



Institut für Qualitätssicherung und  
Transparenz im Gesundheitswesen

# **Spezifikation für die Nutzung der Sozialdaten bei den Krankenkassen**

Technische Dokumentation

Erstellt im Auftrag des  
Gemeinsamen Bundesausschusses

Stand: 30. November 2023 (V22)

---

# Impressum

**Thema:**

Spezifikation für die Nutzung der Sozialdaten bei den Krankenkassen.  
Technische Dokumentation

**Auftraggeber:**

Gemeinsamer Bundesausschuss

**Datum der Veröffentlichung:**

30. November 2023 (V22)

**Herausgeber:**

IQTIG – Institut für Qualitätssicherung  
und Transparenz im Gesundheitswesen

Katharina-Heinroth-Ufer 1  
10787 Berlin

Telefon: (030) 58 58 26-0  
Telefax: (030) 58 58 26-999

[info@iqtig.org](mailto:info@iqtig.org)

<https://www.iqtig.org>

**Hinweis:**

Aus Gründen der leichten Lesbarkeit wird im Folgenden auf eine geschlechtsspezifische Differenzierung verzichtet. Entsprechende Begriffe gelten im Sinne der Gleichbehandlung für beide Geschlechter.

# Inhaltsverzeichnis

|   |    |
|---|----|
| Tabellenverzeichnis.....  | 6  |
| Abbildungsverzeichnis.....  | 8  |
| Abkürzungsverzeichnis.....  | 11 |
| Glossar.....  | 13 |
| 1    Einleitung.....  | 15 |
| 1.1    Zielsetzung der Technischen Dokumentation.....   | 15 |
| 1.2    Spezifikationsbegriff.....   | 15 |
| 1.2.1    Gültigkeit von Spezifikationspaketen.....  | 17 |
| 1.3    Datenflüsse.....   | 17 |
| 1.3.1    Datenflüsse in der Qualitätssicherung gemäß DeQS-RL .....                                      | 18 |
| 1.3.2    Datenflüsse in der Methodenbewertung gemäß oKFE-RL.....  | 18 |
| 2    Prozesse bei den Krankenkassen.....  | 20 |
| 2.1    Datenselektion .....   | 21 |
| 2.1.1    Anforderungen an die Datenselektion .....  | 21 |
| 2.1.2    Referenzierung der Datenfelder .....   | 21 |
| 2.1.3    Selektionsstufen .....   | 26 |
| 2.1.4    Filter .....   | 29 |
| 2.1.5    Externe Parameter des Filters.....   | 33 |
| 2.1.6    Ergebnis der Selektion.....  | 34 |
| 2.1.7    Beispiel für Datenselektion .....  | 35 |
| 2.1.8    Beispiel zwei für Datenselektion .....   | 37 |
| 2.2    Datenexport .....  | 39 |
| 2.2.1    Ziel und Umfang des Datenexports.....  | 39 |
| 2.2.2    Voraussetzungen für den Datenexport.....   | 39 |
| 2.2.3    Exportdateien.....   | 40 |
| 2.2.4    XML-Struktur .....   | 42 |
| 2.2.5    Überleitung von den Daten der Leistungserbringer zu den Exportdaten<br>(Mapping-Tabelle) ..... | 43 |
| 2.2.6    Beispiel für Datenexport .....   | 45 |
| 2.2.7    Segmentierung .....  | 48 |

|       |   |     |
|-------|---|-----|
| 2.2.8 | Nullmeldung .....   | 48  |
| 2.3   | Datenprüfung .....  | 49  |
| 2.3.1 | Struktur- und Formatprüfungen .....                             | 49  |
| 2.3.2 | Plausibilitätsregeln .....                                      | 51  |
| 2.3.3 | Administrative Prüfungen .....                                  | 53  |
| 2.4   | Verschlüsselung und Übermittlung an die DAS-KK .....            | 57  |
| 2.4.1 | Vorgaben für die Dateiverschlüsselung und Übermittlung.....     | 57  |
| 2.4.2 | Konsequenzen für den Datenfluss für Sozialdaten .....           | 58  |
| 2.4.3 | Dateiverschlüsselung.....                                       | 60  |
| 2.4.4 | Übermittlung .....  | 60  |
| 2.5   | Rückprotokollierung.....  | 62  |
| 2.5.1 | Empfangsbestätigung der DAS-KK.....                             | 62  |
| 2.5.2 | Datenflussprotokoll .....                                       | 63  |
| 2.6   | Bestätigung und Aufstellung gemäß DeQS-RL .....                 | 65  |
| 2.7   | Informationen über die Güte gelieferter Daten .....             | 68  |
| 3     | Prozesse bei der Datenannahmestelle.....                        | 71  |
| 4     | Prozesse bei der Vertrauensstelle.....                          | 78  |
| 5     | Prozesse bei der (Bundes-)Auswertungsstelle .....               | 81  |
| 6     | Anhang .....  | 82  |
| 6.1   | Herkunft der Datenbestände nach Abrechnungskontext.....         | 82  |
| 6.1.1 | Datenbestände und Lieferfristen .....                           | 82  |
| 6.1.2 | Export der Datenbestände .....                                  | 83  |
| 6.2   | Datensatzbeschreibung.....                                      | 84  |
| 6.2.1 | Hierarchie der Elemente für den Export .....                    | 84  |
| 6.2.2 | Elemente und Attribute der Nutzdaten in der QS-Datei.....       | 86  |
| 6.2.3 | Elemente und Attribute der Nutzdaten in der PID-Datei.....      | 103 |
| 6.2.4 | Schlüssel .....   | 104 |
| 6.3   | Spezifikationsdatenbank .....                                   | 112 |
| 6.4   | XML-Schemata .....  | 115 |
| 6.4.1 | Kompositionsmodell.....   | 115 |
| 6.4.2 | Schnittstellen.....   | 116 |
| 6.4.3 | XML-Struktur des Datencontainers (KK/DAS-KK-Schnittstelle)..... | 117 |

|       |   |     |
|-------|---|-----|
| 6.4.4 | XML-Struktur der QS-Daten .....                                 | 125 |
| 6.4.5 | XML-Struktur der PID .....                                      | 132 |
| 6.4.6 | XML-Struktur der gemeinsamen QS- und PID-Dateien .....          | 133 |
| 6.4.7 | XML-Struktur der Rückprotokollierung (Datenflussprotokoll)..... | 134 |
| 6.4.8 | Versionierung der XML-Schemata.....                             | 142 |
| 6.5   | XML-Verschlüsselung .....                                       | 142 |
| 6.6   | Die Rückprotokollierung .....                                   | 142 |
| 6.6.1 | Empfangsbestätigung.....  | 143 |
| 6.6.2 | Datenflussprotokoll.....  | 143 |
| 6.6.3 | Fehlermeldungen .....   | 144 |
| 6.6.4 | Prüfergebnisse und Ergebnisprotokollierung .....                | 146 |
| 6.7   | Verfahrensbezogene Spezifikationen .....                        | 149 |
| 6.7.1 | Allgemeines .....   | 150 |
| 6.7.2 | Modul .....   | 151 |
| 6.7.3 | Selektionsfilter .....  | 151 |
| 6.7.4 | Kodelisten.....   | 151 |
| 6.7.5 | Wertebereiche .....   | 152 |
| 6.7.6 | Plausibilitätsregeln .....                                      | 152 |
| 6.7.7 | Lieferfristen .....   | 152 |
| 6.7.8 | Verwendete Datenfelder.....                                     | 153 |
| 6.8   | Releaseplanung .....  | 153 |
| 6.8.1 | Definition von Begriffen .....                                  | 153 |
| 6.8.2 | Informationen zu Beginn des Veröffentlichungsprozesses.....     | 154 |
| 6.8.3 | Entscheidungsverfahren.....                                     | 155 |
| 6.8.4 | Praktische Umsetzung.....                                       | 155 |
| 6.8.5 | Termine für die Releaseplanung .....                            | 155 |
| 6.8.6 | Fehler- und Changemanagement.....                               | 156 |

## Tabellenverzeichnis

|   |     |
|---|-----|
| Tabelle 1: Komponenten eines Spezifikationspakets.....  | 16  |
| Tabelle 2: Struktur der Tabelle <code>Version</code> .....  | 17  |
| Tabelle 3: Struktur der Abfrage <code>Referenz</code> .....   | 22  |
| Tabelle 4: Inhalt der Tabelle <code>Funktion</code> .....   | 23  |
| Tabelle 5: Struktur der Tabelle <code>SchlüsselWert</code> .....  | 26  |
| Tabelle 6: Ausschnitt aus der Tabelle <code>ModulErfassungsjahr</code> .....                              | 29  |
| Tabelle 7: Zu den Operatoren des Pseudocodes analoge SQL-Operatoren .....                                 | 30  |
| Tabelle 8: Operatoren des Pseudocodes .....   | 31  |
| Tabelle 9: Verwendung der Mengen- und Wertebereichoperatoren .....  | 32  |
| Tabelle 10: Struktur der Tabelle <code>Codeliste</code> .....   | 33  |
| Tabelle 11: Struktur der Tabelle <code>Wertebereich</code> .....  | 34  |
| Tabelle 12: Referenz auf ein Datenfeld.....   | 36  |
| Tabelle 13: Datenkategorien im Datenfluss.....  | 40  |
| Tabelle 14: XML-Schemata für die Krankenkassen.....   | 42  |
| Tabelle 15: Struktur der Abfragen <code>MappingTabellePID</code> bzw. <code>MappingTabelleQS</code> ..... | 43  |
| Tabelle 16: Beispiel für ein selektiertes Datenfeld .....   | 45  |
| Tabelle 17: Beispiel für ein berechnetes Datenfeld .....  | 47  |
| Tabelle 18: Struktur der Abfrage <code>FeldDefinition</code> .....  | 49  |
| Tabelle 19: Beispiel der Definition eines Feldes .....  | 50  |
| Tabelle 20: Struktur der Abfrage <code>RegelDatenfeld</code> .....  | 51  |
| Tabelle 21: Arten der Plausibilitätsregeln.....   | 52  |
| Tabelle 22: Struktur der Abfrage <code>PruefschrittPruefung</code> .....                                  | 53  |
| Tabelle 23: Administrative Prüfungen für die Krankenkassen.....   | 55  |
| Tabelle 24: Beispiele administrativer Prüfungen vor dem Export .....                                      | 56  |
| Tabelle 25: Teildateien der Datenlieferungen .....  | 58  |
| Tabelle 26: Verschlüsselung der Exportdateien.....  | 60  |
| Tabelle 27: Verfahrenskennung im Auftragsatz .....  | 61  |
| Tabelle 28: Status des Dokuments.....   | 63  |
| Tabelle 29: Status des Datensatzes.....   | 64  |
| Tabelle 30: Kodierung der Landesgeschäftsstellen der Kassenärztlichen Vereinigungen .....                 | 74  |
| Tabelle 31: Relevante XML-Schemata für die DAS-KK .....   | 76  |
| Tabelle 32: XML-Schemata für die VST .....  | 79  |
| Tabelle 33: Übersicht der Datenbestände .....   | 82  |
| Tabelle 34: Ziel-Elemente der Datenbestände beim Export .....   | 83  |
| Tabelle 35: Tabellen in der Spezifikationsdatenbank .....   | 112 |
| Tabelle 36: Abfragen in der Spezifikationsdatenbank .....   | 114 |
| Tabelle 37: Verwendbare Schemata, Ablageort und Verwendungszweck.....                                     | 116 |
| Tabelle 38: Symbole in den XML-Schema-Diagrammen .....  | 117 |
| Tabelle 39: Attribute: <code>root</code> -Element.....  | 119 |

|  |     |
|--|-----|
| Tabelle 40: Attribute: header-Element .....  | 120 |
| Tabelle 41: Attribute: body-Element .....  | 123 |
| Tabelle 42: Attribute: case-Element.....   | 124 |
| Tabelle 43: Attribute: sent-Element.....   | 125 |
| Tabelle 44: Elemente: Beispielverfahren .....  | 126 |
| Tabelle 45: Attribute: versicherter-Element.....   | 127 |
| Tabelle 46: Attribute: quartalsstammdaten-Element .....                                    | 127 |
| Tabelle 47: Attribute: fall_301-Element .....  | 128 |
| Tabelle 48: Attribute: care_provider-Element .....   | 129 |
| Tabelle 49: mögliche Werte für Anzahl in einer Aufstellung .....                           | 131 |
| Tabelle 50: Attribute: protocol-Element .....  | 136 |
| Tabelle 51: Attribute: validation_provider-Element.....                                    | 136 |
| Tabelle 52: Attribute: validation_item-Element.....  | 137 |
| Tabelle 53: Attribute: status_document-Element .....                                       | 138 |
| Tabelle 54: Attribute: Status-Element .....  | 139 |
| Tabelle 55: Attribute Error-Element .....  | 140 |
| Tabelle 56: Attribute: Error-Element .....   | 140 |
| Tabelle 57: Aufbau und Attribute des Elements <error> .....                                | 144 |
| Tabelle 58: Kind-Elemente des <error>-Elements.....  | 144 |
| Tabelle 59: Verwendete XPath-Syntax.....   | 145 |
| Tabelle 60: Meilensteine der Releaseplanung der Spezifikation 2024 für den Regelbetrieb... | 156 |

# Abbildungsverzeichnis

|  |     |
|--|-----|
| Abbildung 1: Datenfluss im Rahmen der DeQS-RL bei den Leistungserbringern .....  | 18  |
| Abbildung 2: Datenfluss im Rahmen der DeQS-RL bei den Krankenkassen .....  | 18  |
| Abbildung 3: Datenfluss im Rahmen der oKFE-RL bei den Leistungserbringern .....  | 18  |
| Abbildung 4: Datenfluss im Rahmen der oKFE-RL bei den Krankenkassen .....  | 19  |
| Abbildung 5: Prozesse bei den Krankenkassen .....  | 20  |
| Abbildung 6: Referenz-Tabelle .....  | 22  |
| Abbildung 7: Beispiel Filterstufen – Ausgangssituation .....   | 27  |
| Abbildung 8: Beispiel Filterstufen – Patientenfilter .....   | 28  |
| Abbildung 9: Beispiel Filterstufen – Leistungs- und Medikationsfilter .....  | 28  |
| Abbildung 10: Funktion der Mapping-Tabelle .....   | 40  |
| Abbildung 11: Mapping-Tabellen in der Spezifikationsdatenbank .....  | 43  |
| Abbildung 12: Attribut Datentyp SCHLUESSEL .....   | 51  |
| Abbildung 13: Serieller Datenfluss mit PID gemäß DeQS-RL .....   | 59  |
| Abbildung 14: Serieller Datenfluss ohne PID gemäß DeQS-RL .....  | 59  |
| Abbildung 15: Mappen einer Fehlermeldung eines Rückprotokolls auf eine Prüfung .....   | 65  |
| Abbildung 16: Bestätigung & Aufstellung für den Zeitraum 1. Jan bis 31. Dez 2018 (mehrere<br>Module können in einer Bestätigung und einer Aufstellung aufgeführt sein) ..... | 66  |
| Abbildung 17: Element header mit Aufstellung im Attribut /root/header/@data_flow .....   | 67  |
| Abbildung 18: Beispiel des Elementes body für eine Aufstellung .....   | 67  |
| Abbildung 19: Prozesse bei der DAS-KK .....  | 71  |
| Abbildung 20: <data_container>-Element mit enthaltenem <pid_container>-Element .....   | 72  |
| Abbildung 21: Transformation des Headers durch die DAS-KK .....  | 73  |
| Abbildung 22: Aufbau der BSNR .....  | 75  |
| Abbildung 23: Transformation der Bestätigung durch Entfernen der vertretungsberechtigten<br>Person .....   | 77  |
| Abbildung 24: Prozesse bei der VST .....   | 78  |
| Abbildung 25: <case>-Element nach Verarbeitung durch die VST .....   | 78  |
| Abbildung 26: Umgang mit Widersprüchen gemäß § 8 oKFE-RL .....   | 79  |
| Abbildung 27: Prozesse bei der BAS .....   | 81  |
| Abbildung 28: Hierarchie der Elemente für den Export – Element: case .....   | 85  |
| Abbildung 29: Hierarchie der Elemente für den Export – Elemente: versicherter, fall_301,<br>fall_khambo .....  | 85  |
| Abbildung 30: Hierarchie der Elemente für den Export – Elemente: fall_295, fall_300, fall_302<br>.....   | 86  |
| Abbildung 31: Hierarchie der Elemente für den Export – Elemente: care_provider, position... ..   | 86  |
| Abbildung 32: Dateiordner der Schnittstellen-Schemata .....  | 115 |
| Abbildung 33: root-Element und Kind-Elemente header und body .....   | 119 |
| Abbildung 34: Header-Element und Kind-Element protocol .....   | 120 |
| Abbildung 35: body-Element und Kind-Elemente case und statistic .....  | 123 |



|   |     |
|---|-----|
| Abbildung 36: case-Element und Kind-Element qs_data .....   | 124 |
| Abbildung 37: Element statistic und Kind-Element sent .....   | 125 |
| Abbildung 38: Element sent.....   | 125 |
| Abbildung 39: Datentyp qs_data_type .....   | 125 |
| Abbildung 40: Element BSP und Kind-Elemente versicherter, fall_301,<br>fall_khambo, fall_295, fall_300 und fall_302.....          | 126 |
| Abbildung 41: Element versicherter .....  | 127 |
| Abbildung 42: Element quartalsstammdaten.....   | 127 |
| Abbildung 43: Element fall_301 und Kind-Elemente care_provider,<br>hauptdiagnose, nebendiagnose, prozedur und fachabteilung ..... | 128 |
| Abbildung 44: Element care_provider und Kind-Element nummer.....  | 129 |
| Abbildung 45: qs_data_aufst_type für Aufstellung .....  | 130 |
| Abbildung 46: modulkennung_type.....  | 130 |
| Abbildung 47: erfassungsjahr_type.....  | 131 |
| Abbildung 48: lieferung_type.....   | 131 |
| Abbildung 49: bestaetigung_type.....  | 132 |
| Abbildung 50: bestaetigungstext_type.....   | 132 |
| Abbildung 51: vertretungsberechtigteperson_type .....   | 132 |
| Abbildung 52: XML-Struktur der PID-Datei.....   | 133 |
| Abbildung 53: Zusammenführung der QS- und PID-Dateien.....  | 133 |
| Abbildung 54: Zusammenführung der QS- und PID-Datensätze .....  | 134 |
| Abbildung 55: Schema des Datenflussprotokolls „response_DAS_KK.xsd“ .....   | 135 |
| Abbildung 56: Element protocol und Kind-Elemente validation_provider und<br>status_document.....                                  | 135 |
| Abbildung 57: Element validation_provider und Kind-Element validation_item<br>.....   | 136 |
| Abbildung 58: Element validation_item und Kind-Element status.....  | 137 |
| Abbildung 59: Element status_document .....   | 138 |
| Abbildung 60: Element status und Kind-Element error .....   | 139 |
| Abbildung 61: Element error und Kind-Elemente rule_id, rule_type, list und error_message  | 139 |
| Abbildung 62: Element protocol und Kind-Elemente validation_item und<br>status_case .....   | 140 |
| Abbildung 63: Element INFO und Vorfahren-Elemente qs_data sowie case.....   | 141 |
| Abbildung 64: Beispiel einer Empfangsbestätigung der DAS-KK an eine Krankenkasse .....  | 143 |
| Abbildung 65: Beispiel eines <error>-Elements.....  | 145 |
| Abbildung 66: Beziehungen zwischen <validation_item> im header und <validation_item> im<br>body über die id .....                 | 147 |
| Abbildung 67: Allgemeine Angaben.....   | 150 |
| Abbildung 68: Versionierung.....  | 150 |
| Abbildung 69: Modulbeschreibung.....  | 151 |
| Abbildung 70: Filter .....  | 151 |
| Abbildung 71: Kodelisten .....  | 151 |

|   |     |
|---|-----|
| Abbildung 72: Wertebereiche .....           | 152 |
| Abbildung 73: Plausibilitätsregeln .....    | 152 |
| Abbildung 74: Lieferfristen.....            | 152 |
| Abbildung 75: Verwendete Datenfelder .....  | 153 |
| Abbildung 76: Begriffe Releaseplanung ..... | 153 |

## Abkürzungsverzeichnis

| Abkürzung | Beschreibung   |
|-----------|--|
| AES       | Advanced Encryption Standard   |
| BAS       | (Bundes-)Auswertungsstelle   |
| BfArM     | Bundesinstitut für Arzneimittel und Medizinprodukte                              |
| DA        | Datenaustausch   |
| DAS       | Datenannahmestelle   |
| DAS-KK    | Datenannahmestelle für die Krankenkassen   |
| DeQS-RL   | Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung     |
| DFP       | Datenflussprotokoll  |
| G-BA      | Gemeinsamer Bundesausschuss  |
| GFL       | Gesundheitsforen Leipzig GmbH  |
| GKV-SV    | Spitzenverband Bund der Krankenkassen (Gesetzliche Krankenversicherung)          |
| GUID      | Globally Unique Identifier   |
| ICD       | International Statistical Classification of Diseases and Related Health Problems |
| IKNR      | Institutionskennzeichen  |
| IQTIG     | Institut für Qualitätssicherung und Transparenz im Gesundheitswesen              |
| KID       | Krankenkassenidentifizierende Daten  |
| KK        | Krankenkasse   |
| KKS       | Krankenkassenkommunikationssystem  |
| LID       | Leistungserbringeridentifizierende Daten   |
| oKFE-RL   | Richtlinie für organisierte Krebsfrüherkennungsprogramme                         |
| OPS       | Operationen- und Prozedurenschlüssel   |
| PID       | Patientenidentifizierende Daten  |
| PKCS      | Public Key Cryptography Standard   |
| PKI       | Public-Key-Infrastruktur   |
| PZN       | Pharmazentralnummer  |
| QS        | Qualitätssicherung   |

| Abkürzung | Beschreibung  |
|-----------|---|
| QSKH-RL   | Richtlinie (des G-BA) gemäß §§ 136 ff. V i.V.m. § 135a SGB V über Maßnahmen der Qualitätssicherung für nach § 108 SGB V zugelassene Krankenhäuser |
| RL        | Richtlinie  |
| SECON     | Security-Schnittstelle für den Datenaustausch im Gesundheits- und Sozialwesen   |
| SFTP      | SSH File Transfer Protocol  |
| SGB       | Sozialgesetzbuch  |
| VST       | Vertrauensstelle  |
| W3C       | World Wide Web Consortium   |
| XML       | Extensible Markup Language  |
| XSLT      | Extensible Stylesheet Language Transformations  |

## Glossar

| Begriff                            | Beschreibung  |
|------------------------------------|---|
| Auslösekriterien                   | Algorithmus zur Auslösung der Dokumentationspflicht bei Leistungserbringern (Filter).   |
| Erfassungsjahr                     | <p>Das Erfassungsjahr bezeichnet das Jahr, in dem die Indexleistung stattfand. Bezogen auf die Spezifikation für die Nutzung der Sozialdaten bei den Krankenkassen meint der Begriff Erfassungsjahr das Jahr des Entlassungsdatums bei stationärer Behandlung bzw. das Jahr des Behandlungs(ende)datums oder den Tag des letzten Abrechnungsdatums bei ambulanten Eingriffen der jeweils dokumentationspflichtigen Behandlungsfälle.</p> <p>Der Zeitraum des Indexjahres (Zeitraum für den Patientenfilter) kann bei ausgewählten Modulen mehrere Jahre umfassen (z. B. bei NTXS, HTXS, LUTXS und LTXS). Hierbei bezieht sich das Erfassungsjahr auf das aktuellste, also das zuletzt in der Spezifikation aufgenommene Erfassungsjahr.</p> |
| Erhebungsinstrument                | Spezifizierung der Art und Weise der Erhebung von Daten (für die externe Qualitätssicherung) bzw. der Erschließung/Nutzbarmachung von Datenquellen. Beispiele: Dokumentationsbögen zur Datenerhebung durch die Leistungserbringer (QS-Dokumentation), Fragebögen zur Datenerhebung bei Patienten, technische Spezifikationen zur Erhebung von Routinedaten (z. B. aus den Sozialdaten bei den Krankenkassen).   |
| Gemeinsamer Bundesausschuss (G-BA) | Oberstes Beschlussgremium der gemeinsamen Selbstverwaltung der Ärzte, Zahnärzte, Psychotherapeuten, Krankenhäuser und Krankenkassen in Deutschland. Er bestimmt in Form von Richtlinien den Leistungskatalog der Gesetzlichen Krankenversicherung (GKV) für mehr als 70 Millionen Versicherte und legt damit fest, welche Leistungen der medizinischen Versorgung von der GKV erstattet werden. Darüber hinaus beschließt der G-BA Maßnahmen der Qualitätssicherung für den ambulanten und stationären Bereich des Gesundheitswesens.   |
| ICD-GM                             | engl.: International Classification of Diseases – German Modification. Auf deutsche Verhältnisse angepasste Version der ICD, die vom Deutschen Institut für Medizinische Dokumentation und Information (BfArM) erstellt wird.   |
| IK-Nummer                          | Institutionskennzeichen. Wird gemäß § 293 SGB V bei der Datenübermittlung zwischen den gesetzlichen Krankenkassen und den Leistungserbringern als eindeutige Identifizierung verwendet.   |
| Indexleistung                      | Medizinische Leistung, die den Startpunkt für ein Verfahren (die Auslösung) genau definiert (im Regelfall OPS-Kodes).   |

| Begriff   | Beschreibung  |
|---|---|
| Krankenkassen-identifizierende Daten (KID)      | Daten, die eindeutig eine bestimmte Krankenkasse identifizieren (z. B. Institutionskennzeichen der Krankenkasse).   |
| Leistungserbringer-identifizierende Daten (LID) | Daten, die eindeutig einen bestimmten Leistungserbringer identifizieren (z. B. Institutionskennzeichen oder Betriebsstättennummer).   |
| Matching  | Eindeutiger Verweis von Ausgangs- auf Zieldaten (1:1-Beziehung)   |
| Operationen- und Prozedurenschlüssel (OPS)      | Kodierungssystem für medizinische Leistungen bzw. verbindlicher Abrechnungsschlüssel zwischen medizinischen Leistungserbringern und Kostenträgern.  |
| Patientenidentifizierende Daten (PID)           | Daten, die eindeutig einen bestimmten Versicherten identifizieren (z. B. Versichertennummer).   |
| Plausibilitätsprüfung                           | Statistisches Verfahren, mit dem die Dokumentationsdaten auf erlaubte und/oder fehlende Werte, Widerspruchsfreiheit, Werteverteilung und bekannte Korrelationen geprüft werden.   |
| Pseudocode                                      | Programmcode, der das zugrundeliegende Prinzip eines Algorithmus beschreibt, selbst aber nicht lauffähig ist. Er dient zur Veranschaulichung, unabhängig von der konkret zu verwendenden Programmiersprache.  |
| Filter  | Der Filter ist ein Algorithmus, der auf Grundlage festgelegter Kriterien (zumeist Abrechnungsdaten, z. B. § 301 Daten) die für die Qualitätssicherung durch die Leistungserbringer zu dokumentierenden Patienten und deren Daten „filtert“. Die Kriterien hierzu werden in einer Spezifikation definiert. |
| Selektion                                       | Auswahl bestimmter Daten aus einem Datenpool anhand zuvor definierter Kriterien.  |
| XSLT  | Extensible Stylesheet Language Transformations. Programmiersprache zur Transformation von XML-Dokumenten in andere XML-Dokumente oder andere Dokumentformate wie HTML. Im QS-Kontext kann es auch für Datenprüfung und Protokollerstellung verwendet werden.  |

# 1 Einleitung

## 1.1 Zielsetzung der Technischen Dokumentation

Nach § 299 Absatz 1a SGB V sind die Krankenkassen befugt und verpflichtet, einen zweckgebundenen Ausschnitt aus den von ihnen gem. § 284 SGB V erhobenen Daten („Sozialdaten bei den Krankenkassen“) für die gesetzliche Qualitätssicherung (QS), also die Durchführung von Verfahren gemäß §§ 136 ff., und für die Methodenbewertung gemäß §§ 135, 137c und 137e SGB V zur Verfügung zu stellen. Aus datenschutzrechtlichen Gründen und den allgemeinen Grundsätzen der Datensparsamkeit und der Datenvermeidung (§ 3a BDSG) folgend, müssen die für die Zwecke der Qualitätssicherung und der Methodenbewertung erforderlichen Daten und ihre jeweiligen Empfänger spezifiziert und in Richtlinien des G-BA festgelegt werden.

Das vorliegende Dokument ist Teil der Allgemeinen Spezifikation zur Nutzung von Daten gemäß § 299 Abs. 1a SGB V. Es beschreibt die Auswahl von Daten, die Erzeugung von XML-Exportdateien, deren Aufbereitung für eine Übertragung und die Interpretation des Rückprotokolls.

Die XML-Exportdatei verwendet die Daten, die nach den Vorgaben des G-BA aus den Sozialdaten bei den Krankenkassen für die Qualitätssicherung an das Institut für Qualitätssicherung und Transparenz im Gesundheitswesen (IQTIG), dem Institut nach § 137a SGB V sowie für die Methodenbewertung an die Gesundheitsforen Leipzig GmbH (GFL) als die dafür zuständige Auswertungsstelle zu übermitteln sind.

Im vorliegenden Dokument werden die Begriffe QS-Daten und QS-Datei stellvertretend für Inhalte im Kontext der vorgenannten gesetzlichen Qualitätssicherung (konkret gem. DeQS-RL) sowie der vorgenannten Methodenbewertung (konkret der Programmbeurteilung gem. oKFE-RL) verwendet.

## 1.2 Spezifikationsbegriff

Die Spezifikation ist die Gesamtheit aller Vorgaben, nach denen für die einzelnen Verfahren die Selektion der benötigten Daten, deren Export sowie die Übermittlung erfolgen sollen, bezogen auf das Jahr der auslösenden Leistung (Indexleistung), das sogenannte Erfassungsjahr. Das Erfassungsjahr bezeichnet also das Jahr, in dem die Indexleistung stattfand. Bezogen auf die Spezifikation für die Nutzung der Sozialdaten bei den Krankenkassen meint der Begriff Erfassungsjahr das Jahr des Entlassungsdatums bei stationärer Behandlung bzw. das Jahr des Behandlungs(ende)datums oder den Tag des letzten Abrechnungsdatums bei ambulanten Eingriffen der jeweils dokumentationspflichtigen Behandlungsfälle. Das Erfassungsjahr deckt sich mit einem Kalenderjahr. Der Zeitraum des Indexjahres (Zeitraum für den Patientenfilter) kann bei ausgewählten Modulen mehrere Jahre enthalten (z. B. bei NTXS, HTXS, LUTXS und LTXS). Hierbei bezieht sich das Erfassungsjahr auf das aktuellste, also das zuletzt in der Spezifikation aufgenommene Erfassungsjahr.

Die Spezifikation wird jeweils als Spezifikationspaket zum Download zur Verfügung gestellt, das aus einzelnen Komponenten besteht. Komponenten sind dabei die Spezifikationsdatenbank, Technische Dokumentation, XML-Schemata und weitere Anwendersichten:

*Tabelle 1: Komponenten eines Spezifikationspakets*

| Komponente               | Inhalt   |
|--------------------------|--|
| Spezifikationsdatenbank  | Definitionen zur Datenselektion, Plausibilitätsprüfung und Datenübermittlung   |
| Technische Dokumentation | Dokumentation der Inhalte der Datenbank und Beschreibung der Selektion von Daten, der Erzeugung von XML-Exportdateien, deren Aufbereitung für eine Übertragung und der Interpretation des Rückprotokolls.    |
| XML-Schemata             | Definitionen für Exportdateien und Protokolle im Datenfluss  |
| Komponentenübersicht     | Jedem Paket liegt eine Auflistung der enthaltenen Komponenten mit Bezug zu Versionsnummer und Veröffentlichungsdatum bei (CSV).  |
| Übersichtänderungen      | Übersicht über die Änderungen zu allen vorhergehenden Versionen. Die Übersicht wird als Text-Datei gepflegt.   |
| Filterbeschreibungen     | Verfahrensspezifische XML-Dateien mit Informationen zu Filterregeln und verwendeten Kodelisten   |
| Plausibilitätsregeln     | Verfahrensspezifische Excel-Dateien mit Informationen zu Plausibilitätsprüfungen für die einzelnen Datenfelder   |
| Benennungsschema         | Jedem Paket liegt ein Dokument bereit, welches über das Benennungsschema der Spezifikation informiert. Das Benennungsschema ist für alle Spezifikationen des IQTIG gültig und wird demnach zentral gepflegt. |

Jede Komponente dieser Spezifikationspakete kann auf diese Weise unabhängig und parallel zu anderen aktualisiert werden.

Die technische Umsetzung der Anforderungen an die Verfahren zur Datenselektion, Plausibilitätsprüfung und Datenübermittlung werden in einer Spezifikationsdatenbank definiert. Sie enthält zusammen mit der Technischen Dokumentation alle Informationen, die die Krankenkassen für die Selektion und Übermittlung der Sozialdaten brauchen. Von der Spezifikationsdatenbank automatisch abgeleitet, können für bestimmte, prozessspezifische Anwendungen weitere Sichten bereitgestellt werden.

Sowohl die Spezifikationspakete als auch die einzelnen Komponenten werden nach einem einheitlichen Muster benannt, das bereits im Namen die relevanten Informationen wie Betriebsart (Regel-/Testbetrieb), Erfassungsjahr und Versionierung enthält. Dieses Schema ist an das Benennungsschema der QS-Spezifikationen für Leistungserbringer angelehnt und wird im nächsten Abschnitt detailliert erläutert. Durch die Versionierung sowohl auf der Ebene der Pakete als auch auf der Ebene der Komponenten ist gewährleistet, dass der aktuelle Stand leicht ersichtlich ist.



Zudem wird die Kommunikation über die anzuwendenden Bestandteile der Spezifikation erleichtert.

Jedem Paket liegt eine Auflistung der enthaltenen Komponenten und eine Übersicht über die Änderungen zur vorhergehenden Version bei.

Das Benennungsschema für Spezifikationspakete und Spezifikationskomponenten wurde ab der Sozialdatenspezifikation 2023 V01 angepasst. Das angepasste Benennungsschema ist für alle IQTIG-Spezifikationen gültig und wird seit dem 30.06.2022 separat als Spezifikationskomponente (PDF-Dokument) zur Verfügung gestellt (siehe Tabelle 1). Demnach wurden die entsprechenden Textabschnitte aus der Technischen Dokumentation entfernt.

### 1.2.1 Gültigkeit von Spezifikationspaketen

In der Spezifikationsdatenbank für Sozialdaten sind in der Tabelle `Version` Informationen hinterlegt, welche Klarheit bezüglich der Gültigkeit des jeweiligen Spezifikationspaketes bringen. Mit jedem Release erfolgt ein neuer Eintrag in diese Tabelle. Beschreibende Erläuterungen der einzelnen Feldinhalte können nachstehender Tabelle entnommen werden.

Tabelle 2: Struktur der Tabelle `Version`

| Datenfeld                 | Beschreibende Erläuterung   |
|---------------------------|---|
| <code>id</code>           | Eindeutige Nummer   |
| <code>name</code>         | Bezeichnung des Spezifikationspakets entsprechend dem Benennungsschema für Spezifikationspakete   |
| <code>bezeichnung</code>  | Kurzbeschreibung des Spezifikationspaketes  |
| <code>ab</code>           | Beschreibt den Zeitpunkt, ab wann dieses Release Gültigkeit erlangt (und somit das vorige Release abgelöst wird)  |
| <code>bis</code>          | Entspricht der Angabe im Feld <code>ab</code> eines nachfolgenden Releases. Bei neuen Einträgen bleibt diese Angabe zunächst leer und wird erst im Rahmen eines Folgereleases befüllt. Ein Spezifikationspaket bleibt demzufolge gültig bis es durch ein aktuelleres Release zu einem späteren Zeitpunkt abgelöst wird. |
| <code>pub</code>          | Datum der Veröffentlichung. Diese Angabe liegt zeitlich stets mindestens 3 Monate vor der Angabe im Feld <code>ab</code>  |
| <code>fkVorversion</code> | Referenz auf das vormals gültige Release  |
| <code>fkVersStatus</code> | Einordnung des Paketinhalts (Entwicklung, final, Service-Release)   |

## 1.3 Datenflüsse

Dieser Abschnitt vermittelt einen Überblick über die aktuellen Datenflüsse im Rahmen der Qualitätssicherung gemäß DeQS-RL und der Methodenbewertung gemäß oKFE-RL.

### 1.3.1 Datenflüsse in der Qualitätssicherung gemäß DeQS-RL

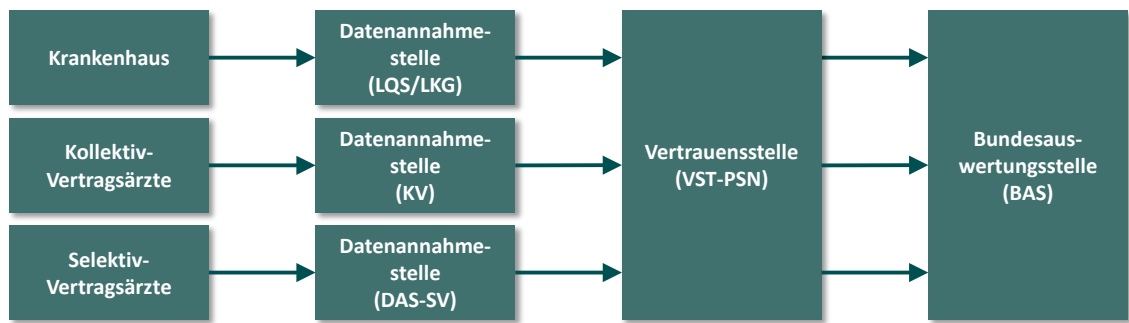


Abbildung 1: Datenfluss im Rahmen der DeQS-RL bei den Leistungserbringern

Für die Sozialdaten bei den Krankenkassen wurde ein weiterer Datenfluss unter der Maßgabe der DeQS-RL entwickelt. Dieser entspricht in seinen Grundzügen dem Datenfluss der Daten von den Leistungserbringern. Abweichende technische Festlegungen werden im vorliegenden Dokument ausführlich beschrieben. Die Datenflüsse unterscheiden sich im Wesentlichen in der zuständigen Datenannahmestelle, der technischen Beschaffenheit der Daten und deren Transport- und Verschlüsselungsverfahren. Die Einbindung einer pseudonymisierenden Vertrauensstelle ist nur in Verfahren notwendig, welche PID bzw. die daraus erzeugten Pseudonyme zur Zusammenführung von Datensätzen benötigen. Bei Nicht-PID-Verfahren erfolgt der Transport der QS-Daten direkt von der Datenannahmestelle für die Krankenkassen (DAS-KK) zur (Bundes-)Auswertungsstelle (BAS). Die Bundesauswertungsstelle für Verfahren gemäß DeQS-RL ist das IQTIG.

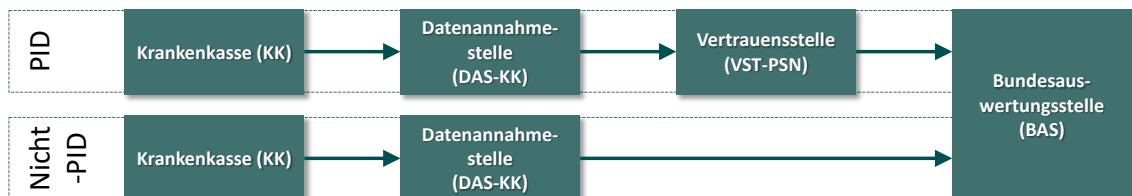


Abbildung 2: Datenfluss im Rahmen der DeQS-RL bei den Krankenkassen

### 1.3.2 Datenflüsse in der Methodenbewertung gemäß oKFE-RL

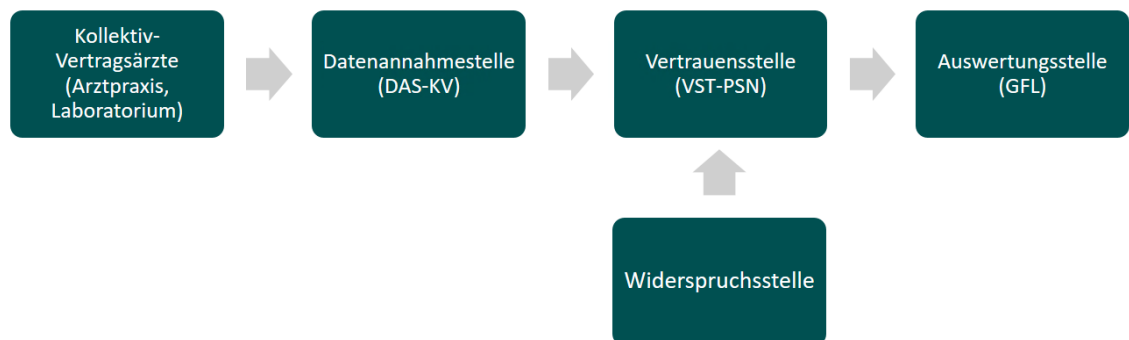


Abbildung 3: Datenfluss im Rahmen der oKFE-RL bei den Leistungserbringern

Für die Sozialdaten bei den Krankenkassen wurde ein weiterer Datenfluss unter der Maßgabe der oKFE-RL entwickelt. Dieser entspricht in seinen Grundzügen dem Datenfluss der Daten von den Leistungserbringern. Abweichende technische Festlegungen werden im vorliegenden Dokument ausführlich beschrieben. Die Datenflüsse zwischen fallbezogener Dokumentation und der Nutzung der Sozialdaten bei den Krankenkassen unterscheiden sich im Wesentlichen in der zuständigen Datenannahmestelle, der technischen Beschaffenheit der Daten und deren Transport- und Verschlüsselungsverfahren. Die Einbindung einer pseudonymisierenden Vertrauensstelle wird sowohl für den Datenfluss bei den Leistungserbringern als auch für den Datenfluss bei den Krankenkassen benötigt, da es sich in jedem Fall um PID-Datensätze handelt.

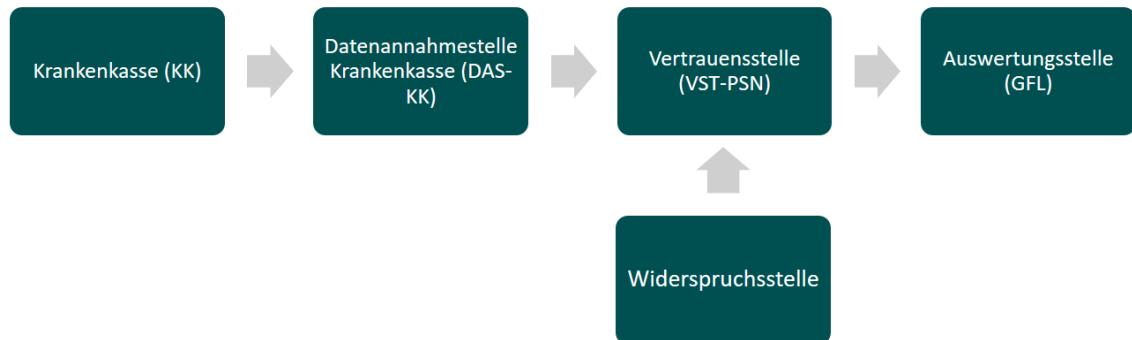


Abbildung 4: Datenfluss im Rahmen der oKFE-RL bei den Krankenkassen

Gemäß § 8 oKFE-RL können Versicherte Widerspruch gegen die Verarbeitung der Daten zum Zweck der Programmbeurteilung gegenüber einer vom G-BA bestimmten zentralen Widerspruchsstelle erklären. Von der Widerspruchsstelle werden eventuelle Widersprüche an die Vertrauensstelle gemeldet. Sollte ein Widerspruch für einen Versicherten vorliegen, muss die Vertrauensstelle die Programmbeurteilungsdaten (Inhalte des XML-Elementes `qs_data`) für diesen Fall löschen. Details können Kapitel 4 Prozesse bei der Vertrauensstelle entnommen werden.

## 2 Prozesse bei den Krankenkassen

Im folgenden Abschnitt werden die Prozesse beschrieben, die notwendig sind, um aus den Sozialdaten bei den Krankenkassen nach Vorgaben des G-BA selektierte Daten an das Institut nach § 137a SGB V bzw. an die Auswertungsstelle zu übermitteln.

Der Datenfluss entspricht einem seriellen Prozessmodell wie in Abbildung 5 dargestellt.

