>	xs:string
<i>Benutzt von</i>	xcpt:Application

3.2.1.11 Element Manufacturer

xcpt:Manufacturer

<i>Inhalt</i>	Herstellerbezeichnung des Software-Produktes des Senders
<i>Datentyp</i>	xs:string
<i>Benutzt von</i>	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.

Datentyp `xs:anyURI`
Benutzt von `xcpt:RequestDetails`

3.2.1.14 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 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="TransportResponseType">
  <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 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. ‚2015-10-30T15:09:00‘ 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 dieser Lieferung
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 konnten

`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 im Element RequestDetails DataType und im Element ResponseDetails beziehen sich innerhalb dieses Headers jedoch auf die Sendung, die im Package ausgeliefert wird und nicht auf die Response als Ganzes. Auf die Unterschiede wird in den Unterkapiteln eingegangen.

```

<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. 5 Aufbau Package-Header

3.2.3.1 DataType

Anstelle von DataType „DataRequest“ wird hier der Datentyp der Sendung mitgeteilt, z.B. A1. Da der Arbeitgeber die Möglichkeit hat, bei einem Abholvorgang mehrere, bzw. alle Datentypen von rvBEA gleichzeitig abzuholen, ist es notwendig, die Sendung von außen zu kennzeichnen, damit der Abholer sofort erkennen kann, welchem Unterverfahren von rvBEA er die Daten zuordnen muss.

3.2.3.2 ResponseID

Bei der ResponseID handelt es sich um die ID der Sendung, die im jeweiligen Package ausgeliefert wird und nicht um die ResponseID für den gesamten Request. Die ResponseID wird bei Erstellung der Sendung vergeben und ist daher immer kleiner als die ResponseID im TransportHeader, welche bei Erstellung des Requests erzeugt wird.

3.2.3.3 Timestamp

Beim Timestamp handelt es sich um den Verarbeitungs-Zeitstempel der Sendung, bzw. um den Bereitstellungszeitpunkt der Sendung zur Abholung und nicht um die Erstellungszeit des Requests.

3.2.3.4 Report

Im Report wird der Status der Sendung hinterlegt und nicht der Status des Abrufs von Anfragen.

3.3 Aufbau der Plugins

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

Benutzt von xplg:Compression

@id

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

Datentyp xs:anyURI

3.3.1.4 Element InputData

xplg:InputData

Inhalt Enthält die Größe der Nutzdaten vor der Komprimierung

Datentyp ##elements

Benutzt von xplg:Encryption

@bytes

Inhalt Dateigröße in Bytes

Datentyp xs:nonNegativeInteger

3.3.1.5 Element Encryption

xplg:Encryption

Inhalt Enthält die Verschlüsselungsinformation

Datentyp ##elements

Benutzt von xplg:DataTransforms

@order

Inhalt 2

Datentyp xs:positiveInteger

3.3.1.6 Element Algorithm

xplg:Algorithm

Inhalt Verwendeter Verschlüsselungs-Algorithmus

Datentyp ##elements

Benutzt von xplg:Encryption

@id

Inhalt Bezeichnung der Verschlüsselung Der zulässige Inhalt wird in Teil A unter [2.4.3 Verschlüsselung / Komprimierung \(DataTransforms\)](#) beschrieben.

Datentyp 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

3.3.2 Aufbau des DataSourcePlugins

Im DataSource-Plugin werden die Informationen zu Dateiname und -format hinterlegt

3.3.2.1 Element DataSource

xplg:DataSource

<i>Inhalt</i>	Enthält zusätzliche Informationen zu den fachlichen Daten
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xreq:TransportPlugIns

@version

<i>Inhalt</i>	1.0
<i>Datentyp</i>	xs:string

3.3.2.2 Element DataContainer

xplg>DataContainer

<i>Inhalt</i>	Enthält als Attribute zusätzliche Informationen zu den fachlichen Daten
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xplg:DataSource

@type

<i>Inhalt</i>	http://www.extra-standard.de/container/FILE
<i>Datentyp</i>	xs:anyURI

@name

<i>Inhalt</i>	(Datei-)Name der fachlichen Daten. Der zulässige Inhalt wird in Teil A unter 2.4.2.2 Dateiname beschrieben
<i>Datentyp</i>	xs:string

@created

Inhalt Erstellungsdatum der fachlichen Daten, z.B. „2008-11-28T13:46:58“

Datentyp `xs:dateTime`

@encoding

Inhalt Zeichensatz der fachlichen Daten im TransportBody beim Request durch den Arbeitgeber und PackageBody beim Response

Datentyp `xs:string`

Die zulässigen Werte für das **encoding** finden Sie unter Teil A [2.4.2.1 Encoding](#) beschrieben.

3.3.3 Aufbau des Contact Plugins

Hier werden die Kontaktinformationen vom Sender hinterlegt.

3.3.3.1 Element Contacts

xplg:Contacts

Inhalt Enthält die Kontaktdaten des Absenders. Angabe ist optional

Datentyp `##elements`

Benutzt von `xreq:TransportPlugIns`

@version

Inhalt 1.0

Datentyp `xs:string`

3.3.3.2 Element SenderContacts

xplg:SenderContacts

Inhalt Enthält die Kontaktdaten des Absenders

Datentyp `##elements`

Benutzt von `xplg:Contacts`

@version

Inhalt 1.0

Datentyp `xs:string`

3.3.3.1 Element Endpoint

xplg:SenderContacts

Inhalt Enthält die eMail-Adresse des Absenders

Datentyp ##elements
Benutzt von xplg:SenderContacts

Inhalt @version 1.0
Datentyp xs:string

3.4 Request Senden von rvBEA Daten