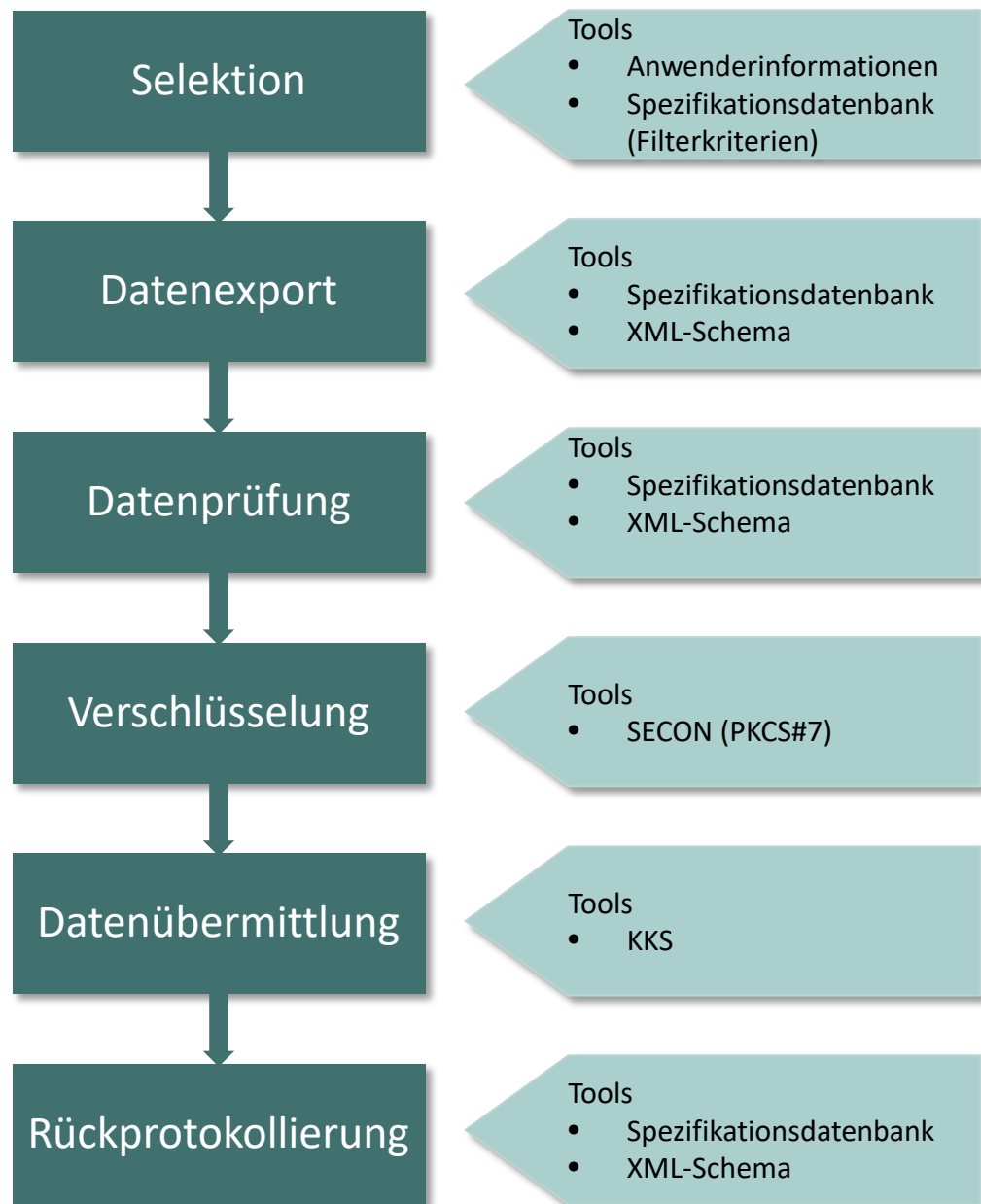


Abbildung 5: Prozesse bei den Krankenkassen

## 2.1 Datenselektion

Bei der Nutzung von Sozialdaten bei den Krankenkassen für Zwecke der gesetzlichen Qualitätssicherung und für die Methodenbewertung ist die Datensparsamkeit von höchster Bedeutung. Daher gilt grundsätzlich, dass nur Daten solcher Fälle selektiert und übermittelt werden dürfen, die für das jeweilige Verfahren unerlässlich sind.

### 2.1.1 Anforderungen an die Datenselektion

Um die Sozialdaten sinnvoll auswerten zu können, muss sichergestellt sein, dass alle Krankenkassen die gleichen Inhalte in gleicher Form übermitteln. Da die Datenbestände der Krankenkassen unterschiedlich strukturiert sind, muss die Spezifikation es ermöglichen, die benötigten Datenfelder unabhängig von der konkreten Struktur der Datenbestände einer Krankenkasse eindeutig zu identifizieren und das Format für die weitere Verarbeitung festzulegen.

Bedingt durch die Heterogenität der Struktur der Krankenkassendaten ist es nicht möglich, im Rahmen dieser Spezifikation einen direkten Bezug zu den Datenfeldern in den Datenbeständen der einzelnen Krankenkassen herzustellen. Stattdessen wird zur eindeutigen Beschreibung der Datenfelder die Tatsache genutzt, dass in den entsprechenden technischen Anlagen der einzelnen Datenaustauschverfahren (DA) zwischen Leistungserbringern (LE) und Krankenkassen alle dort übermittelten Datenfelder (Eingangsdaten) eindeutig benannt und beschrieben werden.<sup>1</sup> An diesem Punkt des Datenflusses sind also die Datenbestände und -strukturen noch einheitlich, die Unterschiede entstehen erst im weiteren Verarbeitungsprozess, wenn die Krankenkassen die Daten in ihre jeweiligen Systeme einspielen. Diese Beschreibungen des DA-Verfahrens werden in der vorliegenden Spezifikation für Sozialdaten verwendet. Auf dieser Grundlage ist es den Krankenkassen möglich, den Bezug eines referenzierten Datenfeldes zu ihren eigenen Datenbeständen herzustellen.

### 2.1.2 Referenzierung der Datenfelder

Die Selektion von Daten für den Export an das Institut nach § 137a SGB V bzw. an die Auswertungsstelle geschieht immer für ein konkretes Verfahren.<sup>2</sup> Die Grundlage der Selektion ist eine Referenz-Tabelle, die für jedes Datenfeld eine eindeutige Kennung und die genaue Herkunft in den Eingangsdaten spezifiziert, also den Daten, die die Krankenkasse von den LE erhalten hat. Wie im vorherigen Abschnitt beschrieben, basiert die Referenz-Tabelle auf den jeweiligen Technischen Anlagen des GKV-SV. Das IQTIG verweist in der jeweiligen Eingangskennung der Datenfelder stets auf die aktuellste Version der Technischen Anlagen, welche zum Zeitpunkt der Veröffentlichung des Spezifikationspakets vorliegt. Weiterhin wird in dieser Tabelle das genaue Format des Datenfeldes spezifiziert, das für die weitere Verarbeitung erforderlich ist. Gegebenenfalls müssen Daten aus im Datenbestand der Krankenkasse vorliegenden Formaten in das

<sup>1</sup> Siehe <http://www.gkv-datenaustausch.de/leistungserbringer/leistungserbringer.jsp> (abgerufen: 05.06.2023)

<sup>2</sup> Definitionen in der Spezifikationsdatenbank sind meist modulbezogen. In der Regel entspricht ein Modul einem bestimmten Verfahren. Es gibt jedoch auch Verfahren, die sich über mehrere Module erstrecken. In diesem Fall ist jedes Modul von den Krankenkassen separat zu bearbeiten. Die Zusammenführung der Module zu einem Verfahren findet erst bei den im Datenfluss folgenden Institutionen statt.

spezifizierte Format konvertiert werden. Die Referenz-Tabelle ist als Abfrage in der Spezifikationsdatenbank hinterlegt:

| modul | eingangskennung  | berechnet                           | funktion         | funktionBe      |
|-------|--|-------------------------------------|------------------|-----------------|
| BSP   | Admin@kassenknr  |                                     |                  |                 |
| BSP   | sequential_nr(Admin)@lfidnr  | <input checked="" type="checkbox"/> | sequential_nr    | Laufende Numm   |
| BSP   | Stamm@geschlecht   | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | Stamm@gebjahr  | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | Stamm@sterbedatum  | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | Stamm@v  | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | Stamm@versicherungsdatum   | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | Stamm@versicherungsstatus  | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | source(301)@quelle   | <input checked="" type="checkbox"/> | source           | Kennzeichen der |
| BSP   | 301.Aufnahmesatz.AUF.Aufnahmetag@aufdatum  | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | 301.Aufnahmesatz.AUF.Aufnahmegrund@aufgrund  | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | 301.Entlassungsanzeige.ETL.Tag der Entlassung/Verlegung@entldatum                          | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | 301.Entlassungsanzeige.ETL.Entlassungs-/Verlegungsgrund@entlgrund                          | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | inpatient_interru(301.Entlassungsanzeige.ETL.Entlassungs-/Verlegungsgrund)@khunterbrechung | <input checked="" type="checkbox"/> | inpatient_interr | Unterbrechung d |
| BSP   | state_key(301.Entlassungsanzeige.FKT.IK des Absenders)@bundesland                          | <input checked="" type="checkbox"/> | state_key        | Bundeslandschl  |
| BSP   | 301.Entlassungsanzeige.FKT.IK des Absenders@nummer   | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | 301.Entlassungsanzeige.FAB.Operation.Prozedurschlüssel@ops                                 | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | 301.Entlassungsanzeige.FAB.Operation.Lokalisation@lokalisation                             | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | 301.Entlassungsanzeige.FAB.Operation.tag@datum   | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | 301.Entlassungsanzeige.ETL.Hauptdiagnose.Diagnoseschlüssel@icd                             | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | 301.Entlassungsanzeige.ETL.Hauptdiagnose.Lokalisation@lokalisation                         | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | 301.Entlassungsanzeige.ETL.Sekundär-Diagnose.Diagnoseschlüssel@icd_sek                     | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | 301.Entlassungsanzeige.ETL.Sekundär-Diagnose.Lokalisation@lokalisation_sek                 | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | 301.Entlassungsanzeige.NDG.Nebendiagnose.Diagnoseschlüssel@icd                             | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | 301.Entlassungsanzeige.NDG.Nebendiagnose.Lokalisation@lokalisation                         | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | 301.Entlassungsanzeige.NDG.Sekundär-Diagnose.Diagnoseschlüssel@icd_sek                     | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | 301.Entlassungsanzeige.NDG.Sekundär-Diagnose.Lokalisation@lokalisation_sek                 | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | source(kh_ambo)@quelle   | <input checked="" type="checkbox"/> | source           | Kennzeichen der |
| BSP   | kh_ambo.Ambulante.Operation.REC.Tag des Zugangs@zugangsdatum                               | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | state_key(kh_ambo.Ambulante.Operation.FKT.IK des Absenders)@bundesland                     | <input checked="" type="checkbox"/> | state_key        | Bundeslandschl  |
| BSP   | kh_ambo.Ambulante.Operation.FKT.IK des Absenders@nummer                                    | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | kh_ambo.Ambulante.Operation.PRZ.Prozedur.Prozedurschlüssel@ops                             | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | kh_ambo.Ambulante.Operation.PRZ.Prozedur.Lokalisation@lokalisation                         | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | kh_ambo.Ambulante.Operation.PRZ.Prozedur.tag@datum   | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | kh_ambo.Ambulante.Operation.BDG.Behandlungsdiaagnose.Diagnoseschlüssel@icd                 | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | kh_ambo.Ambulante.Operation.BDG.Behandlungsdiaagnose.Lokalisation@lokalisation             | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | kh_ambo.Ambulante.Operation.BDG.Behandlungsdiaagnose.Diagnosesicherheit@sicherheit         | <input checked="" type="checkbox"/> |                  |                 |
| BSP   | kh_ambo.Ambulante.Operation.BDG.Sekundär-Diagnose.Diagnoseschlüssel@icd_sek                | <input checked="" type="checkbox"/> |                  |                 |

Abbildung 6: Referenz-Tabelle

Die Felder der Referenz-Tabelle sind in der Tabelle 3 dargestellt:

Tabelle 3: Struktur der Abfrage Referenz

| Datenfeld             | Bedeutung   | Verwendung                 |
|-----------------------|---|----------------------------|
| modul                 | Kennung des Moduls                                      | Filterung nach Modul       |
| eingangskennung       | Eindeutige Kennung des Datenfeldes im Eingangsdatensatz | Referenz auf Eingangsdaten |
| berechnet             | Das Feld wird mit einer Funktion berechnet              | Hinweis auf eine Funktion  |
| funktion              | Enthält ggf. eine Referenz auf diese Funktion           |                            |
| funktion Beschreibung | Beschreibung dieser Funktion                            |                            |
| funktionFormel        | Berechnungsvorschrift dieser Funktion                   |                            |
| datenquelle           | Herkunft des Datenbestandes (z. B. § 301)               | Auffinden der Quelldaten   |
| datensatz             | Referenz auf den Herkunftsdatensatz                     |                            |
| segment               | Referenz auf das Herkunftssegment                       |                            |
| gruppe                | Referenz auf die Herkunftsgruppe                        |                            |

| Datenfeld              | Bedeutung   | Verwendung                                      |
|------------------------|---|---|
| feldkennung            | Referenz auf die Herkunftsfeldkennung                             |   |
| beabsichtigter Inhalt  | Beschreibung des beabsichtigten Feldinhalts                       | Prüfbarkeit                                     |
| basisTyp               | Name des Attributtyps   | Datenformat                                     |
| basisTyp Bezeichnung   | Beschreibung des Attributtyps                                     |   |
| schluessel             | Enthält ggf. eine Referenz auf eine Liste von Schlüsselwerten     |   |
| schluessel Bezeichnung | Beschreibung des Schlüssels                                       |   |
| format                 | Formatanweisung (zur schnellen Übersicht)                         |   |
| formatRegExp           | Formatanweisung als Regulärer Ausdruck (bindende Formatanweisung) |   |
| patientenFilter        | Datenfeld wird im Patientenfilter verwendet                       | Hinweis auf Verwendung für Filter (siehe 2.1.4) |
| leistungsMedFilter     | Datenfeld wird im Leistungs- und Medikationsfilter verwendet      |   |

In der weiteren Verarbeitung wird jedes Datenfeld durch seine Eingangskennung eindeutig referenziert und kann von der Krankenkasse eindeutig einem Datenfeld im eigenen Datenbestand zugeordnet werden. Bei der Referenz-Tabelle können zwei Angaben auftreten, die zusätzliche Arbeitsschritte erfordern: Funktionen und Schlüsselwerte.

### Funktionen

Wenn das Feld `berechnet` den Wert `true` enthält, wird auf das referenzierte Datenfeld eine Funktion angewandt. Beschreibung und Berechnungsvorschrift der Funktion können der Tabelle `Funktion` in der Spezifikationsdatenbank entnommen werden (siehe auch Verzeichnis /1+2 ACCESS/Export aus DB/Tabellen).

Tabelle 4: Inhalt der Tabelle *Funktion*

| Funktion | Formel   | Beschreibung                                   |
|----------|--|--|
| year     | Stelle 1 bis 4 aus einem Datum im Format<br>JJJJ-MM-TT   | Jahr aus einem Datum                           |
| source   | Herkunft der Daten entsprechend dem Abrechnungskontext, aus dem Datenbestand der Krankenkasse zu | Kennzeichen der genauen Datenquelle des Falles |

| Funktion            | Formel   | Beschreibung                        |
|---------------------|--|-------------------------------------|
|                     | entnehmen, siehe Schlüssel 'quelle' ('115b','116b','117','118','119', '295k','295_140','295_73b', '295_73c','300','301', '302_heil','302_hilf','25a')  |                                     |
| kv_key              | 1. und 2. Stelle der BSNR  | KV-Regionsschlüssel                 |
| state_key           | Aus dem Datenbestand der Krankenkasse zu entnehmen oder ersatzweise aus der 3. und 4. Stelle der IKNR  | Bundeslandschlüssel                 |
| specialist_key      | 8. und 9. Stelle der LANR  | Fachgruppenschlüssel                |
| inpatient_interrupt | ja: mind. ein Entlass-/Verlegungsgrund (1. und 2. Stelle) 16, 21 oder 23 im Segment ETL; nein: sonst   | Unterbrechung des KH-Aufenthaltes   |
| sequential_nr       | beginnend mit 1, sonst: vorherige laufende Nummer +1   | Laufende Nummer                     |
| ebm_kh_ambo         | nur selektieren, wenn 2. bis 3. Stelle mit "00" gefüllt sind; nur die 4. bis 7. Stelle des Feldes selektieren  | EBM aus Datenquelle § 301 AMBO      |
| cp_type             | "iknr", wenn es sich um ein Institutionskennzeichen handelt; "bsnr", wenn es sich um eine Betriebsstättennummer handelt  | Art des Leistungserbringers         |
| attending           | 3. Stelle der Entgeltart, wenn 1. und 2. Stelle = ,70'   | Art der Belegleistung               |
| drg_in_list         | ja: Letzte nicht wieder gutgeschriebene DRG des KH-Falles ist in der Liste, welche im Filter mittels Semikolon getrennt als zweiter Parameter an die Funktion übergeben wird, enthalten; nein: sonst | DRG in Liste enthalten <sup>3</sup> |
| date_gop            | Datum einer GO-Nr. falls an KK übermittelt, sonst Datum aus vorhergehender GO-Nr. beziehen   | Datum der Leistung <sup>4</sup>     |

<sup>3</sup> Diese Funktion erhält allein in deren technischer Anwendung im Pseudocode des Filters neben dem ersten Parameter, welche die referenzierte Feldkennung angibt, einen zweiten Parameter, welcher den Namen einer Filterliste referenziert. Dadurch bleiben sämtliche Nennungen an anderer Stelle (z. B. verwendete Datenfelder oder in Tabelle DatenfeldAttribut) unberührt. Für die Implementierung dieser Funktion wird dadurch jedoch die Möglichkeit geschaffen, dynamisch auf verschiedene DRG-Listen innerhalb eines Moduls referenzieren zu können.

<sup>4</sup> Diese Funktion greift lediglich auf leere Datenfelder im Bestand der Kassen, soll jedoch keine vorhandene Werte überschreiben. Die Funktion findet im Rahmen des Exports Anwendung und hat somit keine Auswirkung auf den Filter.



| Funktion | Formel  | Beschreibung   |
|----------|---|--|
| ebm_295s | EBM, die der selektivvertraglichen Abrechnungsziffer entspricht (sofern eine solche Entsprechung existiert); sonst LEER | EBM-Entsprechung der selektivvertraglichen Abrechnungsnummer |



### Hinweis

zur Funktion `ebm_295s`:

Der Zugriff und Export von selektivvertraglichen Leistungen ist grundsätzlich vom G-BA im Rahmen der Durchführung der Vorgaben der DeQS-RL ausdrücklich intendiert!

Es existiert jedoch kein zentrales, bundesweit einheitliches Verzeichnis, in welchem die Abrechnungspositionen für sämtliche selektivvertraglich abrechnungsfähigen Leistungen aufgeführt werden. Insofern können über die Spezifikation keine entsprechenden Filterlisten modulspezifisch bereitgestellt werden.

Daher besteht auf Seiten der Krankenkassen die Verpflichtung, die Verfahrensbezogenen Selektionsmechanismen im Rahmen der Umsetzung der Spezifikation für die Nutzung der Sozialdaten bei den Krankenkassen, selbständig über die der Spezifikation möglichen Vorgaben hinaus zu erweitern und entsprechend anzuwenden.

Zur Unterstützung der vorgenannten Aufgabe im Rahmen der Umsetzung auf Seiten der Krankenkassen wurde die Funktion `ebm_295s` entwickelt, welche auf abstrakter Ebene im Rahmen des Pseudocodes für die Filtermechanismen zur Anwendung kommt.

Die konkrete Aufgabe auf Seiten einer einzelnen Krankenkasse besteht folglich darin tabellarisch, in maschinenlesbarer Form sämtliche relevanten Abrechnungspositionen zusammenzutragen, damit diese automatisiert im Rahmen der Selektion von Behandlungsfällen für den selektivvertraglichen Bereich zur Anwendung kommen können. Diese Listen müssen modulspezifisch sein. Für Jedes Modul, in welchem die Funktion `ebm_295s` im Pseudocode einer der beiden Filterstufen genannt ist, muss jeweils eine eigene Liste zusammengetragen werden.

Im Ergebnis hängt der Umfang einer solchen Filterliste ab von:

- Krankenkasse
- ggf. KV-Bereich bzw. Region
- Modul
- ggf. Filterstufe (Patientenfilter vs. Leistungs- und Medikationsfilter)

Für weitere Informationen bzgl. der Selektion von selektivvertraglich erbrachten Leistungen siehe **Abschnitt Auslösung selektivvertraglich erbrachter Leistungen** unter 2.1.4 Filter.



### Hinweis

zur Funktion `specialist_key`:

Die Funktion `specialist_key` wird im Rahmen der QS-Filterdefinition dazu verwendet, die jeweilige Fachgruppe abzurufen. Dazu werden die 8. und 9. Stelle der LANR verwendet.

Im Modul NWITR werden beispielsweise mit Hilfe des `specialist_key` die Fachgruppen anhand der Nummern 06, 07, 08, 10, 11, 12, 13, 15, 16, 17, 18 und 67 in den Filteralgorithmus aufgenommen und selektiert.

### Schlüsselwerte

Für einige Datenfelder sind nur bestimmte Schlüsselwerte zulässig. In diesem Fall ist im Feld `schluessel` ein Verweis auf die entsprechende Werteliste angegeben. Die zulässigen Werte des jeweiligen Schlüssels sind der Tabelle `SchluesseWert` zu entnehmen.

Tabelle 5: Struktur der Tabelle `SchluesseWert`

| Datenfeld                 | Bedeutung                       |
|---------------------------|---------------------------------|
| <code>id</code>           | Feldkennung des Schlüsselwertes |
| <code>fkSchluessel</code> | Bezeichnung des Schlüssels      |
| <code>code</code>         | Schlüsselwert                   |
| <code>bezeichnung</code>  | Bezeichnung des Schlüsselwertes |

#### 2.1.3 Selektionsstufen

Die Selektionskriterien werden durch den G-BA verfahrensspezifisch festgelegt. Dabei handelt es sich in der Regel um eine Liste von Codes (OPS, ICD u. a.), die als Ein- und Ausschlusskriterien auf spezifizierten Datenfeldern zur Anwendung kommen. Hierbei wird ein zweistufiger Filter verwendet.

##### 1. Patientenfilter

In der ersten Filterstufe wird zunächst über den Patientenfilter sichergestellt, dass nur Patienten eingeschlossen werden, die in einem genau definierten Zeitraum mindestens eine bestimmte Indexleistung erhalten haben. Über diese Indexleistung wird die Relevanz des Falles für ein bestimmtes Modul begründet.

Nur diejenigen Patienten, die in der ersten Filterstufe erfasst wurden, werden für die zweite Filterstufe (Leistungs- und Medikationsfilter) berücksichtigt.



#### Hinweis

Versicherte, die vom Patientenfilter erfasst wurden, sind unabhängig vom Ergebnis der zweiten Filterstufe mit ihren Quartalsstammdaten für den gesamten verfahrensspezifisch festgelegten Beobachtungszeitraum (Zeitfilter der zweiten Filterstufe), vgl. Absatz „Quartalsweise Versichertendaten“ unter 6.2.2 Elemente und Attribute der Nutzdaten in der QS-Datei, zu exportieren. Dabei sind auch die gefundenen Indexleistungen zu exportieren und mit dem Attribut `patientenfilter = "1"` zu kennzeichnen. Handelt es sich darüber hinaus um Module mit PID-Bezug, ist ebenfalls die PID zu exportieren.

## 2. Leistungs- und Medikationsfilter

In der zweiten Filterstufe wird der Datenumfang bestimmt, der für die in der ersten Filterstufe ermittelten Patienten (zusätzlich zu deren Indexleistungen, Quartalsstammdaten und ggf. PID) zu exportieren ist. Dieser Filter wird als Leistungs- und Medikationsfilter bezeichnet. Die Daten dieser Filterstufe unterliegen ebenfalls einer zeitlichen Einschränkung, welche dem verfahrensspezifisch festgelegten Beobachtungszeitraum entspricht. Dieser Beobachtungszeitraum geht über den Betrachtungszeitraum für die Suche nach Indexleistungen aus dem Patientenfilter hinaus, da er das sogenannte Follow-up mit einschließt.



### Hinweis

Wenn der Leistungs- und Medikationsfilter leer ist (also keine Filterregeln darin enthalten sind), sind nur die Ergebnisse des Patientenfilters zu exportieren. Weiterhin kann dann für die zeitliche Einschränkung der zu generierenden Quartalsstammdaten ersatzweise lediglich der zeitliche Rahmen des Patientenfilters herangezogen werden.

### Beispiel

Drei weibliche und drei männliche Patienten (senkrechte Achse) besuchen diverse Ärzte und erhalten entsprechende medizinische Leistungen (große Kästen) und Medikationen (kleine Kästen):

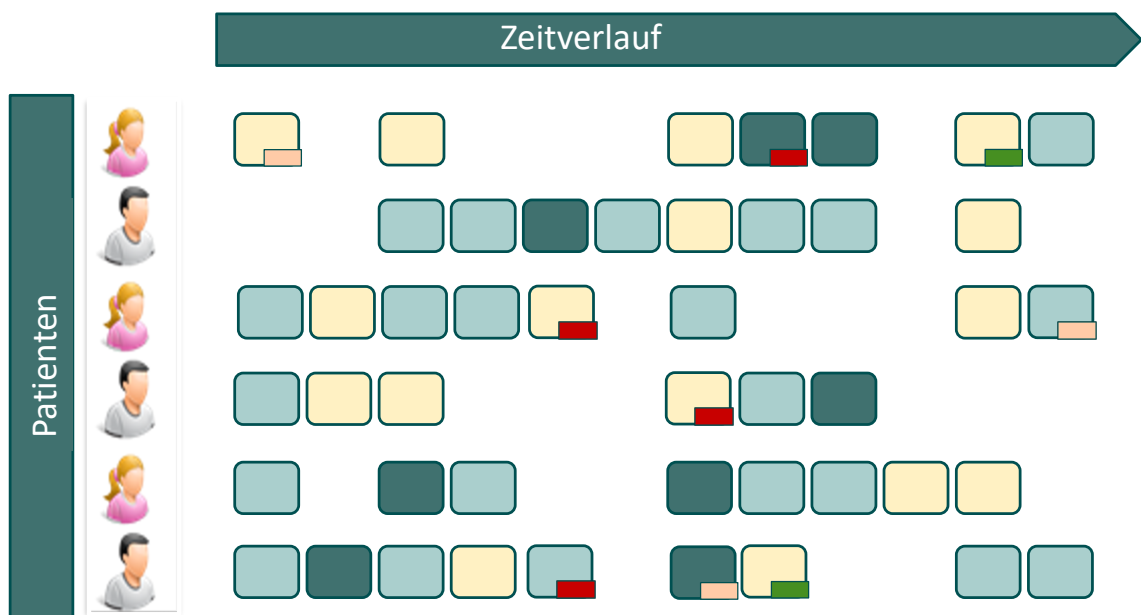


Abbildung 7: Beispiel Filterstufen – Ausgangssituation

Durch den Patientenfilter werden im ersten Schritt ausschließlich Patienten ausgewählt, die in einem bestimmten Zeitraum eine das Verfahren betreffende Indexleistung erhalten haben. Diese ist in Abbildung 8 mit einem „x“ gekennzeichnet. Nur vier Patienten entsprechen den Kriterien des Patientenfilters (farbig hinterlegt):

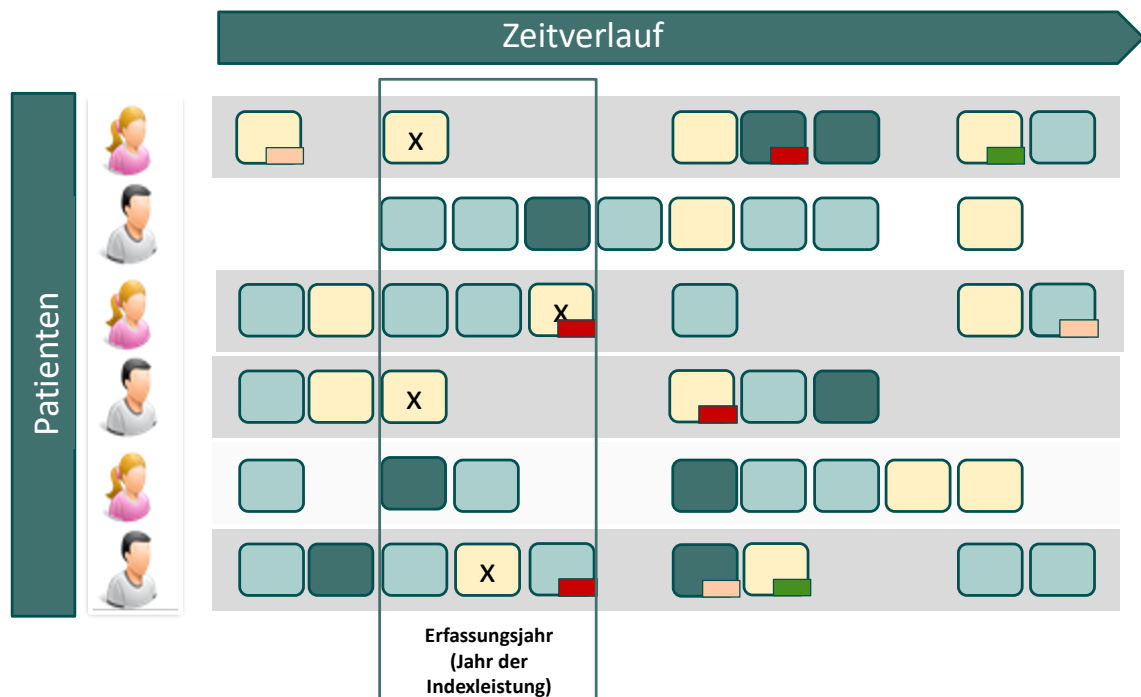


Abbildung 8: Beispiel Filterstufen – Patientenfilter

Die nicht selektierten Versicherten werden von der weiteren Betrachtung ausgeschlossen.

Im zweiten Schritt wird der Leistungs- und Medikationsfilter auf die selektierten Versicherten angewandt:

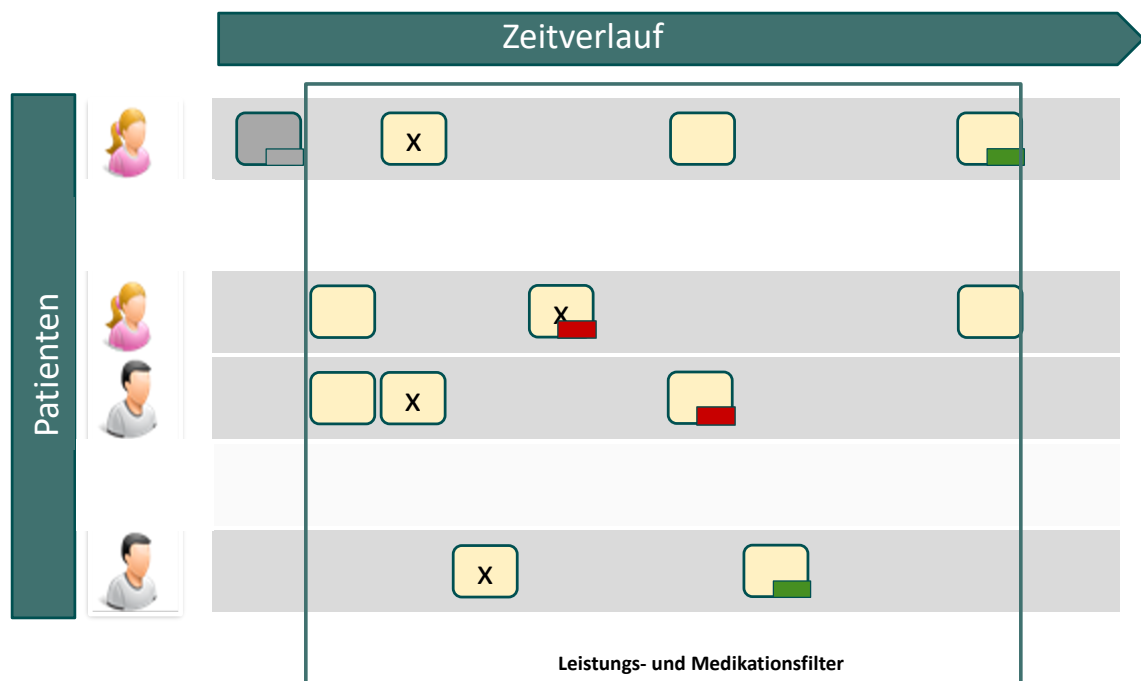


Abbildung 9: Beispiel Filterstufen – Leistungs- und Medikationsfilter

Es verbleiben nur „gelbe“ Leistungen und „grüne“ und „rote“ Medikationen, die beide den Kriterien des Filters entsprechen. Der im Schaubild ausgegraute Kasten ist eine Leistung mit Medikation, die nicht durch die zeitliche Komponente eingeschlossen wird. Für die vier durch den Patientenfilter bestimmten Patienten sind nun also nur die Daten zu den verbleibenden Leistungen und Medikationen im definierten Zeitraum zu exportieren.

#### 2.1.4 Filter

Die in Abschnitt 2.1.3 beschriebenen Filterschritte bilden gemeinsam den Filter und sind für jedes Verfahren in der Spezifikations-Datenbank abgebildet.

##### Definition der Filter

Die Definition der Filterkriterien ist in der Tabelle `ModulErfassungsjahr` abgelegt. Darin befinden sich folgende Felder mit Bezug auf die Filter:

Tabelle 6: Ausschnitt aus der Tabelle `ModulErfassungsjahr`

| Datenfeld                                   | Bedeutung   | Verwendung   |
|---|---|--|
| <code>fkModul</code>                        | Name des Moduls   | Eingrenzung auf das Exportmodul  |
| <code>erfassungsjahr</code>                 | Erfassungsjahr  | Eingrenzung auf das Erfassungsjahr der Indexleistung beim Leistungserbringer   |
| <code>patientenFilter</code>                | Pseudocode des Patientenfilters   | Logischer Aufbau des Filters mit dem die Patientenauswahl stattfindet  |
| <code>patientenFilterText</code>            | Textbeschreibung des Patientenfilters   | Filter erklären  |
| <code>leistungsMedikationsFilter</code>     | Pseudocode des Leistungs- und Medikationsfilters  | Logischer Aufbau des Filters mit dem die Routinedaten für bestimmte Leistungen und Medikationen für die Patientenmenge aus dem Patientenfilter ermittelt wird. |
| <code>leistungsMedikationsFilterText</code> | Textbeschreibung des Leistungs- und Medikationsfilters  | Filter erklären  |
| <code>regelErrorIsWarning</code>            | Wenn true, dann werden harte Regeln aus Tabelle Regel weich interpretiert. (Hinweis: Prüfungen aus Tabelle Prüfung bleiben hart.) | Bspw. bei Einführung eines neuen Verfahrens  |

Da die genaue Anwendung der Filter auf die Datenbestände der Krankenkassen durch diese individuell umgesetzt werden muss, sind die Filter in Form eines Pseudocodes angegeben, der im Folgenden beschrieben wird.

**Pseudocode der Filter**

Die Selektionskriterien der Filter werden in einem Pseudocode dargestellt, in dem zum einen die Filter benannt werden, zum anderen die Datenfelder, die geprüft werden.

Die Syntax des Pseudocodes lehnt sich an die logischen Ausdrücke in bekannten Programmier- und Skriptsprachen an. Jedoch haben die Operatoren deutsche Namen, z. B. UND statt AND oder ODER statt OR (weitere Beispiele sind in der Tabelle 7 aufgelistet).



**Hinweis**

Durch Anwendung einer Filterstufe werden einzelne Fälle einzelner Patienten nacheinander fokussiert, um deren Relevanz für das Verfahren festzustellen. Somit kann ausgeschlossen werden, dass mehr als ein Fall gleichzeitig adressiert werden soll, wodurch leistungsarten-übergreifende Verknüpfungen ausschließlich mittels logischem ODER verknüpft werden dürfen.

Dieser Pseudocode verwendet Referenzen auf Datenfelder sowie externe Parameter (siehe Abschnitt 2.1.5) und verknüpft diese mittels Operatoren, die im Folgenden beschrieben sind.

Jede Bedingung (Filterkriterium) hat folgenden Aufbau:

<Operand> <Operator> <Operand>

- Der linke Operand ist das Referenzfeld.
- Als Operator kann jeder der Operatoren in Tabelle 8 verwendet werden.
- Die auf der rechten Seite erlaubten Operanden sind nachfolgend aufgelistet:
  - Literale (z. B. Ganzzahlen, Zeichenketten)
  - Kodelisten, in denen auch die Codes eines Schlüssels referenziert werden können
  - ICD-Listen oder OPS-Listen (Alle Codes sind terminal. Die Diagnosecodes enthalten nicht die Suffixe #, †, +, \* oder !.)

*Tabelle 7: Zu den Operatoren des Pseudocodes analoge SQL-Operatoren*

| Operator   | SQL-Analogie                        |
|------------|-------------------------------------|
| IN         | IN                                  |
| NICHTIN    | NOT IN                              |
| EINSIN     | IN                                  |
| KEINSIN    | NOT IN                              |
| INNERHALB  | BETWEEN <Minimum> AND <Maximum>     |
| AUSSERHALB | NOT BETWEEN <Minimum> AND <Maximum> |

Tabelle 8: Operatoren des Pseudocodes

| Präzedenz* | Assoziativität | Operator    | Erläuterung  |
|------------|----------------|-------------|--|
| 0          | links          | IN          | Die Variable und die Feldelemente müssen gleichen Typs sein. Prüft, ob sich das genannte Element der linken Seite in der Liste auf der rechten Seite befindet. |
| 0          | links          | NICHTIN     | Mengenoperator (komplementär zu IN)  |
| 0          | links          | EINSIN      | Mengenoperator. Prüft, ob sich mindestens eines der Elemente der linken Seite in der Liste auf der rechten Seite befindet.                                     |
| 0          | links          | JEDESIN     | Mengenoperator. Prüft, ob sich alle Elemente der linken Seite der Liste auf der rechten Seite befinden.  |
| 0          | links          | EINSNICHTIN | Mengenoperator (komplementär zu JEDESIN)   |
| 0          | links          | KEINSIN     | Mengenoperator (komplementär zu EINSIN)  |
| 0          | links          | INNERHALB   | Wertebereich. Vergleich eines Wertes bzw. eines Datums gegen eine untere und eine obere Grenze, wobei die Grenzen eingeschlossen sind                          |
| 0          | links          | AUSSERHALB  | Wertebereich (komplementär zu INNERHALB)   |
| 1          | links          | *           | Multiplikation   |
| 1          | links          | /           | Division   |
| 2          | links          | +           | Addition   |
| 2          | links          | -           | Subtraktion  |
| 3          | links          | <           | Vergleichsoperator „kleiner“   |
| 3          | links          | >           | Vergleichsoperator „größer“  |
| 3          | links          | <=          | Vergleichsoperator „kleiner gleich“  |
| 3          | links          | >=          | Vergleichsoperator „größer gleich“   |
| 4          | links          | =           | Vergleichsoperator „gleich“  |
| 4          | links          | <>          | Vergleichsoperator „ungleich“  |
| 5          | rechts         | NICHT       | logisches Nicht  |
| 6          | links          | UND         | Logische UND-Operation:<br>Beide Seiten müssen erfüllt sein.   |

| Präzedenz* | Assoziativität | Operator | Erläuterung   |
|------------|----------------|----------|---|
| 7          | links          | ODER     | Logische ODER-Operation:<br>Eine oder beide Seiten müssen erfüllt sein. |

\* In Tabelle 8 hat jeder einzelne Operator eine Präzedenzstufe (höchste Präzedenzstufe ist „0“). Operatoren, welche die gleiche Stufe haben, werden stattdessen nach den Regeln der Assoziativität aufgelöst. Präzedenzen und Assoziativität können mittels Klammerung punktuell außer Kraft gesetzt werden. Dabei werden Ausdrücke innerhalb der Klammern zuerst ausgewertet.

Tabelle 9: Verwendung der Mengen- und Wertebereichoperatoren

| Operand (links) | Operator    | Operand (rechts)   | Beschreibung / Komplement-Darstellung   | Beispiel für positives Prüfergebnis                                  |
|-----------------|-------------|--------------------|---|--|
| Wert            | IN          | Liste <sup>5</sup> | Prüft, ob sich das Element der linken Seite in der Liste auf der rechten Seite befindet.                                    | '01234' IN<br>( '00112', '01234' )                                   |
| Wert            | NICHTIN     | Liste              | NICHT IN  | '01234' NICHTIN<br>( '00112', '43210' )                              |
| Liste           | EINSIN      | Liste              | Prüft, ob sich mindestens eines der Elemente der linken Seite in der Liste auf der rechten Seite befindet                   | ( '01234', '23456' )<br>EINSIN<br>( '00112', '01234', '54321' )      |
| Liste           | JEDESIN     | Liste              | Prüft, ob sich alle Elemente auf der linken Seite in der Liste auf der rechten Seite befinden                               | ( '01234', '23456' )<br>JEDESIN<br>( '54321', '01234', '23456' )     |
| Liste           | EINSNICHTIN | Liste              | NICHT JEDESIN   | ( '01234', '23456' )<br>EINSNICHTIN<br>( '00112', '01234', '54321' ) |
| Liste           | KEINSIN     | Liste              | NICHT EINSIN  | ( '01234', '23456' )<br>KEINSIN<br>( '00112', '54321', '11223' )     |
| Wert            | INNERHALB   | Wertebereich       | Min-Max-Vergleich eines Wertes (z. B. Datum) gegen eine untere und eine obere Grenze, wobei die Grenzen eingeschlossen sind | '2015-01-01'<br>INNERHALB<br>( '2015-01-01', '2015-12-31' )          |
| Wert            | AUSSERHALB  | Wertebereich       | NICHT INNERHALB   | '2016-01-01'<br>AUSSERHALB<br>( '2015-01-01', '2015-12-31' )         |

<sup>5</sup> Mengenoperatoren behandeln Listen immer als unsortierte Mengen. Die Reihenfolge der Elemente spielt also keine Rolle.



**String-Vergleich bei Leistungen, Diagnosen und Prozeduren:**

Die Mengenoperatoren basieren für Leistungen, Diagnosen oder Prozeduren auf einem Vergleich von Zeichenketten (String-Vergleich). Hierfür müssen in den vorliegenden ICD-Kodes die Suffixe „#“, „+“, „+“, „\*“ und „!“ ignoriert werden. OPS-Kodes werden in den Kodelisten ohne die Sonderzeichen „.“ und „-“ angegeben. Ebenso müssen Zusatzkennzeichen zu eigentlich 5-stelligen EBM-Ziffern (z. B. 01234A) ignoriert werden.

**Beispiel**

Folgende Operationen führen zum gleichen Ergebnis:

- ('Z37.9') EINSIN ('Z37.9'; 'Z37.0'; 'Z37.1'; 'Z37.2'; 'Z37.3')
- ('Z37.9!') EINSIN ('Z37.9'; 'Z37.0'; 'Z37.1'; 'Z37.2'; 'Z37.3')

**NULL-Werte im Pseudocode:**

Das Vorhandensein eines Attributs wird mit der Schreibweise `attributname <> NULL` gekennzeichnet. Analog bezeichnet `attributname = NULL` das Fehlen eines Attributs.

**Auslösung selektivvertraglich erbrachter Leistungen**

Leistungen im Rahmen von Selektivverträgen gemäß §§ 73b, 73c oder 140a SGB V (Datenquelle 295s) können abweichend vom EBM-Katalog gemäß vertragsspezifischen Regelungen dokumentiert werden. Es ist nicht möglich, alle individuellen Kodierungen in den Kodelisten zu berücksichtigen bzw. es sind nicht alle diese Kodierungen bekannt, die für die Selektion im Pseudocode verwendet werden. Stattdessen müssen die Krankenkassen ggf. anhand der angegebenen Abrechnungsziffern die analog in den Verträgen verwendeten Codes identifizieren und für die Auslösung eines Falles heranziehen. Solche vertragsspezifischen Codes werden jedoch selbst nicht mit den ausgelösten Fällen exportiert. Für weitere Information bzgl. der Selektion von selektivvertraglich erbrachten Leistungen siehe **Hinweis zur Funktion ebm\_295s**.

**2.1.5 Externe Parameter des Filters**

Im Pseudocode der Filter wird auf zwei Arten von externen Parametern Bezug genommen: Kodelisten und Wertebereiche. Diese Entkopplung der Parameter aus dem Pseudocode dient dazu, den Pseudocode übersichtlicher zu halten und eine Pflege der Filter und der zugehörigen Datenbankabfragen bei Änderungen von Kodelisten und Betrachtungszeiträumen zu erleichtern.

**Kodelisten**

Die in den Ein- und Ausschlusskriterien verwendeten Kodelisten sind in der Tabelle `Codeliste` enthalten, die als Abfrage in der Spezifikationsdatenbank vorliegt (siehe auch Verzeichnis /1+2 ACCESS/Export aus DB/Abfragen).

Tabelle 10: Struktur der Tabelle `Codeliste`

| Datenfeld          | Bedeutung              | Verwendung                                |
|--------------------|------------------------|---|
| <code>modul</code> | Bezeichnung des Moduls | Eingrenzung auf ein bestimmtes Modul      |
| <code>liste</code> | Bezeichnung der Liste  | Ist im Pseudocode der Filter referenziert |

| Datenfeld | Bedeutung         | Verwendung  |
|-----------|-------------------|---|
| code      | Wert des Kodes    | Wert, gegen den auf Gleichheit geprüft wird.                      |
| version   | Version des Kodes | Zur Überprüfung der Aktualität und dem Anwendungsgebiet der Liste |

### Wertebereiche

Die in den Ein- und Ausschlusskriterien verwendeten Wertebereiche (zumeist Zeiträume) sind in der Tabelle *Wertebereich* enthalten.

Tabelle 11: Struktur der Tabelle *Wertebereich*

| Datenfeld        | Bedeutung                           | Verwendung  |
|------------------|-------------------------------------|---|
| modul            | Bezeichnung des Moduls              | Eingrenzung auf ein bestimmtes Modul                |
| erfassungsjahr   | Angabe des Erfassungsjahres         | Eingrenzung auf ein bestimmtes Erfassungsjahr       |
| name             | Bezeichnung des Wertebereichs       | Ist im Pseudocode der Filter referenziert           |
| min              | Kleinster Wert (inklusive)          | Wertebereich, gegen den auf Einschluss geprüft wird |
| max              | Größter Wert (inklusive)            |   |
| version          | Version des Wertebereichs           | Überprüfung der Aktualität                          |
| datum            | Änderungsdatum                      | Intern  |
| fkVersionRelease | Release-Version der aktuellen Liste | Versionskontrolle                                   |

Die Minimum- und Maximumwerte eines Wertebereichs können im Pseudocode mit der Schreibweise `NAME.min` bzw. `NAME.max` referenziert werden.

#### 2.1.6 Ergebnis der Selektion

Die Anzahl der zu exportierenden Datensätze ergibt sich aus der Anwendung des Patientenfilters. Jeder Datensatz entspricht dabei einem Versicherten. Der Umfang der zu exportierenden Daten je Datensatz ergibt sich dagegen aus der Anwendung des Leistungs- und Medikationsfilters. Die Selektion durch die Filterkriterien erfolgt dabei jeweils auf Ebene eines Falles (Behandlungs- bzw. Verordnungsfall). Wenn ein Fall die Filterbedingungen erfüllt, ist er zu exportieren. Unterelemente (Prozeduren, Diagnosen usw.) werden nur dann mit exportiert, wenn sie:

- als Pflichtfelder in der Struktur der Exportdaten vorgesehen sind (siehe Abschnitt 6.2)
- durch einen Filter selektiert wurden
- Sekundärkodierungen zu einer durch den Filter selektierten Angabe sind

Fälle, die durch den Patientenfilter selektiert wurden, sind beim Export mit dem zusätzlichen Attribut `patientenfilter="1"` zu versehen.

Wird für einen Patienten ein und der gleiche Fall sowohl im Patientenfilter als auch im Leistungs- und Medikationsfilter jeweils mit einem positiven Prüfergebnis aufgegriffen, so ist dieser Fall (möglichst) nicht doppelt in den Exportdaten aufzuführen. Vielmehr ist eine Zusammenführung der durch die jeweilige Filterstufe selektierten Daten vorzunehmen. Das zusätzliche Attribut `patientenfilter="1"` ist für den zusammengesetzten Fall anzugeben.

---

### Beispiel

Im Modul NWITR stellt die leistungsartenbezogene Definition des Leistungs- und Medikationsfilters eine Erweiterung der Definition des Patientenfilters dar. Hintergrund ist, dass im Leistungs- und Medikationsfilter prinzipiell die gleichen Fälle erneut betrachtet werden sollen, welche durch Anwendung des Patientenfilters überhaupt erst zum Zugriff des Patienten geführt haben. Zusätzlich sollen jedoch - sofern vorhanden - bestimmte Diagnosen, welche in den Eli-xhauser Komorbiditätsindex fallen, zum Zwecke einer Risikoadjustierung zusätzlich übermittelt werden. Diese Diagnosen wiederum sind einerseits nicht erforderlich für den Zugriff des Patienten und andererseits darf durch das Nicht-Vorliegen dieser Diagnosen der Zugriff des Patienten nicht verhindert werden. Somit sind diese Diagnosen nicht in den Patientenfilter integrierbar.

Liegen also in Fällen, welche zum Zugriff eines Patienten führen (positives Prüfergebnis des Patientenfilters) auch Diagnosen zur Risikoadjustierung vor, so können bzw. sollen diese gemeinsam innerhalb eines Falles exportiert und übermittelt werden.

Somit wird die Übermittlung redundanter Informationen vermieden, wodurch insbesondere für Module mit hohen Fallzahlen (z. B. NWITR ohne PID) die Größe der zu übermittelnden Transaktionen so gering wie möglich gehalten wird.

---

### 2.1.7 Beispiel für Datenselektion

Für dieses Beispiel wird auf das fiktive Modul BSP zurückgegriffen.

#### Patientenfilter

Das Modul BSP enthält den folgenden Patientenfilter:

```
(301.Entlassungsanzeige.FAB.Operation.Prozedurenschlüssel@ops
EINSIN Codeliste.BSP_OPS)
UND
(301.Entlassungsanzeige.ETL.Tag der Entlassung/Verlegung@entlda-
tum INNERHALB Wertebereich.BSP_INDEXJAHR)
```

Gemäß Abschnitt 2.1.2 finden sich in der Referenz-Tabelle unter der Kennung `301.Entlassungsanzeige.FAB.Operation.Prozedurenschlüssel@ops` die weiteren fünf Spalten `datenquelle`, `datensatz`, `segment`, `gruppe`, `feldkennung`.

Tabelle 12: Referenz auf ein Datenfeld

| Referenz    |                    |         |           |                     |
|-------------|--------------------|---------|-----------|---------------------|
| datenquelle | datensatz          | segment | gruppe    | feldkennung         |
| 301         | Entlassungsanzeige | FAB     | Operation | Prozedurenschlüssel |

Mit diesen fünf Spalten kann die Krankenkasse das referenzierte Feld eindeutig zuordnen und entsprechend den Formatangaben formatieren. Analog wird mit dem referenzierten Feld 301.Entlassungsanzeige.ETL.Tag der Entlassung/Verlegung@entldatum verfahren, das gegebenenfalls in das Format JJJJ-MM-TT konvertiert werden muss (siehe Tabelle 3).

Auf die so identifizierten und formatierten Datenfelder werden nun die Filterkriterien des Patientenfilters angewandt. Die Filterbedingung ist genau dann für einen konkreten Behandlungsfall nach § 301 erfüllt, wenn mindestens eine Prozedur mit einem OPS-Kode aus der Liste BSP OPS darin enthalten ist und das Datum der Entlassung im Wertebereich BSP\_INDEXJAHR liegt. Ein Versicherter, für den wenigstens ein Behandlungsfall die Bedingungen des Patientenfilters erfüllt, wird für den Export ausgewählt und der Leistungs- und Medikationsfilter auf seine weiteren Behandlungsfälle und Verordnungen angewandt. Entsprechend der Beschreibung in Abschnitt 2.1.3 werden lediglich die fallbezogenen Indexleistungen exportiert.

**Hinweis**

Alle Behandlungsfälle oder Verordnungen, die die Bedingungen des Patientenfilters erfüllen, sind mit zu exportieren. Solche Fälle sind im Export mit dem zusätzlichen Attribut `patientenfilter = "1"` zu markieren.

**Leistungs- und Medikationsfilter**

Das Modul BSP enthält den folgenden Leistungs- und Medikationsfilter:

```
(295k.DIA.Diagnose.4/4.2.1@icd EINSIN Codeliste.BSP_ICD_KOMPL
UND
295k.RND.Behandlungszeitraum.3/3.3.2@endedatum INNERHALB Wertebereich.BSP_ZEITFILTER)
ODER
(301.Entlassungsanzeige.FAB.Operation.Prozedurenschlüssel@ops EINSIN
Codeliste.BSP_OPS_KOMPL
UND
301.Entlassungsanzeige.ETL.Tag der Entlassung/Verlegung@entldatum INNERHALB Wertebereich.BSP_ZEITFILTER)
```

Die Bedingungen beschreiben analog zum Patientenfilter Ein- und Ausschlusskriterien für die Selektion eines Falles. Innerhalb eines ausgewählten Falles nach § 295 sind in diesem Beispiel nur solche Diagnosen mit zu exportieren, die ICD-Kodes aus der Liste BSP\_ICD\_KOMPL enthalten. Prozeduren, Leistungen und andere Diagnosen, deren ICD-Kodes nicht in der Liste enthalten

sind, werden in diesem Beispiel nicht exportiert. Analog werden in den selektierten Behandlungsfällen nach § 301 in diesem Beispiel nur Prozeduren exportiert, die in der Liste BSP\_OPS\_KOMPL enthalten sind.

### 2.1.8 Beispiel zwei für Datenselektion

Für dieses Beispiel wird auf Mechanismen der Module NWITR bzw. NWIWI zurückgegriffen. Insbesondere sollen die Wirkweisen der Anwendung des Operators KEINSIN sowie der Funktion `drg_in_list()` fokussiert werden.

#### Patientenfilter

Bezogen auf die Anwendung von Filterlisten werden diese im Pseudocode grundsätzlich mittels der Operatoren EINSIN bzw. KEINSIN verknüpft. Ausnahme kann die Verwendung einer Funktion darstellen. Beispielsweise ist die Anwendung des Operators EINSIN in die Funktionsweise von `drg_in_list()` inkludiert.

Der Pseudocode für eine Filterstufe eines Moduls kann über Teilbedingungen verfügen. Als **atomare Teilbedingung** kann hierbei

- die Referenzierung auf **ein DatenfeldAttribut**
- unter Verwendung **eines Operators**
- für die Verknüpfung mit **einer Filterliste**

verstanden werden.

Atomare Teilbedingungen können zu verschachtelten Teilbedingungen unter Anwendung der aus der Logik bekannten Operatoren Konjunktion (UND), Disjunktion (ODER) sowie Negation (NICHT) zusammengesetzt werden. Auch Umklammerungen folgen den bekannten Regeln der Logik. Nicht atomare Teilbedingungen können ebenfalls zu komplexeren Teilbedingungen verschachtelt werden.

Werden also mehrere (atomare) Teilbedingungen unter Anwendung des Operators EINSIN konjunktiv verknüpft, so müssen sämtliche Teilbedingungen erfüllt sein, wohingegen bei disjunktiven Verknüpfungen lediglich eine der Teilbedingungen erfüllt sein muss, damit ein Fall durch Anwendung des gesamten Pseudocodes einer Filterstufe eines Moduls mit einem positiven Prüfergebnis bewertet wird.

Sind zusätzlich (atomare) Teilbedingungen unter Verwendung des Operators KEINSIN konjunktiv mit (atomaren) Teilbedingungen unter Verwendung des Operators EINSIN verknüpft, so bedeutet dies, dass selbst bei positivem Prüfergebnis der Teilbedingung, welche ausschließlich den Operator EINSIN verwendet, der gesamte Pseudocode mit einem negativen Prüfergebnis zu bewerten ist, sobald eine Kodierung aus einer Filterliste, welche unter Verwendung des Operators KEINSIN definiert ist, in dem jeweiligen Fall vorliegt.

Das Vorliegen von Kodierungen aus Filterlisten, welche mittels des Operators KEINSIN zur Anwendung gebracht werden, wirkt lediglich auf das Prüfergebnis bezogen auf den einzelnen Fall, also auch nicht auf einen anderen Fall der gleichen Leistungsart (z. B. Fälle nach § 295). Das Prü-

Ergebnis anderer Fälle desselben Patienten wird dadurch nicht beeinflusst. Dies wird beispielsweise durch die ausschließliche Verwendung disjunktiver Verknüpfungen für leistungsartenübergreifende Teilbedingungen sichergestellt (vgl. Hinweis im Abschnitt 2.1.4).

Hintergrund für die Erforderlichkeit der Anwendung des Operators KEINSIN ist im Verfahren QS WI der Aspekt, dass eine exakte Eingrenzung auf relevante Fälle allein durch die Verwendung konkreter Kodierungen mittels EINSIN aufgrund des Nicht-Vorhandenseins solch konkreter Kodierungen nicht möglich ist. Daher mussten die Filterlisten für die Anwendung des Operators EINSIN entsprechend weiter gefasst werden. Im Ergebnis der empirischen Prüfung konnten jedoch Kriterien erarbeitet werden, welche die zu betrachtende Grundmenge dahingehend reduziert, dass bei Vorliegen bestimmter weiterer Kodierungen ein Ausschluss dieser Fälle aus der Grundmenge sichergestellt werden kann. Diese Kriterien werden als (atomare) Teilbedingungen mit dem Operator KEINSIN konjunktiv verknüpft.

Für weitere Information bzgl. der Selektion von selektivvertraglich erbrachten Leistungen siehe **Hinweis zur Funktion ebm\_295s** bzw. **Abschnitt Auslösung selektivvertraglich erbrachter Leistungen**.

#### **Umgang mit der Funktion drg\_in\_list()**

Die konkrete abrechnungsrelevante Information über eine abgerechnete DRG darf aus datenschutzrechtlichen Gründen nicht an die BAS übermittelt werden. Dennoch konnten auf fachlicher Seite Kriterien auf Basis der DRG entwickelt werden. Bezüglich der Anwendung dieser Kriterien wird auf eine Funktion zurückgegriffen, deren Ergebnis durch Abbildung auf einen Wahrheitswert ein entsprechendes Surrogat beinhaltet.

Wird also die Funktion `drg_in_list()` zur Anwendung gebracht, so ergibt sich in Folge der Überprüfung auf das Vorliegen bestimmter Kodierungen aus einer Filterliste mit DRG-Kodes im Ergebnis entweder WAHR oder FALSCH. Im Modul NWITR soll zusätzlich zum Wahrheitswert der Funktion `drg_in_list()` die Negation zur Anwendung gebracht. Das bedeutet also bei Vorliegen einer DRG-Kodierung aus der DRG-Filterliste in einem konkreten Fall wird das Ergebnis der Funktion auf WAHR abgebildet. Durch die zusätzliche Negation wird der Wahrheitswert entsprechend umgekehrt, so dass diese Teilbedingung mit einem negativen Prüfergebnis zu bewerten ist. Sofern - wie im Modul NWITR vorliegend - diese negierte Teilbedingung konjunktiv mit weiteren Teilbedingungen verknüpft ist, kann die gesamte Filterbedingung nicht mehr zu einem positiven Filterergebnis führen, was faktisch zur Folge hat, dass dieser Fall nicht zum Zugriff eines Patienten führen darf. Diese atomare Teilbedingung (einschließlich boolsche Bewertung durch die Funktion `drg_in_list()` sowie deren Negation) kann also nur dann zu einem positiven Prüfergebnis führen, wenn keine Kodierung aus der DRG-Filterliste im konkreten Fall vorliegt.

#### **Leistungs- und Medikationsfilter**

Auch im Rahmen des Leistungs- und Medikationsfilters ist die Verwendung des Operators KEINSIN zulässig. Hierbei gilt ebenfalls der Grundsatz, dass das Vorliegen von Kodierungen aus

Filterlisten, welche mittels des Operators `KEINSIN` zur Anwendung gebracht werden, lediglich auf das Prüfergebnis bezogen auf den einzelnen Fall wirkt.

Für weitere Information bzgl. der Selektion von selektivvertraglich erbrachten Leistungen siehe **Hinweis zur Funktion `ebm_295s`** bzw. **Abschnitt Auslösung selektivvertraglich erbrachter Leistungen**.

### **Umgang mit der Funktion `drg_in_list()`**

Um für Patienten - deren Aufgriff durch ein positives Prüfergebnis des Patientenfilters bereits gesichert ist - ggf. vorliegende, auswertungsrelevante Informationen über sog. „Ausschlüsse im (Behandlungs-) Verlauf“ zu erhalten, soll hierfür u.a. die Funktion `drg_in_list()` zur Anwendung gebracht werden. An dieser Stelle wird das Ergebnis der Funktion `drg_in_list()` einerseits nicht negiert und andererseits mittels Disjunktion mit weiteren Teilbedingungen verknüpft. Dadurch wird ein positives Prüfergebnis des Filters bewirkt, sobald für einen Fall eine Kodierung aus der DRG-Filterliste vorliegt. In diesem Falle wird als Surrogat entsprechend der Wert für den Wahrheitswert `WAHR` exportiert.

## **2.2      Datenexport**

### **2.2.1      Ziel und Umfang des Datenexports**

Der Datenexport hat zum Ziel, die im vorherigen Schritt selektierten Daten der Krankenkassen im von der BAS spezifizierten Format in Exportdateien für die weitere Verarbeitung und Übermittlung abzulegen.

Für jedes Modul (an einem Verfahren können mehrere Module beteiligt sein) mit Sozialdaten ist von jeder Krankenkasse eine separate Datenlieferung je Erfassungsjahr zu exportieren.

Für Fälle, zu denen sowohl durch Anwendung des Patienten- als auch der Leistungs- und Medikationsfilters ein positives Prüfergebnis einhergeht, ist eine Zusammenführung, entsprechend Abschnitt 2.1.6, im Rahmen des Exports zu beachten.

### **2.2.2      Voraussetzungen für den Datenexport**

#### **Dateien und Dateiformate**

Um die Daten für die weitere Verarbeitung exportieren zu können, müssen Art, Format und Struktur der zu erzeugenden Dateien klar definiert sein.

#### **Mapping-Tabelle**

Wie schon im Zusammenhang mit der Datenselektion bemerkt, kann sich die Spezifikation nur auf die Datenstruktur der von Leistungserbringern an die Krankenkassen übermittelten Daten (Eingangsdaten) beziehen und nicht auf die individuell verschiedenen Datenspeichersysteme der Krankenkassen. Die Umsetzung des Datenexports erfordert deswegen, dass die einzelne Krankenkasse in der Lage ist zu identifizieren, wo und in welcher Form die in ihrem System selektierten Daten innerhalb der Datenstruktur der Exportdateien abzulegen sind. Dazu ist eine Überleitungstabelle (Mapping-Tabelle) von den Daten der Leistungserbringer zu den Exportdaten notwendig.

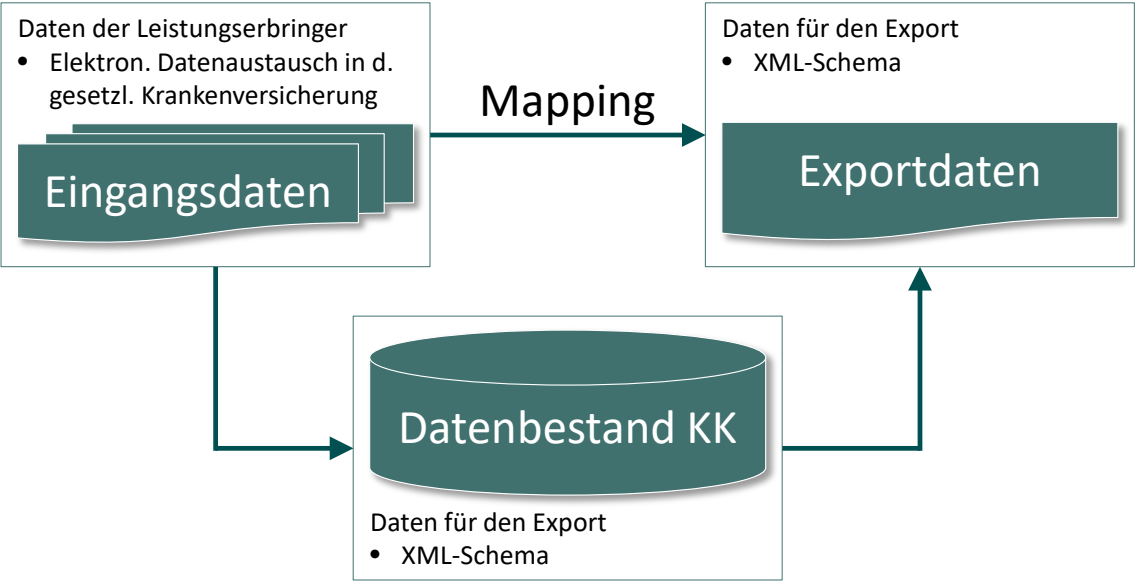


Abbildung 10: Funktion der Mapping-Tabelle

Mit Hilfe dieser Mapping-Tabelle können die einzelnen Datenfelder der Eingangsdaten eindeutig einem entsprechenden Datenfeld in den Exportdateien zugeordnet werden.

2.2.3      Exportdateien

Art und Anzahl der Exportdateien in einer Datenlieferung ergeben sich aus den verschiedenen Datenkategorien und ihrem Empfänger im Datenfluss.

In der Datenlieferung sind fünf Kategorien zu unterscheiden:

Tabelle 13: Datenkategorien im Datenfluss

| Datenkategorie                            | XML-Element | Beschreibung  |
|---|-------------|---|
| Admin-Daten/<br>Kopfdaten                 | header      | Enthält Meta-Informationen zum Datenlieferanten, über das Datenpaket, Informationen zur Organisation des Datenflusses und zur Identifizierung der Transaktion. Pro Datei ist der Header nur einmal vorhanden. Jede Stelle im Datenfluss muss diese Daten einsehen können. |
| Krankenkassenidentifizierende Daten (KID) | body        | Enthält die Haupt-IK-Nummer der Krankenkasse. Die Datenannahmestelle der Krankenkassen (DAS-KK) ersetzt die KID durch ein Pseudonym, sofern in den themenspezifischen Bestimmungen nichts anderes vorgesehen ist.   |
| Patientenidentifizierende Daten (PID)     | patient     | Die PID müssen mit dem öffentlichen Schlüssel der VST (PKCS#7) verschlüsselt werden. Nur die Vertrauensstelle (VST) kann diesen   |



| Datenkategorie                                 | XML-Element   | Beschreibung  |
|--|---------------|---|
|  |               | Bereich einsehen und ersetzt die PID durch ein (verfahrensspezifisches) Pseudonym.  |
| Leistungserbringeridentifizierende Daten (LID) | care_provider | Die LID müssen gemeinsam mit den QS-Daten mit dem öffentlichen Schlüssel der DAS-KK (PKCS#7) verschlüsselt werden. Nur die DAS-KK kann diesen Bereich einsehen. Die DAS-KK anonymisiert die LID oder pseudonymisiert sie dem bundesweit einheitlichen Pseudonymisierungsverfahren gemäß, wie in den themenspezifischen Bestimmungen festgelegt. |
| QS-Daten                                       | qs_data       | Dies sind die Fall- und Versichertendaten ohne PID. Diese Daten dürfen von der DAS-KK und der BAS eingesehen werden. Sie müssen von der absendenden Krankenkasse mit dem öffentlichen Schlüssel der DAS-KK (PKCS#7) und von der DAS-KK mit dem öffentlichen Schlüssel der BAS (W3C-XML-Encryption) verschlüsselt werden.                        |

### Pseudonymisierte Dienstleisterkennung und Versichertenzahl der KK als Admin-Daten

Die Krankenkassen haben in Teilen Dienstleister mit der Datenselektion beauftragt. Ein Dienstleister kann dann Daten mehrerer Kassen verarbeiten. Im Sinne einer genauen Modellierung der Auffälligkeit ist es notwendig, die Sozialdatenlieferungen dienstleisterspezifisch zu betrachten. Dies erfolgt in Analogie zur Software-Anbieterkennung in den QS-Daten. Daher ist zur Bestimmung der Auffälligkeit einzelner Krankenkassen im Rahmen der Sozialdatenvalidierung gemäß § 16 der DeQS-Richtlinie eine pseudonymisierte Dienstleisterkennung erforderlich. Als Dienstleister sind die Stellen im Prozessverlauf gemeint, die die Filter der Sozialdatenspezifikation umsetzen. Als Dienstleisterkennung ist die Handelsregisternummer einzutragen. Sofern eine Krankenkasse keinen Dienstleister mit der Auslösung beauftragt hat, kann das Feld frei bleiben. Die DAS-KK ersetzt die Dienstleisterkennung analog zur KID (siehe Tabelle 13) durch ein Pseudonym.

Das IQTIG soll gemäß der Tragenden Gründe zu § 16 Absatz 6 die Vollständigkeit der Sozialdaten plausibilisieren. Dazu soll die Versichertenzahl der Krankenkasse übermittelt werden. Zur Prüfung, ob die Krankenkassen entsprechend ihrer Größe Daten geliefert hat, benötigt das IQTIG die Versichertenzahl jeder Krankenkasse. Als Versichertenzahl ist die aktuellste Zahl zu übermitteln, welche im Rahmen der GKV-Statistik KM1 ermittelt wird. Da es sich bei beiden Exportdaten (Dienstleisterkennung und Versichertenzahl) um übergreifende Informationen der Krankenkassen handelt, werden diese der Datenkategorie Admin-Daten zugeordnet. Die Übermittlung der Versichertenzahl ist für alle Datenlieferungen für Module nach DeQS-RL Pflicht und wird durch administrative Prüfungen sichergestellt (siehe Abschnitt 2.3.3).

Bedingt durch den Einsatz des Krankenkassenkommunikationssystems (KKS) müssen Daten, die für verschiedene Empfänger verschlüsselt werden sollen, in getrennten Dateien vorliegen (siehe Abschnitt 2.4). Das bedeutet, dass die PID, die nur von der Vertrauensstelle (VST) eingesehen werden dürfen, von den für die Datenannahmestelle der Krankenkassen (DAS-KK) einsehbaren Daten getrennt exportiert werden müssen. Es sind also von jeder Krankenkasse jeweils zwei Dateien zu erzeugen: eine QS-Datei, die alle für die DAS-KK bestimmten Daten (inklusive der krankenkassen- und leistungserbringeridentifizierenden Daten) enthält sowie eine zugehörige PID-Datei, die die patientenidentifizierenden Daten der QS-Datensätze enthält und nur von der VST eingesehen werden kann. Der detaillierte Datenfluss ist ebenfalls in Abschnitt 2.4 dargestellt. Es sei bereits an dieser Stelle darauf hingewiesen, dass die Separation von QS- und PID-Datei ein gemeinsames Merkmal erfordert, auf dessen Grundlage eine spätere Zusammenführbarkeit dieser beiden Dokumente gewährleistet wird.

Bei Verfahren, die keine PID verwenden (Nicht-PID-Verfahren) entfällt die Beteiligung der Vertrauensstelle im Datenfluss. In diesem Fall wird keine separate PID-Datei erzeugt, sondern nur eine QS-Datei.

**2.2.4      XML-Struktur**

Als Exportformat für PID- und QS-Dateien wird XML genutzt. Die Struktur der XML-Dateien wird jeweils durch ein XML-Schema aus der Spezifikationsdatenbank eindeutig definiert. Da die Datenbestände der Krankenkassensysteme nicht einheitlich gestaltet sind, kann die Erzeugung der Exportdatei nicht im Detail beschrieben werden und muss individuell durch die Krankenkassen erfolgen. Hier wird stattdessen die Struktur der zu erzeugenden Dateien erläutert. Das Mapping der selektierten Daten in diese Struktur muss jeweils in Abhängigkeit von der Struktur der Datenbestände der Krankenkassen erfolgen.

Die grundlegende XML-Struktur ist im Abschnitt 6.4 beschrieben. Die spezifische XML-Struktur im Bereich der PID und QS-Daten ergibt sich erst im Kontext eines bestimmten Moduls.

Folgende Tabelle zeigt an, nach welchen XML-Schemata die Krankenkassen den Datenexport durchführen:

*Tabelle 14: XML-Schemata für die Krankenkassen*

| Ablageort    | XML-Schema           | Verwendungszweck                            |
|--------------|----------------------|---|
| interface_KK | interface_KK_QS.xsd  | Datenstruktur und Validierung der QS-Datei  |
|              | interface_KK_PID.xsd | Datenstruktur und Validierung der PID-Datei |

## 2.2.5 Überleitung von den Daten der Leistungserbringer zu den Exportdaten (Mapping-Tabelle)

Die Mapping-Tabellen für die PID- und die QS-Datei sind als Abfragen in der Spezifikationsdatenbank hinterlegt.

| modul  | eingangskennung  | berechnet                           | funktion       | funktionBesch     |
|--|--|-------------------------------------|----------------|-------------------|
| Admin@kassenknr  | Admin@kassenknr  | <input type="checkbox"/>            |                |                   |
| sequential_nr(Admin)@lfndr   | sequential_nr(Admin)@lfndr   | <input checked="" type="checkbox"/> | sequential_nr  | Laufende Nummer   |
| Stamm@geschlecht   | Stamm@geschlecht   | <input type="checkbox"/>            |                |                   |
| Stamm@gebjahr  | Stamm@gebjahr  | <input type="checkbox"/>            |                |                   |
| Stamm@sterbedatum  | Stamm@sterbedatum  | <input type="checkbox"/>            |                |                   |
| Stamm@versicherungsdatum   | Stamm@versicherungsdatum   | <input type="checkbox"/>            |                |                   |
| Stamm@versicherungsstatus  | Stamm@versicherungsstatus  | <input type="checkbox"/>            |                |                   |
| source(301)@quelle   | source(301)@quelle   | <input checked="" type="checkbox"/> | source         | Kennzeichen der g |
| 301.Aufnahmesatz.Auf.Aufnahmetag@aufdatum  | 301.Aufnahmesatz.Auf.Aufnahmetag@aufdatum  | <input type="checkbox"/>            |                |                   |
| 301.Aufnahmesatz.Auf.Aufnahmegrund@aufgrund  | 301.Aufnahmesatz.Auf.Aufnahmegrund@aufgrund  | <input type="checkbox"/>            |                |                   |
| 301.Entlassungsanzeige.ETL.Tag der Entlassung/Verlegung@entldatum                            | 301.Entlassungsanzeige.ETL.Tag der Entlassung/Verlegung@entldatum                            | <input type="checkbox"/>            |                |                   |
| 301.Entlassungsanzeige.ETL.Entlassungs-/Verlegungsgrund@entlgrund                            | 301.Entlassungsanzeige.ETL.Entlassungs-/Verlegungsgrund@entlgrund                            | <input type="checkbox"/>            |                |                   |
| inpatient_interrupt(301.Entlassungsanzeige.ETL.Entlassungs-/Verlegungsgrund)@khunterbrechung | inpatient_interrupt(301.Entlassungsanzeige.ETL.Entlassungs-/Verlegungsgrund)@khunterbrechung | <input checked="" type="checkbox"/> | inpatient_inte | Unterbrechung des |
| state_key(301.Entlassungsanzeige.FKT.IK des Absenders)@bundesland                            | state_key(301.Entlassungsanzeige.FKT.IK des Absenders)@bundesland                            | <input checked="" type="checkbox"/> | state_key      | Bundeslandschlüss |
| 301.Entlassungsanzeige.FKT.IK des Absenders@nummer   | 301.Entlassungsanzeige.FKT.IK des Absenders@nummer   | <input type="checkbox"/>            |                |                   |
| 301.Entlassungsanzeige.FAB.Operation.Prozedurenschlüssel@ops                                 | 301.Entlassungsanzeige.FAB.Operation.Prozedurenschlüssel@ops                                 | <input type="checkbox"/>            |                |                   |
| 301.Entlassungsanzeige.FAB.Operation.Lokalisation@lokalisation                               | 301.Entlassungsanzeige.FAB.Operation.Lokalisation@lokalisation                               | <input type="checkbox"/>            |                |                   |
| 301.Entlassungsanzeige.FAB.Operation.tag@datum   | 301.Entlassungsanzeige.FAB.Operation.tag@datum   | <input type="checkbox"/>            |                |                   |
| 301.Entlassungsanzeige.ETL.Hauptdiagnose.Diagnoseschlüssel@icd                               | 301.Entlassungsanzeige.ETL.Hauptdiagnose.Diagnoseschlüssel@icd                               | <input type="checkbox"/>            |                |                   |
| 301.Entlassungsanzeige.ETL.Hauptdiagnose.Lokalisation@lokalisation                           | 301.Entlassungsanzeige.ETL.Hauptdiagnose.Lokalisation@lokalisation                           | <input type="checkbox"/>            |                |                   |
| 301.Entlassungsanzeige.ETL.Sekundär-Diagnose.Diagnoseschlüssel@icd_sek                       | 301.Entlassungsanzeige.ETL.Sekundär-Diagnose.Diagnoseschlüssel@icd_sek                       | <input type="checkbox"/>            |                |                   |
| 301.Entlassungsanzeige.ETL.Sekundär-Diagnose.Lokalisation@lokalisation_sek                   | 301.Entlassungsanzeige.ETL.Sekundär-Diagnose.Lokalisation@lokalisation_sek                   | <input type="checkbox"/>            |                |                   |
| 301.Entlassungsanzeige.NDG.Nebendiagnose.Diagnoseschlüssel@icd                               | 301.Entlassungsanzeige.NDG.Nebendiagnose.Diagnoseschlüssel@icd                               | <input type="checkbox"/>            |                |                   |
| 301.Entlassungsanzeige.NDG.Nebendiagnose.Lokalisation@lokalisation                           | 301.Entlassungsanzeige.NDG.Nebendiagnose.Lokalisation@lokalisation                           | <input type="checkbox"/>            |                |                   |
| 301.Entlassungsanzeige.NDG.Sekundär-Diagnose.Diagnoseschlüssel@icd_sek                       | 301.Entlassungsanzeige.NDG.Sekundär-Diagnose.Diagnoseschlüssel@icd_sek                       | <input type="checkbox"/>            |                |                   |
| 301.Entlassungsanzeige.NDG.Sekundär-Diagnose.Lokalisation@lokalisation_sek                   | 301.Entlassungsanzeige.NDG.Sekundär-Diagnose.Lokalisation@lokalisation_sek                   | <input type="checkbox"/>            |                |                   |
| 301.Entlassungsanzeige.ETL.Fachabteilung@fachabteilung                                       | 301.Entlassungsanzeige.ETL.Fachabteilung@fachabteilung                                       | <input type="checkbox"/>            |                |                   |
| source(kh_ambo)@quelle   | source(kh_ambo)@quelle   | <input checked="" type="checkbox"/> | source         | Kennzeichen der g |
| kh_ambo.Ambulante Operation.REC.Tag des Zugangs@zugangsdatum                                 | kh_ambo.Ambulante Operation.REC.Tag des Zugangs@zugangsdatum                                 | <input type="checkbox"/>            |                |                   |
| state_key(kh_ambo.Ambulante Operation.FKT.IK des Absenders)@bundesland                       | state_key(kh_ambo.Ambulante Operation.FKT.IK des Absenders)@bundesland                       | <input checked="" type="checkbox"/> | state_key      | Bundeslandschlüss |
| kh_ambo.Ambulante Operation.FKT.IK des Absenders@nummer                                      | kh_ambo.Ambulante Operation.FKT.IK des Absenders@nummer                                      | <input type="checkbox"/>            |                |                   |
| kh_ambo.Ambulante Operation.PRZ.Prozedur.Prozedurenschlüssel@ops                             | kh_ambo.Ambulante Operation.PRZ.Prozedur.Prozedurenschlüssel@ops                             | <input type="checkbox"/>            |                |                   |
| kh_ambo.Ambulante Operation.PRZ.Prozedur.Lokalisation@lokalisation                           | kh_ambo.Ambulante Operation.PRZ.Prozedur.Lokalisation@lokalisation                           | <input type="checkbox"/>            |                |                   |
| kh_ambo.Ambulante Operation.PRZ.Prozedur.tag@datum   | kh_ambo.Ambulante Operation.PRZ.Prozedur.tag@datum   | <input type="checkbox"/>            |                |                   |
| kh_ambo.Ambulante Operation.BDG.Behandlungsdiagnose.Diagnoseschlüssel@icd                    | kh_ambo.Ambulante Operation.BDG.Behandlungsdiagnose.Diagnoseschlüssel@icd                    | <input type="checkbox"/>            |                |                   |
| kh_ambo.Ambulante Operation.BDG.Behandlungsdiagnose.Lokalisation@lokalisation                | kh_ambo.Ambulante Operation.BDG.Behandlungsdiagnose.Lokalisation@lokalisation                | <input type="checkbox"/>            |                |                   |
| kh_ambo.Ambulante Operation.BDG.Behandlungsdiagnose.Diagnosesicherheit@sicicherheit          | kh_ambo.Ambulante Operation.BDG.Behandlungsdiagnose.Diagnosesicherheit@sicicherheit          | <input type="checkbox"/>            |                |                   |
| kh_ambo.Ambulante Operation.BDG.Sekundär-Diagnose.Diagnoseschlüssel@icd_sek                  | kh_ambo.Ambulante Operation.BDG.Sekundär-Diagnose.Diagnoseschlüssel@icd_sek                  | <input type="checkbox"/>            |                |                   |
| kh_ambo.Ambulante Operation.BDG.Sekundär-Diagnose.Lokalisation@lokalisation_sek              | kh_ambo.Ambulante Operation.BDG.Sekundär-Diagnose.Lokalisation@lokalisation_sek              | <input type="checkbox"/>            |                |                   |

Abbildung 11: Mapping-Tabellen in der Spezifikationsdatenbank

Die Felder der Mapping-Tabelle sind in Tabelle 15 dargestellt:

Tabelle 15: Struktur der Abfragen *MappingTabellePID* bzw. *MappingTabelleQS*

| Datenfeld             | Bedeutung  | Verwendung   |
|-----------------------|--|--|
| modul                 | Zugehörigkeit zu einem Modul   | Filterung  |
| eingangskennung       | Eindeutige Kennung des Datums im Eingangsdatensatz. Diese Kennung setzt sich zusammen aus den Feldern datenquelle, datensatz, segment, gruppe und feldkennung sowie dem Namen des Zielattributs. | Referenz auf Eingangsdaten                             |
| berechnet             | Das Feld wird mit einer Funktion berechnet   | Hinweis auf eine Funktion (siehe Tabelle 4)            |
| funktion              | Enthält ggf. eine Referenz auf die Funktion  |  |
| funktion Beschreibung | Beschreibung der Funktion  |  |
| funktionFormel        | Berechnungsvorschrift der Funktion   |  |
| datenquelle           | Herkunft des Datenbestandes (z. B. § 301)  | Auffinden der Quelldaten gemäß den Technischen Anlagen |
| datensatz             | Referenz auf den Herkunftsdatensatz  |  |
| segment               | Referenz auf das Herkunftssegment  |  |
| gruppe                | Referenz auf die Herkunftsgruppe   |  |

| Datenfeld                 | Bedeutung  | Verwendung                                |
|---------------------------|--|---|
| feldkennung               | Referenz auf die Herkunftsfeldkennung  | zum Datenaustausch der Leistungserbringer |
| kennungElement            | Eindeutige Kennung des Elements im Exportdatensatz, das das Zielattribut enthält. Die Kennung enthält den Pfad zum gesuchten Element in der XML-Datei. | Referenz auf Exportdaten                  |
| elementParent             | Name des Elternelements  |   |
| kardinalitaet             | Anzahl der Elemente des angegebenen Typs unterhalb des Elternelements  |   |
| element                   | Zielelement im XML-Format  |   |
| element<br>Bezeichnung    | Beschreibung des Elements  |   |
| elementHinweis            | ggfs. Hinweis auf Besonderheiten des Elements  |   |
| elementDatentyp           | Datentyp des Elements im XML-Schema  |   |
| attribut                  | Zielattribut im XML-Format, in dem das Datenfeld exportiert wird   |   |
| attribut<br>Bezeichnung   | Beschreibung des Attributs   | Beschreibung des Zielattributs            |
| laenge                    | Maximale Stellenzahl des Attributwerts   |   |
| nachKommaLaenge           | Maximale Zahl der Nachkommastellen   |   |
| min                       | Mindestwert  |   |
| max                       | Maximalwert  |   |
| basisTyp                  | Name des Attributtyps (siehe Tabelle BasisTyp in der Spezifikationsdatenbank)  |   |
| basisTyp<br>Bezeichnung   | Beschreibung des Attributtyps  |   |
| format                    | Formatanweisung zur schnellen Übersicht  |   |
| formatRegExp              | Formatanweisung als Regulärer Ausdruck (Extended Regular Expression), bindende Formatanweisung   |   |
| schluessel                | Name des zu verwendenden Schlüssels (siehe Tabellen Schluessel und SchluesselWert in der Spezifikationsdatenbank)                                      |   |
| schluessel<br>Bezeichnung | Beschreibung des Schlüssels  |   |
| mussKann                  | Muss- (M) oder Kann-Feld (K)   |   |

| Datenfeld             | Bedeutung                                   | Verwendung |
|-----------------------|---|------------|
| beabsichtigter Inhalt | Beschreibung des beabsichtigten Feldinhalts |            |

Nach dem passenden Modul gefiltert, enthält die Tabelle alle in die jeweilige Datei aufzunehmenden Attribute und beschreibt genau deren Herkunft, Ziel und Format. Die eindeutige Kennung im Feld `eingangskennung`, die in den Feldern `datenquelle` bis `feldkennung` aufgeschlüsselt ist, erlaubt die Identifizierung des jeweiligen Datenfeldes in den Daten der Leistungserbringer und damit die Referenzierung innerhalb der Datenbestände der Krankenkasse. Die Felder `kennungElement` und `Folgende` beschreiben die Position des Elements in der XML-Struktur der jeweiligen Exportdatei (siehe Abschnitt 2.2.4). Dies ist das Element, das das Datenfeld als Attribut mit dem im Feld `attribut` angegebenen Namen aufnimmt. Die Felder ab `attributBezeichnung` beschreiben das Attribut und sein vorgeschriebenes Format. Sofern für ein Attribut ein bestimmter Schlüssel vorgeschrieben ist, können die hierfür zulässigen Schlüsselwerte der Tabelle `SchluesselWert` entnommen werden (siehe Abschnitt 2.1.2).

Einige Datenfelder des Exportformats entsprechen nicht unmittelbar einem Datenfeld der Eingangsdaten, sondern werden mittels einer Funktion von diesem abgeleitet oder auch unabhängig von den Eingangsdaten bestimmt. Dies ist der Fall, wenn das Feld `berechnet` den Wert `true` enthält. In diesem Fall wird im Feld `funktion` die konkrete anzuwendende Funktion referenziert.

### 2.2.6 Beispiel für Datenexport

In Tabelle 16 ist ein Beispiel für ein selektiertes Datenfeld dargestellt. Hierin wird ein Datenfeld aus den Eingangsdaten der Leistungserbringer im Abrechnungsverfahren nach § 301 referenziert. Es handelt sich um das Feld `Aufnahmetag` im AUF-Segment des Aufnahmesatzes. Dieses Feld wird mit dem Attributnamen `aufndatum` in das Element `fall_301` exportiert, welches beliebig oft unterhalb des Elternelementes `BSP` auftreten darf. Das Feld wird als 10-stelliges Datum im Format `JJJJ-MM-TT` ausgegeben.

Tabelle 16: Beispiel für ein selektiertes Datenfeld

| Datenfeld                         | Inhalt                                     |
|-----------------------------------|--|
| <code>modul</code>                | BSP  |
| <code>eingangskennung</code>      | 301.Aufnahmesatz.AUF.Aufnahmetag@aufndatum |
| <code>berechnet</code>            | false                                      |
| <code>funktion</code>             |  |
| <code>funktionBeschreibung</code> |  |
| <code>funktionFormel</code>       |  |
| <code>datenquelle</code>          | 301  |

| Datenfeld             | Inhalt  |
|-----------------------|---|
| datensatz             | Aufnahmesatz  |
| segment               | AUF   |
| gruppe                |   |
| feldkennung           | Aufnahmetag   |
| kennungElement        | root.body.case.qs_data.BSP.fall_301   |
| elementParent         | BSP   |
| kardinalitaet         | *   |
| element               | fall_301  |
| elementBezeichnung    | Fall nach § 301   |
| elementHinweis        |   |
| elementDatentyp       | fall301_type  |
| attribut              | aufndatum   |
| attributBezeichnung   | Aufnahmedatum Krankenhaus   |
| laenge                | 10  |
| nachKommaLaenge       |   |
| min                   |   |
| max                   |   |
| basisTyp              | DATUM   |
| basisTypBezeichnung   | Zehnstelliges Datum   |
| format                | JJJJ-MM-TT  |
| formatRegExp          | [12][0-9]{3}-((0[1-9]) (1[0-2]))-((0[1-9]) ([12][0-9]) (3[01]))               |
| schluessel            |   |
| schluesselBezeichnung |   |
| mussKann              | M   |
| beabsichtigterInhalt  | Erster Aufnahmetag des Falles, an dem der Versicherte ins KH aufgenommen wird |

Das Beispiel in Tabelle 17 zeigt ein berechnetes Datenfeld. Die Referenzierung des Ausgangsdatenfelds erfolgt analog zum vorherigen Beispiel (hier auf das Institutionskennzeichen des Absenders). Allerdings wird in diesem Beispiel nicht der Inhalt des Datenfelds direkt exportiert, sondern die Funktion `state_key` angewendet, die den Bundeslandschlüssel für das Krankenhaus berechnet. Hierbei ist als abstrakte Berechnungsvorschrift primär der Datenbestand der Krankenkasse angegeben, da dieser zuverlässiger ist als die direkte Berechnung aus der 3. und

4. Stelle der IKNR. Der zu verwendende Schlüssel `bundesland` entspricht der in der IKNR verwendeten Kodierung und ist in der Tabelle `SchlüsselWert` erläutert.

Tabelle 17: Beispiel für ein berechnetes Datenfeld

| Datenfeld                         | Inhalt  |
|-----------------------------------|---|
| <code>modul</code>                | BSP   |
| <code>eingangskennung</code>      | <code>state_key(301.Entlassungsanzeige.FKT.IK des Absenders)@bundesland</code>                        |
| <code>berechnet</code>            | true  |
| <code>funktion</code>             | <code>state_key</code>  |
| <code>funktionBeschreibung</code> | Bundeslandschlüssel   |
| <code>funktionFormel</code>       | Aus dem Datenbestand der Krankenkasse zu entnehmen oder ersatzweise aus der 3. und 4. Stelle der IKNR |
| <code>datenquelle</code>          | 301   |
| <code>datensatz</code>            | Entlassungsanzeige  |
| <code>segment</code>              | FKT   |
| <code>gruppe</code>               |   |
| <code>feldkennung</code>          | IK des Absenders  |
| <code>kennungElement</code>       | <code>root.body.case.qs_data.BSP.fall_301.care_provider</code>  |
| <code>elementParent</code>        | <code>fall_301</code>   |
| <code>kardinalitaet</code>        | 1   |
| <code>element</code>              | <code>care_provider</code>  |
| <code>elementBezeichnung</code>   | Leistungserbringer nach § 301   |
| <code>elementHinweis</code>       |   |
| <code>elementDatentyp</code>      | <code>care_provider_type</code>   |
| <code>attribut</code>             | <code>bundesland</code>   |
| <code>attributBezeichnung</code>  | Bundesland des Krankenhauses  |
| <code>laenge</code>               | 2   |
| <code>nachKommaLaenge</code>      |   |
| <code>min</code>                  |   |
| <code>max</code>                  |   |
| <code>basisTyp</code>             | SCHLUESSEL  |
| <code>basisTypBezeichnung</code>  | Schlüssel mit alphanumerischen Schlüsselcodes   |

| Datenfeld             | Inhalt   |
|-----------------------|--|
| format                |  |
| formatRegExp          |  |
| schluessel            | bundesland   |
| schluesselBezeichnung | Bundesland   |
| mussKann              | M  |
| beabsichtigterInhalt  | Bundesland aus der IKNR des Krankenhauses (Stellen 3+4)<br>bzw. aus Datenbestand der Kasse |

Weitere XML-Beispiele befinden sich im Paket der XML-Schemata. Nähere Informationen sind im Abschnitt 6.4 zu finden.