```

<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>

```

Abb. 6 Aufbau Request Arbeitgeberanforderung- und –antwort

Die folgende Beschreibung der Elemente gilt für den Sendeprozess des Szenario 1 Geschäftsprozess 1 und Szenario 2 Geschäftsprozess 3.

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 das DataTransforms. Der Aufbau wird in den Abschnitten „Aufbau des DataTransforms-Plugins“ beschrieben.
<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>	Base64-Zeichenfolge der verschlüsselten Nutzdaten
<i>Datentyp</i>	xs:base64Binary
<i>Benutzt von</i>	xcpt:Data

3.5 Response Bestätigung über Erhalt der gesendeten rvBEA Daten

Die folgende Beschreibung der Elemente gilt für den in Szenario 1 beschriebenen Geschäftsprozess 3, sowie Szenario 2 Geschäftsprozesse 2 und 5.

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 „Geschäftsprozess 1 - Senden von Registrierung“, enthält nur eine technische Bestätigung, jedoch keine fachlichen Daten des Fachverfahrens. Deshalb ist das Element TransportBody leer.

3.6 Request Abholen von rvBEA Daten

```

<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>

```

Abb. 7 Aufbau Request Datenabholung

Die folgende Beschreibung der Elemente gilt für den Abholprozess des Szenario 1 Geschäftsprozess 2, sowie Szenario 2 Geschäftsprozesse 1 und 4.

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 Anforderung der Anfragen.

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>	Kann eine Reihe von Argumenten zur Einschränkung der Arbeitsergebnisse enthalten. Eine nähere Beschreibung kann unter Szenario 1 Geschäftsprozess 2 – Abholen von rvBEA Daten und Szenario 2 Geschäftsprozess 1 - Abholen von rvBEA Daten nachgelesen werden.
<i>Datentyp</i>	xs:string
<i>Benutzt von</i>	xmsg:DataRequest

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

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

Zulässig sind hierbei:

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

@type

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

3.6.9 Element XX

xmsg:XX

<i>Inhalt</i>	Element zum Einschränken der Auswahl XX wird ersetzt durch einen der folgenden Operanden: GT = Greater than, also größer als der angegebene Inhalt; ist nur bei Argument RequestID zu verwenden. EQ = Equal, also gleich dem angegebenen Inhalt; ist nur bei Argument DataType zu verwenden.
---------------	---

Datentyp `xs:string`
Benutzt von `xmsg:Argument`

3.6.10 Element Control

xmsg:Control

Inhalt Optionales Element dass die maximale Anzahl der auszuliefernden Pakete enthält

Datentyp `xs:string`

Benutzt von `xmsg:DataRequest`

3.6.11 Element MaximumPackages

xmsg:MaximumPackages

Inhalt Anzahl der maximal zu liefernden Pakete innerhalb der Rückmeldung

Datentyp `xs:positiveInteger`

Benutzt von `xmsg:Control`

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

3.7 Response Abholen der Anforderung von rvBEA Daten

```

<xs:complexType name="TransportResponseBodyType">
  <xs:complexContent>
    <xs:extension base="xcpt:ElementWithOptionalVersionType">
      <xs:choice minOccurs="0" maxOccurs="1">
        <xs:element ref="Package" minOccurs="1" maxOccurs="unbounded"/>
      </xs:choice>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="PackageResponseType">
  <xs:complexContent>
    <xs:extension base="xcpt:ElementWithOptionalVersionType">
      <xs:sequence minOccurs="1" maxOccurs="1">
        <xs:element ref="PackageHeader" minOccurs="1" maxOccurs="1"/>
        <xs:element ref="PackagePlugins" minOccurs="0" maxOccurs="1"/>
        <xs:element ref="PackageBody" minOccurs="1" maxOccurs="1"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