### 2.2.7 Segmentierung

Um die Größe der zu übertragenden Dateien zu begrenzen, dürfen für Verfahren gemäß DeQS-RL nicht mehr als 10.000 Datensätze in einer Exportdatei enthalten sein. Für Verfahren der Methodenbewertung können maximal 50.000 Datensätze in einer Exportdatei enthalten sein. Diese Vorgaben sind modulspezifisch in der Spezifikationsdatenbank hinterlegt in Tabelle Modul in der Spalte `segmentierung`. Die Richtlinienzugehörigkeit der einzelnen Module ist in Spalte `richtlinie` hinterlegt. Als Datensatz für die Zählung gilt ein Versicherter (Patient) mit allen dazugehörigen Falldaten (Behandlungsfälle und Verordnungen). Wurden für den gesetzten Filter eines Moduls mehr als die maximale Anzahl der Datensätze selektiert, sind diese auf mehrere Dateien (Segmente) aufzuteilen. Bei der Aufteilung einer Datenlieferung auf mehrere Segmente müssen folgende Besonderheiten beachtet werden:

- Die in der XML-Struktur vorgesehene laufende Nummer des Datensatzes `/root/body/case/@lfdnr` muss segmentübergreifend eindeutig vergeben werden.
- Das Header-Attribut `set_id` muss bei allen einer Transaktion zugehörigen Segmenten den gleichen Wert aufweisen. Es dient insbesondere der Zusammenführung sämtlicher Segmente zu einer Transaktion in der BAS. Darüber hinaus kommt dieser Wert bei der transaktionsbezogenen (also segmentübergreifenden) Rückprotokollierung in Form des Datenflussprotokolls von der BAS für die KK zur Anwendung.
- Das Header-Attribut `total_count` muss in allen Segment-Dateien die Gesamtzahl aller Datensätze der Gesamtlieferung enthalten. Es muss somit analog zum Header-Attribut `set_id` bei allen einer Transaktion zugehörigen Segmenten den gleichen Wert aufweisen.
- Bei PID-Modulen werden für QS- und PID-Dateien segmentweise die gleichen Kennungen in `/root/header/@id` verwendet.

### 2.2.8 Nullmeldung

Sollte für die Kombination aus einem Modul und einem Erfassungsjahr vom Patientenfilter kein Versicherter aufgegriffen worden sein, so ist dennoch eine Transaktion im Sinne einer Nullmel-



dung durchzuführen. Im Unterschied zu einer Transaktion deren Ergebnismenge nach Anwendung des Patientenfilters nicht leer ist, enthält eine Nullmeldung keine `case`-Elemente. Weiterhin ist eine Nullmeldung grundsätzlich als R-Lieferung zu kennzeichnen, d.h. lediglich der Versand einer Q-Datei mit 0 Elementen ist erforderlich. Der Versand einer zugehörigen P-Datei mit 0 `case`-Elementen ist nicht gestattet (genauere Informationen zu R-Lieferung, Q- sowie P-Datei sind im Kapitel 2.4.4 unter „Physische Dateinamen“ zu finden).

Dadurch wird sichergestellt, dass zu jedem Lieferzeitfenster von jeder Krankenkasse eine Transaktion durchgeführt werden kann.

## 2.3 Datenprüfung

Bei der Datenprüfung sind drei Arten von Prüfungen zu unterscheiden:

1. Struktur- und Formatprüfungen
2. Plausibilitätsregeln
3. administrative Prüfungen

Plausibilitätsregeln beziehen sich auf QS- und leistungserbringeridentifizierende Daten. Diese sind im Gegensatz zu den administrativen Prüfungen jeweils verfahrensbezogen.

### 2.3.1 Struktur- und Formatprüfungen

Vor Prüfung der Plausibilitätsregeln muss eine Struktur- und Formatprüfung durchgeführt werden, die sicherstellt, dass die zu überprüfenden Felder im richtigen Format und ihre Werte im richtigen Gültigkeitsbereich liegen. Liegt das Dokument als XML vor, wird als Strukturprüfung eine Schemaprüfung durchgeführt. Liegt das Dokument zum Zeitpunkt der Prüfung noch nicht als XML vor, sind die Felder anderweitig über die Felddefinitionen (Datenformat, Feldlänge, Wertebereich, etc.) zu prüfen, um sicherstellen zu können, dass die Überführung der Daten in XML fehlerfrei durchgeführt werden kann.

Die Felddefinitionen sind entweder dem Schema (XML-Schema-SDAT\_<versionsnummer>/interface\_KK/Interface\_KK\_QS.xsd und XML-Schema-SDAT\_<versionsnummer>/interface\_KK/Interface\_KK\_PID.xsd) oder alternativ der Abfrage `FeldDefinition` der Spezifikationsdatenbank zu entnehmen.

Die Abfrage `FeldDefinition` enthält dazu folgende Angaben:

Tabelle 18: Struktur der Abfrage `FeldDefinition`

| Spalte                       | Bedeutung  |
|------------------------------|--|
| <code>modul</code>           | Verfahren  |
| <code>eingangskennung</code> | Eindeutige Identifikation des Datenfeldes                        |
| <code>attribut</code>        | XML-Attribut eines Elements                                      |
| <code>basisTyp</code>        | Datentyp <sup>6</sup> (ZAHL, GANZEZAHL, DATUM, SCHLUESSEL, etc.) |

<sup>6</sup> Details siehe Tabelle `BasisTyp` in der Spezifikationsdatenbank

| Spalte              | Bedeutung  |
|---------------------|--|
| schluessel          | Falls basisTyp dem Wert „SCHLUESSEL“ entspricht, wird der konkrete Schlüsselname angegeben. Die einzelnen Schlüsselcodes befinden sich in der Spezifikationsdatenbank in der Tabelle SchluesselWert. |
| basisTypBezeichnung | Bezeichnung des Datentyps  |
| format              | Format   |
| formatRegExp        | Regulärer Ausdruck   |
| laenge              | Feldlänge:<br>x = genaue Länge<br>..x = maximale Länge   |
| nachKommaLaenge     | Maximale Nachkommastellenlänge   |
| min                 | Minimalwert eines Gültigkeitsbereichs  |
| max                 | Maximalwert eines Gültigkeitsbereichs  |
| mussKann            | Angabe, ob Pflichtfeld oder optional:<br>K = Kann-Feld (optional)<br>M = Muss-Feld (Pflichtfeld)   |

Aus dieser Abfrage ergeben sich folgende Prüfungen:

- Überprüfung des Formats (Feldinhalt)
- Überprüfung der Feldlänge
- Überprüfung der Schlüsselcodes
- Überprüfung der numerischen Wertebereiche
- Überprüfung der Muss-Kann-Eigenschaft

### Beispiel:

Das Feld entldatum ist in der Spezifikationsdatenbank wie folgt definiert:

Tabelle 19: Beispiel der Definition eines Feldes

| modul          | attribut  | basisTyp               | basisTyp<br>Bezeichnung      | Format                  | formatRegExp  | laenge                  |
|----------------|-----------|------------------------|------------------------------|-------------------------|---|-------------------------|
| BSP            | entldatum | DATUM                  | Zehnstelliges<br>Datum       | JJJJ-MM-TT              | [12][0-9]{3}-<br>((0[1-9]) (1[0-2]))-<br>((0[1-9]) ([12]<br>[0-9]) (3[01])) | 10                      |
| Erklär-<br>ung | Feldname  | Datentyp<br>des Feldes | Bezeichnung<br>des Datentyps | Format des<br>Datentyps | Regulärer Ausdruck  | Feste<br>Feld-<br>länge |

Ist das Attribut vom Datentyp **SCHLUESSEL**, können die möglichen Werte im Unterdatenblatt wie in der folgenden Abbildung visualisiert werden:

| attribut            | basisTyp   | schluessel          | basisTypBezeichnung        | Format |
|---------------------|------------|---------------------|----------------------------|--------|
| ops                 | OPS        |                     | Prozedurenschlüssel in der |        |
| lokalisierung       | SCHLUESSEL | seitenlokalisierung | Schlüssel mit alphanumeris |        |
| fkSchluessel        | code       | bezeichnung         |                            |        |
| seitenlokalisierung | B          | beidseits           |                            |        |
| seitenlokalisierung | L          | links               |                            |        |
| seitenlokalisierung | R          | rechts              |                            |        |

Abbildung 12: Attribut Datentyp **SCHLUESSEL**

### 2.3.2 Plausibilitätsregeln

Nach den Struktur- und Formatprüfungen schließen sich Plausibilitätsprüfungen an. Diese befinden sich in der Spezifikationsdatenbank in der Abfrage **RegelDatenfeld**. Fehlende oder fehlerhafte Angaben in den QS-Datensätzen sollen dadurch verhindert werden.

Tabelle 20: Struktur der Abfrage **RegelDatenfeld**

| Spalte             | Bedeutung   | Verwendung   |
|--------------------|---|--|
| ModulName          | Name des Moduls, in dem die Regel verwendet wird        | Sortieren/Filtern  |
| Datenquelle Name   | Name der Datenquelle                                    | Sortieren/Filtern  |
| idRegel            | ID der Regel  | Referenzierung   |
| bedingung          | Bedingung, die die zugehörige Meldung der Regel auslöst | Plausibilitätsprüfung  |
| meldung            | Text der Fehlermeldung                                  | Rückmeldung und Beschreibung des Fehlers   |
| alternativ Meldung | Alternativer Text der Fehlermeldung                     | Nicht verwendet  |
| strenge            | Grad der Strenge  | „Harte“ Plausibilitätsregeln müssen von den Krankenkassen geprüft werden. „Weiche“ Regeln sind optional. |
| attributName       | Bezeichnung des verwendeten Attributs                   | Kann mehrfach pro Regel vorkommen. Bezeichnet die einzelnen verwendeten Datenfelder.                     |
| eingangskennung    | Eingangskennung des Datenfeldes                         | Identifizierung des Datenfeldes gemäß den technischen Anlagen zum Datenaustausch der Leistungserbringer  |

| Spalte         | Bedeutung                          | Verwendung  |
|----------------|------------------------------------|---|
| elementKennung | Kennung des Elements im XML-Export | Identifizierung des Ausgangselements zur Prüfung in XML |

Es wird zwischen zwei Arten von Plausibilitätsregeln unterschieden, die der Spalte `strenge` zu entnehmen sind: harte Regeln und weiche bzw. warnende Regeln. Die zwei Arten der Plausibilitätsregeln werden in unterschiedlichen Kontexten (Datenexport bzw. Datenannahme) und an unterschiedlichen Stellen durchgeführt und haben unterschiedliche Konsequenzen. Die Tabelle 21 gibt einen Überblick.

Weiche bzw. warnende Plausibilitätsregeln können im gesamten Datenfluss durchgeführt werden. Im entsprechenden Fehlerfall erhält das XML-Protokoll auf Datensatzebene den Eintrag `WARNING`. Der Gesamtstatus auf Dokumentenebene einer Datenlieferung ist `WARNING` sobald auf Datensatzebene ein `WARNING` - aber kein `ERROR` - vorliegt.

Harte Plausibilitätsregeln sind für alle am Datenfluss beteiligten Stellen obligat. Regelverletzungen mit der Strenge `hart` führen zu einem `ERROR` auf Datensatzebene. `//status_document/@V` entspricht `ERROR`, wenn alle Datensätze den Status `ERROR` in `case/case_admin/protocol/status_case/@V` aufweisen. Ist ein `status_case/@V` `WARNING` oder `OK`, dann ist auch `//status_document/@V` `WARNING`.



#### Hinweis

Entspricht das Attribut `regelErrorIsWarning` aus der Tabelle `DB-SDAT.ModulErfassungsjahr` dem Wert `true`, dann werden Regeln mit der Strenge `hart` für dieses Modul und Erfassungsjahr weich interpretiert. In den XML-Datenflussprotokollen markiert der `rule_type` „D“ die entsprechenden Regelverletzungen. `regelErrorIsWarning` hat im Gegensatz zu Regeln aus der Tabelle `DB-SDAT.Regel` keinen Einfluss auf administrative Prüfungen aus `DB-SDAT.Pruefung`.

Tabelle 21: Arten der Plausibilitätsregeln

| Art der Prüfung        | Kürzel (rule_type) | XML-Protokoll | Prüfung durch KK | Prüfung durch DAS-KK/BAS | verhindert Datenentgegennahme des Datensatzes |
|------------------------|--------------------|---------------|------------------|--------------------------|---|
| hart                   | H                  | ERROR         | ja               | ja                       | ja  |
| weich, eigentlich hart | D                  | WARNING       | ja               | ja                       | nein  |
| weich/warnend          | W                  | WARNING       | optional         | optional                 | nein  |

**Fehlende Attribute**

Prüfregeln werden in der Regel nicht auf fehlende Attribute angewendet. Die einzige Ausnahme hiervon stellen explizite Prüfungen auf das Vorhandensein eines Attributs dar (`attribut-name=NULL`).

**Beispiel:**

Die Prüfregel `ID=10 hauptdiagnose.icd_sek NICHTIN PCI_ICD_KOMPL` wird nur angewendet, wenn das entsprechende Attribut `icd_sek` im `<hauptdiagnose>`-Element des Falles vorhanden ist. Andernfalls wird die Regel ignoriert.

**Beispiel:**

Die Prüfregel `ID=49 fall_300.care_provider.art='bsnr' UND kvregion=NULL` prüft ausdrücklich die Existenz des Attributs `kvregion`, wenn als Art des Leistungserbringers der Wert `bsnr` angegeben ist. Diese Regel muss auch dann angewendet werden, wenn das Attribut fehlt.

**2.3.3 Administrative Prüfungen**

Administrative Prüfungen sind in der Regel verfahrensunabhängig und definieren, welche Prüfungen im Rahmen des Exports (von der Krankenkasse bis zur (Bundes-)Auswertungsstelle (BAS)) innerhalb einer Einrichtung in welcher zeitlichen Reihenfolge durchzuführen sind. Administrative Prüfungen beziehen sich nicht auf Datenfelder innerhalb der QS- und PID-Daten, sondern auf übergeordnete Prüfungen, wie die Feststellung korrekter Dateinamen, Einhaltung des Schemas, Prüfung auf fehlende Segmente, etc.

Zu finden sind diese Prüfungen in der Abfrage `PruefschrittPruefung`. Über die Spalte `einrichtung` können sie je nach Einrichtung (KK, DAS-KK, VST und BAS) gefiltert werden. Bestandteil der Abfrage sind auch die Fehlermeldungen, die als Ergebnis einer Prüfung mit negativem Ergebnis angezeigt werden.

Die folgende Tabelle beschreibt die Struktur der Abfrage `PruefschrittPruefung`. Die Fehlermeldungen und weitere Informationen im Falle eines negativen Prüfergebnisses werden nicht durch die Krankenkasse, sondern durch die Daten entgegennehmende Stelle in das Datenflussprotokoll eingetragen. Die Krankenkasse kann jedoch durch eigene Prüfungen sicherstellen, dass keine fehlerhaften Dokumente exportiert werden.

*Tabelle 22: Struktur der Abfrage `PruefschrittPruefung`*

| Datenfeld                   | Bedeutung   | Anwendungsrelevanz                                     |
|-----------------------------|---|--|
| <code>prozessschritt</code> | Ordnet die Prüfung in den zeitlichen Kontext des Prozessschrittes ein | Implementierung der Prüfprozesse in einem Datenservice |
| <code>pruefung</code>       | Beschreibung der Prüfung  | Datenservice   |

| Datenfeld              | Bedeutung   | Anwendungsrelevanz                |
|------------------------|---|-----------------------------------|
| einrichtung            | Einrichtung, die die Prüfung durchzuführen hat  | Datenservice und Anwender         |
| pruefbereich           | Kategorisierung der Prüfung nach: Spezifikation, Transaktion, Dechiffrierung, Schema, PID_Pseudonym, Sonstige Prüfung   | Rückprotokollierung               |
| prozessschrittSort     | Legt die Reihenfolge der Prozessschritte fest   | Datenservice                      |
| pruefungSort           | Stellt die Prüfung innerhalb des Prozessschrittes in zeitliche Reihenfolge  | Datenservice                      |
| fehlermeldungId        | Id der Fehlermeldung  | Rückprotokollierung               |
| fehlermeldung          | Konkrete Fehlermeldung  | Rückprotokollierung               |
| fehlermeldungParameter | Fehlermeldungen sind parametrisierbar. Parameter befinden sich in <spitzen Klammern> innerhalb der Fehlermeldung. Im Feld fehlermeldungParameter werden alle gültigen Parameter aufgelistet und erläutert. Beispiel:<br><br><params details="Technische Details" /> beschreibt den Parameter „details“ in der Fehlermeldung „Die QS-Datei konnte nicht entschlüsselt werden: <details>“ | Rückprotokollierung               |
| strenge                | Strenge einer negativ verlaufenen Prüfung: Mögliche Werte sind „hart“ und „weich“   | Rückprotokollierung               |
| verursacher_wenn_vst   | Gibt den Verursacher einer negativ verlaufenden Prüfung an, wenn die prüfende Einrichtung die VST ist und der Verursacher des Fehlers eindeutig feststeht   | Datenservice, Rückprotokollierung |
| verursacher_wenn_bas   | Gibt den Verursacher einer negativ verlaufenden Prüfung an, wenn die prüfende Einrichtung die BAS ist und der Verursacher des Fehlers eindeutig feststeht   | Datenservice, Rückprotokollierung |
| verursacher_wenn_das   | Gibt den Verursacher einer negativ verlaufenden Prüfung an, wenn die prüfende Einrichtung die DAS-KK ist und der Verursacher des Fehlers eindeutig feststeht  | Datenservice, Rückprotokollierung |

Durch die Spalten `prozessschrittSort` und `pruefungSort` ergibt sich eine komplett geordnete Reihenfolge der Prüfungen innerhalb einer Einrichtung.

Eine Filterung der Abfrage in der Spalte `einrichtung` nach „KK“ ergibt alle für die Krankenkasse vor dem Export durchzuführenden Prüfungen. Schlägt hierbei eine Prüfung mit der Strenge „hart“ fehl, ist die Ursache dafür zu beheben und der Export neu durchzuführen.

Für die Krankenkassen ergeben sich folgende administrative Prüfungen:

Tabelle 23: Administrative Prüfungen für die Krankenkassen

| Prozess-Schritt                  | Prüfung  | Strenge | Bedeutung  |
|----------------------------------|--|---------|--|
| Vor Erstellung der XML-Dokumente | Struktur- und Formatprüfungen  | hart    | Die Strukturprüfung wird in der Regel durch den Prüfschritt „Schemavalidierung“ durchgeführt. Sollten die Daten bei einer Krankenkasse noch nicht als XML vorliegen, sind die Feldprüfungen anderweitig durchzuführen (siehe Abschnitt 2.3.2).   |
| Formale Dateiprüfungen           | Wurde die Dateinamenkonvention für die PID-Datei eingehalten?                                    | hart    | Es ist sicherzustellen, dass die PID-Datei nach Konvention benannt ist.  |
| Formale Dateiprüfungen           | Wurde die Dateinamenkonvention für die QS-Datei eingehalten?                                     | hart    | Es ist sicherzustellen, dass die QS-Datei nach Konvention benannt ist.   |
| Formale Dateiprüfungen           | Prüfung auf bereits gesendete Dateien  | hart    | Es ist sicherzustellen, dass Dateien mit der gleichen Benennung nicht mehrfach exportiert werden.  |
| Formale Dateiprüfungen           | Liegt ein komplettes Paar von QS- und PID-Datei vor?   | hart    | Es ist sicherzustellen, dass pro Segment jeweils eine PID- und eine QS-Datei übermittelt werden.   |
| Formale Dateiprüfungen           | Vollständigkeitsprüfung: Sind alle Segmente einer Transaktion vorhanden?                         | hart    | Es ist sicherzustellen, dass alle Segmente einer Transaktion vorhanden sind.   |
| Dokumentprüfungen                | Besitzt die Exportdatei dieselbe Dokumenten-ID (GUID) wie eine bereits zuvor übermittelte Datei? | hart    | Es ist sicherzustellen, dass die Dokumenten-ID bei jeder Export-XML-Datei einmalig ist. Bei PID-Modulen ist zu beachten, dass diese ID erst nach Zusammenführung von QS- und PID-Daten für eine Datei eindeutig ist. Solange im Datenfluss beide Dateien separiert vorliegen, ist die Dokumenten-ID als Segment-ID zu verstehen. |

| Prozess-Schritt           | Prüfung  | Strenge | Bedeutung   |
|---------------------------|--|---------|---|
| Dokument-<br>prüfungen    | Liegt das Erstelldatum nach dem Modifikationsdatum?  | weich   | Es ist darauf zu achten, dass das Modifikationsdatum (root/header@modification_dttm) nicht größer als das Erstelldatum ist (root/header@origin_dttm)  |
| Dokument-<br>prüfungen    | Sind Datensätze gleicher ID mehrfach in der Datei angegeben?   | hart    | Es ist sicherzustellen, dass jeder Datensatz innerhalb eines Segments mit einer eindeutigen ID gekennzeichnet ist.  |
| Dokument-<br>prüfungen    | Wird ein Datensatz mit einer ID übermittelt, der in einem anderen Segment dieser Transaktion bereits exportiert wurde? | hart    | Es ist sicherzustellen, dass jeder Datensatz innerhalb einer Transaktion (die mehrere Segmente enthalten kann) durch seine ID eindeutig ist.  |
| Plausibilitäts-<br>regeln | Wohlgeformtheit  | hart    | Es ist sicherzustellen, dass jedes XML-Dokument wohlgeformt ist.  |
| Plausibilitäts-<br>regeln | Schemavalidierung  | hart    | Es ist sicherzustellen, dass jedes XML-Dokument schemavalide nach dem Schema XML-Schema-SDAT_<versionsnummer>/Interface_KK/Interface_KK_QS.xsd bzw. XML-Schema-SDAT_<versionsnummer>/Interface_KK/Interface_KK_PID.xsd ist. |
| Plausibilitäts-<br>regeln | Prüfung der Plausibilitätsregeln   | -       | Durchführung der Plausibilitätsprüfungen. Jede Regel definiert ihre eigene Strenge.   |

Beispielhaft folgen in Tabelle 24 administrative Prüfungen für die Krankenkasse und Konsequenzen, falls diese vor dem Export fehlschlagen.

Tabelle 24: Beispiele administrativer Prüfungen vor dem Export

| Prüfung  | Strenge | Konsequenz   |
|--|---------|--|
| Sind Datensätze gleicher ID mehrfach in der Datei angegeben?   | hart    | Export abbrechen, Ursache für den Fehler beheben und Export neu durchführen.   |
| Wird ein Datensatz mit einer ID übermittelt, der in einem anderen Segment dieser Lieferung bereits exportiert wurde? | hart    | Export abbrechen, überprüfen, ob der Datensatz bereits übermittelt wurde oder eine falsche Kennzeichnung (ID) für einen noch nicht exportierten Datensatz verwendet wird und ggf. neu exportieren. |



Bekommt die Krankenkasse ein Rückprotokoll für bereits exportierte Daten, ist zu überprüfen, ob dieses Fehler auf Datensatz- oder Dokumentenebene auflistet und ob die Krankenkasse dafür verantwortlich ist (siehe Abschnitt 2.5).



#### Hinweis

Bei den Prüfungen mit der id 32 und 33 zu der `versichertenzahl` (siehe Abschnitt 2.2.3) handelt es sich um verfahrensspezifische Prüfungen. Es werden nur die Module überprüft, die einem QS-Verfahren nach DeQS-RL entsprechen. Die Prüfungen unterscheiden sich nach der jeweiligen Stelle im Datenfluss.

Die DAS-KK müssen anhand der Prüfung 32 (harte Prüfung) feststellen können, ob das Attribut `versichertenzahl` im `<body>` angegeben ist. Die BAS prüft anhand der Prüfung 33, ob die `versichertenzahl` im `<data_provider>` übermittelt wurde.

Welche Module der Prüfung unterliegen und nach DeQS-RL spezifiziert werden, ist der Tabelle `Modul` aus der Spezifikationsdatenbank zu entnehmen (Modul.richtlinie DeQS-RL).

Die `versichertenzahl` wird nicht für das Modul AUFST übermittelt.

## 2.4 Verschlüsselung und Übermittlung an die DAS-KK

Die Übertragung von Sozialdaten zur Qualitätssicherung und zur Methodenbewertung enthält schützenswerte Datenbestände, welche nur für den berechtigten Empfänger bzw. die auswertende Stelle lesbar sein dürfen. Zu diesem Zweck sind, unter Berücksichtigung vorhandener Infrastrukturen, geeignete sichere Übertragungswege zu wählen und Verschlüsselungen einzusetzen. Bei der Dateiverschlüsselung und -übermittlung sind im Folgenden genannte Standards im Bereich der gesetzlichen Krankenversicherung bereits etabliert, auf die im Rahmen der Spezifikation für Sozialdaten zurückgegriffen wird.

### 2.4.1 Vorgaben für die Dateiverschlüsselung und Übermittlung

#### Krankenkassenkommunikationssystem (KKS)

Zur Datenübertragung zwischen Krankenkassen und der Datenannahmestelle wird das Krankenkassenkommunikationssystem (KKS)<sup>7</sup> verwendet, ein bei allen gesetzlichen Krankenkassen vorhandener Standard, der die gesetzlichen Vorgaben zur Sicherheit und Datenschutz erfüllt.

Die Kodierung der Datenübertragungen vom KKS (ausgehende PID-Dateien, QS-Dateien) und zum KKS (eingehende Empfangsprotokolle DAS-KK und Rückprotokolle BAS) erfolgt in ISO 8859-1.

#### Security-Schnittstelle (SECON)

Die für die Verwendung im KKS vorgeschriebene Security-Schnittstelle (SECON)<sup>8</sup> sieht, anders als die bislang von der BAS spezifizierte Verschlüsselung, ausschließlich die Verschlüsselung vollständiger Dateien – nicht einzelner Bereiche der Dateien – nach dem Standard PKCS#7 vor.

<sup>7</sup> Siehe <https://www.gkv-datenaustausch.de> (Stand: 06.06.2023, abgerufen: 06.06.2023).

<sup>8</sup> Siehe <https://www.gkv-datenaustausch.de> (Stand: 06.06.2023, abgerufen: 06.06.2023).

## 2.4.2 Konsequenzen für den Datenfluss für Sozialdaten

Aus den Möglichkeiten der in der Security-Schnittstelle genutzten Verschlüsselungsverfahren und den der sektorenübergreifenden QS bzw. Methodenbewertung zugrunde liegenden Richtlinien ergibt sich die Notwendigkeit, die patientenidentifizierenden Daten in einer separaten Datei zu den QS-Daten zu versenden. Daraus resultiert, dass Datenlieferungen, welche patientenidentifizierende Daten enthalten, stets aus zwei Dateien bestehen:

Tabelle 25: Teildateien der Datenlieferungen

| Datei     | verarbeitende Stelle | Inhalt   |
|-----------|----------------------|--|
| PID-Datei | VST                  | <ul style="list-style-type: none"> <li>Admin-Daten</li> <li>patientenidentifizierende Daten</li> </ul>   |
| QS-Datei  | DAS-KK               | <ul style="list-style-type: none"> <li>Admin-Daten</li> <li>leistungserbringeridentifizierende Daten</li> <li>krankenkassenidentifizierende Daten</li> <li>QS-Daten</li> </ul> |

Die beiden Dateien müssen durch entsprechende Angaben innerhalb der Admin-Daten einander eindeutig zugeordnet werden können. Hierfür wird die Verwendung einer gemeinsamen Dokumenten-ID (Attribut `id` im Element `header`) für zwei zueinander gehörende PID- und QS-Dateien empfohlen, welche in diesem Kontext als eindeutige Segmentkennung dient. Weiterhin ist sicherzustellen, dass die in der PID-Datei enthaltenen patientenidentifizierenden Daten (bzw. nach Bearbeitung durch die VST die Patientenpseudonyme) eindeutig den in der QS-Datei enthaltenen Fällen zugeordnet werden können. Hierfür müssen in zwei zueinander gehörenden PID- und QS-Dateien identische Nummerierungen der Datensätze (Attribut `lfdrnr` im Element `case`) hinterlegt sein. Da die DAS-KK die Admin-Daten der verschlüsselten PID-Datei nicht einsehen kann, muss mit den Mitteln des KKS ein eindeutiger Bezug der QS-Datei zur jeweiligen PID-Datei hergestellt werden. Hierfür ist das Feld `DATEINAME` des Auftragsatzes der KKS-Übertragung zu verwenden.

Einhergehend mit dem jeweiligen seriellen Datenfluss-Modell für Verfahren mit patientenidentifizierenden Daten (Abbildung 13) und ohne patientenidentifizierende Daten (Abbildung 14), werden Maßnahmen zur Verschlüsselung der Datendateien und des Datentransports getroffen.

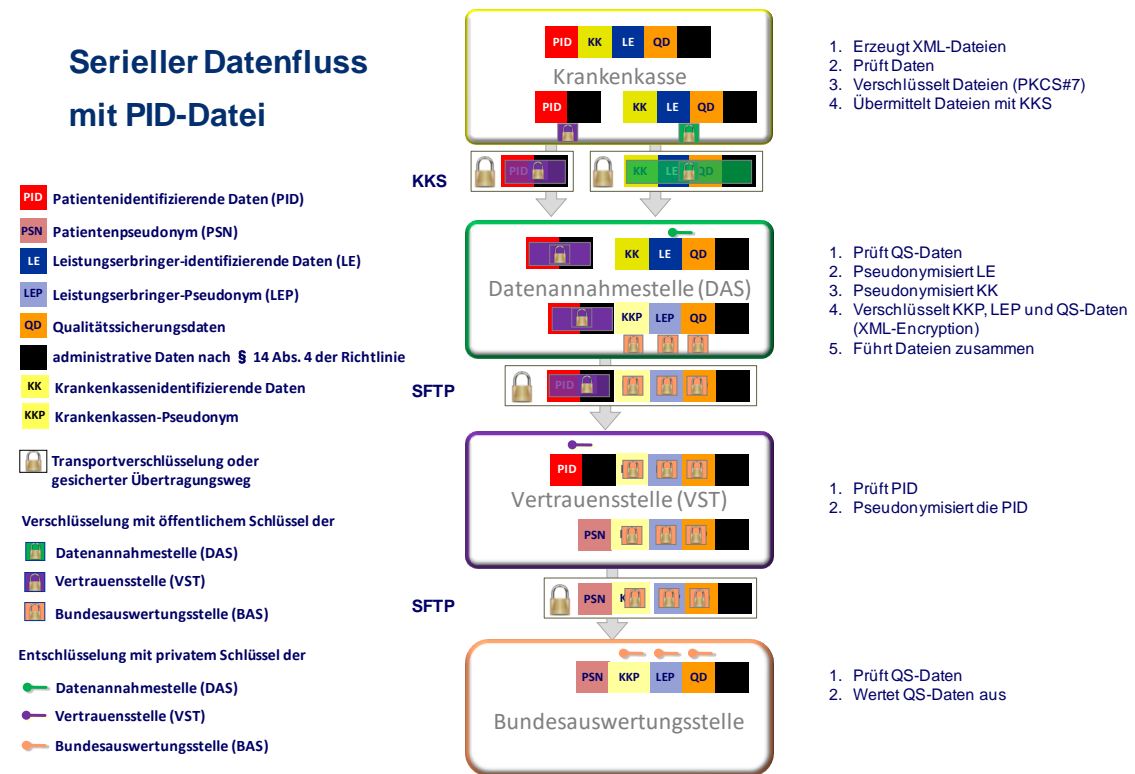


Abbildung 13: Serieller Datenfluss mit PID gemäß DeQS-RL

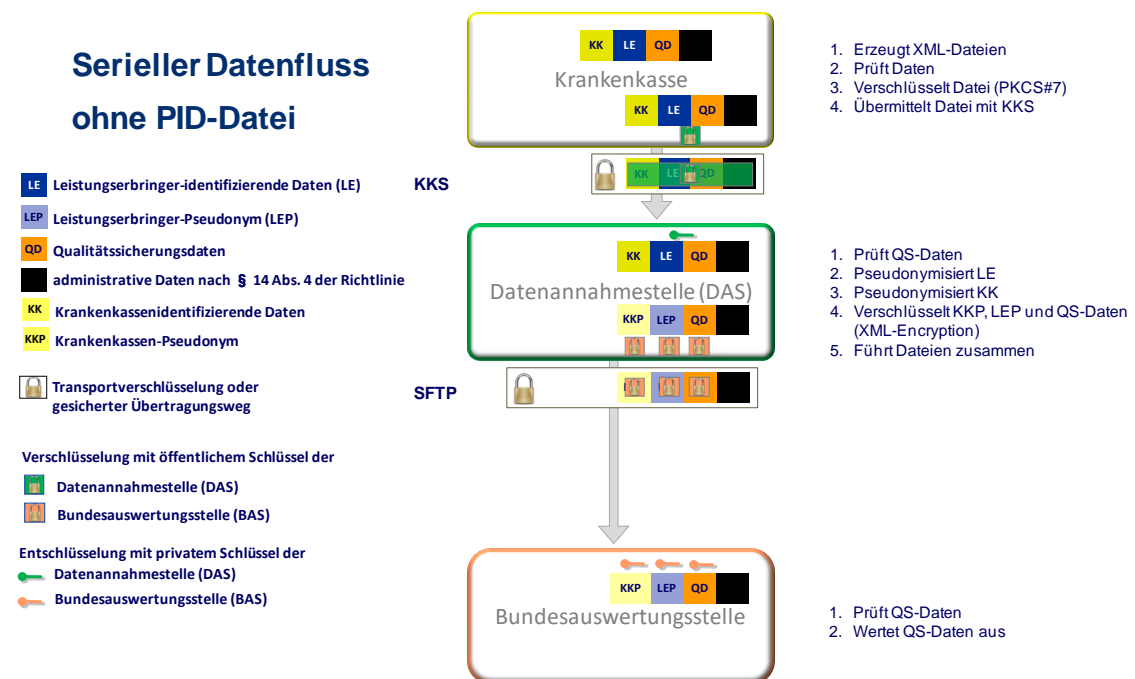


Abbildung 14: Serieller Datenfluss ohne PID gemäß DeQS-RL

### 2.4.3 Dateiverschlüsselung

Die PID-Datei enthält die patientenidentifizierenden Daten und wird von den Krankenkassen vor dem Versand mit dem öffentlichen Schlüssel der Vertrauensstelle (VST) verschlüsselt. Die QS-Datei für Verfahren enthält die krankenkassen- und die leistungserbringeridentifizierenden Daten sowie die QS-Daten und wird zunächst von den Krankenkassen mit dem öffentlichen Schlüssel der Datenannahmestelle der Krankenkassen (DAS-KK) verschlüsselt. Die QS-Datei für die Verfahren der Methodenbewertung enthält im Vergleich zu den Verfahren keine leistungserbringeridentifizierenden Daten. Die beiden Dateien sind also für unterschiedliche Empfänger bestimmt und können nur von diesen eingesehen und bearbeitet werden: Die DAS-KK ist in der Lage, die QS-Datei zu entschlüsseln und die erforderlichen Prüfungen vorzunehmen, die PID-Datei kann nur von der VST entschlüsselt werden.

Tabelle 26: Verschlüsselung der Exportdateien

| Datei     | Zu verwendender Schlüssel                              |
|-----------|--|
| PID-Datei | Öffentlicher Schlüssel der Vertrauensstelle (VST)      |
| QS-Datei  | Öffentlicher Schlüssel der Datenannahmestelle (DAS-KK) |

Die QS-Dateien sind gemäß der Security-Schnittstelle für den Datenaustausch im Gesundheits- und Sozialwesen (SECON) nach dem Standard PKCS#7 zu verschlüsseln und zu signieren.

### 2.4.4 Übermittlung

#### Übertragungsweg

Zur Übermittlung der verschlüsselten Dateien an die DAS-KK wird das KKS genutzt. Dazu stellt die Krankenkasse eine SFTP-Verbindung zum Server der DAS-KK her. Der Verbindungsaufbau erfolgt mittels Benutzername und Passwort bzw. Schlüsseln in der Public-Key-Infrastruktur (PKI) der Vertrauensstelle.

#### Physische Dateinamen

Gemäß den Richtlinien des KKS wird der 8-stellige Transferdateiname für die Übertragung nach folgendem Schema gebildet:

<[ E/T ]><XXXX><999>

Erläuterung der Dateinamensbausteine:

<[ E/T ]>: Echtdaten (E) oder Testdaten (T)

<XXXX>: 4-stellige Verfahrenskennung inkl. Versionsnummer (hier immer „SQS0“)

<999>: 3-stellige laufende Nummer

Der Name der zugehörigen Auftragssatzdatei wird aus dem Dateinamen der Transferdatei und dem Zusatz „.AUF“ gebildet.

### Logische Dateinamen

Damit die DAS-KK die jeweils zusammengehörenden PID- und QS-Dateien zusammenführen kann, ist eine eindeutige Kennung erforderlich. Die in der XML-Struktur vorgesehene Dokument-ID kann hierfür nicht genutzt werden, da die PID-Datei für die DAS-KK nicht einsehbar ist. Stattdessen wird das 11-stellige Feld `DATEINAME` des gemäß KKS zu übertragenden Auftragssatzes für die Zuordnung verwendet.

Dabei gilt das folgende Schema:

`<XXXXXX><[P/Q/R/A]><99999>`

Erläuterung der Dateinamensbausteine:

`<XXXXXX>` : 5-stellige Modulbezeichnung, fehlende Stellen werden rechts mit Unterstrichen („\_“) aufgefüllt (z. B. „PCI\_\_“)

`<[P/Q/R/A]>` : PID-Datei (P) oder  
 QS-Datei (Q) bei PID-Verfahren bzw.  
 QS-Datei (R) für Nicht-PID-Verfahren und Nullmeldungen, bei denen keine zugehörige PID-Datei übermittelt wird sowie  
 Aufstellung (A)

`<99999>` : 5-stellige laufende Nummer mit führenden Nullen

Die beiden zusammengehörigen Dateien bei PID-Verfahren erhalten die gleiche Modulbezeichnung und die gleiche laufende Nummer. Sie unterscheiden sich lediglich durch die Kennung der Dateiarart.

---

#### Beispiel:

PID-Datei: PCI\_\_P00001

QS-Datei: PCI\_\_Q00001

---

### Unterscheidung von Test- und Echtdaten

In den administrativen Daten der XML-Dateien wird zwischen drei Datenzielen (`data_target`) unterschieden: Testdatenpool, Probedatenpool und Echtdatenpool (siehe Abschnitt 6.4). Im Feld `VERFAHREN_KENNUNG` des Auftragssatzes wird nur zwischen Echtdaten (E) und Testdaten (T) unterschieden. Hier ist eine der Art der Daten entsprechende Kennung zu nutzen.

Tabelle 27: Verfahrenskennung im Auftragssatz

| <code>data_target</code> | Art der Daten | <code>VERFAHREN_KENNUNG</code> |
|--------------------------|---------------|--------------------------------|
| Echtdatenpool            | Echtdaten     | ESQS0                          |
| Probedatenpool           | Echtdaten     | ESQS0                          |
| Testdatenpool            | Testdaten     | TSQS0                          |

**Kennzeichnung der Datenart**

Zusätzlich zur Angabe im logischen Dateinamen kann die Datenart auch im Feld VERFAHREN\_KENNUNG\_SPEZIFIKATION des Auftragsatzes gekennzeichnet werden. Dabei wird die gleiche Kodierung wie im logischen Dateinamen verwendet: PID-Datei (P), QS-Datei (Q) bei PID-Verfahren bzw. QS-Datei (R) für Nicht-PID-Verfahren und Nullmeldungen sowie Aufstellungen (A). Diese Angabe ist optional. Alternativ kann das Feld mit Leerzeichen gefüllt sein.

**Komprimierung**

Die Dateien können komprimiert an die DAS-KK übermittelt werden. Als Komprimierungsverfahren ist das Produkt „gzip“ zulässig. Der fakultative Einsatz einer Komprimierung einer Transferdatei muss im zugehörigen Auftragsatz im Feld KOMPRIMIERUNG angegeben werden. Die zulässigen Werte lauten:

- 00: keine Komprimierung
- 02: gzip

**Hinweis**

Im Rahmen der Rückübermittlung von Datenflussprotokollen ist die Anwendung einer Komprimierung obligatorisch, vgl. Abschnitt Komprimierung unter Absatz 2.5.2.

---

**Segmentierung**

Eine Segmentierung von Datenlieferungen (mit mehr als 10.000 Datensätzen für Verfahren gemäß DeQS-RL bzw. 50.000 Datensätze für Verfahren der Methodenbewertung) findet erst im Rahmen der Zusammenführung zu gemeinsamen Transaktionen in der BAS Beachtung. Die DAS-KK bzw. die VST gewährleisten keine transaktions-, sondern lediglich eine segmentbezogene Verarbeitung. Insofern wird empfohlen im Auftragsatz sämtlicher Datenlieferungen im Feld SEQUENZ-NR stets den Wert „000“ anzugeben, selbst wenn – entgegen der KKS-Spezifikation – eine Nachricht segmentiert von der KK übermittelt wird.

**2.5 Rückprotokollierung****2.5.1 Empfangsbestätigung der DAS-KK**

Nach erfolgreicher Eingangsverarbeitung durch die DAS-KK erhält die Krankenkasse von der DAS-KK eine Empfangsbestätigung, die den Erhalt eines Segmentes einer Transaktion bestätigt. Ist eine Datenlieferung nicht bearbeitbar, wird anstelle einer Empfangsbestätigung ein Datenflussprotokoll an die Krankenkasse übermittelt.

Die genaue Struktur und ein Beispiel ist Abschnitt 6.6 zu entnehmen.

**Logischer Dateiname**

Analog zur Übertragung der Daten von den Krankenkassen zur DAS-KK wird im Feld DATEINAME des Auftragsatzes die Empfangsbestätigung kenntlich gemacht.

Dabei gilt das folgende Schema:

```
<XXXXXX><[ B ]><99999>
```

Erläuterung der Dateinamensbausteine:

- <XXXXXX> : 5-stellige Modulbezeichnung, fehlende Stellen werden rechts mit Unterstrichen („\_“) aufgefüllt (z. B. „PCI\_\_“)
- <[ B ]> : Bestätigung des vollständigen Eingangs der Datenlieferung
- <99999> : 5-stellige laufende Nummer mit führenden Nullen

### Kennzeichnung der Datenart

Zusätzlich zur Angabe im logischen Dateinamen wird die Datenart auch im Feld VERFAHREN\_KENNUNG\_SPEZIFIKATION des Auftragssatzes gekennzeichnet. Dabei wird die gleiche Kodierung wie im logischen Dateinamen verwendet: Bestätigung (B). Diese Angabe ist nicht optional.

### 2.5.2 Datenflussprotokoll

Ein Datenflussprotokoll (DFP) wird durch folgende Stellen und in folgenden Fällen erstellt und an die Krankenkasse übermittelt:

- DAS-KK: Wenn eine Weiterverarbeitung aufgrund von Fehlern nicht möglich ist (der Dokumentenstatus wurde auf „ERROR“ gesetzt). Diese DFP werden segmentbezogen unter Verwendung des Header-Attributs `id` erstellt.
- BAS: Wenn die BAS die Eingangsdatei schemakonform von der VST erhält. In diesem Fall enthält das Protokoll die Prüfergebnisse aller Prüfinstanzen (DAS-KK, VST, BAS). Diese DFP werden transaktionsbezogen unter Verwendung des Header-Attributs `set_id` erstellt.

Dem Datenflussprotokoll kann anhand des Attributs `V` des Elements `<status_document>` entnommen werden, ob die Übermittlung der Transaktion (BAS) bzw. eines Segments (DAS-KK) erfolgreich war:

Tabelle 28: Status des Dokuments

| Ergebnis | Bedeutung  | Konsequenz   |
|----------|--|--|
| OK       | Transaktion erfolgreich:<br>Keine Auffälligkeiten  | Keine  |
| WARNING  | Transaktion erfolgreich:<br>Es sind Verstöße gegen weiche, aber nicht gegen harte Prüfungen aufgetreten.<br>.                      | Abhängig von den aufgetretenen Fehlern ist von der KK zu entscheiden, ob die Lieferung nach Beheben der Fehler erneut durchzuführen ist. |
| ERROR    | Transaktion nicht erfolgreich:<br>Es sind Verstöße gegen harte Prüfungen aufgetreten, die die Weiterverarbeitung verhindert haben. | Die Fehler sind zu beheben und die Datensätze sind anschließend erneut zu übermitteln.   |

Wenn der Dokumentenstatus „WARNING“ oder „ERROR“ ist, bedeutet dies, dass entweder auf Headerebene oder auf Datensatzebene Fehler aufgetreten sind. Analog zum Status des Dokuments werden die Prüfergebnisse auf Datensatzebene im Element `<status_case>` festgehalten:

Tabelle 29: Status des Datensatzes

| Ergebnis | Bedeutung  | Konsequenz  |
|----------|--|---|
| OK       | Datensatz angenommen:<br>Keine Auffälligkeiten   | Keine   |
| WARNING  | Datensatz angenommen:<br>Es sind weiche Fehler aufgetreten.  | Abhängig von den aufgetretenen Fehlern ist von der KK zu entscheiden, ob die Lieferung nach Beheben der Fehler erneut durchzuführen ist.  |
| ERROR    | Datensatz abgelehnt:<br>Es sind harte Fehler aufgetreten, die eine Aufnahme in den Bundesdatenpool verhindert haben. | Sofern der Status des Dokuments „WARNING“ ist, ist in Abhängigkeit von den aufgetretenen Fehlern durch die KK zu entscheiden, ob die Lieferung nach Beheben der Fehler erneut durchzuführen ist. Bei Dokumentstatus „ERROR“ muss der Datensatz neu exportiert werden. |

Ist ein Fehler aufgetreten, ist zu überprüfen, ob die Krankenkasse dafür verantwortlich war. Den Hinweis hierzu gibt das optionale Attribut `originator` im `<error>`-Tag, falls dieses gesetzt ist und den Wert „KK“ enthält. Da dieses Attribut jedoch nur gesetzt werden darf, wenn ein Verursacher eindeutig feststeht, ist auch bei Fehlern ohne `originator`-Attribut zu überprüfen, ob der Fehler durch die Krankenkasse verursacht wurde.

#### Beispiel:

Ein Rückprotokoll enthält einen Fehler auf Dokumentenebene mit der ID 100010 und der Fehlermeldung „Der Datensatz für den Patienten mit der laufenden Nummer wurde bereits in dieser Datei übermittelt.“

Der Abfrage `PruefschrittPruefung` (siehe Abschnitt 2.3) ist zu entnehmen, dass dieser Fehler Folge des Scheiterns der Prüfung „Sind Datensätze gleicher IDs mehrfach in der Datei angegeben?“ ist und die Krankenkasse diesen Fehler durch eine eigene Prüfung vor dem Export hätte verhindern können:



| prozessschritt           | prüfung  | fehler | fehler   |
|--------------------------|--|--------|--|
| Vor Erstellung der XML-D | Struktur- und Formatprüfungen                            | KK     |  |
| Formale Dateiprüfungen   | Wurde die Dateinamenkonvention für die PID-Datei ein     | KK     | 100004 Die PID-Datei <dateiname> wurde falsch benannt.                     |
| Formale Dateiprüfungen   | Wurde die Dateinamenkonvention für die QS-Datei ein      | KK     | 100005 Die QS-Datei <dateiname> wurde falsch benannt.                      |
| Formale Dateiprüfungen   | Prüfung auf bereits gesendete Dateien                    | KK     | 100006 Die Datei <dateiname> wurde bereits exportiert.                     |
| Formale Dateiprüfungen   | Liegt ein komplettes Paar von QS- und PID-Datei vor?     | KK     | 100014 Zu der Datei <dateiname> fehlt die Datei <dateinamePendant>.        |
| Formale Dateiprüfungen   | Vollständigkeitsprüfung: Sind alle Segmente einer Tran   | KK     | 100013 Die Transaktion <transaktionsId> wurde nicht vollständig abgeschlos |
| Dokumentprüfungen        | Besitzt die Exportdatei dieselbe Dokumenten-ID (GUID)    | KK     | 100007 Die übermittelte GUID (<guid>) wurde bereits verwendet.             |
| Dokumentprüfungen        | Liegt das Erstelldatum nach dem Modifikationsdatum?      | KK     | 100009 Das Erstelldatum (<erstellDatum>) liegt nach dem Modifikationsdatu  |
| Dokumentprüfungen        | Sind Datensätze gleicher ID mehrfach in der Datei ange   | KK     | 100010 Der Datensatz für den Patienten mit der laufenden Nummer <ldnr>     |
| Dokumentprüfungen        | Wird ein Datensatz mit einer ID übermittelt, der in eine | KK     | 100011 Der Datensatz für den Patienten mit der laufenden Nummer <ldnr>     |
| Vor Übermittlung         | Wohlgeformtheit  | KK     | 100000 Die XML-Datei ist nicht wohlgeformt.                                |
| Vor Übermittlung         | Schemavalidierung  | KK     | 100001 Die XML-Datei ist nicht schemakonform.                              |
| Vor Übermittlung         | Prüfung der Plausibilitätsregeln                         | KK     |  |

Abbildung 15: Mappen einer Fehlermeldung eines Rückprotokolls auf eine Prüfung

Einzelheiten zur Rückprotokollierung können dem Abschnitt 6.6, die XML-Struktur dem Abschnitt 6.4 entnommen werden.

### Logischer Dateiname

Der Status des Dokumentes wird auch im Feld „DATEINAME“ des Auftragssatzes angegeben:

Dabei gilt das folgende Schema:

<XXXXXX><[O/W/E]><99999>

Erläuterung der Dateinamensbausteine:

<XXXXXX>: 5-stellige Modulbezeichnung, fehlende Stellen werden rechts mit Unterstrichen („\_“) aufgefüllt (z. B. „PCI\_\_“)

<[O/W/E]>: Dokumentstatus OK (O), WARNING (W) oder ERROR (E)

<99999>: 5-stellige laufende Nummer mit führenden Nullen

### Kennzeichnung der Datenart

Zusätzlich zur Angabe im logischen Dateinamen wird der Status des Dokuments auch im Feld VERFAHREN\_KENNUNG\_SPEZIFIKATION des Auftragssatzes gekennzeichnet. Dabei wird die gleiche Kodierung wie im logischen Dateinamen verwendet: OK (O), WARNING (W) oder ERROR (E). Diese Angabe ist nicht optional.

### Komprimierung

Die Dateien werden komprimiert von der DAS-KK übermittelt. Als Komprimierungsverfahren ist das Produkt „gzip“ zulässig. Der obligatorische Einsatz einer Komprimierung einer Transferdatei, welche das Datenflussprotokoll beinhaltet, muss im zugehörigen Auftragssatz im Feld KOMPRIMIERUNG angegeben werden. Der zulässige Wert lautet:

- 02: gzip

## 2.6 Bestätigung und Aufstellung gemäß DeQS-RL

Einmal jährlich bestätigen die Krankenkassen gegenüber der DAS-KK und der BAS, dass die erforderlichen Daten nach Vorgabe der DeQS-RL und der themenspezifischen Bestimmungen übermittelt worden sind. Diese Anforderung ergibt sich aus § 16 Abs. 5 DeQS-RL, Teil 1 (Rahmenbestimmungen) und jeweils § 16 der themenspezifischen Bestimmungen.

§ 16 Abs. 5 DeQS-RL, Teil 1 differenziert zwischen Bestätigung und Aufstellung.

Bestätigung und Aufstellung zu einem Jahr X umfassen alle Lieferungen, die vom 01.01.X bis zum 31.12.X erfolgt sind. Sie können frühestens nach Abschluss der letzten erfolgreich durchgeführten Transaktion im 4. Quartal des Jahres X erstellt und übermittelt werden. Spätestens bis zum 31.10.(X+1) sind jedoch Aufstellung und Bestätigung zu Jahr X an die DAS-KK zu liefern. Beispielhaft ist in Abbildung 16 dargestellt, welche Lieferungen für das Modul PCI in der Bestätigung und Aufstellung zu dem Jahr 2018 aufzuführen sind. Diese müssen in diesem Beispiel bis zum 31.10.2019 übermittelt werden.

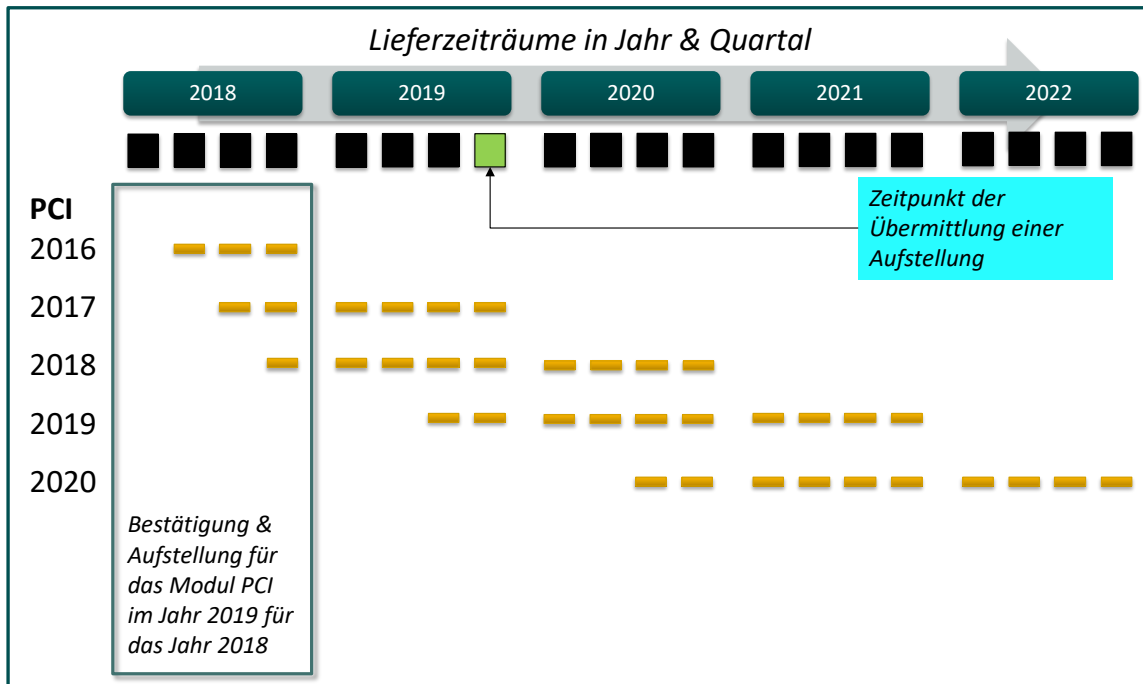


Abbildung 16: Bestätigung & Aufstellung für den Zeitraum 1. Jan bis 31. Dez 2018 (mehrere Module können in einer Bestätigung und einer Aufstellung aufgeführt sein)

Eine Aufstellung zeigt, wie viele Datensätze für jeweils ein Modul und Erfassungsjahr in einem Jahr übermittelt wurden. Dabei kann eine Aufstellung mehrere Module enthalten. Die Aufstellung wird in elektronischer Form analog zum Datenfluss von QS-Daten im XML-Format an die DAS-KK übermittelt.

Eine Aufstellung im XML-Format erfolgt grundsätzlich entsprechend der Beschreibung in Abschnitt 6.4.3 XML-Struktur des Datencontainers (KK/DAS-KK-Schnittstelle).

Im Element `header` wird diese durch den Wert `Aufstellung` im Attribut `data_flow` kenntlich gemacht. Die Attribute `validity_start_date` und `validity_end_date` im Element `/root/header` definieren das Jahr der abgeschlossenen Lieferquartale. Das Attribut `total_count` wird in der Aufstellung mit dem Wert 1 belegt. Das Element `header` ist exemplarisch in Abbildung 17 gezeigt.

```
<header id="{0b9f1a6d-6f68-45a3-93a1-8f03ae38e69e}"
  set_id="{0b9f1a6d-6f68-45a3-93a1-8f03ae38e69e}"
  origination_dttm="2019-10-15T14:22:31"
  modification_dttm="2019-10-15T14:22:31"
  validity_start_date="2018-01-01"
  validity_end_date="2018-12-31"
  data_target="Testdatenpool"
  data_flow="Aufstellung"
  specification="2019_Qesue_SozDat_KK_RB_XML_V02"
  total_count="1" />
```

Abbildung 17: Element header mit Aufstellung im Attribut /root/header/@data\_flow

Grundsätzlich erfolgt der Aufbau des Elementes body analog zu einer Transaktion von QS-Daten. Eine detaillierte Erklärung zur XML-Struktur einer Aufstellung ist zu finden unter der Zwischenüberschrift body/case\_qs\_data/AUFST im Abschnitt 6.4.4 XML-Struktur der QS-Daten. Exemplarisch zeigt Abbildung 18 Daten für eine Aufstellung.

```
<body kasseiknr="103456789" module="AUFST">
  <case lfdnr="1">
    <qs_data>
      <AUFST>
        <modulkennnung V="PCI">
          <erfassungsjahr V="2016">
            <lieferung V="22018" Anzahl="13531"/>
            <lieferung V="32018" Anzahl="-2"/>
            <lieferung V="42018" Anzahl="13554"/>
          </erfassungsjahr>
          <erfassungsjahr V="2017">
            <lieferung V="32018" Anzahl="9345"/>
            <lieferung V="42018" Anzahl="10959"/>
          </erfassungsjahr>
          <erfassungsjahr V="2018">
            <lieferung V="42018" Anzahl="10959"/>
          </erfassungsjahr>
        </modulkennnung>
        <modulkennnung V="NWITR">
          <erfassungsjahr V="2017"> [5 lines]
          <erfassungsjahr V="2018">
            <lieferung V="12018" Anzahl="-1"/>
            <lieferung V="22018" Anzahl="-1"/>
            <lieferung V="32018" Anzahl="-1"/>
            <lieferung V="42018" Anzahl="23547"/>
          </erfassungsjahr>
        </modulkennnung>
        <modulkennnung V="NWIWI"> [8 lines]
        <bestaetigung>
          <bestaetigungstext V="Hiermit wird bestätigt, dass die
            <vertretungsberechtigteperson V="Nachname, Vorname"/>
          </bestaetigungstext>
        </bestaetigung>
      </AUFST>
    </qs_data>
  </case>
  <statistic>
    <sent count="1"/>
  </statistic>
</body>
```

Abbildung 18: Beispiel des Elementes body für eine Aufstellung

Die Bestätigung, zusammengesetzt aus dem Bestätigungstext sowie dem Namen der vertretungsberechtigten Person, wird gemeinsam mit der Aufstellung an die DAS-KK übermittelt. Analog zu den QS-Daten wird in der DAS-KK das IK der Krankenkasse pseudonymisiert. Aus diesem Grunde erfolgt auch keine Weiterleitung des Namens der vertretungsberechtigten Person an die BAS.

Die Übertragung von Aufstellung und Bestätigung in Form einer Transaktion des Moduls AUFST wird, analog zu versicherten- bzw. fallbezogenen Datenlieferungen, mit einer zugehörigen Empfangsbestätigung und einem Datenflussprotokoll quittiert.

## 2.7 Informationen über die Güte gelieferter Daten

Das IQTIG prüft die übermittelten Daten vor deren Verwendung gemäß § 16 DeQS-RL für die Berechnung von Qualitätsindikatoren auf deren Güte. Prinzipiell besteht die Möglichkeit zur Prüfung der übermittelten Daten vor deren Verwendung für die Programmbeurteilung gem. oKFE-RL gleichermaßen für die GFL. Die Prüfungen erfolgen transaktionsspezifisch je Modul.

Die Ergebnisse dieser Prüfungen werden in Form von Berichten zusammengefasst und krankenkassenspezifisch aufbereitet an die Krankenkassen zurück gespiegelt. Bei der Übermittlung krankenkassenspezifischer Rückmeldeberichte muss berücksichtigt werden, dass in der BAS lediglich pseudonymisierte Institutionskennzeichen der Krankenkassen vorliegen. Insofern sind die Adressaten der Berichte in der BAS nicht bekannt, so dass keine direkte Kontaktaufnahme möglich ist.

Unter Nutzung etablierter Kommunikationskanäle können krankenkassenspezifische Rückmeldeberichte über die DAS-KK von der BAS an die einzelnen KK übermittelt werden. Das hierfür zu verwendende Grundgerüst entspricht jenem von Datenflussprotokollen. Da eine Normierung der Inhalte der krankenkassenspezifischen Rückmeldeberichte nicht sinnvoll ausgestaltet werden kann, werden die vollständigen Berichte im Portable Document Format (PDF) bereitgestellt. Diese werden vollständig base64-kodiert in die Struktur eines Datenflussprotokolls eingebettet und im Rahmen eines gesonderten Prozesses unter Nutzung der Verfahrenskennung [I] (wie Informationen an die Krankenkasse) übertragen. Grundsätzlich ist eine Transaktion beschränkt auf Informationen für ein Modul und Erfassungsjahr. Unter Einhaltung dieser Maßgabe ergibt sich, dass bei der Übertragung von Informationen über die Güte der gelieferten Daten nur jeweils ein Bericht in Form einer PDF-Datei base64-kodiert eingebettet in ein case-Element von der BAS über die DAS-KK an die Krankenkasse übermittelt werden darf. In den Rückmeldeberichten gibt der <protokoll\status\_document> an, ob für die jeweilige Transaktion (Modul-Erfassungsjahr) kein Prüfbedarf vorliegt (OK), bei mindestens einem Prüfkriterium ein Prüfbedarf vorliegt (WARNING) bzw. keine Lieferung vorliegt (ERROR). Im Fall von WARNING und ERROR, sind die Fehler innerhalb von 2 Lieferungen zu beheben. Weitere Informationen zur Übertragungsstruktur (`./interface_KK_DAS/response_DAS_KK.xsd`) können dem Abschnitt 6.4.7 XML-Struktur der Rückprotokollierung (Datenflussprotokoll) (Zwischenüberschrift: Element `case/qs_data/INFO`) entnommen werden. Eine automatisierte Rückantwort seitens der Krankenkassen im Sinne einer technischen Quittung ist nicht vorgesehen. Daher ergeben sich im Zusammenhang mit der Übertragung krankenkassenspezifischer Rückmeldeberichte auf Seiten der Krankenkassen insbesondere folgende zwei Aufgaben:

- Datenentgegennahme
- Extraktion des base64-kodierten Inhaltes sowie Reproduktion der Inhalte als PDF mit anschließender interner Weiterleitung entsprechend der Zuständigkeiten

Für die Übertragung der krankenkassenspezifischen Rückmeldeberichte als Informationen für die Krankenkassen gelten folgenden Festlegungen im Rahmen der Nutzung des KKS.

**Physische Dateinamen**

Gemäß den Richtlinien des KKS wird der 8-stellige Transferdateiname für die Übertragung nach folgendem Schema gebildet:

< [ E / T ] > < XXXX > < 999 >

Erläuterung der Dateinamensbausteine:

< [ E / T ] > :      Echtdaten (E) oder Testdaten (T)

< XXXX > :      4-stellige Verfahrenskennung inkl. Versionsnummer (hier immer „SQS0“)

< 999 > :      3-stellige laufende Nummer

Der Name der zugehörigen Auftragssatzdatei wird aus dem Dateinamen der Transferdatei und dem Zusatz „.AUF“ gebildet.

**Logischer Dateiname**

Analog zur Übertragung der Daten von den Krankenkassen zur DAS-KK wird im Feld DATEINAME des Auftragssatzes die Empfangsbestätigung kenntlich gemacht.

Dabei gilt das folgende Schema:

< XXXXXX > < [ I ] > < 99999 >

Erläuterung der Dateinamensbausteine:

< XXXXXX > :      5-stellige Modulbezeichnung, fehlende Stellen werden rechts mit Unterstrichen („\_“) aufgefüllt (z. B. „PCI\_“, alternativ „INFO\_“ bei modulübergreifenden bzw. –unabhängigen Rückmeldungen)

< [ I ] > :      Informationen über die Güte übermittelter Daten, krankenkassenspezifische Rückmeldeberichte

< 99999 > :      5-stellige laufende Nummer mit führenden Nullen

**Kennzeichnung der Datenart**

Zusätzlich zur Angabe im logischen Dateinamen wird die Datenart auch im Feld VERFAHREN\_KENNUNG\_SPEZIFIKATION des Auftragssatzes gekennzeichnet. Dabei wird die gleiche Kodierung wie im logischen Dateinamen verwendet: Information - krankenkassenspezifischer Rückmeldebericht (I). Diese Angabe ist nicht optional.

**Komprimierung**

Die Dateien werden komprimiert von der DAS-KK übermittelt. Als Komprimierungsverfahren ist das Produkt „gzip“ zulässig. Der obligatorische Einsatz einer Komprimierung einer Transferdatei, welche das Datenflussprotokoll beinhaltet, muss im zugehörigen Auftragssatz im Feld KOMPRIMIERUNG angegeben werden. Der zulässige Wert lautet:

- 02: gzip

Sollten Verarbeitungsprobleme jeglicher Art (beispielsweise Auftragssatz nicht lesbar, Nutzdaten nicht verarbeitbar, Dekodierung von base64 nach PDF nicht erfolgreich, etc.) auftreten, muss

zur Fehleranalyse und Klärung eine telefonische Kontaktaufnahme wahlweise zur Datenannahmestelle oder alternativ zur (Bundes-)Auswertungsstelle (sofern zumindest die Transaktionskennung lesbar war) erfolgen. Die Datenannahmestelle kann insbesondere im Problemfall ihrerseits selbst Kontakt zur (Bundes-)Auswertungsstelle aufnehmen. Eine technische Quittung in Form einer Empfangsbestätigung oder eines Datenflussprotokolls ist für die Datenart [I] nicht vorgesehen. Somit wird die Nutzung des etablierten Kommunikationskanals für die Übertragung von QS-Daten nicht überstrapaziert und gleichzeitig der Prozessaufwand so gering wie möglich gehalten.

### 3 Prozesse bei der Datenannahmestelle

Die folgenden Prozesse der Datenannahmestelle der Krankenkassen (DAS-KK) werden durch die Rahmenvorgaben des G-BA umrissen und im weiteren Verlauf dieses Kapitels näher erläutert:

- Um eine Datenprüfung in der DAS-KK durchführen zu können, müssen die QS-Daten durch die Krankenkasse mit dem öffentlichen Schlüssel der DAS-KK verschlüsselt worden sein.
- Die Krankenkasse muss zudem die patientenidentifizierenden Daten (PID) vor der Übertragung an die DAS-KK mit dem öffentlichen Schlüssel der Vertrauensstelle (VST) verschlüsseln.
- Die DAS-KK muss alle für sie verschlüsselten Daten prüfen.
- Die DAS-KK muss die Pseudonymisierung der leistungserbringeridentifizierenden Daten mit den DAS der bestehenden Datenflüsse nach der DeQS-RL so abstimmen, dass die BAS die Pseudonyme der Leistungserbringer zusammenführen kann. Derselbe Leistungserbringer muss also auf Verfahrensebene von allen DAS identisch pseudonymisiert werden. Die Abläufe sind in der Datenflussspezifikation für die systemweit einheitliche Leistungserbringerpseudonymisierung in der Basisspezifikation definiert.
- Die DAS-KK muss die Sozialdaten (QS-Daten) nach erfolgreicher Prüfung mit dem öffentlichen Schlüssel der (Bundes-)Auswertungsstelle (BAS) verschlüsseln.

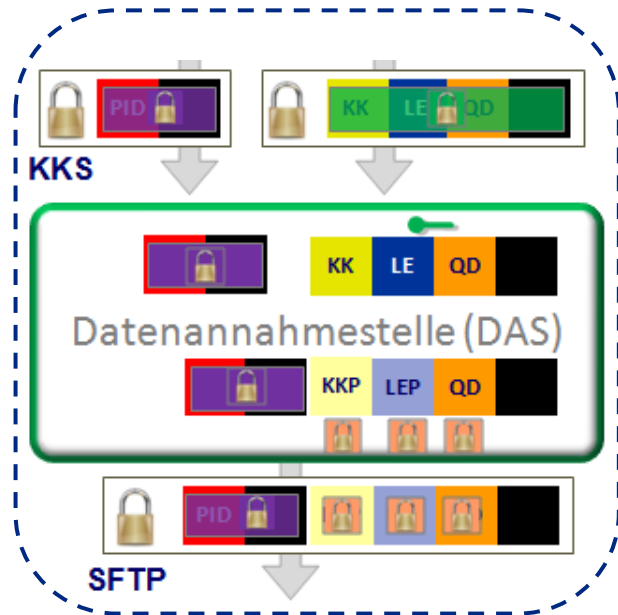


Abbildung 19: Prozesse bei der DAS-KK

#### Eingangskontrolle und Vollständigkeitsprüfung

Empfangene Dateien mit QS-Daten werden von der DAS-KK einer Schemaprüfung entsprechend Abschnitt 2.3.1 unterzogen. Bei Verfahren mit PID ist zusätzlich eine Prüfung auf das Vorhandensein einer zur QS-Datei gehörigen PID-Datei notwendig.

#### Entschlüsselung der QS-Daten

Die DAS-KK entschlüsselt die gesamten QS-Daten. Für Details siehe Abschnitt 2.4.

#### Umkodierung von ISO 8859-1 zu UTF-8

Der Export der Daten erfolgt von den Krankenkassen in der Zeichensatzkodierung ISO 8859-1. Die übermittelte QS-Datei muss deswegen bei der DAS-KK auf UTF-8 umcodiert werden, um kompatibel zu den nachfolgenden Datenflüssen zu werden.

## Datenprüfung

Die Datenprüfung setzt auf den Plausibilitätsregeln auf, wie sie im Abschnitt 2.3.2 erläutert werden. Dabei werden nur strukturelle Prüfungen vorgenommen, die harte Fehler verursachen. Fachliche Prüfungen, die weiche Fehler verursachen, werden nicht vorgenommen. Alle Daten werden weiter an die nachfolgenden Stellen im Datenfluss übermittelt, sofern mindestens ein Datensatz den Status „OK“ oder „WARNING“ hat. Die Krankenkasse erhält eine Empfangsbestätigung. Im Rahmen der Verarbeitung einer Aufstellung des Moduls AUFST entfällt dieser Schritt.

## Administrative Prüfungen

Analog zum Vorgehen bei den Krankenkassen erfolgen auch bei der DAS-KK administrative Prüfungen, wie in Abschnitt 2.3.3 beschrieben.

## Zusammenführung PID und QS-Dateien

Der Inhalt der übermittelten PID-Datei muss eingelesen, nach BASE64 codiert und abschließend in das Element `data_container` eingefügt werden.

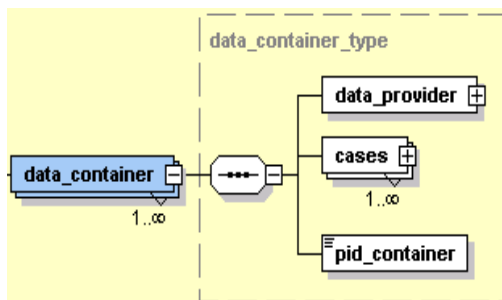


Abbildung 20: `<data_container>`-Element mit enthaltenem `<pid_container>`-Element

Die DAS-KK kann in Absprache mit der VST von diesem Vorgehen abweichen, solange dabei sichergestellt ist, dass PID und QS-Daten gemäß den Vorgaben der einschlägigen Richtlinie(n) (DeQS-RL, oKFE-RL) gemeinsam an die VST übertragen werden. Im Rahmen der Verarbeitung einer Aufstellung des Moduls AUFST entfällt dieser Schritt.

## Transformation KK-Header zu SQG-Header und KK-Body zu SQG-Body

Die korrekte Durchführung dieser Arbeitsschritte ist ausschlaggebend für das Gelingen der weiteren Verarbeitung (siehe Abbildung 21). Die durchzuführende Transformation ist in den jeweiligen XML-Schnittstellen (siehe Tabelle 31) dargestellt.



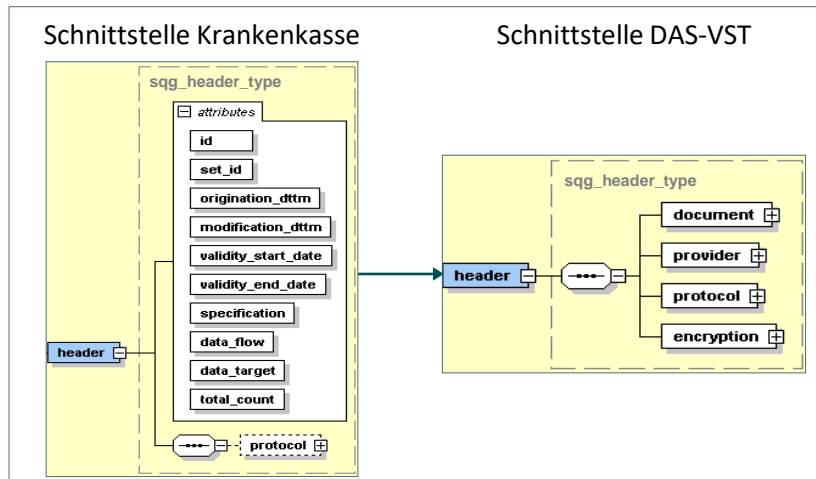


Abbildung 21: Transformation des Headers durch die DAS-KK

### Pseudonymisierung der LE-Daten

Zu den Aufgaben der DAS-KK gehört bei Verfahren gemäß DeQS-RL die Pseudonymisierung der leistungserbringeridentifizierenden Daten (LID) vor der Weiterleitung an die BAS. Hier ist die eindeutige Kennzeichnung der Leistungserbringer (IK-Nummer bei Krankenhäusern, Betriebsstättennummer bei Praxen/MVZ) durch ein Pseudonym zu ersetzen.

Das Pseudonym des Leistungserbringers (LE-Pseudonym) wird unter Entfernung der IKNR/BSNR in den XML-Code an vorgesehener Stelle eingetragen.

Bei der Pseudonymisierung ist die richtlinienspezifische Regelung zu berücksichtigen: Gemäß DeQS-RL ist die LE-Pseudonymisierung verfahrensspezifisch. Welche Module ein bestimmtes Pseudonymisierungsverfahren bilden, ist der Datenbanktabelle „Modul“ aus Spalte „pseud\_procedure\_le“ zu entnehmen.

Eine modul- und landesspezifische Auflistung inklusive der jeweils zuständigen Datenannahmestellen und der damit verbundenen Leistungserbringerpseudonymisierung wird maschinenlesbar und elektronisch verarbeitbar in den Datenserviceinformationen auf der IQTIG-Website unter folgendem Link in der jeweils aktuellen Fassung bereitgestellt:

<https://iqtig.org/datenerfassung/spezifikationen/spezifikation-zu-datenserviceinformationen/>

Für die systemweit einheitliche Leistungserbringerpseudonymisierung muss das Pseudonymisierungsprogramm (PSP) verwendet werden.

Im Rahmen der Verarbeitung einer Aufstellung des Moduls AUFST entfällt dieser Schritt.

### Pseudonymisierung der Krankenkassen und Dienstleisterkennung

Anstelle des Institutskennzeichens der Krankenkasse (kasseiknr) und der Dienstleisterkennung (dienstleister) wird je ein Pseudonym an die BAS übermittelt (siehe Erläuterung zur pseudonymisierten Dienstleisterkennung in Abschnitt 2.2.3).

Die DAS-KK muss in der Lage sein, die erstellten Pseudonyme zum Institutskennzeichen der Krankenkasse z. B. vor der Weiterleitung der Rückprotokolle zu depseudonymisieren. Zudem muss das Pseudonymisierungsverfahren konstant sein, um Datenzusammenführungen mit Vorjahren zu erlauben. Die Depseudonymisierung der Dienstleisterkennung ist nicht vorgesehen.

### Angabe des Pseudonymisierungsverfahrens für PID

Die Pseudonymisierung der patientenidentifizierenden Daten durch die VST ist vom jeweiligen Verfahren abhängig. Da ein Verfahren mehrere Module umfassen kann, ist die Angabe des Moduls alleine nicht ausreichend, um die Pseudonymisierung durchzuführen. Die DAS-KK bestimmt aus dem in den Daten angegebenen Modul das zugehörige Pseudonymisierungsverfahren und trägt es in das Feld `pseud_procedure` ein. Im Rahmen der Verarbeitung einer Aufstellung des Moduls AUFST bzw. von Modulen ohne PID entfällt dieser Schritt.

### Mapping der KV-Regionen aus BSNR

Für eine landesbezogene Auswertung der vertragsärztlichen Leistungen gemäß DeQS-RL ist eine eindeutige Zuordnung zu den Bundesländern erforderlich. Hierfür ist eine bereits von der KBV spezifizierte Kodierung zu verwenden (KV-Region)<sup>9</sup>.

Die Krankenkassen haben hierbei die ersten beiden Stellen der Betriebsstättennummer des Leistungserbringers im Klartext anzugeben (Attribut `kvregion` im Element `<care_provider>`). Diese beiden Stellen enthalten entweder einen Schlüsselwert auf KV-Ebene oder in einigen Fällen auf untergeordneter Bezirksebene einer KV. Ist ein Wert auf Bezirksebene einer KV enthalten muss dieser vor der Weiterleitung der Daten an die BAS durch den Schlüsselwert der jeweiligen KV-Region ersetzt werden. Der entsprechende Code kann der nachfolgenden Tabelle 30 entnommen werden (2. Spalte, „Kode“) und ist von der DAS-KK in das Attribut `kvregion` eintragen.

Tabelle 30: Kodierung der Landesgeschäftsstellen der Kassenärztlichen Vereinigungen

| KV-Region          | Kode | BSNR (1. und 2. Stelle)        |
|--------------------|------|--------------------------------|
| Schleswig-Holstein | 01   | 01                             |
| Hamburg            | 02   | 02                             |
| Bremen             | 03   | 03                             |
| Niedersachsen      | 17   | 06-17                          |
| Westfalen-Lippe    | 20   | 18-20                          |
| Nordrhein          | 38   | 21, 24, 25, 27, 28, 31, 37, 38 |
| Hessen             | 46   | 39-46                          |
| Rheinland-Pfalz    | 51   | 47-51                          |

<sup>9</sup> <http://applications.kbv.de> (abgerufen: 06.06.2023)

| KV-Region              | Kode | BSNR (1. und 2. Stelle) |
|------------------------|------|-------------------------|
| Baden-Württemberg      | 52   | 52-62                   |
| Bayern                 | 71   | 63-71                   |
| Berlin                 | 72   | 72                      |
| Saarland               | 73   | 73                      |
| Mecklenburg-Vorpommern | 78   | 78                      |
| Brandenburg            | 83   | 79, 80, 81, 83          |
| Sachsen-Anhalt         | 88   | 85-88                   |
| Thüringen              | 93   | 89, 90, 91, 93          |
| Sachsen                | 98   | 94, 95, 96, 98          |

Wenn in Folge des Vorliegens eines unbekannten KV-Bezirks keiner der KV-Schlüssel zur Leistungserbringerpseudonymisierung zur Anwendung gebracht werden kann, soll dennoch eine ersatzweise Pseudonymisierung in den Datenverarbeitungsprozess der DAS-KK ermöglicht sein. Ist ein unbekannter Wert auf Bezirksebene einer KV enthalten, muss dieser vor der Weiterleitung der Daten an die BAS durch den Schlüsselwert „00“ ersetzt werden.

Folgende Abbildung 22 zeigt, in welcher Relation die ersten beiden Stellen der BSNR zur KV-Region stehen.

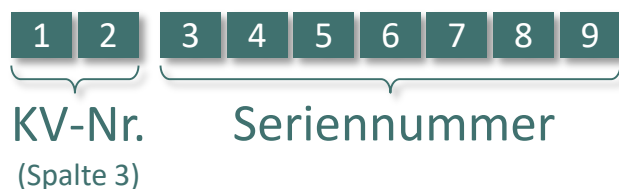


Abbildung 22: Aufbau der BSNR

Im Rahmen der Verarbeitung einer Aufstellung des Moduls AUFST entfällt dieser Schritt.

### QS-Verschlüsselung

Nach erfolgreicher Prüfung und LE-Pseudonymisierung sind die QS-Daten mit dem durch die Basispezifikation vorgesehenen Verschlüsselungsverfahren nach W3C-XML-Encryption zu verschlüsseln. Für die Verschlüsselung ist der öffentliche Schlüssel der BAS zu verwenden. Nähere Details zur W3C-Verschlüsselung sind Abschnitt 6.5 zu entnehmen.

Folgende Tabelle zeigt, welche XML-Schemata für die DAS-KK relevant sind:<sup>10</sup>

Tabelle 31: Relevante XML-Schemata für die DAS-KK

| Schnittstelle/Ablageort      | Schema                           | Verwendungszweck   |
|------------------------------|----------------------------------|--|
| KK/DAS<br>Interface_KK_DAS   | response_receipt_<br>DAS_KK.xsd  | Empfangsbestätigung der DAS  |
|                              | response_DAS_KK.xsd              | Datenflussprotokoll der DAS<br>sowie Übertragung krankenkassenspezifischer Rückmeldungen                                       |
| DAS<br>Interface_DAS         | interface_DAS.xsd                | Zusammenführung der QS- und<br>PID-Dateien   |
|                              |                                  | Pseudonymisierung der LE- und<br>KK-Daten  |
| DAS/VST<br>Interface_DAS_VST | interface_DAS_VST.xsd            | Datenübermittlung von der DAS-<br>KK an die VST  |
|                              | response_VST_DAS.xsd             | Datenflussprotokoll von der VST<br>an die DAS-KK   |
|                              | response_receipt_<br>VST_DAS.xsd | Empfangsbestätigung von der VST<br>an die DAS-KK   |
| DAS/BAS<br>Interface_DAS_BAS | interface_DAS_BAS.xsd            | Datenübermittlung von der DAS-<br>KK an die BAS  |
|                              | response_BAS_DAS.xsd             | Datenflussprotokoll der BAS (nur<br>bei Nicht-PID-Verfahren) sowie<br>Übertragung krankenkassenspezi-<br>fischer Rückmeldungen |

### Transformation der Bestätigung/Aufstellung

Bei der Übertragung einer Aufstellung und Bestätigung nach § 16 Abs. 5 DeQS-RL besteht die Erfordernis den Namen der vertretungsberechtigten Person aus der von den Krankenkassen übertragenen XML-Datei zu entfernen. Eine Aufstellung und Bestätigung wird in einem qs\_data-Element <AUFST> übertragen und ist an der Kennung der Dateiart („A“) in der Angabe zum logischen Dateinamen im Auftragssatz erkennbar.

Vor Weiterleitung der Bestätigung/Aufstellung an die BAS ist neben der Pseudonymisierung der Krankenkasse anstelle der Pseudonymisierung der LE-Daten (in Bestätigung/Aufstellung nicht enthalten) das Entfernen des Namens der vertretungsberechtigten Person erforderlich. Hierfür ist das dem <bestaetigung>-Element untergeordnete Element <vertretungsbe-rechtigtepersion>, einschließlich des den Namen der entsprechenden Person beinhalten- den Attributes V, aus der direkt an die BAS weiterzuleitenden Datei zu entfernen und in der DAS

<sup>10</sup> Die Zielgruppe ist der Schnittstelle zu entnehmen.

zu persistieren. Abbildung 23 veranschaulicht den Transformationsprozess durch Darstellung von Ausgangs- und Zielstruktur.

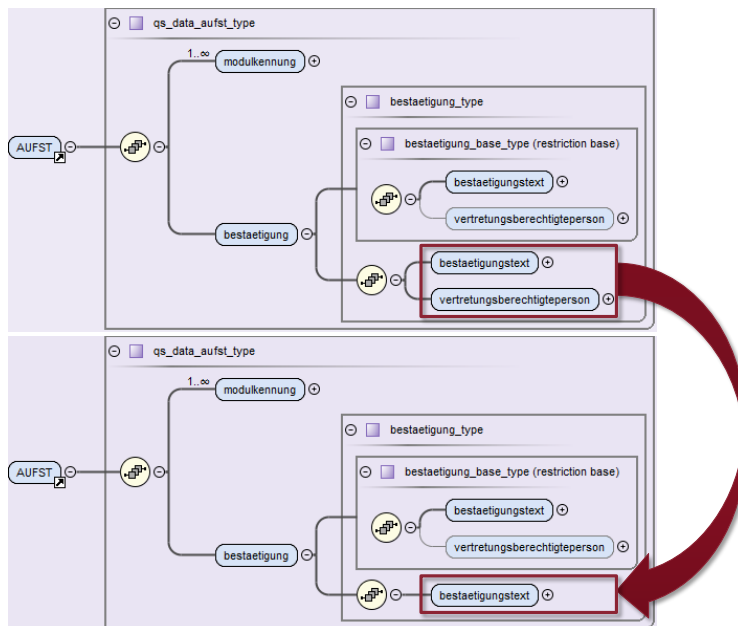


Abbildung 23: Transformation der Bestätigung durch Entfernen der vertretungsberechtigten Person

### Weiterleitung krankenkassenspezifischer Rückmeldeberichte

Analog zur Weiterleitung von Datenflussprotokollen von der BAS an die KK, sind anlassbezogen krankenkassenspezifische Rückmeldeberichte zur Validierung der Sozialdaten von der BAS an die KK zu übertragen. Hierbei ist vor Weiterleitung insbesondere die Depseudonymisierung des Institutionskennzeichens der jeweiligen Krankenkasse erforderlich. Die Berichte werden in einem XML-Element <INFO> base64-kodiert innerhalb von `./case/qs_data/INFO` in die Struktur eines Datenflussprotokolls integriert.

In Abgrenzung zur Übertragung von Datenflussprotokollen, enthalten die XML-Dateien mit einem Krankenkassenspezifischen Rückmeldebericht im Header unter `/root/header/document/data_flow/@V` den Wert „Information“ und werden im KKS mit der Inhalts- bzw. Verfahrenskennung I geführt.

Ein Beispiel für einen Dateinamen für die Übertragung von BAS zu DAS-KK lautet:

`0b9f1a6d-6f68-45a3-93a1-8f03ae38e69e_I_BAS.xml`

Weitere zur Nutzung des KKS erforderlichen Details zur Dateibenennung sowie zum Auftragsatz sind Abschnitt 2.7 Informationen über die Güte gelieferter Daten zu entnehmen. Der Wert für die 5-stellige Modulbezeichnung im logischen Dateinamen des Auftragsatzes kann aus der von der BAS übermittelten XML-Datei entnommen werden aus der Angabe im Attribut `/root/body/data_container/cases/@module`.

## 4 Prozesse bei der Vertrauensstelle

### Eingangsprüfung/Empfangsbestätigung

Die von der DAS-KK erhaltenen Daten werden auf Schemakonformität geprüft. Nach Feststellen der Schemakonformität wird eine Empfangsbestätigung (entsprechend der Spezifikation) erzeugt und an die DAS-KK übermittelt.

Das Übertragungsverfahren ist SFTP.

### Administrative Prüfungen

Analog zum Vorgehen bei den Krankenkassen erfolgen auch bei der Vertrauensstelle (VST) administrative Prüfungen, wie in Abschnitt 2.3.3 beschrieben.

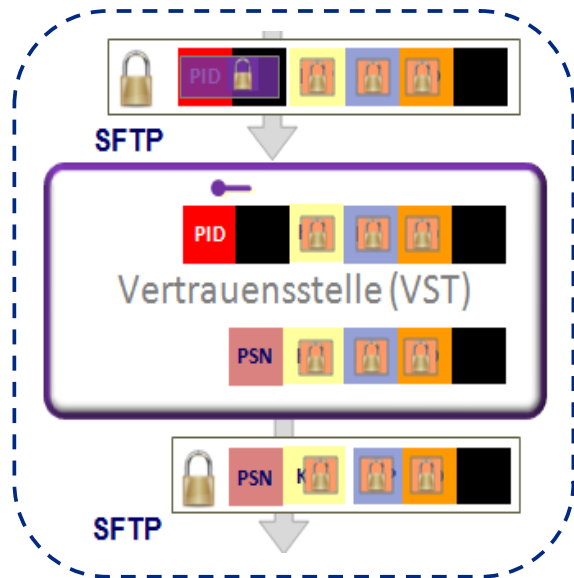


Abbildung 24: Prozesse bei der VST

### Herauslösung und Entschlüsselung des PID-Blocks

Der ins XML-Dokument eingebettete, BASE64-kodierte PID-Block (siehe Abschnitt 2.2.3) wird aus dem XML-Dokument herausgelöst und entsprechend dem Verschlüsselungsverfahren zwischen Krankenkasse und VST mit dem privaten Schlüssel der VST in einem XML-Strom entschlüsselt.

### Einfügen der patient-Elemente

Über ein Matching der Datensatznummern in den nun parallel zu verarbeitenden XML-Dateien (QS- und PID-Datei) erfolgt die Zuordnung der PID zu den jeweiligen Leistungs- und Medikationsdaten auf Fallebene. Dazu wird das Attribut `lfidnr` des `<case>`-Elements in der PID-Datei der ID im `<case_admin>`-Element der QS-Datei zugeordnet.

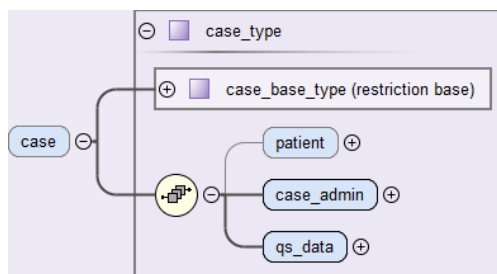


Abbildung 25: `<case>`-Element nach Verarbeitung durch die VST

## Pseudonymisierung der PID

Die daraus entstehende XML-Struktur stimmt nun mit dem XML überein, das in der Basisspezifikation für diese Schnittstelle definiert ist. Die weitere Verarbeitung der VST zur Pseudonymisierung orientiert sich an dem Verfahren, welches über vertragliche Regelungen zwischen der VST und dem G-BA definiert ist und auf der Dokumentenstruktur der Spezifikation für die Sozialdaten bei den Krankenkassen aufsetzt.

Tabelle 32 zeigt, nach welchem XML-Schema die VST die Exportdateien aufbereiten muss:

Tabelle 32: XML-Schemata für die VST

| Ablageort             | XML-Schema                | Verwendungszwecke   |
|-----------------------|---------------------------|---|
| interface_<br>VST_BAS | interface_<br>VST_BAS.xsd | <ul style="list-style-type: none"> <li>▪ Zusammenführung der QS- und der PID Datensätze</li> <li>▪ Integration der Patientenpseudonyme in die entsprechenden QS-Datensätze</li> <li>▪ Datenübermittlung von der VST an die BAS</li> </ul> |

## Umgang mit Widersprüchen gemäß § 8 oKFE-RL

Wenn Versicherte einer Verarbeitung Ihrer Daten zum Zweck der Programmbeurteilung nach § 6 oKFE-RL wirksam widersprochen haben, müssen gem. § 8 Abs. 4 Satz 4 in der VST die dazugehörigen Programmbeurteilungsdaten gelöscht und durch eine Kennung, dass ein Widerspruch vorliegt ersetzt werden.

Bei Vorliegen entsprechender Widersprüche, erfolgt das „Einfügen der patient-Elemente“ sowie die „Pseudonymisierung der PID“ wie gehabt. Zusätzlich müssen jedoch zwei weitere Schritte vollzogen werden, deren strukturelle Zugehörigkeiten sich auf das Element `qs_data` beschränken, so wie in Abbildung 26 dargestellt.