Abb. 8 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 Rückantwort (also fachliche Daten) des Fachverfahrens auf die Anforderung des Senders.
<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 das Verschlüsselungs- und Dateiinformatons-Plugin. Der Aufbau wird im Abschnitt „Aufbau des DataTransforms-Plugins“ beschrieben.
<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>	<p>Enthält die fachlichen Daten, die der Sender mittels DataRequest angefordert hat.</p> <p>Die Daten werden wie unter 2.4.3 Verschlüsselung / Komprimierung (DataTransforms) beschrieben verschlüsselt und arbeitgeberspezifisch komprimiert oder unkomprimiert an den Arbeitgeber ausgeliefert (siehe 2.5.1.2 Geschäftsprozess 2 – Abholen von rvBEA Daten, 2.5.2.1 Geschäftsprozess 1 – Abholen von rvBEA Daten).</p>
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xreq:PackageBody

3.7.9 Element Base64CharSequence

xcpt:Base64CharSequence

<i>Inhalt</i>	Base64-Zeichenfolge
---------------	---------------------

Datentyp xs:base64Binary

Benutzt von xcpt:Data

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

3.8 Request Bestätigung der abgeholten rvBEA Daten

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 rvBEA Daten

Datentyp ##elements
Benutzt von xreq:TransportBody

3.8.5 Element ElementSequence

xcpt:ElementSequence

Inhalt Enthält das Element ConfirmationOfReceipt
Datentyp xs:any
Benutzt von xcpt:Data

3.8.6 Element ConfirmationOfReceipt

xmsg:ConfirmationOfReceipt

Inhalt Enthält die Informationen, welche ResponseIDs erfolgreich empfangen wurden
Datentyp xs:string
Benutzt von 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
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 verbleiben bereit zur Abholung bis diese bestätigt werden.

3.9 Response Bestätigung der abgeholten rvBEA Daten

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

Inhalt Enthält die relevanten Steuerungsinformationen, die zwischen Sender und der DSRV als Empfänger auszutauschen sind. Der Aufbau ist unter „Transport-Header der Response“ beschrieben.

Datentyp ##elements

Benutzt von 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

Inhalt Enthält einen leeren Body der Transportebene

Datentyp ##elements

Benutzt von xres:Transport

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

4 Anhang

4.1 StatusCodes

Statuscode	Text	Erläuterung	
000	Daten wurden erfolgreich verarbeitet	Daten wurden erfolgreich verarbeitet	I
C99	Die Nachricht kann nicht verarbeitet werden	Die eingegangene Nachricht kann nicht verarbeitet werden, da nicht alle notwendigen Daten mitgeliefert wurden.	E
C00	Die Nachricht entspricht dem geforderten Aufbau und kann im Fachverfahren verwendet werden	Sendung wurde erfolgreich bei der DSRV gespeichert	I
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 zur Zeit im Wartungsmodus. Bitte den Fehlertext beachten	E
E86	Empfang der Nachricht konnte nicht bestätigt werden.	Der Empfang der Datensendungen konnte nicht korrekt quittiert werden.	E
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
E94	Fehler beim Entpacken	Daten konnten nicht entpackt werden	E
E97	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
E98	Anfrage erfolgreich	Die Anfrage nach Anforderungen des Rentenversicherungsträgers auf dem Kommunikationsserver der DSRV war erfolgreich. Die Anforderungen können den einzelnen Packages entnommen 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
SpocWSDL1.4.zip	Zip-Datei mit der Webservice WSDL
Schema.zip	Zip-Datei mit den Schematas für die eXTra - Kommunikation

4.3 Abbildungsverzeichnis

Abb. 1 Grafischer Ablauf der Geschäftsprozesse Szenario 1, Kommunikationsinitiative durch AG	6
Abb. 2 Grafischer Ablauf der Geschäftsprozesse Szenario 2, Kommunikationsinitiative durch RV	7
Abb. 3 Aufbau Request-Header	17
Abb. 4 Aufbau Transport-Header Response	21
Abb. 5 Aufbau Package-Header	26
Abb. 6 Aufbau Request Arbeitgeberanforderung- und –antwort	32
Abb. 7 Aufbau Request Datenabholung	34
Abb. 8 Aufbau Response Datenabholung	37