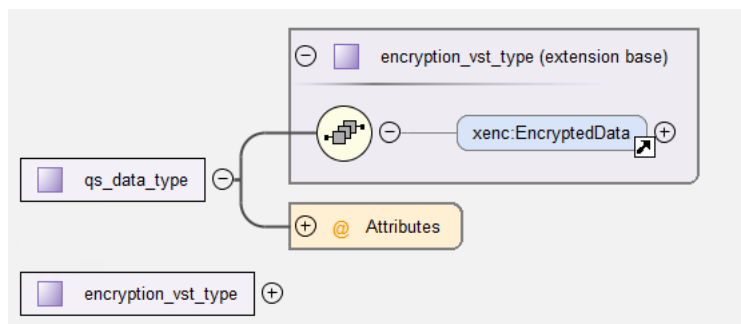


Abbildung 26: Umgang mit Widersprüchen gemäß § 8 oKFE-RL

In einem ersten Schritt erfolgt die Löschung der Programmbeurteilungsdaten. Hierfür muss der gesamte XML-verschlüsselte Inhalt des `qs_data`-Elementes entfernt werden. In einem zweiten Schritt ist die zuvor erfolgte Löschung der Programmbeurteilungsdaten im XML-Attribut `widerspruch` des `qs_data`-Elementes anzuzeigen, so dass die Auswertungsstelle dies erkennen kann.

### **Übermittlung an BAS**

Die Weiterleitung der Daten an die BAS erfolgt mittels SFTP. Hierbei ist zu beachten, dass Daten für Module gemäß DeQS-RL an das IQTIG, wohingegen Daten für Module gemäß oKFE-RL an die GFL zu übermitteln sind.

### **Rückmeldung der BAS**

Die von der VST empfangenen Daten werden von der BAS auf Schemakonformität geprüft. Nach Feststellen der Schemakonformität wird eine Empfangsbestätigung (entsprechend der Spezifikation) erzeugt und per SFTP an die VST übermittelt.

Bei einem Schemaverstoß wird anstelle der Eingangsbestätigung ein Protokoll mit entsprechenden Fehlermeldungen zurückgeschickt.



## 5 Prozesse bei der (Bundes-)Auswertungsstelle

Abweichend von der Regelverarbeitung, die in der Basisspezifikation festgelegt ist, hat die (Bundes-)Auswertungsstelle (BAS) in Bezug auf die Verarbeitung der Sozialdaten folgende Punkte zu berücksichtigen:

1. Eine Überprüfung der Daten auf Duplikate zwischen der Datenlieferung und den bereits im Bestand der BAS vorhandenen Daten entfällt.
2. Jede Datenlieferung wird als komplett für den ausgewiesenen Zeitraum, das ausgewiesene Exportmodul, das Erfassungsjahr und die einsendende Krankenkasse betrachtet.
3. Das bedeutet, dass die BAS bei einer Lieferung alle Daten, die diesem Rahmen entsprechen, aus dem Datenpool entfernt und die neue Datenlieferung in den Datenpool einspielt.
4. Die Zeichensatzkodierung des Rückprotokolls muss in ISO 8859-1 erfolgen.
5. Die Datenlieferungen sind für Verfahren gemäß DeQS-RL in Dateien zu jeweils maximal 10.000 Datensätzen segmentiert. Für Verfahren gemäß oKFE-RL dürfen maximal 50.000 Datensätze innerhalb eines Segmentes exportiert werden. Rückprotokolle beziehen sich immer auf die Gesamtdatenlieferung.
6. Die Annahme der Daten und der Versand der Rückprotokolle erfolgt per SFTP.
7. Anlassbezogene (insbesondere nach erstmaliger Übermittlung von Daten zu einem „neuen“ Modul) Erstellung krankenkassenspezifischer Rückmeldungen (Informationen/Berichte) nebst Übertragung an DAS-KK zum Zweck der Weiterleitung nach Depseudonymisierung des Institutionskennzeichens der jeweiligen Krankenkasse.

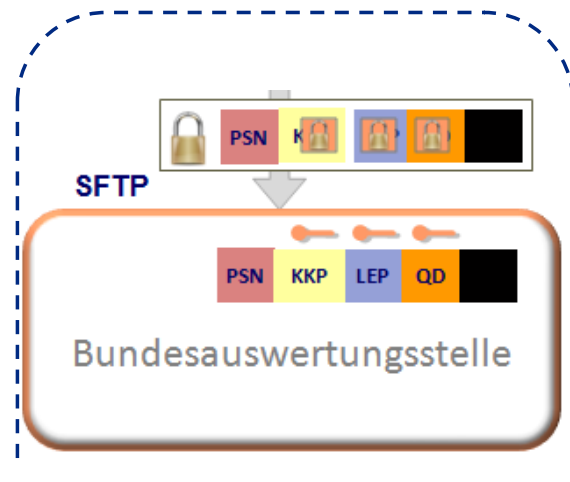


Abbildung 27: Prozesse bei der BAS

## 6 Anhang

### 6.1 Herkunft der Datenbestände nach Abrechnungskontext

#### 6.1.1 Datenbestände und Lieferfristen

Die Sozialdaten bei den Krankenkassen untergliedern sich in verschiedene Datenbestände, die sich hinsichtlich ihrer Herkunft unterscheiden. Je nach Abrechnungskontext sind die Datenaustauschverfahren zwischen Leistungserbringern und Krankenkassen unterschiedlich geregelt. Details dazu finden sich in den jeweiligen technischen Anlagen zu den Datenaustauschverfahren.<sup>11</sup> Aufgrund ihrer Verarbeitungswege kommt es bei den verschiedenen Datenbeständen zu unterschiedlichen Verzögerungsfristen, bis die entsprechenden Daten vollständig bei den Krankenkassen vorliegen. Aktuelle Annahmen zu den Verzögerungsfristen sind in Tabelle 33 aufgeführt.

Tabelle 33: Übersicht der Datenbestände

| Datenbestand nach Abrechnungskontext (jeweils SGB V) | Beschreibung  | Verzögerung bis zur Verfügbarkeit bei den Krankenkassen |
|--|---|---|
| § 301  | Stationäre Behandlung im Krankenhaus                              | max. 3 Monate nach Entlassung                           |
| § 115b   | Ambulantes Operieren im Krankenhaus                               | ca. 8 Monate nach Quartalsende                          |
| § 116b   | Ambulante Spezialfachärztliche Versorgung                         | ca. 9 Monate nach Quartalsende                          |
| § 117  | Hochschulambulanz   | ca. 8 Monate nach Quartalsende                          |
| § 118  | Psychiatrische Institutsambulanz                                  | ca. 8 Monate nach Quartalsende                          |
| § 119  | Sozialpädiatrisches Zentrum                                       | ca. 8 Monate nach Quartalsende                          |
| § 295 (kollektiv)                                    | Vertragsärzte mit kollektivvertraglicher Abrechnung               | ca. 8 Monate nach Quartalsende                          |
| § 295 1b (§ 140a)                                    | Vertragsärzte mit Abrechnungsform „Integrierte Versorgungsformen“ | ca. 8 Monate nach Leistungserbringung                   |
| § 295 1b (§ 73b)                                     | Vertragsärzte mit Abrechnungsform „Hausarztzentrierte Versorgung“ | ca. 8 Monate nach Quartalsende                          |

<sup>11</sup> Siehe <http://www.gkv-datenaustausch.de/leistungserbringer/leistungserbringer.jsp> (abgerufen: 05.06.2023)

| Datenbestand nach Abrechnungskontext (jeweils SGB V) | Beschreibung   | Verzögerung bis zur Verfügbarkeit bei den Krankenkassen            |
|--|--|--|
| § 295 1b (§73c)                                      | Vertragsärzte mit Abrechnungsförm „Besondere ambulante ärztliche Versorgung“ | ca. 8 Monatenach Leistungserbringung                               |
| § 300  | Apotheken  | ungeprüft ca. 2 Monate nach Abgabe, geprüft ca. 1 Jahr nach Abgabe |
| § 302 (Heilmittel)                                   | Heilmittelverordnung nach § 302  | ca. 7 Monate nach Leistungserbringung                              |
| § 302 (Hilfsmittel)                                  | Hilfsmittelverordnung nach § 302   | ca. 7 Monate nach Abgabe   |
| § 284 (Stammdaten)                                   | Versichertendaten nach § 284   | max. 2 Monate  |
| § 25a  | Organisierte Früherkennungsprogramme   |  |

Die entstehenden Verzögerungen müssen in Abhängigkeit von den jeweils verwendeten Datenbeständen in den Auswertungskonzepten der einzelnen Verfahren berücksichtigt werden.

### 6.1.2 Export der Datenbestände

Die von den Leistungserbringern an die Krankenkassen gelieferten Daten beziehen sich auf konkrete Behandlungs- bzw. Verordnungsfälle. In den Exportdateien finden sich diese Fälle als eigene Elemente wieder. Bei gleicher oder ähnlicher Struktur können Fälle aus verschiedenen Datenbeständen in den gleichen Elementtypen abgebildet werden. Dies ist in Tabelle 34 dargestellt.

Tabelle 34: Ziel-Elemente der Datenbestände beim Export

| Datenbestand | Beschreibung                              | XML-Element in der Export-Datei* |
|--------------|---|----------------------------------|
| § 301        | Stationäre Behandlung im Krankenhaus      | fall_301                         |
| § 115b       | Ambulantes Operieren im Krankenhaus       | fall_kh_ambo                     |
| § 116b       | Ambulante Spezialfachärztliche Versorgung |                                  |
| § 117        | Hochschulambulanz                         |                                  |
| § 118        | Psychiatrische Institutsambulanz          |                                  |
| § 119        | Sozialpädiatrisches Zentrum               |                                  |

| Datenbestand        | Beschreibung   | XML-Element in der Export-Datei* |
|---------------------|--|----------------------------------|
| § 295 (kollektiv)   | Vertragsärzte mit kollektiv-vertraglicher Abrechnung                         | fall_295                         |
| § 295 1b (§ 140a)   | Vertragsärzte mit Abrechnungsform „Integrierte Versorgungsformen“            |                                  |
| § 295 1b (§ 73b)    | Vertragsärzte mit Abrechnungsform „Hausarzt-zentrierte Versorgung“           |                                  |
| § 295 1b (§ 73c)    | Vertragsärzte mit Abrechnungsform „Besondere ambulante ärztliche Versorgung“ |                                  |
| § 300               | Apotheken  | fall_300                         |
| § 302 (Heilmittel)  | Heilmittelverordnung nach § 302  | fall_302                         |
| § 302 (Hilfsmittel) | Hilfsmittelverordnung nach § 302   |                                  |
| § 284 (Stammdaten)  | Versichertendaten nach § 284   | versicherter_type (QS-Datei)     |
|                     |  | pid_data_type (PID-Datei)        |
| § 25a               | Organisierte Früherkennungsprogramme   | fall_25a                         |

\* Hier wird das in der Hierarchie der XML-Struktur oberste Element genannt, das für diesen Datenpool spezifisch ist. Die aus dem Datenpool stammenden Daten können teilweise in untergeordnete Elemente gegliedert sein.

Die jeweiligen Elemente sind entsprechend der in den XML-Schemata definierten Struktur zu exportieren. Diese Struktur ist in Abschnitt 6.4 genauer beschrieben.

## 6.2 Datensatzbeschreibung

### 6.2.1 Hierarchie der Elemente für den Export

Die Exportelemente sind hierarchisch aufgebaut. Das bedeutet, dass ein Element außer seinen Attributen noch ein oder mehrere weitere Elemente enthalten kann. Dies ist in den folgenden Abbildungen dargestellt:

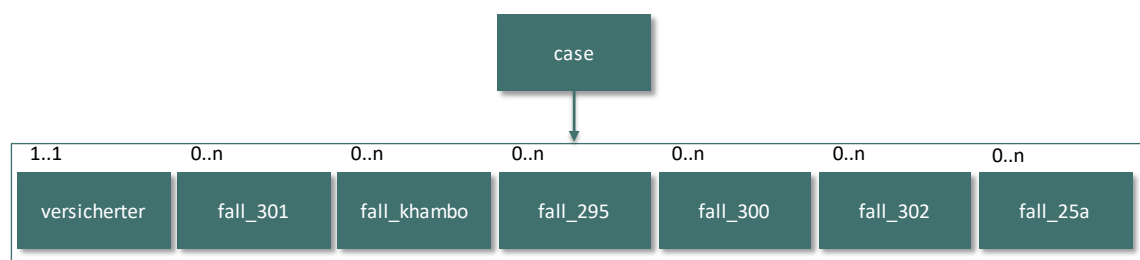


Abbildung 28: Hierarchie der Elemente für den Export – Element: case

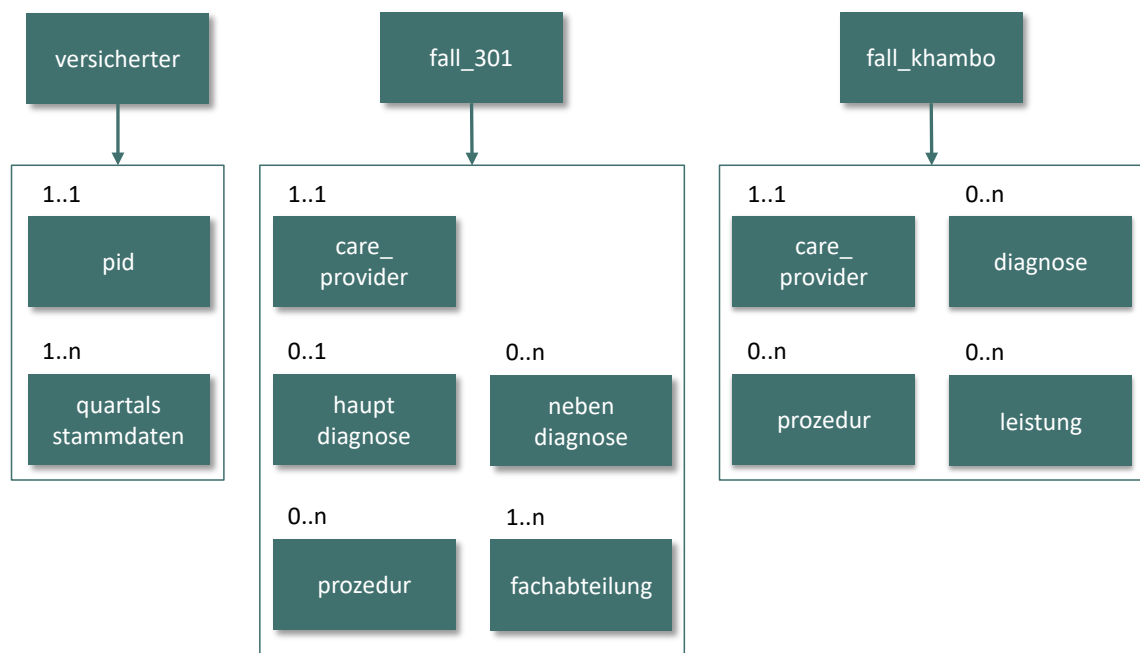


Abbildung 29: Hierarchie der Elemente für den Export – Elemente: versicherter, fall\_301, fall\_khambo

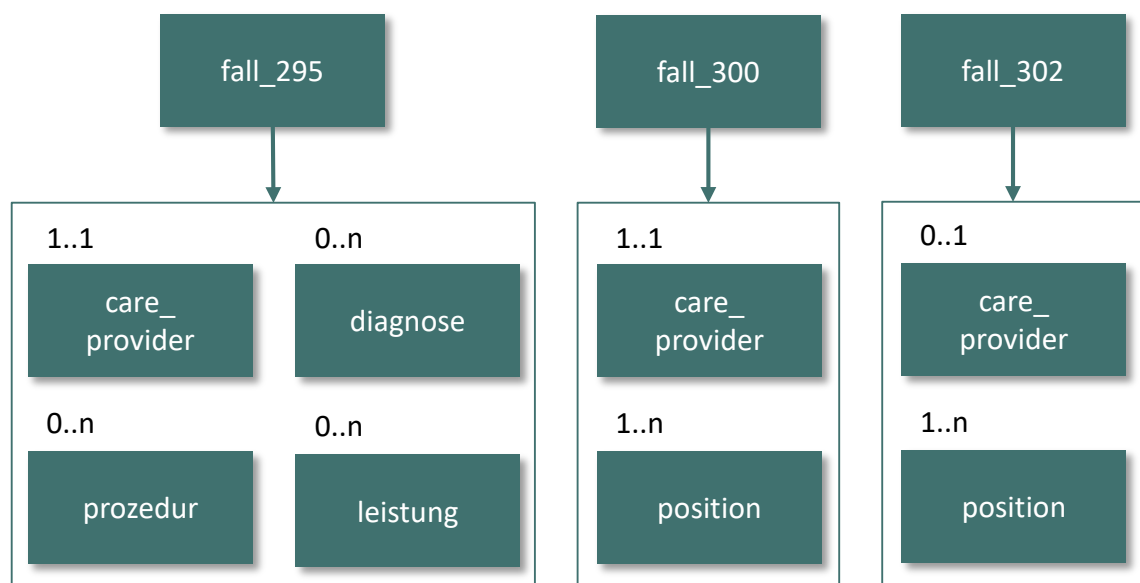


Abbildung 30: Hierarchie der Elemente für den Export – Elemente: fall\_295, fall\_300, fall\_302

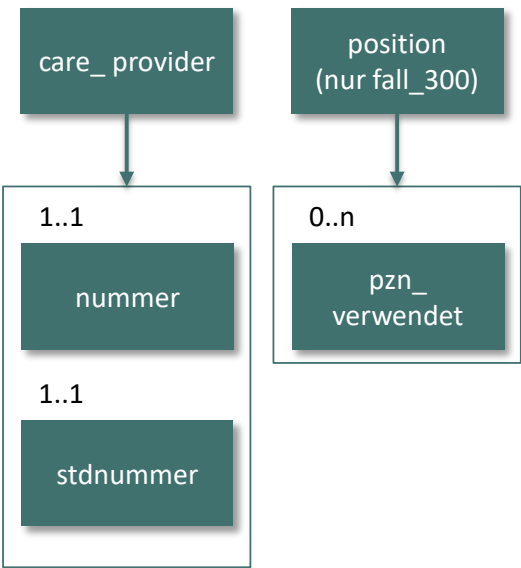


Abbildung 31: Hierarchie der Elemente für den Export – Elemente: care\_provider, position

Darüber hinaus sind noch weitere Elemente und Attribute in der XML-Datei enthalten, wie im Abschnitt 6.4 beschrieben.

**6.2.2      Elemente und Attribute der Nutzdaten in der QS-Datei**

Im Folgenden werden die zu exportierenden Elemente im `body`-Bereich der QS-Datei beschrieben. Dabei werden Elementname und Elementdatentyp nach folgendem Schema angegeben:

[ <Elementname> : <Datentyp> ]

Elementnamen können mehrfach mit unterschiedlichem Datentyp vorkommen. Bei den angegebenen Elementdatentypen handelt es sich jeweils um den Basisdatentyp mit allen für dieses Element möglichen Attributen. Je nach Modul können diese Datentypen im XML-Schema eingeschränkt sein.

**[body : sqg\_body\_type] – Nutzdaten-Element**

| Lfd. Nr. | Techn. Feld-name | Feldbezeichnung  | Feld-typ      | Max. Stellen-zahl | M /K | Schlüs-sel | Bemerkun-gen |
|----------|------------------|--|---------------|-------------------|------|------------|--------------|
| 1        | kasseiknr        | Institutionskenn-<br>zeichen der<br>Krankenkasse<br>(Hauptkassen-IK<br>ohne<br>Erstreckung-IK) | IKNR_<br>BSNR | 9                 | M    | -          | -            |

|   |                    |  |              |   |   |   |   |
|---|--------------------|--|--------------|---|---|---|---|
| 2 | module             | Modulkennung   | -            |   | M |   | Name des Moduls aus der Tabelle Modul                     |
| 3 | dienstleister      | (Pseudonymisierte) Dienstleisterkennung                                  | STRING       | - | K | - | Handelsregisternummer, wird im Datenfluss pseudonymisiert |
| 4 | Versichertenanzahl | Aktuellste Anzahl der Versicherten gemäß KM1-Statistik zum Lieferfenster | POSGANZEZAHL | - | K | - | -   |

Dieses Element enthält den gesamten Nutzdatenbereich der Datei. Es enthält als Attribut die IKNR der liefernden Krankenkasse und die Dienstleisterkennung als Handelsregisternummer des Softwareanbieters, der für die Datenselektion und -lieferung von Sozialdaten bei den Krankenkassen beauftragt wurde. Diese Attribute werden von der DAS-KK vor der weiteren Übermittlung der Datei durch ein Pseudonym ersetzt und als Kindelemente des Elements `<data_provider>` umstrukturiert. Zudem enthält das Element `<body>` das optionale Attribut `versichertenanzahl`, das die aktuellste Anzahl aller Versicherten der Krankenkasse gemäß KM1-Statistik zum festgelegten Lieferfenster angibt und für alle Module nach DeQS-RL zu befüllen ist (siehe Abschnitt 2.2.3). Weiteres Attribut ist das Modul `module`. Diese Kennung wird nicht aus den Datenbeständen der Krankenkasse ermittelt, sondern der Tabelle `Modul` in der Spezifikationsdatenbank entnommen.

#### [case : case\_type] – Datensatz eines Versicherten

| Lfd. Nr. | Techn. Feld-name | Feldbezeichnung                              | Feldtyp      | Max. Stellenzahl | M/K | Schlüssel | Bemerkungen |
|----------|------------------|--|--------------|------------------|-----|-----------|-------------|
| 1        | lfdnr            | Laufende Nummer des Versicherten-datensatzes | POSGANZEZAHL |                  | M   | -         | -           |

Dieses Element enthält genau einen Datensatz, der alle exportierten Daten zu einem bestimmten Versicherten umfasst.

#### [qs\_data: qs\_data\_type] – Übergreifendes Container-Element für QS-Daten

Dieses Element dient als Container für die Daten gem. DeQS-RL bzw. oKFE-RL eines Versicherten.

**[BSP: qs\_data\_beispiel\_type] – Container-Element für QS-Daten eines bestimmten Moduls**

Vorhandene Container-Elemente:

- BSP: qs\_data\_beispiel\_type
- PCI: qs\_data\_pci\_type
- NWITR: qs\_data\_nwitr\_type
- NWIWI: qs\_data\_nwiwi\_type
- CHOL: qs\_data\_chol\_type
- AUFST: qs\_data\_aufst\_type
- KCHK: qs\_data\_kchk\_type
- DIALS: qs\_data\_dials\_type
- NTXS: qs\_data\_ntxs\_type
- INFO: qs\_data\_info\_type
- KFEDK: qs\_data\_kfedk\_type
- KFEZK: qs\_data\_kfezk\_type
- HTXS: qs\_data\_htxs\_type
- LUTXS: qs\_data\_lutxs\_type
- LTXS: qs\_data\_ltxs\_type

Jedes Container-Element ist spezifisch für ein Modul. Es ermöglicht die Anwendung unterschiedlicher XML-Schemata innerhalb des übergeordneten Elementes `qs_data`. Mit Einführung weiterer Module werden zusätzliche Elementnamen (und entsprechende Datentypen) hinzukommen, die an dieser Stelle austauschbar sind. Dabei ist jedoch zu beachten, dass jede Datenlieferung immer nur Daten zu einem einzigen Modul beinhalten darf. Der Elementname entspricht dem Kürzel des zugehörigen Moduls.

**[versicherter: versicherter\_type] – Versichertendaten**

| Lfd. Nr. | Techn. Feldname | Feldbezeichnung | Feldtyp    | Max. Stellenzahl | M/K | Schlüssel  | Bemerkungen  |
|----------|-----------------|-----------------|------------|------------------|-----|------------|--|
| 1        | geschlecht      | Geschlecht      | SCHLUESSEL | 1                | M   | geschlecht | (m: männlich, w: weiblich, u: unbestimmt, d: divers)                                   |
| 2        | gebdatum        | Geburtsdatum    | DATUM      | 10               | M   |            | JJJJ-MM-TT<br>(Hinweis: nur bei Verfahren, die das tagesgenaue Geburtsdatum erfordern) |



| Lfd. Nr. | Techn. Feldname | Feldbezeichnung              | Feldtyp | Max. Stellenzahl | M/K | Schlüssel | Bemerkungen  |
|----------|-----------------|------------------------------|---------|------------------|-----|-----------|--|
|          | gebjahr         | Geburtsjahr des Versicherten | JAHRTUM | 4                | M   |           | JJJJ<br>(Hinweis: bei Verfahren, die kein tagesgenaues Geburtsdatum erfordern) |
| 3        | sterbedatum     | Sterbedatum des Versicherten | DATUM   | 10               | K   |           | JJJJ-MM-TT   |

Dieses Element enthält die Attribute des Versicherten. Dabei ist je nach Modul zu unterscheiden, ob das taggenaue Geburtsdatum `gebdatum` oder das Geburtsjahr `gebjahr` enthalten ist. Nur eines dieser beiden Datenfelder ist in dem konkreten Element enthalten.



#### Hinweis

Das Sterbedatum `sterbedatum` ist auf Ebene des Versicherten einmalig als Kann-Feld definiert. Diese Kann-Eigenschaft bezieht sich jedoch nicht auf die grundsätzlich fakultative Übermittlung dieser Information, sondern auf die Tatsache, dass nicht jeder Patient zum Zeitpunkt einer Datenselektion verstorben sein muss.

**[quartalsstammdaten : quartalsstammdaten\_type] – Quartalsweise Versichertendaten**

| Lfd. Nr. | Techn. Feld-name    | Feldbezeichnung                      | Feldtyp              | Max. Stellenzahl | M / K | Schlüssel | Bemerkungen  |
|----------|---------------------|--------------------------------------|----------------------|------------------|-------|-----------|--|
| 1        | versicherungsdatum  | Datumsangabe zum Versicherungsstatus | DATUM-QUARTALS-MITTE | 10               | M     |           | Stichtag ist jeweils die Mitte des Quartals (Q1: 15.02.; Q2: 15.05.; Q3: 15.08.; Q4: 15.11.) |
| 2        | versicherungsstatus | Patient ist versichert (ja/nein)     | SCHLUESSEL           | 1                | M     | jn        | 1: ja; 0: nein   |

Dieses Element enthält Angaben zum Mitgliedsstatus des Versicherten in einem bestimmten Quartal. Es ist für jedes Quartal im von der Datenlieferung umfassten Zeitraum zu exportieren und enthält den genauen Stichtag (jeweils zur Mitte des Quartals) sowie die Angabe, ob der Versicherte an diesem Stichtag bei der Krankenkasse versichert war.

**[care\_provider: care\_provider\_type] – Leistungserbringer**

| Lfd. Nr. | Techn. Feld-name | Feldbezeichnung                                       | Feldtyp    | Max. Stellenzahl | M / K | Schlüssel  | Bemerkungen  |
|----------|------------------|---|------------|------------------|-------|------------|--|
| 1        | art              | Art der Identifikationsnummer des Leistungserbringers | SCHLUESSEL | 4                | M     | le_kennung | „bsnr“ oder „iknr“   |
| 2        | bundesland       | Bundesland des Krankenhauses                          | SCHLUESSEL | 2                | K     | bundesland | Bundeslandschlüssel aus der IKNR des Leistungserbringers bzw. dem Datenbestand der Kasse |
| 3        | kvregion         | KV-Region der Praxis                                  | GANZE-ZAHL | 2                | K     |            | KV-Regions- bzw. Bezirks- schlüssel aus der BSNR   |

| Lfd. Nr. | Techn. Feld-name | Feldbezeichnung                | Feldtyp     | Max. Stellenzahl | M /K | Schlüssel | Bemerkungen                   |
|----------|------------------|--------------------------------|-------------|------------------|------|-----------|-------------------------------|
| 4        | fachgruppe       | Fachgruppe des Arztes, kodiert | FACH-GRUPPE | 2                | K    |           | Fachgruppenschlüssel aus LANR |

Dieses Element kommt in allen Behandlungs- und Verordnungsfällen vor und enthält Informationen über den verordnenden bzw. ausführenden Leistungserbringer. Abhängig von der Art des Leistungserbringers werden entweder das Bundesland oder die KV-Region und die Fachgruppe des Arztes angegeben.

**[nummer] – Identifikationsnummer und Standortnummer des Leistungserbringers**

| Lfd. Nr. | Techn. Feldname | Feldbezeichnung                               | Feldtyp   | Max. Stellenzahl | M/K | Schlüssel | Bemerkungen    |
|----------|-----------------|---|-----------|------------------|-----|-----------|----------------|
| 1        | nummer          | Identifikationsnummer des Leistungserbringers | IKNR_BSNR | 9                | M   |           | BSNR bzw. IKNR |
| 2        | stdnummer       | Standortnummer der entlassenden Einrichtung   | STANDORT  | 9                | M   |           |                |

Dieses Element enthält die konkrete Identifikationsnummer des Leistungserbringers sowie die Standortnummer der entlassenden Einrichtung, die von der DAS-KK durch ein Pseudonym ersetzt wird. `nummer` und `stdnummer` haben jeweils einen eigenen Datentyp.

**[fall\_301: fall\_301\_type] – Fall nach § 301**

| Lfd. Nr. | Techn. Feld-name | Feldbezeichnung | Feldtyp | Max. Stellenzahl | M / K | Schlüssel | Bemerkungen   |
|----------|------------------|-----------------|---------|------------------|-------|-----------|---|
| 1        | patientenfilter  | -               | BOOLEAN | 1                | K     |           | Wird mit Wert „1“ exportiert, wenn der Fall dem Patientenfilter |

| Lfd. Nr. | Techn. Feldname | Feldbezeichnung                           | Feldtyp     | Max. Stellenzahl | M / K | Schlüssel | Bemerkungen  |
|----------|-----------------|---|-------------|------------------|-------|-----------|--|
|          |                 |   |             |                  |       |           | des Moduls entspricht. Andernfalls wird das Attribut nicht exportiert.                             |
| 2        | aufngrund       | Aufnahmegrund                             | AUFN-GRUND  | 4                | M     |           |  |
| 3        | aufndatum       | Aufnahmedatum Krankenhaus                 | DATUM       | 10               | M     |           |  |
| 4        | entldatum       | Entlassungsdatum Krankenhaus              | DATUM       | 10               | M     |           |  |
| 5        | entlgrund       | Entlassungsgrund                          | ENT-LGRUND  | 3                | M     |           |  |
| 6        | khunterbrechung | Unterbrechung des Krankenhausaufenthaltes | SCHLU-ESSEL | 1                | M     | JN        | (1: mind. ein Entlass-/Verlegungsgrund (1. und 2. Stelle) 16, 21 oder 23 im Segment ETL; 0: sonst) |
| 7        | belegleistung   | Art der Belegleistung                     | SCHLU-ESSEL | 1                | M     | belegart  |  |
| 8        | drg_aus_liste   | DRG des KH-Falles in Liste                | BOOLEAN     | 1                | M     |           | Geprüft wird die letzte nicht wieder gutgeschriebene DRG.  |

Dieses Element enthält die Daten eines stationären Behandlungsfalles nach § 301.

**[fachabteilung: fachabteilung\_type] – Fachabteilung eines stationären Falles nach § 301**

| Lfd. Nr. | Techn. Feld-name | Feldbezeichnung          | Feldtyp    | Max. Stellenzahl | M/K | Schlüssel | Bemerkungen                 |
|----------|------------------|--------------------------|------------|------------------|-----|-----------|-----------------------------|
| 1        | fachabteilung    | Beteiligte Fachabteilung | SCHLUESSEL | 4                | M   | fachabt   | Fachabteilung des KH-Falles |

Für jede beteiligte Fachabteilung eines stationären KH-Falles wird ein Element fachabteilung erzeugt.

**[fall\_khambo: fall\_khambo\_type] – Fall nach § 301 (AMBO)**

| Lfd. Nr. | Techn. Feld-name | Feldbezeichnung        | Feldtyp    | Max. Stellenzahl | M / K | Schlüssel | Bemerkungen  |
|----------|------------------|------------------------|------------|------------------|-------|-----------|--|
| 1        | patientenfilter  | -                      | BOOLEAN    | 1                | K     |           | Wird mit Wert „1“ exportiert, wenn der Fall dem Patientenfilter des Moduls entspricht. Andernfalls wird das Attribut nicht exportiert. |
| 2        | quelle           | Datenquelle des Falles | SCHLUESSEL | 8                | M     | quelle    | Kennzeichen der Datenquelle des Falles   |
| 3        | zugangsdatum     | Tag des Zugangs        | DATUM      | 10               | M     |           |  |

Dieses Element enthält die Daten eines nach § 301 (AMBO) abgerechneten Behandlungsfalles. Dabei kann es sich um eine Abrechnung nach den §§ 115b, 116b, 117, 118 oder 119 handeln.

**[fall\_295: fall\_295\_type] – Quartalsfall nach § 295 (kollektiv- oder selektivvertraglich)**

| Lfd. Nr. | Techn. Feld-name | Feldbezeichnung | Feldtyp | Max. Stellenzahl | M/K | Schlüssel | Bemerkungen                   |
|----------|------------------|-----------------|---------|------------------|-----|-----------|-------------------------------|
| 1        | patientenfilter  | -               | BOOLEAN | 1                | K   |           | Wird mit Wert „1“ exportiert, |

| Lfd. Nr. | Techn. Feldname | Feldbezeichnung                         | Feldtyp    | Max. Stellenzahl | M/K | Schlüssel  | Bemerkungen  |
|----------|-----------------|---|------------|------------------|-----|------------|--|
|          |                 |   |            |                  |     |            | wenn der Fall dem Patientenfilter des Moduls entspricht. Andernfalls wird das Attribut nicht exportiert. |
| 2        | quelle          | Datenquelle des Falles                  | SCHLUESSEL | 8                | M   | quelle     | Kennzeichen der Datenquelle des Falles   |
| 3        | inanspruchnahme | Art der Inanspruchnahme                 | SCHLUESSEL | 1                | M   | inanspruch | Art der Inanspruchnahme des niedergelassenen Arztes  |
| 4        | beginn datum    | Erstes Behandlungsdatum im Quartal      | DATUM      | 10               | M   |            |  |
| 5        | ende datum      | Letztes Behandlungsdatum im Quartal     | DATUM      | 10               | M   |            |  |
| 6        | behandlungsart  | Art der Behandlung (ambulant/stationär) | SCHLUESSEL | 1                | K   | behandart  |  |

Dieses Element enthält die Daten eines nach § 295 abgerechneten Behandlungsfalles. Dabei kann es sich um eine kollektivvertragliche Abrechnung oder um eine selektivvertragliche Abrechnung nach den §§ 140a, 73b oder 73c handeln.

**[fall\_300: fall\_300\_type] – Arznei-/Hilfsmittelverordnung nach § 300**

| Lfd. Nr. | Techn. Feld-name   | Feldbezeichnung                                 | Feldtyp    | Max. Stellenzahl | M /K | Schlüssel | Bemerkungen  |
|----------|--------------------|---|------------|------------------|------|-----------|--|
| 1        | patientenfilter    | -   | BOOLEAN    | 1                | K    |           | Wird mit Wert „1“ exportiert, wenn der Fall dem Patientenfilter des Moduls entspricht. Andernfalls wird das Attribut nicht exportiert. |
| 2        | verordnungsdatum   | Verordnungsdatum des Arznei-/Heil-/Hilfsmittels | DATUM      | 10               | M    |           |  |
| 3        | begründungspflicht | Begründungspflicht der Verordnung               | SCHLUESSEL | 1                | M    | jn        | Handelt es sich um einen Zahnarzt?   |

Dieses Element enthält eine Arzneimittel- bzw. Hilfsmittelverordnungsabrechnung einer Apotheke.

**[fall\_302: fall\_302\_type] – Heil-/Hilfsmittelverordnung nach § 302**

| Lfd. Nr. | Techn. Feld-name | Feldbezeichnung | Feldtyp | Max. Stellenzahl | M /K | Schlüssel | Bemerkungen  |
|----------|------------------|-----------------|---------|------------------|------|-----------|--|
| 1        | patientenfilter  | -               | BOOLEAN | 1                | K    |           | Wird mit Wert „1“ exportiert, wenn der Fall dem Patientenfilter des Moduls entspricht. Andernfalls wird das Attribut nicht exportiert. |

| Lfd. Nr. | Techn. Feldname      | Feldbezeichnung                                 | Feldtyp     | Max. Stellenzahl | M / K | Schlüssel | Bemerkungen  |
|----------|----------------------|---|-------------|------------------|-------|-----------|--|
| 2        | quelle               | Datenquelle des Falles                          | SCHLU ESSEL | 5                | M     | quelle    | Kennzeichen der Datenquelle des Falles   |
| 3        | verordnungsdatum     | Verordnungsdatum des Arznei-/Heil-/Hilfsmittels | DA-TUM      | 10               | M     |           |  |
| 4        | indikationsschlüssel | Indikationsschlüssel                            | INDIKATION  | 4                | K     |           | Indikationsschlüssel oder 9999   |
| 5        | verord_art           | Verordnungsart                                  | SCHLU ESSEL | 2                | K     | verordart | 01=Erstverordnung (Regelfall);<br>02= Folgeverordnung (Regelfall);<br>10=Verordnung außerhalb des Regelfalls |
| 6        | verord_bes           | Verordnungsbesonderheiten                       | SCHLU ESSEL | 1                | K     | verordbes | 1=Zahnarzt/Kieferorthopäde;<br>2=Verordnung im Zusammenhang mit Schwangerschaft oder Entbindung              |

Dieses Element enthält eine nach § 302 abgerechnete Heil- oder Hilfsmittelverordnung. Hierbei wird anhand des Feldes `quelle` zwischen den beiden Verordnungsformen unterschieden.



**[fall\_25a: fall\_25a\_type] – Einladungswesen organisierte Krebsfrüherkennung nach § 25a**

| Lfd. Nr. | Techn. Feldname   | Feldbezeichnung                 | Feldtyp    | Max. Stellenzahl | M /K | Schlüssel     | Bemerkungen  |
|----------|-------------------|---------------------------------|------------|------------------|------|---------------|--|
| 1        | patientenfilter   | -                               | BOOLEAN    | 1                | K    |               | Wird mit Wert „1“ exportiert, wenn der Fall dem Patientenfilter des Moduls entspricht. Andernfalls wird das Attribut nicht exportiert. |
| 2        | programmnummer    | Programmnummer gemäß oKFE-RL    | SCHLUESSEL | 2                | M    | okfe_programm |  |
| 3        | einladungsdatum   | Einladungsdatum gemäß oKFE-RL   | DATUM      | 10               | K    |               |  |
| 4        | widerspruchsdatum | Widerspruchsdatum gemäß oKFE-RL | DATUM      | 10               | K    |               |  |

Dieses Element enthält Daten zum Einladungswesen gemäß oKFE-RL. Für das Feld `einladungsdatum` ist das tatsächliche und nicht das berechnete Einladungsdatum der/des Versicherten zu übermitteln.

**[prozedur : prozedur\_stat\_type] – OPS mit obligatorischem OP-Datum**

| Lfd. Nr. | Techn. Feldname | Feldbezeichnung                   | Feldtyp    | Max. Stellenzahl | M /K | Schlüssel           | Bemerkungen                                      |
|----------|-----------------|-----------------------------------|------------|------------------|------|---------------------|--|
| 1        | ops             | Operation nach OPS                | OPS        | 11               | M    |                     | Prozedurenschlüssel in der gültigen OPS- Version |
| 2        | lokalisierung   | Operation nach OPS – Lokalisation | SCHLUESSEL | 1                | K    | seitenlokalisierung | -  |

| Lfd. Nr. | Techn. Feldname | Feldbezeichnung | Feldtyp | Max. Stellenzahl | M/K | Schlüssel | Bemerkungen |
|----------|-----------------|-----------------|---------|------------------|-----|-----------|-------------|
| 3        | datum           | Operationsdatum | DATUM   | 10               | M   |           | -           |

Prozedur mit Pflichtangabe des OP-Datums zur Verwendung in Abrechnungsfällen nach § 301 und § 301 (AMBO).

**[prozedur : prozedur\_amb\_type] – OPS mit optionalem OP-Datum**

| Lfd. Nr. | Techn. Feldname | Feldbezeichnung                   | Feldtyp     | Max. Stellenzahl | M/K | Schlüssel           | Bemerkungen                                      |
|----------|-----------------|-----------------------------------|-------------|------------------|-----|---------------------|--|
| 1        | ops             | Operation nach OPS                | OPS         | 11               | M   |                     | Prozedurenschlüssel in der gültigen OPS- Version |
| 2        | lokalisierung   | Operation nach OPS – Lokalisation | SCHLU-ESSEL | 1                | K   | seitenlokalisierung | -  |
| 3        | datum           | Operationsdatum                   | DATUM       | 10               | K   |                     | -  |

Prozedur mit optionaler Angabe des OP-Datums zur Verwendung in ambulanten Abrechnungsfällen nach § 295.

**[hauptdiagnose : diagnose\_stat\_type]/[nebendiagnose : diagnose\_stat\_type] – Haupt- bzw. Nebendiagnose nach § 301**

| Lfd. Nr. | Techn. Feldname | Feldbezeichnung              | Feldtyp     | Max. Stellenzahl | M/K | Schlüssel           | Bemerkungen   |
|----------|-----------------|------------------------------|-------------|------------------|-----|---------------------|---|
| 1        | icd             | Primäre Diagnose nach ICD-10 | ICD         | 9                | M   |                     | Behandlungsrelevante Diagnose aus Fachabteilung ,0000‘ bzw. einziger Fachabteilung (siehe TA5 TZ 1.2.7) |
| 2        | lokalisation    | Primäre Diagnose –           | SCHLU-ESSEL | 1                | K   | seitenlokalisierung |   |

| Lfd. Nr. | Techn. Feld-name | Feldbezeichnung                   | Feldtyp     | Max. Stellenzahl | M / K | Schlüssel           | Bemerkungen |
|----------|------------------|-----------------------------------|-------------|------------------|-------|---------------------|-------------|
|          |                  | Lokalisation                      |             |                  |       |                     |             |
| 3        | icd_sek          | Sekundäre Diagnose nach ICD-10    | ICD         | 9                | K     |                     |             |
| 4        | lokalisation_sek | Sekundäre Diagnose – Lokalisation | SCHLU-ESSEL | 1                | K     | seitenlokalisierung |             |

Die Haupt- bzw. Nebendiagnose ist immer mit dem kompletten Paar der primären und sekundären Diagnose zu exportieren, sofern eine sekundäre Diagnose vorliegt. Dies gilt auch, wenn nur eine der beiden Diagnosen den Selektionskriterien entspricht. Die Angabe der Kreuz-Stern-Schlüssel kann im Export entfallen.

**[diagnose : diagnose\_kh\_amb\_type] – Behandlungsdiagnose nach § 301 (AMBO)**

| Lfd. Nr. | Techn. Feld-name | Feldbezeichnung                         | Feldtyp    | Max. Stellenzahl | M / K | Schlüssel           | Bemerkungen |
|----------|------------------|---|------------|------------------|-------|---------------------|-------------|
| 1        | icd              | Primäre Diagnose nach ICD-10            | ICD        | 9                | M     |                     |             |
| 2        | lokalisation     | Primäre Diagnose – Lokalisation         | SCHLUESSEL | 1                | K     | seitenlokalisierung |             |
| 3        | sicherheit       | Primäre Diagnose – Diagnosesicherheit   | SCHLUESSEL | 1                | K     | diagsicherheit      |             |
| 4        | icd_sek          | Sekundäre Diagnose nach ICD-10          | ICD        | 9                | K     |                     |             |
| 5        | lokalisation_sek | Sekundäre Diagnose – Lokalisation       | SCHLUESSEL | 1                | K     | seitenlokalisierung |             |
| 6        | sicherheit_sek   | Sekundäre Diagnose – Diagnosesicherheit | SCHLUESSEL | 1                | K     | diagsicherheit      |             |

| Lfd. Nr. | Techn. Feldname | Feldbezeichnung  | Feldtyp     | Max. Stellenzahl | M / K | Schlüssel      | Bemerkungen              |
|----------|-----------------|------------------|-------------|------------------|-------|----------------|--------------------------|
| 7        | diagnose art    | Art der Diagnose | SCHLU ESSEL | 1                | K     | diagart_khambo | (Behandlung/Überweisung) |

Die Haupt- bzw. Nebendiagnose ist immer mit dem kompletten Paar der primären und sekundären Diagnose zu exportieren, sofern eine sekundäre Diagnose vorliegt. Dies gilt auch, wenn nur eine der beiden Diagnosen den Filterkriterien entspricht. Die Angabe der Kreuz-Stern-Schlüssel kann im Export entfallen.

**[diagnose : diagnose\_amb\_type] – Diagnose nach § 295**

| Lfd. Nr. | Techn. Feldname | Feldbezeichnung      | Feldtyp     | Max. Stellenzahl | M / K | Schlüssel           | Bemerkungen |
|----------|-----------------|----------------------|-------------|------------------|-------|---------------------|-------------|
| 1        | icd             | Diagnose nach ICD-10 | ICD         | 9                | M     |                     |             |
| 2        | lokalisierung   | Lokalisation         | SCHLU ESSEL | 1                | K     | seitenlokalisierung |             |
| 3        | sicherheit      | Diagnosesicherheit   | SCHLU ESSEL | 1                | K     | diagsicherheit      |             |
| 4        | datum           | Datum der Diagnose   | DATUM       | 10               | K     |                     |             |

Diese Diagnose wird bei allen kollektivvertraglichen oder selektivvertraglichen ambulanten Behandlungsfällen nach § 295 verwendet.

**[leistung : leistung\_kh\_amb\_type] – Leistungsabrechnung EBM nach § 301 (AMBO)**

| Lfd. Nr. | Techn. Feldname | Feldbezeichnung        | Feldtyp | Max. Stellenzahl | M/K | Schlüssel | Bemerkungen   |
|----------|-----------------|------------------------|---------|------------------|-----|-----------|---|
| 1        | ebm             | Gebührenordnungsziffer | EBM     | 7                | M   |           | nur selektieren, wenn 2. bis 3. Stelle mit "00" gefüllt sind; nur die 4. bis 7. Stelle des Feldes selektieren |
| 2        | datum           | Behandlungsdatum       | DA TUM  | 10               | M   |           |   |

**[leistung : leistung\_amb\_type] – Leistungsabrechnung EBM nach § 295**

| Lfd. Nr. | Techn. Feldname | Feldbezeichnung                | Feldtyp        | Max. Stellenzahl | M/K | Schlüssel | Bemerkungen                   |
|----------|-----------------|--------------------------------|----------------|------------------|-----|-----------|-------------------------------|
| 1        | ebm             | Gebührenordnungsnummer         | EBM            | 7                | M   |           |                               |
| 2        | datum           | Behandlungsdatum               | DA<br>TUM      | 10               | M   |           |                               |
| 3        | fachgruppe      | Fachgruppe des Arztes, kodiert | FACH<br>GRUPPE | 2                | K   |           | Fachgruppenschlüssel aus LANR |

Falls in einer kollektivvertraglichen Abrechnung kein Datum der Leistungserbringung angegeben ist, muss das Datum der vorhergehenden, gemeinsam abgerechneten Leistung entnommen werden. Gegebenenfalls ist die Liste der Leistungen so lange zu durchlaufen, bis ein Behandlungsdatum gefunden wurde.

**[position : fall300\_position\_type] – Verordnungsposition nach § 300**

| Lfd. Nr. | Techn. Feldname      | Feldbezeichnung  | Feldtyp        | Max. Stellenzahl | M / K | Schlüssel      | Bemerkungen  |
|----------|----------------------|--|----------------|------------------|-------|----------------|--|
| 1        | pzn<br>him<br>sonder | Pharmazentral- oder Hilfsmittelnummer oder Sonderkennzeichen | PZN_<br>HIM    | 8 oder 10        | M     |                | Kennzeichen nach § 4 der Vereinbarung nach § 300 SGB V   |
| 2        | kennzeichen<br>typ   | Kennzeichentyp Arzneimittel oder Sonderkennzeichen           | SCHLU<br>ESSEL | 1                | M     | kennzeichentyp | 1=Hilfsmittelpositionsnummer, 2=PZN, 3=Sonderkennzeichen |
| 3        | anzahl               | Anzahl/Menge der abgegebenen Einheiten                       | ZAHL           | 6                | M     |                |  |

Dieses Element enthält eine Verordnungsposition für eine Arzneimittel- oder Hilfsmittelverordnung nach § 300. Das Attribut `kennzeichentyp` gibt die genaue Art der Verordnungsposition an.

**[pzn\_verwendet] – Verordnungsposition nach § 300**

| Lfd. Nr. | Techn. Feldname | Feldbezeichnung               | Feldtyp | Max. Stellenzahl | M / K | Schlüssel | Bemerkungen               |
|----------|-----------------|-------------------------------|---------|------------------|-------|-----------|---------------------------|
| 1        | pzn_verwendet   | PZN der verwendeten Packungen | PZN     | 8                | M     |           | Liste der verwendeten PZN |

Dieses Element enthält eine für eine Rezeptur verwendete PZN. Es ist für jede verwendete PZN ein eigenes Element zu exportieren. `pzn_verwendet` hat keinen eigenen Datentyp sondern ist im Typ `fall300_position_type` definiert.

**[position : fall302\_position\_type] – Verordnungsposition nach § 302**

| Lfd. Nr. | Techn. Feldname | Feldbezeichnung                        | Feldtyp          | Max. Stellenzahl | M / K | Schlüssel          | Bemerkungen   |
|----------|-----------------|--|------------------|------------------|-------|--------------------|---|
| 1        | abrechnungscod  | Art des Leistungserbringers            | ABRECHNUNGS CODE | 2                | M     |                    |   |
| 2        | leistungsart    | Art der abgegebenen Leistung           | POSGANZE ZAH     | 10               | M     |                    | Heil- (5-stellig) oder Hilfsmittelnummer (10-stellig) |
| 3        | anzahl          | Anzahl/Menge der abgegebenen Einheiten | ZAH              | 6                | M     |                    | max. 2 Nachkommastellen möglich                       |
| 4        | leistungsdatum  | Tag der Abgabe/Leistungserbringung     | DATUM            | 10               | M     |                    |   |
| 5        | anwendungsor    | Spezifikation Anwendungsort            | SCHLUESSEL       | 1                | K     | seitenlokalisation |   |
| 6        | versorgungvon   | Versorgungszeitraum von                | DATUM            | 10               | K     |                    |   |

| Lfd. Nr. | Techn. Feldname | Feldbezeichnung         | Feldtyp | Max. Stellenzahl | M / K | Schlüssel | Bemerkungen |
|----------|-----------------|-------------------------|---------|------------------|-------|-----------|-------------|
| 7        | versorgungbis   | Versorgungszeitraum bis | DATUM   | 10               | K     |           |             |

Die genauen Inhalte dieses Elementes hängen von der im Feld `quelle` angegebenen Datenquelle des übergeordneten Falles vom Typ `fall_302_type` ab.

### 6.2.3 Elemente und Attribute der Nutzdaten in der PID-Datei

Im Folgenden werden die zu exportierenden Elemente im `body`-Bereich der PID-Datei beschrieben.

#### [`body : sqg_body_type`] – Nutzdaten-Element

| Lfd. Nr. | Techn. Feldname | Feldbezeichnung | Feldtyp | Max. Stellenzahl | M / K | Schlüssel | Bemerkungen  |
|----------|-----------------|-----------------|---------|------------------|-------|-----------|--|
| 1        | module          | Modulkennung    | -       |                  | M     |           | Name des Moduls aus der Tabelle <code>Modul</code> |

Dieses Element enthält den gesamten Nutzdatenbereich der Datei. Es enthält als Attribut das Modul `module`. Diese Kennung wird nicht aus den Datenbeständen der Krankenkasse ermittelt, sondern der Tabelle `Modul` in der Spezifikationsdatenbank entnommen.

#### [`case : case_type`] – Datensatz eines Versicherten

| Lfd. Nr. | Techn. Feldname | Feldbezeichnung                             | Feldtyp       | Max. Stellenzahl | M / K | Schlüssel | Bemerkungen |
|----------|-----------------|---|---------------|------------------|-------|-----------|-------------|
| 1        | lfdnr           | Laufende Nummer des Versichertendatensatzes | POSGANZE ZAHN |                  | M     | -         | -           |

Dieses Element enthält genau einen Datensatz, der alle exportierten Daten eines bestimmten Versicherten umfasst.

#### [`patient`] – Container-Element für Patientendaten

Dies ist ein Container-Element ohne eigene Attribute. Es dient dazu, eine zu den aktuellen Datenflüssen analoge XML-Struktur herzustellen und die Verarbeitung bei der Vertrauensstelle zu vereinfachen.

**[pid] – Container-Element für PID-Daten**

Dies ist ein Container-Element ohne eigene Attribute. Es dient dazu, eine zu den aktuellen Datenflüssen analoge XML-Struktur herzustellen und die Verarbeitung bei der Vertrauensstelle zu vereinfachen.

**[VERSICHERTENIDNEU: pid\_data\_type] – Versichertendaten**

| Lfd. Nr. | Techn. Feld-name | Feldbezeichnung    | Feldtyp | Max. Stellenzahl | M/K | Schlüssel | Bemerkungen                                 |
|----------|------------------|--------------------|---------|------------------|-----|-----------|---|
| 1        | V                | Versichertennummer | an      | 10               | M   |           | Lebenslange Versichertennummer (10-stellig) |

Dieses Element enthält die Versichertennummer des Patienten. Die Versichertennummer wird von der Vertrauensstelle durch ein Pseudonym ersetzt.

**6.2.4 Schlüssel****behandart**

| Kode | Bezeichnung        |
|------|--------------------|
| 1    | ambulant (default) |
| 2    | stationär          |

**belegart**

| Kode | Bezeichnung   |
|------|---|
| 0    | keine   |
| 1    | Hauptabteilung  |
| 2    | Hauptabteilung und Beleghebamme                                   |
| 3    | Belegoperator   |
| 4    | Belegoperator und Beleganästhesist                                |
| 5    | Belegoperator und Beleghebamme                                    |
| 6    | Belegoperator, Beleganästhesist und Beleghebamme                  |
| 7    | Teilstationäre Versorgung (für teilstationäre DRG-Fallpauschalen) |
| 8    | Belegarzt mit Honorarvertrag                                      |
| A    | Hauptabteilung  |
| B    | reserviert  |
| C    | Belegoperator   |



| Kode | Bezeichnung  |
|------|--|
| D    | Belegoperator und Beleganästhesist                 |
| E    | reserviert   |
| F    | reserviert   |
| G    | reserviert   |
| H    | Belegarzt mit Honorarvertrag (§ 18 Abs. 3 KHEntgG) |

**bundesland**

| Kode | Bezeichnung            |
|------|------------------------|
| 00   | Ausland                |
| 01   | Schleswig-Holstein     |
| 02   | Hamburg                |
| 03   | Niedersachsen          |
| 04   | Bremen                 |
| 05   | Nordrhein-Westfalen    |
| 06   | Hessen                 |
| 07   | Rheinland-Pfalz        |
| 08   | Baden-Württemberg      |
| 09   | Bayern                 |
| 10   | Saarland               |
| 11   | Berlin                 |
| 12   | Brandenburg            |
| 13   | Mecklenburg-Vorpommern |
| 14   | Sachsen                |
| 15   | Sachsen-Anhalt         |
| 16   | Thüringen              |

**Diagart (wird derzeit nicht verwendet)**

| Kode | Bezeichnung               |
|------|---------------------------|
| AD   | Ambulante Diagnose        |
| BD   | Behandlungsdiagnose       |
| HD   | Hauptdiagnose (stationär) |

| Kode | Bezeichnung               |
|------|---------------------------|
| ND   | Nebendiagnose (stationär) |

**diagart\_khambo**

| Kode | Bezeichnung  |
|------|--|
| 1    | Behandlungsdiagnose                                    |
| 2    | bei § 116b: Diagnose der Überweisung innerhalb der ASV |

**diagsicherheit**

| Kode | Bezeichnung   |
|------|---|
| A    | Ausgeschlossene Diagnose                            |
| G    | Gesicherte Diagnose                                 |
| V    | Verdachtsdiagnose                                   |
| Z    | Symptomloser Zustand nach der betreffenden Diagnose |

**geschlecht**

| Kode | Bezeichnung |
|------|-------------|
| d    | divers      |
| m    | männlich    |
| u    | unbestimmt  |
| w    | weiblich    |

**inanspruch**

| Kode | Bezeichnung   |
|------|---|
| D    | Direkt (nur selektivvertraglich, Wert bei SV = 0)                                       |
| H    | HZV: Abrechnung Verträge nach §§ 73b,c SGB V (nur selektivvertraglich, Wert bei SV = 7) |
| I    | IV: Abrechnung Verträge nach § 140a SGB V (nur selektivvertraglich, Wert bei SV = 8)    |
| K    | Konsiliarbehandlung (Wert bei SV = 5)   |
| M    | Mit-/Weiterbehandlung (Wert bei SV = 6)   |
| N    | Notfallschein (Wert bei SV = 3)   |
| O    | Originalschein (Wert bei SV = 1)  |

| Kode | Bezeichnung                       |
|------|-----------------------------------|
| V    | Vertreterschein (Wert bei SV = 2) |
| Z    | Zielauftrag (Wert bei SV = 4)     |

**Hinweis**

Für die Verwendung der auf Ziffern basierenden Werte aus den Abrechnungsdaten im selektivvertraglichen Bereich ist ein Mapping auf Buchstaben durchzuführen. Die Zuordnungen sind explizit in den Bezeichnungen der einzelnen Schlüsselwerte aufgeführt.

Im kollektivvertraglichen Bereich dürfen die Codes „D“, „H“ und „I“ nicht zur Anwendung kommen, da hierfür keine Entsprechungen im Abrechnungskontext existieren.

**jn**

| Kode | Bezeichnung |
|------|-------------|
| 0    | nein        |
| 1    | ja          |

**kennzeichentyp**

| Kode | Bezeichnung                |
|------|----------------------------|
| 1    | Hilfsmittelpositionsnummer |
| 2    | Pharmazentralnummer        |
| 3    | Sonderkennzeichen          |

**kvregion**

| Kode | Bezeichnung        |
|------|--------------------|
| 01   | Schleswig-Holstein |
| 02   | Hamburg            |
| 03   | Bremen             |
| 17   | Niedersachsen      |
| 20   | Westfalen-Lippe    |
| 38   | Nordrhein          |
| 46   | Hessen             |
| 51   | Rheinland-Pfalz    |
| 52   | Baden-Württemberg  |

| Kode | Bezeichnung            |
|------|------------------------|
| 71   | Bayern                 |
| 72   | Berlin                 |
| 73   | Saarland               |
| 74   | KBV                    |
| 78   | Mecklenburg-Vorpommern |
| 83   | Brandenburg            |
| 88   | Sachsen-Anhalt         |
| 93   | Thüringen              |
| 98   | Sachsen                |
| 00   | Platzhalter            |
| 26   | Platzhalter            |
| 35   | Platzhalter            |
| 75   | Platzhalter            |

**le\_kennung**

| Kode | Bezeichnung             |
|------|-------------------------|
| bsnr | Betriebsstättennummer   |
| iknr | Institutionskennzeichen |

**Leistungserbringerart (wird derzeit nicht verwendet)**

| Kode | Bezeichnung   |
|------|---|
| 11   | Apotheke (mit gesonderter Zulassung nach § 126 SGB V) |
| 12   | Augenoptiker  |
| 13   | Augenarzt   |
| 14   | Hörgeräteakustiker                                    |
| 15   | Orthopädiemechaniker, Bandagist, Sanitätshaus         |
| 16   | Orthopädieschuhmacher                                 |
| 17   | Orthopäde   |
| 18   | Sanitätshaus  |
| 19   | sonstiger Hilfsmittellieferant                        |
| 21   | Masseur/Medizinischer Badebetrieb                     |

| Kode | Bezeichnung   |
|------|---|
| 22   | Krankengymnast/Physiotherapeut  |
| 23   | Logopäde, Atem-, Sprech- und Stimmlehrer, staatl. anerkannter Sprachtherapeut                           |
| 24   | Sprachheilpädagoge, Dipl. Pädagoge  |
| 25   | Sonstiger Sprachtherapeut   |
| 26   | Ergotherapeut   |
| 27   | Krankenhaus   |
| 28   | Kurbetrieb  |
| 29   | Sonstige therapeutische Heilperson  |
| 31   | freigemeinnützige Anbieter (Sozialstation)  |
| 32   | privatgewerbliche Anbieter  |
| 33   | öffentliche Anbieter  |
| 34   | Sonstige Pflegedienste  |
| 41   | Öffentlicher Anbieter von qualifizierten Krankentransportleistungen (z. B. Feuerwehr)                   |
| 42   | Deutsches Rotes Kreuz (DRK)   |
| 43   | Arbeiter-Samariter-Bund (ASB)   |
| 44   | Johanniter-Unfall-Hilfe (JUH)   |
| 45   | Malteser-Hilfsdienst (MHD)  |
| 46   | Sonstiger Leistungserbringer von nichtqualifizierten Krankentransportleistungen (z. B. Taxi/Mietwagen)  |
| 47   | Leistungserbringer von Flugrettungs- und Transportleistungen  |
| 48   | Sonstiger nichtöffentlicher Anbieter von qualifizierten Krankentransport- bzw. Rettungsdienstleistungen |
| 49   | Sonstiger Anbieter von Krankentransportleistungen (z. B. Bergwacht, Wasserschutz etc.)                  |
| 50   | Hebamme/Entbindungspfleger  |
| 55   | Sonstiger Leistungserbringer von nichtärztlichen Dialysesachleistungen                                  |
| 56   | Kuratorium für Heimdialyse (KfH)  |
| 57   | Patienten-Heimversorgung (PHV)  |
| 60   | Betriebshilfe   |
| 61   | Leistungserbringer von Rehabilitationssport   |

| Kode | Bezeichnung  |
|------|--|
| 62   | Leistungserbringer von Funktionstraining   |
| 63   | Leistungserbringer für ergänzende Rehabilitationsmaßnahmen   |
| 64   | - nicht besetzt -  |
| 65   | Sonstiger Leistungserbringer   |
| 66   | Leistungserbringer von Präventions- und Gesundheitsförderungsmaßnahmen im Rahmen von ambulanten Vorsorgeleistungen |
| 67   | Ambulantes Rehasentrum / Mobile Rehabilitationseinrichtung   |
| 68   | Sozialpädiatrische Zentren/Frühförderstellen   |
| 69   | Soziotherapeutischer Leistungserbringer  |
| 71   | Podologen  |
| 72   | Med. Fußpfleger (gemäß § 10 Abs. 4 bis 6 PodG)   |
| 75   | Spezialisierte ambulante Palliativversorgung (SAPV)  |

**modulkennung**

| Kode  | Bezeichnung   |
|-------|---|
| AUFST | Aufstellung nach § 16 Abs. 5 DeQS-RL  |
| BSP   | Beispiel-Modul (dient nur zur Erläuterung der Spezifikation)  |
| CHOL  | Cholezystektomie  |
| DIALS | Nierenersatztherapie bei chronischem Nierenversagen einschließlich Pan-<br>kreastransplantation - Dialyse (Sozialdaten) |
| KCHK  | Koronarchirurgie und Eingriffe an Herzklappen (Sozialdaten)   |
| NTXS  | Nierenersatztherapie bei chronischem Nierenversagen einschließlich Pan-<br>kreastransplantation - Dialyse (Sozialdaten) |
| PCI   | Perkutane Koronarintervention und Koronarangiographie   |
| NWITR | Vermeidung nosokomialer Wundinfektionen - Tracer  |
| NWIWI | Vermeidung nosokomialer Wundinfektionen - Wundinfektionen   |
| KFEDK | Programmbeurteilung zur organisierten Krebsfrüherkennung von Darmkrebs  |
| KFEZK | Programmbeurteilung zur organisierten Krebsfrüherkennung von Zervixkarzi-<br>nomen                                      |
| HTXS  | Transplantationsmedizin - Herztransplantation (Sozialdaten)   |
| LUTXS | Transplantationsmedizin - Lungen- und Herz-Lungentransplantation (Sozialda-<br>ten)                                     |

| Kode | Bezeichnung  |
|------|--|
| LTXS | Transplantationsmedizin - Lebertransplantation (Sozialdaten) |

**okfe\_programm**

| Kode | Bezeichnung                                     |
|------|---|
| DK   | Programm zur Früherkennung von Darmkrebs        |
| ZK   | Programm zur Früherkennung von Zervixkarzinomen |

**quelle**

| Kode     | Bezeichnung  |
|----------|--|
| 115b     | Ambulantes Operieren KH                                  |
| 116b     | Ambulante Spezialfachärztliche Versorgung                |
| 117      | Hochschulambulanz  |
| 118      | Psychiatrische Institutsambulanz                         |
| 119      | Sozialpädiatrisches Zentrum                              |
| 25a      | Einladungsstellen der Krankenkassen                      |
| 295k     | Vertragsärzte (kollektivvertraglich)                     |
| 295_140  | Vertragsärzte (Integrierte Versorgungsformen)            |
| 295_73b  | Vertragsärzte (Hausarztzentrierte Versorgung)            |
| 295_73c  | Vertragsärzte (Besondere ambulante ärztliche Versorgung) |
| 300      | Apotheken  |
| 301      | Krankenhäuser  |
| 302_heil | Heilmittelverordnung                                     |
| 302_hilf | Hilfsmittelverordnung                                    |

**seitenlokalisation**

| Kode | Bezeichnung |
|------|-------------|
| B    | beidseitig  |
| L    | links       |
| R    | rechts      |

**verordart**

| Kode | Bezeichnung                         |
|------|-------------------------------------|
| 01   | Erstverordnung (Regelfall)          |
| 02   | Folgeverordnung (Regelfall)         |
| 10   | Verordnung außerhalb des Regelfalls |

**verordbes**

| Kode | Bezeichnung  |
|------|--|
| 1    | Zahnarzt/Kieferorthopäde                                       |
| 2    | Verordnung im Zusammenhang mit Schwangerschaft oder Entbindung |

**6.3 Spezifikationsdatenbank**

Die Spezifikationsdatenbank ist eine zentrale Komponente der Spezifikation. In der Datenbank werden alle Informationen in Bezug auf die Datenselektion, das Mapping von den Quelldaten auf die Exportdaten sowie die Struktur der Exportdaten verankert.

Die Spezifikationsdatenbank soll dabei ein beständiges maschinenlesbares Format bieten, so dass entsprechende Parser auf der gleichen Struktur die sich dynamisch ändernden fachlichen Spezifikationen immer auslesen können.

Weitere Spezifikationskomponenten werden aus dieser Datenbank ausgelesen bzw. abgeleitet. Dazu gehören auch die spezifischen QS-Datentypen, die im XML-Schema Verwendung finden.

**Bereiche in der Spezifikationsdatenbank**

In der Spezifikationsdatenbank werden die folgenden Bereiche unterschieden:

- Tabellen
- Abfragen

Einzelne Tabellen bzw. Abfragen, die für den Anwender von Bedeutung sind, werden in den entsprechenden Abschnitten im Kapitel 2 beschrieben. Daher werden die einzelnen Tabellen und Abfragen im Folgenden nur grob umrissen.

**Tabellen**

Die Inhalte der Spezifikationsdatenbank sind in Tabellen abgelegt. Alle Tabellen sind auch im Verzeichnis /1+2 ACCESS/Export aus DB/Tabellen/ als Excel-Dateien abgelegt.

*Tabelle 35: Tabellen in der Spezifikationsdatenbank*

| Tabelle  | Bedeutung                                  |
|----------|--|
| Attribut | Liste der verwendeten Exportattribute      |
| BasisTyp | Datentypen der verwendeten Exportattribute |



| <b>Tabelle</b>         | <b>Bedeutung</b>   |
|------------------------|--|
| Datei                  | Label „Exportdatei“  |
| DatenfeldAttribut      | Zuordnung der Eingangsdaten zu den Exportattributen  |
| DatenfeldVerwendung    | Angabe der Verwendung der Eingangsdaten je Modul   |
| Datenquelle            | Label „Datenquelle nach Abrechnungskontext“  |
| Einrichtung            | Label „Einrichtung im Datenfluss“  |
| Element                | Liste der Exportelemente   |
| ElementDatentyp        | Angabe der modulabhängigen XML-Elementtypen der Exportelemente                             |
| ElementRelation        | Hierarchie der Exportelemente je Modul   |
| Fehlermeldung          | Liste der Fehlermeldungen  |
| FilterListe            | Kodelisten zur Verwendung in Selektionsfiltern   |
| FilterListeModul       | Relationentabelle zur Verknüpfung von Filterlisten mit mehreren Modulen                    |
| FilterListeWert        | Inhalt der Kodelisten aus der Tabelle „FilterListe“  |
| Funktion               | Funktionen zur Ableitung von Werten aus Datenfeldern                                       |
| Lieferfrist            | Zeiträume für Datenlieferungen   |
| Modul                  | Angaben zu den Modulen   |
| ModulErfassungsjahr    | Erweiterte Angaben zu den Modulen je Erfassungsjahr  |
| MussKann               | Label „Muss-/Kann-Feld“  |
| Pruefbereich           | Label „Prüfbereiche für Prüfregeln“  |
| Pruefung               | Liste der Fehlerprüfungen  |
| Regel                  | Beschreibung der Plausibilitätsregeln  |
| RegelDatenfeldAttribut | Zuordnung der verwendeten Datenfelder zu den Plausibilitätsregeln                          |
| RegelDatenquelle       | Angabe zu Plausibilitätsregeln, die abhängig von verwendeten Datenquellen verwendet werden |
| RegelModul             | Angabe zu Plausibilitätsregeln, die abhängig von bestimmten Modulen verwendet werden       |
| RelationTyp            | Label „Kardinalität von Element-Relationen“  |
| Schluessel             | Schlüssel zur Verwendung in Attributwerten   |
| SchluesselWert         | Schlüsselwerte zur Tabelle „Schluessel“  |
| Strenge                | Label „Strenge der Prüfregel“  |

| <b>Tabelle</b>                | <b>Bedeutung</b>  |
|-------------------------------|---|
| Verarbeitungsschritt          | Liste von Verarbeitungsschritten für Prüfungen                      |
| Verarbeitungsschritt-Pruefung | Zugehörigkeit der Prüfungen zu den Verarbeitungsschritten           |
| Version                       | Version der Spezifikationsdatenbank                                 |
| VersStatus                    | Label „Versionsstatus“  |
| Wertebereich                  | Wertebereiche (z. B. Zeiträume) zur Verwendung in Selektionsfiltern |

### Abfragen

Abfragen sind Sichten auf Ausschnitte bzw. Verknüpfungen von Tabellen. Alle Abfragen sind auch im Verzeichnis /1+2 ACCESS/Export aus DB/Abfragen/ als Excel-Dateien abgelegt.

Tabelle 36: Abfragen in der Spezifikationsdatenbank

| <b>Abfrage</b>           | <b>Bedeutung</b>                               |
|--------------------------|--|
| AbfrageFeldPlausibilität | Hilfstabelle zur internen Verwendung           |
| AbfragePlausibilität     | Hilfstabelle zur internen Verwendung           |
| Codeliste                | Kodelisten für Selektionsfilter                |
| FeldDefinition           | Formatanweisungen der Datenfelder              |
| KennungAttribut          | Hilfstabelle zur internen Verwendung           |
| KennungElement           | Hilfstabelle zur internen Verwendung           |
| KennungVorbereitung[_nn] | Hilfstabellen zur internen Verwendung          |
| MappingKomplett          | Hilfstabelle für Referenz- und Mappingtabellen |
| MappingTabellePID        | Mapping-Tabelle für die PID-Datei              |
| MappingTabelleQS         | Mapping-Tabelle für die QS-Datei               |
| ModulDatenquelle         | Hilfstabelle zur internen Verwendung           |
| ParentChild              | Hilfstabelle zur internen Verwendung           |
| PruefschrittPruefung     | Übersicht der Prüfschritte                     |
| Referenz                 | Referenz-Tabelle auf Eingangsdaten             |
| RegelDurchDatenquelle    | Hilfstabelle zur internen Verwendung           |
| RegelDurchModul          | Hilfstabelle zur internen Verwendung           |

| Abfrage             | Bedeutung  |
|---------------------|--|
| KennungVorbereitung | Hilfstabelle zur internen Verwendung                           |
| RegelDatenfeld      | Übersicht über Plausibilitätsregeln und verwendete Datenfelder |

## 6.4 XML-Schemata

Das XML-Schema hat die Aufgabe, das aktuelle Datenflussmodell der G-BA-Richtlinie abzubilden:

- Abbildung der zu exportierenden Sozialdaten bei den Krankenkassen
- Abbildung der Schnittstellen an den Institutionen-Übergängen (z. B. DAS-VST)
- Abbildung der vorzunehmenden Datentransformation in den jeweiligen Einrichtungen des Datenflusses (z. B. LE-Pseudonymisierung bei der DAS)
- Abbildung der Rückprotokollierung

Aus diesem Grund gibt es bei der Erstellung von Schemata, welche die Konformität von Richtlinie und Datenschutz sicherstellen sollen, kein „Allround-Schema“, welches alle Anforderungen an alle Beteiligten abdeckt, sondern eine „Schema-Familie“, aus der heraus gezielt für jede Schnittstelle („Interface“) eine passende Datenstruktur definiert wird.

### 6.4.1 Kompositionsmodell

Bei der Schema-Erstellung wurde auf ein Kompositionsmodell zurückgegriffen, in dem sich alle Teilschemata am Ende einen Namensraum teilen, um diese Schema-Familie besser warten zu können und gleiche Teilstrukturen nur einmal definieren zu müssen. Als Bezeichnung des Namensraums wurde „urn:gba:sqg“ gewählt. Zu diesem Namensraum werden die Bausteine je nach Bedarf über „includes“ zusammengestellt.

|                         |                               |
|-------------------------|-------------------------------|
| abstract                | Teilschemata                  |
| imports                 | W3C-Encryption                |
| interface_BAS           | Schnittstelle BAS             |
| interface_DAS           | Schnittstelle DAS             |
| interface_DAS_BAS       | Schnittstelle DAS/BAS         |
| interface_DAS_VST       | Schnittstelle BAS/VST         |
| interface_KK            | Schnittstelle KK              |
| interface_kk_DAS        | Schnittstelle VST/BAS         |
| interface_VST_BAS       | XML-Beispiele                 |
| sqg_admin_datatypes.xsd | Administrative Datentypen     |
| sqg_datatypes.xsd       | Verfahrensbezogene Datentypen |

Abbildung 32: Dateiordner der Schnittstellen-Schemata

Das Kompositionsmodell macht es möglich, Konzepte aus der objektorientierten Programmierung – darunter fallen die Konzepte abstrakter Typ, Ersetzbarkeit von Typen, Wiederverwendung und Polymorphismus – zu nutzen. Dadurch können Schemata erstellt oder genutzt werden, die generische Grundtypen definieren und diese Typen so erweitern, dass sie schnittstellenspezifisch sind, ohne das ursprüngliche Schema zu beeinflussen.

Dieses Kompositionsmodell soll hier näher erläutert werden.

#### Beispiel:

Die krankenkassenidentifizierenden Daten existieren gemäß Datenflussmodell der DeQS-RL in drei Ausprägungen:

- im Klartext (Schnittstelle KK)
- pseudonymisiert (Schnittstelle DAS-KK)
- pseudonymisiert und verschlüsselt (Schnittstelle DAS-VST)

Bei dieser Konzeption werden alle drei Ausprägungen vom selben Basisdatentyp geerbt, in einem zweiten Schritt die drei Ausprägungen konkretisiert und angepasst, und über „includes“ in die jeweilige Schnittstelle integriert.

#### 6.4.2 Schnittstellen

In der folgenden Tabelle werden Schema-Dateien aufgeführt, die im Rahmen der Übermittlung von Sozialdaten Verwendung finden. Andere Dateien haben zwar ebenfalls die Dateiendung .xsd, sind aber keine vollständigen Schemata, sondern Bausteine für Schnittstellen.

Tabelle 37: Verwendbare Schemata, Ablageort und Verwendungszweck

| Schnittstelle <sup>12</sup> /<br>Ablageort | Schema                       | Verwendungszweck  |
|--|------------------------------|---|
| Krankenkasse<br>Interface_KK               | interface_KK_QS.xsd          | Validierung der QS-Datei  |
|  | interface_KK_PID.xsd         | Validierung der PID-Datei   |
| KK/DAS-KK<br>Interface_KK_DAS              | response_receipt_DAS_KK.xsd  | Empfangsbestätigung der DAS-KK  |
|  | response_DAS_KK.xsd          | Datenflussprotokoll der DAS-KK bzw. Datenübermittlung krankenkassenspezifischer Rückmeldungen |
| DAS-KK<br>Interface_DAS                    | interface_DAS.xsd            | Zusammenführung der QS- und PID-Dateien   |
|  |                              | Pseudonymisierung der LE- und KK-Daten  |
| DAS-KK/VST<br>Interface_DAS_VST            | interface_DAS_VST.xsd        | Datenübermittlung von der DAS-KK an die VST   |
|  | response_VST_DAS.xsd         | Datenflussprotokoll von der VST an die DAS-KK   |
|  | response_receipt_VST_DAS.xsd | Empfangsbestätigung von der VST an die DAS-KK   |
| DAS-KK/BAS<br>Interface_DAS_BAS            | interface_DAS_BAS.xsd        | Datenübermittlung von der DAS-KK an die BAS (Nicht-PID-Verfahren)                             |
|  | response_BAS_DAS.xsd         | Datenflussprotokoll der BAS (Nicht-PID-Verfahren) bzw. Datenübermittlung                      |


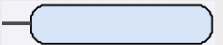

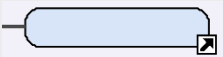
<sup>12</sup> Die Zielgruppe ist aus der Schnittstelle zu entnehmen.





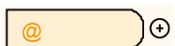
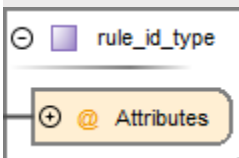
| Schnittstelle <sup>12</sup> /<br>Ablageort | Schema                    | Verwendungszweck  |
|--|---------------------------|---|
|  |                           | krankenkassenspezifischer Rückmeldungen   |
| VST/BAS<br>Interface_VST_BAS               | interface_VST_BAS.xsd     | Datenübermittlung von der VST an die BAS  |
|  | response_BAS_VST.xsd      | Datenflussprotokoll von der BAS an die VST  |
|  | response_receipt_BAS_.xsd | Empfangsbestätigung von der BAS an die VST  |
| BAS  | interface_BAS.xsd         | Datenstruktur der Datenübermittlung nach der Entschlüsselung der QS- und KK-Daten   |
|  | interface_BAS_case.xsd    | Modifiziertes Unterschema für Case-Elemente in interface_BAS.xsd, für die Datenentgegennahme in der BAS                       |
|  | interface_BAS_PB.xsd      | Datenstruktur der Datenübermittlung für die Dateiablage entschlüsselter Daten mit Einschränkung auf die Module gemäß oKFE-RL. |
|  | interface_BAS_QS.xsd      | Datenstruktur der Datenübermittlung für die Dateiablage entschlüsselter Daten mit Einschränkung auf die Module gemäß DeQS-RL. |

#### 6.4.3 XML-Struktur des Datencontainers (KK/DAS-KK-Schnittstelle)

Zur Veranschaulichung der verwendeten XML-Schemata werden Diagramme verwendet, deren Symbole in der folgenden Tabelle kurz dargestellt und erläutert werden.

Tabelle 38: Symbole in den XML-Schema-Diagrammen

| Symbol  | Beschreibung   |
|---|--|
|  | <b>Optionales Element</b><br>Kardinalität 0..1 („0 oder 1“)  |
|  | <b>Obligatorisches Element</b><br>Kardinalität 1: das Element muss genau einmal vorkommen  |
|  | <b>Mehrfach wiederholbares Element</b><br>Kardinalität: die erlaubte Anzahl der Elemente wird unter dem Symbol dargestellt (Beispiel: 1..n, n..m). |
|  | <b>Referenzelement</b>   |

|   |   |
|---|---|
|   | Das referenzierte globale Element ist an anderer Stelle im Schema definiert.  |
|  | <b>Eine Folge von Elementen</b><br>Die Elemente müssen genau in der Reihenfolge vorkommen, in der sie im Schemadiagramm angezeigt sind. |
|  | <b>Eine Auswahl von Elementen</b><br>Nur ein einziges Element aus der Liste kann ausgewählt werden.                                     |
|  | Ein Element mit Kind-Elementen  |
|  | <b>Optionales Attribut</b><br>Kardinalität 0..1 („0 oder 1“)  |
|  | <b>Obligatorisches Attribut</b><br>Kardinalität 1: das Attribut muss genau einmal vorkommen   |
|  | <b>Komplexer Datentyp</b><br>Der komplexe Datentyp wird mit einem Rahmen mit einem weißen bzw. lilafarbenen Hintergrund angezeigt.      |

Die wesentlichen Bestandteile der XML-Schemata werden in tabellarischer Form dargestellt. Die Darstellung umfasst folgende Eigenschaften des betrachteten Elements:

- Grafische Abbildung der Kind-Elemente und -Attribute
- Auflistung der Kind-Elemente
- Auflistung der Attribute sowie ihre Eigenschaften wie:
  - Name
  - XML-Datentyp (technische Bezeichnung: „Type“)
  - Muss-Kann-Feld (technische Bezeichnung: „Use“)
  - Konstante (technische Bezeichnung: „Fixed“)
  - Kurze Beschreibung (technische Bezeichnung: „Annotation“)

Grundsätzlich beginnt jede XML-Exportdatei mit einer Headerzeile (`<?xml version="1.0" encoding="iso-8859-1"?>`), gefolgt vom Wurzelement `<root>`, das den gesamten Inhalt einschließt. Als Zeichensatz wird „iso-8859-1“ von den Krankenkassen verwendet. Erst in der DAS müssen die unverschlüsselten Daten in „UTF-8“ (Unicode) umcodiert werden.

### Wurzelement `<root>`

Das Wurzelement ist eine Art Umschlag für alle XML-Typen in den Verfahren. Das Wurzelement besteht immer aus zwei Kind-Elementen (Zweigen) `<header>` und `<body>`.

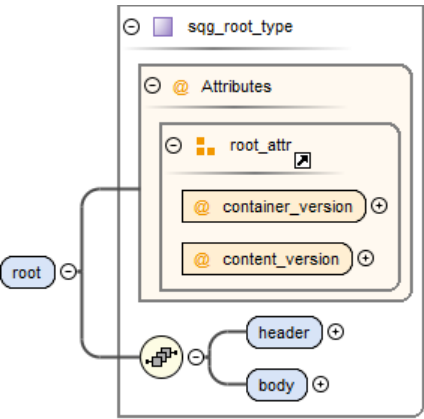


Abbildung 33: root-Element und Kind-Elemente header und body

Das root-Element hat zusätzlich zwei Attribute:

Tabelle 39: Attribute: root-Element

| Attribut          | Type                 | Use      | Fixed | Beschreibung  |
|-------------------|----------------------|----------|-------|---|
| container_version | xs:string            | required | 2.0   | Ist ein fixer Wert und definiert die aktuell gültige Versionsnummer des Containers. Die Versionsnummer wird erhöht, wenn Änderungen am Schema des Containers (Umschlags) gemacht werden. Bei optionalen Änderungen wird die Versionsnummer beibehalten, um die Aufwärtskompatibilität zu gewährleisten.<br><br>Ein XML-Dokument, das einen alten Wert dieses Attributs enthält, muss von der DAS zurückgewiesen werden. |
| content_version   | content_version_type | required |       | Ist ein fixer Wert und definiert die aktuell gültige Versionsnummer des Inhalts.<br><br>Die Versionsnummer wird nur erhöht, wenn das Schema der QS-Daten oder PID unabhängig von der zugrunde liegenden Spezifikation geändert wird.  |

**Header-Bereich <header>**

Das Element header besteht aus Metadaten (administrativen und meldebezogenen Daten) zu den QS-Daten, die im <body> enthalten sind.

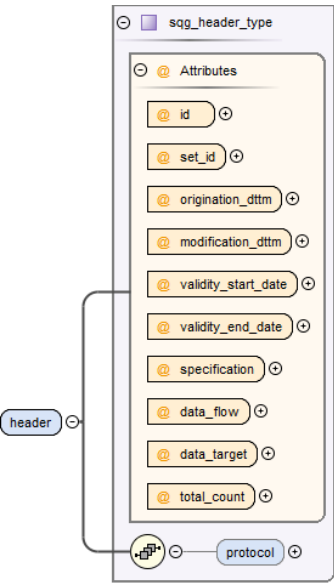


Abbildung 34: Header-Element und Kind-Element protocol

Das Element hat folgende Attribute:

Tabelle 40: Attribute: *header*-Element

| Attribut | Type             | Use      | Beschreibung  |
|----------|------------------|----------|---|
| id       | guid_simple_type | required | Eindeutige ID eines Segmentes, wird vom Dokumentenersteller als GUID <sup>13</sup> erzeugt. Nach Erstellung nicht mehr modifizierbar. Bei PID-Modulen kann diese Angabe in der VST verwendet werden, um die Zusammenführung der in getrennten Dateien an die DAS-KK übermittelten QS- und PID-Daten zu sichern. Diese id wird somit von den beiden Dokumenten eines Segmentes geteilt. Segmentübergreifend muss diese id jedoch unterschiedlich sein. |
| set_id   | guid_simple_type | required | Eindeutige ID für sämtliche Dokumente, einer Transaktion; wird von der KK erzeugt. Nach Erstellung nicht mehr modifizierbar. Bei segmentierten Transaktionen muss die set_id in sämtlichen Segmenten identisch sein. damit diese in der   |

<sup>13</sup> Ein Globally Unique Identifier oder kurz GUID ist eine global eindeutige Zahl mit 128 Bit (16 Bytes).



| Attribut            | Type                      | Use      | Beschreibung   |
|---------------------|---------------------------|----------|--|
|                     |                           |          | BAS zu einer Transaktion zusammengesetzt werden können. Für die Erstellung von transaktionsbezogenen (und somit segmentübergreifenden) Datenflussprotokollen für die KK wird von der BAS diese set_id verwendet.   |
| origination_dttm    | smalldatetime_simple_type | required | <p>Dieses Element ist der Zeitstempel der ursprünglichen Dokumenterzeugung.</p> <p>Es darf nach seiner Erstellung nicht mehr modifiziert werden.</p> <p>Format: CCYY-MM-DDThh:mm:ss</p>  |
| modification_dttm   | smalldatetime_simple_type | required | <p>Dieses Element ist ein Zeitstempel und muss bei jeder Modifikation des Dokuments aktualisiert werden.</p> <p>Das Modifikationsdatum darf nicht vor dem Erstelldatum liegen.</p> <p>Format: CCYY-MM-DDThh:mm:ss</p>  |
| validity_start_date | datum_en_type             | required | <p>Beginn des Bezugszeitraums der übermittelten Daten (gemäß Vorgabe der jeweiligen verfahrensbezogenen Spezifikation).</p> <p>Für die Übermittlung einer Aufstellung (vgl. 2.6 Bestätigung und Aufstellung) ist der 1.1. jenes Jahres einzutragen, für welches die Aufstellung erfolgt.</p> |
| validity_end_date   | datum_en_type             | required | <p>Ende des Bezugszeitraums der übermittelten Daten (gemäß Vorgabe der jeweiligen verfahrensbezogenen Spezifikation).</p> <p>Für die Übermittlung einer Aufstellung (vgl. 2.6 Bestätigung und Aufstellung) ist der 31.12. jenes Jahres einzutragen, für welches die Aufstellung erfolgt.</p> |

| Attribut      | Type                      | Use      | Beschreibung   |
|---------------|---------------------------|----------|--|
| specification | enum_spez_type            | required | Angabe der zum Export eingesetzten Version der Spezifikation. Hieraus ergeben sich insbesondere die eingesetzten Filterkriterien und Kodelisten.   |
| data_flow     | enum_flow_type            | required | <p>Dieses Attribut gibt an, für welchen Datenfluss (DAS) dieses Dokument erzeugt wurde. Für die Übermittlung der Sozialdaten an die DAS-KK ist der Wert „Sozialdaten“ einzutragen.</p> <p>Für die Übermittlung einer Aufstellung (vgl. 2.6 Bestätigung und Aufstellung) ist der Wert „Aufstellung“ einzutragen.</p> <p>Für die Übermittlung krankenkassenspezifischer Rückmeldeberichte (vgl. 2.7 Informationen über die Güte gelieferter Daten) ist der Wert „Information“ einzutragen.</p> |
| data_target   | enum_target_type          | required | <p>Dieses Attribut gibt an, welches Ziel der Datenfluss hat. Mögliche Werte sind:</p> <ul style="list-style-type: none"> <li>▪ Echtdatenpool: Bei Echtdaten für den Echtbetrieb (Regelbetrieb)</li> <li>▪ Probedatenpool: Bei Echtdaten für vorläufige Auswertungen</li> <li>▪ Testdatenpool: Bei Testdaten für Testzwecke</li> </ul> <p>Die Kennzeichnung „Echtdatenpool“ ist dabei ausschließlich für den Echtbetrieb vorgesehen.</p>  |
| total_count   | non_negative_integer_type | required | Gesamtzahl der exportierten case-Elemente in allen XML-Dateien einer Transaktion.  |

Die Attribute „*origination\_dttm*“ und „*modification\_dttm*“ sind vom Datentyp „*smallDateTime*“, der einen Zeitpunkt darstellt (ISO 8601). Es handelt sich um das Format CCYY-MM-DDThh:mm:ss, dabei steht

1. „CC“ für das Jahrhundert,
2. „YY“ für das Jahr,
3. „MM“ für den Monat und
4. „DD“ für den Tag.
5. Der Buchstabe „T“ dient als Trennzeichen zwischen Datum und Zeit,
6. „hh“, „mm“ und „ss“ repräsentieren jeweils Stunden, Minuten und Sekunden.

**Beispiel:**

- 2011-11-01T21:32:52

**Element header/protocol**

Das Element *protocol* wird im Abschnitt 6.4.7 näher erläutert.

**Body-Bereich <body>**

In dem Element <body> liegen die eigentlichen Nutzdaten:

- Patientenidentifizierende - bzw. QS-Daten (Stamm- und Behandlungsdaten)
- leistungserbringeridentifizierende Daten
- krankenkassenidentifizierende Daten

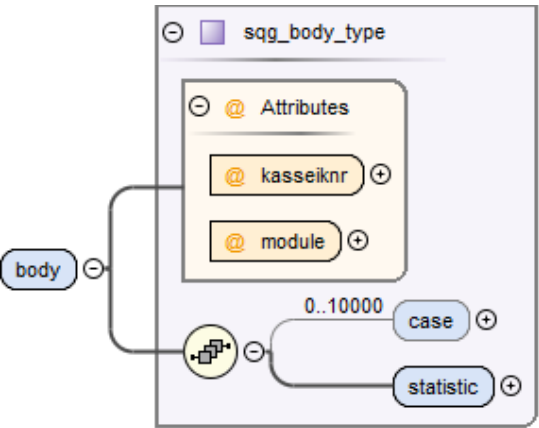


Abbildung 35: *body*-Element und Kind-Elemente *case* und *statistic*

Das *body*-Element hat folgende Attribute:

Tabelle 41: Attribute: *body*-Element

| Attribut | Type | Use | Beschreibung |
|----------|------|-----|--------------|
|----------|------|-----|--------------|

|                  |                        |          |  |
|------------------|------------------------|----------|--|
| kasseiknr        | kasse_iknr_simple_type | required | Das Hauptinstitutionskennzeichen der Krankenkasse  |
| module           | enum_modul_type        | required | Enthält das Modul, aus dem die Datensätze stammen  |
| versichertenzahl | xs:positiveInteger     | optional | Enthält die aktuellste Anzahl der Versicherten der liefernden Krankenkasse gemäß KM1-Statistik   |
| dienstleister    | xs:string              | optional | Handelsregisternummer des Dienstleisters, der für die Datenselektion und -lieferung von Sozialdaten bei den Krankenkassen beauftragt wurde.<br><br>Sofern eine Krankenkasse keinen Dienstleister mit der Auslösung bzw. dem Export beauftragt hat, kann das Feld frei bleiben (siehe Abschnitt 2.2.3). |

Element body/case

Das Element <case> entspricht einem Treffer des Patientenfilters und enthält genau einen QS-Datensatz.

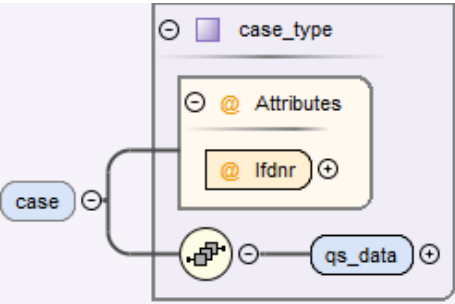


Abbildung 36: case-Element und Kind-Element qs\_data

Das case-Element hat folgende Attribute:

Tabelle 42: Attribute: case-Element

| Attribut | Type                | Use      | Beschreibung  |
|----------|---------------------|----------|---|
| lfdnr    | case_id_simple_type | required | Dateiübergreifende laufende Nummer des Datensatzes und muss eindeutig auf Ebene der gesamten Datenlieferung sein. |

### Element body/statistic

Das Element dient dazu, Statistiken über die Datenlieferung des Absenders und über deren Verarbeitung durch die DAS aufzunehmen. Das Element <statistic> ist nach der Prüfung bei der DAS Teil des Rückprotokolls.

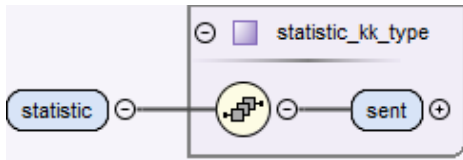


Abbildung 37: Element statistic und Kind-Element sent

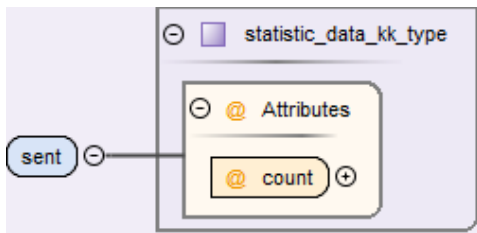


Abbildung 38: Element sent

Tabelle 43: Attribute: sent-Element

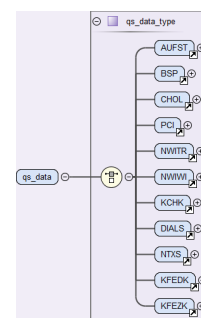
| Attribut     | Type                      | Use      | Beschreibung  |
|--------------|---------------------------|----------|---|
| <u>count</u> | non_negative_integer_type | required | Gesamtzahl von Datensätzen in der XML-Datei, ist vom Datenlieferant auszufüllen |

### 6.4.4 XML-Struktur der QS-Daten

#### Element body/case/qs\_data

Das Element <qs\_data> ist ein Container für die verfahrensspezifischen QS-Daten.

Dieser Datencontainer darf nur ein einziges Verfahren enthalten. Das Verfahren enthält einen weiteren Datencontainer (BSP, PCI usw.)



enthalten.

Abbildung 39: Datentyp qs\_data\_type

Die Inhalte der einzelnen Verfahren können aus der Spezifikationsdatenbank bzw. ihrer begleitenden Dokumentation entnommen werden. Im Folgenden wird das Beispielfahrer (BSP<sup>14</sup>) für die Visualisierung der XML-Struktur verwendet.

Ein Verfahren (hier BSP) wird aus bereits verfahrensübergreifenden, datenbestandsbezogenen (z. B. § 301, § 300 usw.) Datenstrukturen zusammengestellt. Folgende Tabelle zeigt an, welche datenbestandsbezogenen Datensätze bereits definiert wurden:

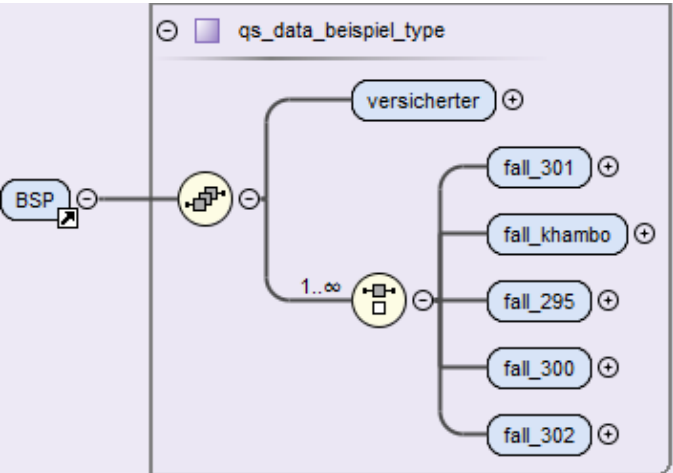


Abbildung 40: Element BSP und Kind-Elemente versicherter, fall\_301, fall\_khambo, fall\_295, fall\_300 und fall\_302

Tabelle 44: Elemente: Beispielfahrer

| Element      | Beschreibung   |
|--------------|--|
| versicherter | Das Element nimmt Versichertendaten (wie Alter und Geschlecht) auf.  |
| fall_301     | § 301  |
| fall_khambo  | § 115b, § 116b, § 117, § 118, § 119  |
| fall_295     | § 295 (kollektivvertraglich), § 295 1b, § 140a (Integrierte Versorgung), § 73b (Hausarztzentrierte Versorgung), § 73c (Besondere ambulante ärztliche Versorgung) |
| fall_300     | Arznei- oder Hilfsmittelverordnung nach § 300  |
| fall_302     | Heil- oder Hilfsmittelverordnung nach § 302  |

<sup>14</sup> BSP ist kein Verfahren, sondern dient nur zur Illustration der XML-Struktur.

Element **qs\_data/BSP/versicherter**

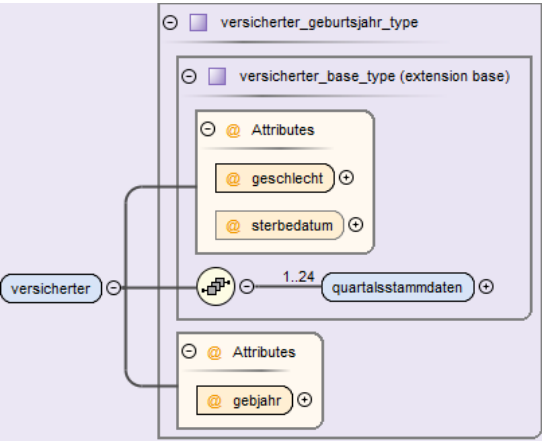


Abbildung 41: Element *versicherter*

Tabelle 45: Attribute: *versicherter*-Element

| Attribut           | Type                 | Use      | Beschreibung                 |
|--------------------|----------------------|----------|------------------------------|
| <u>geschlecht</u>  | enum_geschlecht_type | required | Geschlecht                   |
| <u>sterbedatum</u> | datum_en_type        | optional | Sterbedatum des Versicherten |
| <u>gebjahr</u>     | jahr_type            | required | Geburtsjahr                  |

Element **qs\_data/BSP/versicherter/quartalsstammdaten**

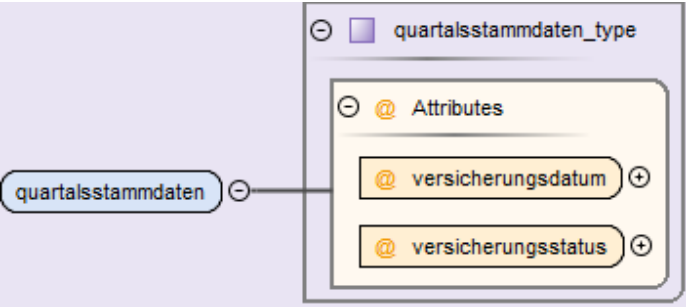


Abbildung 42: Element *quartalsstammdaten*

Tabelle 46: Attribute: *quartalsstammdaten*-Element

| Attribute                        | Type                     | Use      | Beschreibung                         |
|----------------------------------|--------------------------|----------|--------------------------------------|
| <u>versicherungsda-<br/>tum</u>  | Datum_quartalsmitte_type | required | Datumsangabe zum Versicherungsstatus |
| <u>versicherungs-<br/>status</u> | enum_jn_type             | required | Versicherungsstatus                  |

Element qs\_data/BSP/fall\_301

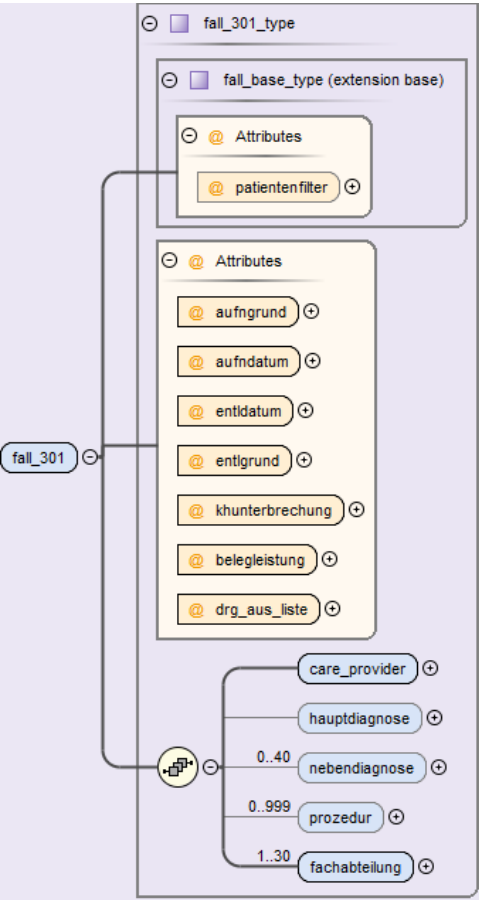


Abbildung 43: Element `fall_301` und Kind-Elemente `care_provider`, `hauptdiagnose`, `nebuldiagnose`, `prozedur` und `fachabteilung`

Das Element hat folgende Attribute:

Tabelle 47: Attribute: `fall_301`-Element

| Attribut                     | Type                           | Use      | Beschreibung  |
|------------------------------|--------------------------------|----------|---|
| <code>patientenfilter</code> | <code>int_based_boolean</code> | optional | Dieses Attribut wird auf „1“ gesetzt, wenn der entsprechende Fall durch den Patientenfilter ausgelöst wurde (im Unterschied zum Leistungs-/Medikationsfilter). Dieser Fall enthält also eine Indexleistung, die zur Selektion des Versicherten geführt hat. |
| <code>aufgrund</code>        | <code>aufgrund_type</code>     | required | Aufnahmegrund   |
| <code>aufndatum</code>       | <code>datum_en_type</code>     | required | Aufnahmedatum Krankenhaus   |



| Attribut        | Type            | Use      | Beschreibung  |
|-----------------|-----------------|----------|---|
| entldatum       | datum_en_type   | required | Entlassungsdatum                                    |
| entlgrund       | entlgrund_type  | required | Entlassungsgrund                                    |
| khunterbrechung | khunterbrechung | required | Unterbrechung des Krankenhausaufenthaltes           |
| belegleistung   | belegleistung   | optional | Art der Belegleistung                               |
| drg_aus_liste   | drg_aus_liste   | optional | Marker, ob DRG des KH-Falles in Liste enthalten ist |

#### Element `qs_data/BSP/fall_301/care_provider`

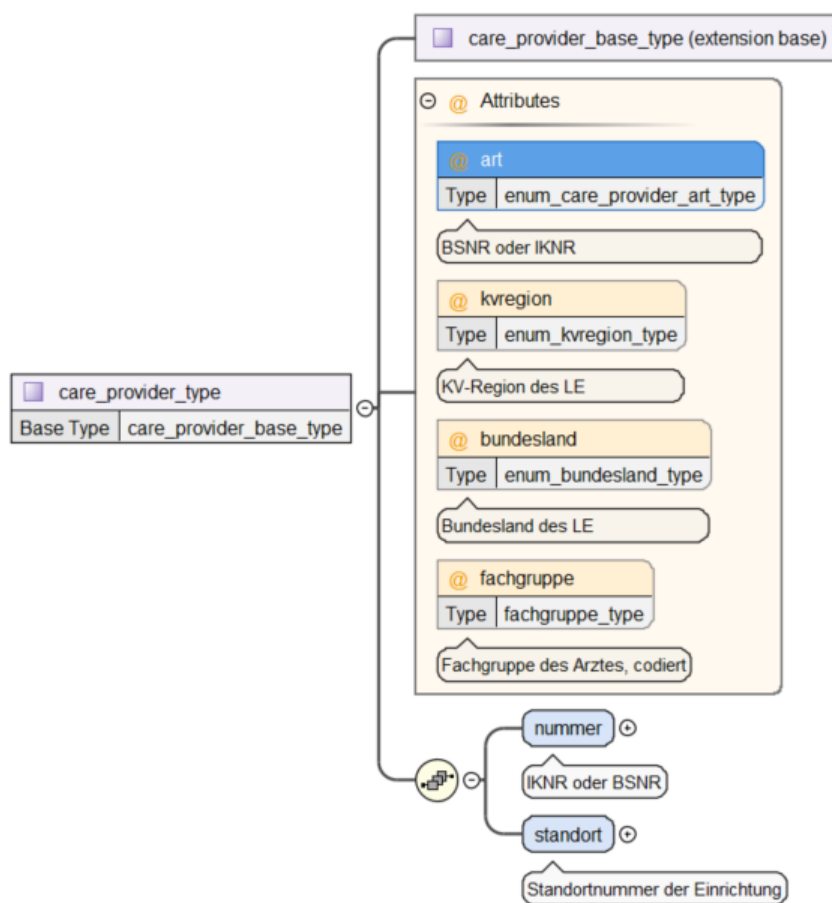


Abbildung 44: Element `care_provider` und Kind-Element `nummer`

Das Element hat folgende Attribute:

Tabelle 48: Attribute: `care_provider`-Element

| Attribut   | Type                        | Use      | Beschreibung  |
|------------|-----------------------------|----------|---|
| art        | enum_care_provider_art_type | required | BSNR oder IKNR (zur Unterscheidung der Art des Leistungserbringers) |
| kvregion   | enum_kvregion_type          | optional | KV-Region des LE  |
| bundesland | enum_bundesland_type        | optional | Bundesland des LE   |
| fachgruppe | fachgruppe_type             | optional | Fachgruppe des Arztes, codiert                                      |

Die Beschreibungen der weiteren Elemente können der Datensatzbeschreibung (Abschnitt 6.2) oder dem XML-Schema entnommen werden.

### Element body/case/qs\_data/AUFST

Die jährlich zu erstellende Aufstellung nach § 16 Abs. 5 DeQS-RL, Teil 1 wird in der XML-Struktur eingebunden wie ein QS-Modul. Das Modul AUFST ist mit dem Typ `qs_data_aufst_type` ein Element der Typdefinition `qs_data_type`. Abbildung 45 zeigt die XML-Struktur für eine Aufstellung.

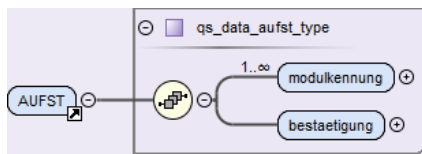


Abbildung 45: `qs_data_aufst_type` für Aufstellung

Eine Aufstellung enthält im Attribut `body/@module` den Wert AUFST. Der Name des Elementes innerhalb des Elementes `body/case/qs_data` entspricht ebenfalls dem Wert AUFST. Im Element AUFST sind zwei Unterelemente definiert. Das Element AUFST/modulkennung ist beliebig oft wiederholbar, jedoch mindestens einmal (je case-Element) aufzuführen. Das Element AUFST/bestaetigung ist genau einmal anzugeben.

### Element body/case/qs\_data/AUFST/modulkennung

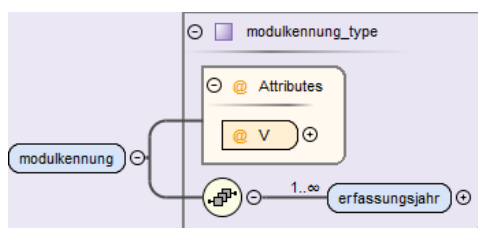
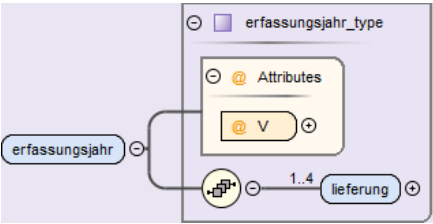


Abbildung 46: `modulkennung_type`

Mittels der Aufstellung sollen modulübergreifende Auflistungen möglich sein. Die Zuordnung zu der Anzahl der übermittelten Datensätze von ggf. verschiedenen Modulen erfolgt innerhalb des Elementes `body/case/qs_data/AUFST` durch das Attribut `V` im Element `./modulkennung`.

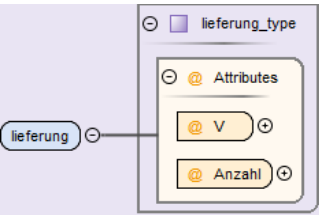
**Element body/case/qs\_data/AUFST/modulkennung/erfassungsjahr**



Pro modulkennung sind beliebig viele Erfassungsjahre möglich. Das Erfassungsjahr ist unter `modulkennung/erfassungsjahr/@V` einzutragen. Das Element `erfassungsjahr` hat bis zu vier Kindelemente `lieferung` jeweils mit den zwei Attributen `V` und `Anzahl`.

Abbildung 47:  
*erfassungsjahr\_type*

**Element body/case/qs\_data/AUFST/modulkennung/erfassungsjahr/lieferung**



Das Attribut `lieferung/@V` entspricht einem Wert des Feldes `quartal` in der Tabelle `Lieferfrist` der Spezifikationsdatenbank. Das Attribut `lieferung/@Anzahl` entspricht im Regelfall dem Wert des XML-Attributes `total_count` der jeweils letzten erfolgreich übermittelten Transaktion des jeweiligen Lieferquartals. Dieser Wert ist auch im zugehörigen Datenflussprotokoll der letzten erfolgreich übermittelten Transaktion unverändert enthalten.

Abbildung 48:  
*lieferung\_type*

Aus Gründen der Lesbarkeit und Nachvollziehbarkeit der Aufstellung sind Lieferquartale auch dann aufzulisten, wenn keine Transaktion erfolgte. Dies kann dadurch begründet sein, dass ein Lieferquartal noch nicht bzw. nicht mehr geöffnet war. Hierfür ist der Wert `-1` vorgesehen. Ein Eintrag mit dem Wert `-1` darf weggelassen werden. Wenn alternativ bei geöffnetem Lieferquartal, beispielsweise aus technischen Gründen, keine Transaktion erfolgte, ist der Wert `-2` einzutragen. Würde sich für ein Modul und Erfassungsjahr in allen vier Lieferquartalen jeweils ausschließlich der Wert `-1` ergeben, so ist die Kombination aus Modul und Erfassungsjahr nicht in der Aufstellung zu erfassen. Alle möglichen Wertangaben für die Anzahl übermittelter Datensätze können Tabelle 49 entnommen werden.

Tabelle 49: mögliche Werte für Anzahl in einer Aufstellung

| Szenario  | möglicher Wert im Attribut Anzahl einer Aufstellung |
|---|---|
| erfolgreiche Transaktion (auch Nullmeldung)   | <code>/root/header/@total_count</code>              |
| Lieferquartal war für dieses Modul und Erfassungsjahr per Spezifikation nicht geöffnet<br>(Kein Eintrag für dieses Quartal in Tabelle Lieferfrist der Spezifikationsdatenbank für die Kombination aus Modul und Erfassungsjahr) | <code>-1</code>                                     |
| keine erfolgte Transaktion  | <code>-2</code>                                     |

### Element body/case/qs\_data/AUFST/bestaetigung

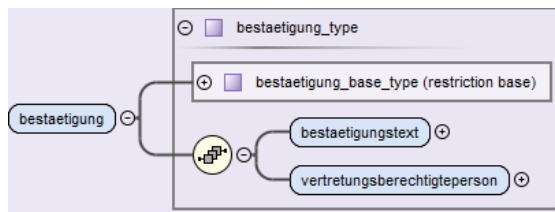


Abbildung 49: *bestaetigung\_type*

Das Element *bestaetigung* dient der Übertragung von Inhalten im Zusammenhang mit der Bestätigung der Krankenkasse darüber, dass alle gem. Richtlinie erforderlichen Daten übermittelt worden sind. Hierzu ist einerseits ein entsprechender Bestätigungstext sowie andererseits der Name einer vertretungsberechtigten Person zu übermitteln. Für diese beiden Angaben sind zwei untergeordnete Elemente vorgesehen. Das Element *bestaetigung* verfügt über keine Attribute.

### Element body/case/qs\_data/AUFST/bestaetigung/bestaetigungstext

Im Attribut *bestaetigungstext/@V* ist Platz für eine freie Formulierung des Bestätigungstextes. Ein Formulierungsvorschlag ist in den Annotationen direkt in der Datei `./abstract/sqg_cases_base.xsd` im XML-Schema hinterlegt.

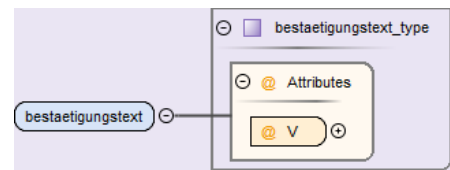


Abbildung 50:  
*bestaetigungstext\_type*

### Element body/case/qs\_data/AUFST/vertretungsberechtigteperson

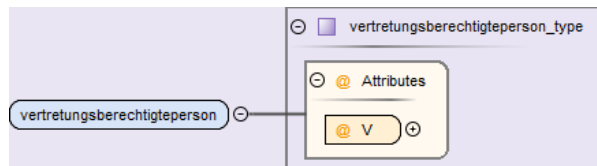


Abbildung 51:  
*vertretungsberechtigteperson\_type*

Das Attribut *vertretungsberechtigteperson/@V* beinhaltet (in Form von Freitext mit mindestens 3 Zeichen) den Namen, der gem. RL-Vorgaben für die Abgabe der Aufstellung und Bestätigung vertretungsberechtigten Person seitens der Krankenkasse. Der Inhalt dieses Attributs

verbleibt in der DAS-KK, d.h. bei der Verarbeitung durch die DAS-KK wird das Element für die vertretungsberechtigte Person entfernt. Eine Weiterleitung ohne diese Veränderung ist nicht zulässig. Dies wird per XML-Schema ab der DAS-KK sichergestellt.

#### 6.4.5 XML-Struktur der PID

Die PID-Datei ist der QS-Datei ähnlich. Anstelle der QS-Daten (`<qs_data>`) werden die PID (`<patient>`) in den Datencontainer aufgenommen.

Das Element `<patient>` enthält die patientenidentifizierenden Daten. Das Kind-Element von `<patient>` ist das Element `<pid>`, das die tatsächliche PID `<VERSICHERTENIDNEU>` aufnimmt. Dies entspricht der bisherigen Struktur im Bereich der DeQS-RL. Somit kann die VST weiterhin ihren bereits bestehenden Verarbeitungsprozess nutzen. Dies betrifft auch folgende Elemente, die nur eine Art Platzhalter für die VST darstellen und dementsprechend nicht von den

Krankenkassen exportiert werden dürfen: <KASSEIKNR>, <VERSICHERTENIDALT> und das Attribut @twodigitik. Ebenso ist das <patient>-Element lediglich aus Kompatibilitätsgründen für die VST als optional definiert. Die Krankenkassen müssen für jeden Datensatz (<case>) immer auch ein <patient>-Element angeben.

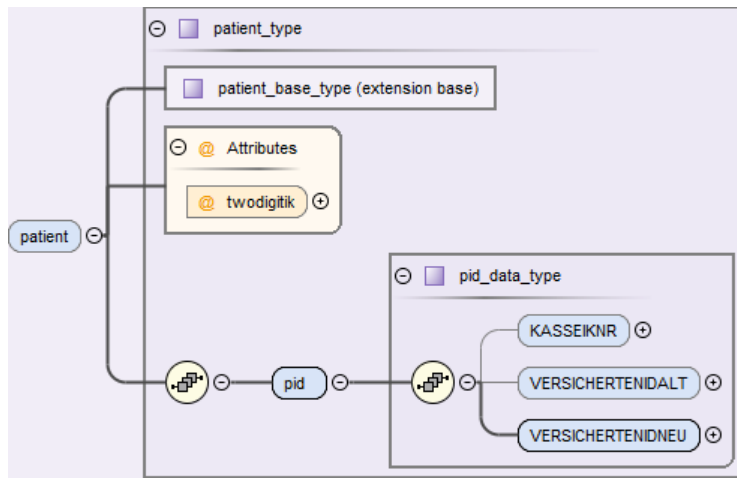


Abbildung 52: XML-Struktur der PID-Datei

<VERSICHERTENIDNEU> nimmt in das Attribut „v“ die lebenslange Versichertennummer des Patienten auf.

#### 6.4.6 XML-Struktur der gemeinsamen QS- und PID-Dateien

Die Zusammenführung der QS- und der PID-Datei erfolgt bei der DAS, anschließend führt die VST die einzelnen Datensätze beider Dateien anhand der Datensatz-IDs (ID im <case\_admin>-Element bzw. lfdnr im <case>-Element) zusammen. Die XML-Schnittstellen „interface\_DAS\_VST“ und „interface\_VST\_BAS“ definieren die Datenstruktur der zusammengeführten Dateien.

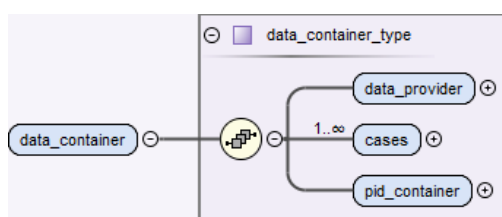


Abbildung 53: Zusammenführung der QS- und PID-Dateien

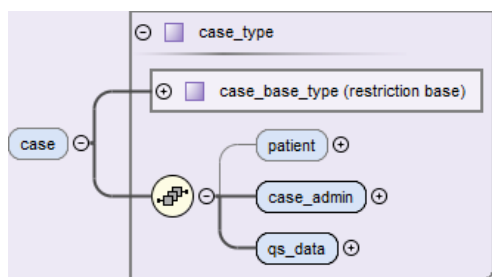


Abbildung 54: Zusammenführung der QS- und PID-Datensätze

Die Strukturen werden hier nicht näher betrachtet und können direkt den jeweiligen XML-Schnittstellen entnommen werden.

#### 6.4.7 XML-Struktur der Rückprotokollierung (Datenflussprotokoll)

Für die Protokollierung der Prüfungen und deren Ergebnissen gibt es auf Dokumentenebene im Header und auf Fallebene im `<case_admin>` das Element `<protocol>`.

Auf Dokumentenebene sind alle Prüfungen zu dokumentieren, einschließlich der Prüfungen, die ausschließlich die Datensatzebene betreffen. Eine prüfende Einrichtung trägt sich als `<validation_provider>` in die entsprechende Auflistung ein und dokumentiert dann die von ihr durchgeführten Prüfungen in der Auflistung `<validation_item>`.

Sollte die ursprüngliche XML-Datei nicht lesbar (z. B. nicht entschlüsselbar oder nicht wohlgeformt) sein, ist ein reduziertes Protokoll („Miniprotokoll“) zu erstellen.

Das Miniprotokoll unterscheidet sich zum vollständigen Protokoll in folgenden Punkten:

- Es enthält kein Header- und kein body-Element (Abbildung 55).
- Die ursprüngliche ID (GUID) der Export-Datei wird nur angegeben, wenn sie zur Verfügung steht und lesbar ist.

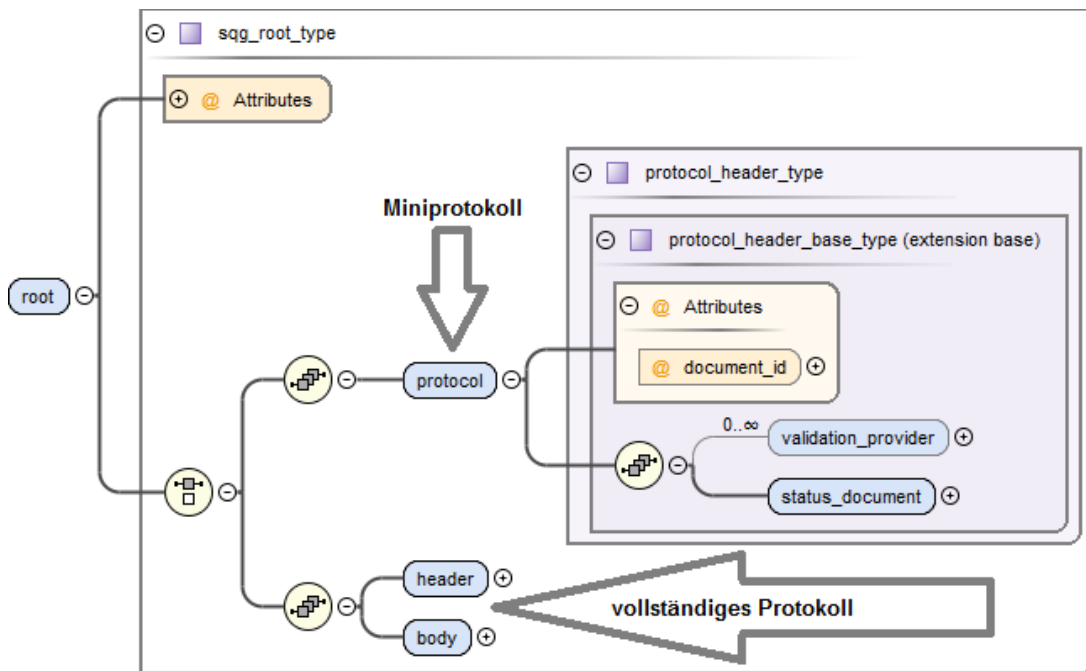


Abbildung 55: Schema des Datenflussprotokolls „response\_DAS\_KK.xsd“

Nähere Informationen zur Rückprotokollierung können dem Abschnitt 6.6 entnommen werden.

Die XML-Struktur der Rückprotokollierung für die Dokumenten- und die Fallebene ist identisch und wird hier näher erläutert:

### Element header/protocol

Das Element `<protocol>` im ursprünglichen XML-Dokument mit den QS-Daten nimmt Informationen zu Prüfungen auf, die im Datenfluss durchgeführt wurden. Es ist Teil der Rückprotokollierung. Es soll gemeinsam mit dem Unterelement `<status_document>` ab der DAS im Datenfluss vorhanden sein, um im nachfolgenden Datenfluss vorgenommene Prüfergebnisse aufzunehmen.

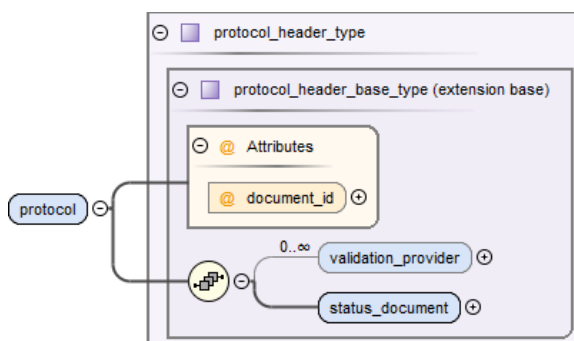


Abbildung 56: Element `protocol` und Kind-Elemente `validation_provider` und `status_document`

Tabelle 50: Attribute: *protocol*-Element

| Attribut    | Type      | Use      | Beschreibung   |
|-------------|-----------|----------|--|
| document_id | xs:string | optional | Wenn die ursprüngliche XML-Datei für die Rückprotokollierung nicht geeignet ist, muss sie in das Attribut document_id eingetragen werden, falls die GUID des Exportdokuments lesbar ist. |

**Element header/protocol/validation\_provider**

Hier wird diejenige Stelle verzeichnet, die einen oder mehrere Prüfungsschritte durchgeführt hat. Die Ergebnisse der Prüfung werden in diesem Container abgelegt und Teil der Rückprotokollierung.

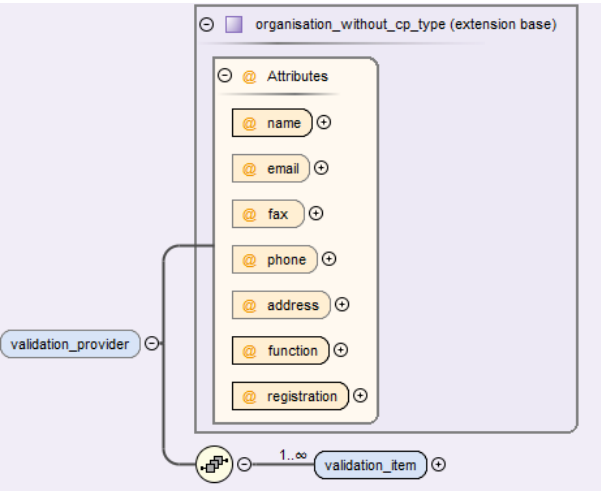


Abbildung 57: Element *validation\_provider* und Kind-Element *validation\_item*

Tabelle 51: Attribute: *validation\_provider*-Element

| Attribut     | Type                          | Use      |
|--------------|-------------------------------|----------|
| name         | xs:string                     | required |
| email        | emailAddress_type             | optional |
| fax          | xs:string                     | optional |
| phone        | xs:string                     | optional |
| address      | xs:string                     | optional |
| function     | enum_validation_provider_type | required |
| registration | registration_type             | required |



**Element header/protocol/validation\_provider/validation\_item**

Auf Dokumentenebene sind alle Prüfungen zu dokumentieren. Eine prüfende Einrichtung trägt sich als <validation\_provider> in die entsprechende Auflistung ein und dokumentiert dann die von ihr durchgeführten Prüfungen in der Auflistung <validation\_item>.

Als Ergebnis jeder Prüfung wird eine der folgenden Aussagen über das geprüfte Dokument getroffen:

- OK (Keine Auffälligkeiten)
- WARNING (Auffälligkeiten, die einer Weiterverarbeitung nicht im Weg stehen)
- ERROR (Auffälligkeiten bzw. Fehler, die eine Weiterverarbeitung des Dokuments ausschließen)

Das Ergebnis der Prüfung wird in das Attribut „V“ des Elements <status> im Element <validation\_item> eingetragen.

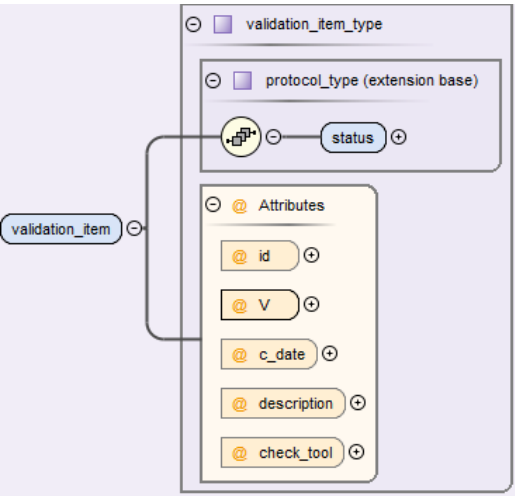


Abbildung 58: Element `validation_item` und Kind-Element `status`

Das Element hat folgende Attribute:

Tabelle 52: Attribute: `validation_item`-Element

| Attribut        | Type                | Use      | Beschreibung   |
|-----------------|---------------------|----------|--|
| <code>id</code> | <code>xs:int</code> | optional | Die ID muss nur dann vergeben werden, wenn eine Prüfung auf Datensatzebene stattfindet. Diese ID ist dokumentweit gültig und darf im Header nur einmal vorkommen. Prüfungen auf Datensatzebene (Element "case"), die zu dieser Prüfung gehören, werden über diese ID zugeordnet. |

| Attribut    | Type                      | Use      | Beschreibung   |
|-------------|---------------------------|----------|--|
| V           | enum_validation_type      | required | Dieser Wert bezeichnet die durchgeführte Prüfung anhand einer „enumeration“, die in sqg_protocol.xsd definiert wird.<br>Gültige Werte sind: Dechiffrierung, LE_Pseudonym, PID_Pseudonym, Protokoll, Schema, Spezifikation, Transaktion, sonstige Prüfung |
| c_date      | smalldatetime_simple_type | optional | Hier kann ein Zeitstempel für die Verarbeitung angegeben werden.   |
| description | xs:string                 | optional | Hier kann bei erklärungsbedürftigen Items eine Beschreibung hinterlassen werden (auf Dokumentenebene).   |
| check_tool  | xs:string                 | optional | Hier können beim Einsatz eines Tools für die Prüfung der XML-Dateien der Name und die Versionsnummer des Tools hinterlassen werden (beim Einsatz des Datenprüfprogramms wird die Versionsnummer des XSL-T-Skripts eingetragen).                          |

**Element header/protocol/status\_document**

Hier wird der Gesamtstatus des Dokuments angegeben. Das Attribut V kann also auf „OK“, „WARNING“ oder „ERROR“ stehen. Dieser Status kann nur geändert werden, wenn sich der Status des Dokuments im Laufe der Prüfung verschlechtert. „ERROR“ hat zur Folge, dass das Dokument komplett zurückgewiesen werden muss.

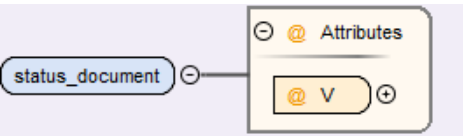


Abbildung 59: Element `status_document`

Tabelle 53: Attribute: `status_document`-Element

| Attribut | Type                     | Use      | Beschreibung                     |
|----------|--------------------------|----------|----------------------------------|
| V        | status_document_Datentyp | required | Mögliche Werte: OK/WARNING/ERROR |

### Element header/protocol/validation\_provider/validation\_item/status

Hier wird angegeben, ob die betroffene Prüfung ohne Fehler („OK“), mit Fehlern („WARNING“) oder mit fatalem Fehler („ERROR“) abgeschlossen wurde. Der Gesamtstatus des Dokuments entspricht jeweils dem schlechtesten Prüfergebnis. Bei der ersten Prüfung mit dem Ergebnis „ERROR“ muss die Weiterverarbeitung abgebrochen werden.

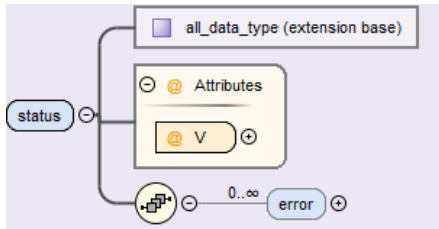


Abbildung 60: Element status und Kind-Element error

Tabelle 54: Attribute: Status-Element

| Attribut | Type        | Use      |
|----------|-------------|----------|
| V        | status_type | required |

Darüber hinaus gibt es die Möglichkeit, eine beliebige Anzahl von `<error>` Elementen mit einer `<error_message>` im `<status>` Element unterzubringen.

### Element header/protocol/validation\_provider/validation\_item/status/error

Ein `<error>` Element nimmt Fehlerdaten auf. Als einziges verpflichtendes Unterelement gilt das `<error_message>`-Element. Die Elemente `<rule_id>` und `<rule_type>` sind spezifisch für die Anwendung von Plausibilitätsregeln der Spezifikation.

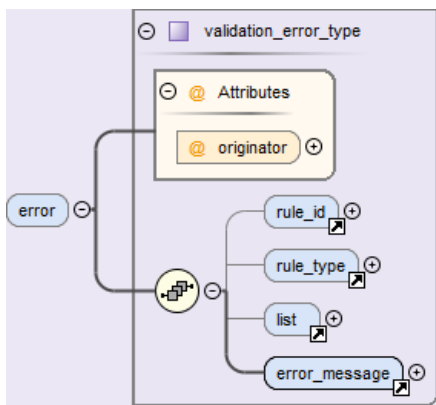


Abbildung 61: Element error und Kind-Elemente rule\_id, rule\_type, list und error\_message

Tabelle 55: Attribute Error-Element

| Attribut   | Type              | Use      | Beschreibung  |
|------------|-------------------|----------|---|
| originator | enum_Organisation | optional | Mögliche Werte: (Bundes-)Auswertungsstelle, Datenannahmestelle, Vertrauensstelle, Krankenkasse. |

Das Element hat folgende Kind-Elemente:

Tabelle 56: Attribute: Error-Element

| Element       | Beschreibung  |
|---------------|---|
| rule_id       | Nummer der Regel bzw. der Fehlermeldung (Fehlercode)                            |
| rule_type     | Werte H (=hart), D (=weich, eigentlich hart) oder W (=weich)                    |
| list          | Liste der fehlerhaften Felder. Das Trennzeichen ist „ “ (siehe Abschnitt 6.6.3) |
| error_message | Fehlermeldung aus der Spezifikationsdatenbank als Freitext                      |

### Element case/case\_admin/protocol

Dieses Element hat auf Datensatzebene eine ähnliche Struktur wie das oben beschriebene Element `<protocol>`.

Die Unterschiede sind:

- Während das `<protocol>`-Element im Header Ergebnisse der Prüfungen, die das Dokument insgesamt betreffen, aufnimmt, nimmt das Protokoll-Element im Body-Bereich die Ergebnisse der Prüfungen auf, die auf Datensatzebene erfolgen.

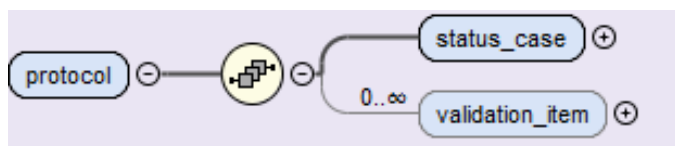


Abbildung 62: Element protocol und Kind-Elemente validation\_item und status\_case

- Entspricht das Attribut `regelErrorIsWarning` aus der Tabelle DB-SDAT.ModulErfassungsjahr dem Wert `true`, dann werden Regeln mit der Strenge hart für dieses Modul und Erfassungsjahr weich interpretiert. Das Attribut `rule_type/@V` mit dem Wert „D“ markiert diese Regelverletzungen. `regelErrorIsWarning` hat im Gegensatz zu Regeln aus der Tabelle DB-SDAT.Regel keinen Einfluss auf administrative Prüfungen aus DB-SDAT.Pruefung.
- `<status_case>` beinhaltet jeweils das schlechteste Ergebnis aller auf Datensatzebene.
- Das Protokoll auf Datensatzebene hat kein Element `<validation_provider>` (Prüfstelle). Damit auch auf dieser Ebene die Ergebnisse der durchgeführten Prüfungen einer Prüfstelle zugeordnet werden können, müssen alle Ergebnisse einer Prüfung auf Fallebene mit

einer gemeinsamen, dokumentweit eindeutigen ID im Attribut ID des Elements <validation\_item> eingetragen werden.

### Element case/qs\_data/INFO

Ein Rücktransport übermittelter Datensätze im Rahmen des Versands von Datenflussprotokollen ist nicht erlaubt. Im Sinne einer Verarbeitungsprotokollierung sind somit die administrativen Informationen auf Datensatzebene (./case/case\_admin) zur Zuordnung sowie die darin enthaltenen Protokollinformationen in einem Datenflussprotokoll als technische Quittung zu einer von der KK initiierten Transaktion hinreichend. Insofern werden in Datenflussprotokollen innerhalb der case-Elemente ausschließlich die darin enthaltenen case\_admin-Elemente übertragen.

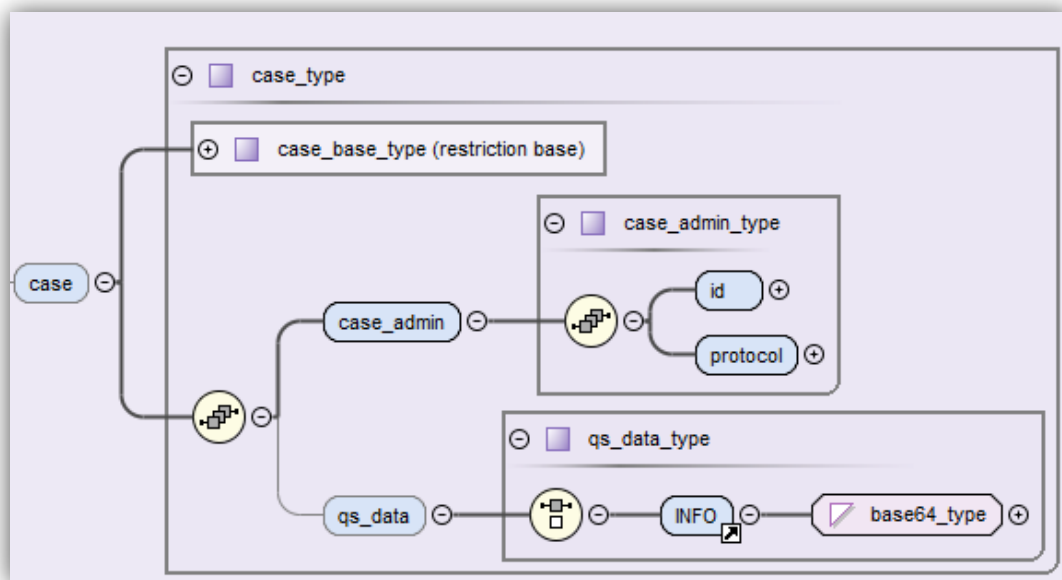


Abbildung 63: Element INFO und Vorfahren-Elemente qs\_data sowie case

Zum Zwecke der Übertragung krankenkassenspezifischer Rückmeldeberichte als eigene Datenart von der BAS an die KK wird das Schema des Datenflussprotokolls verwendet. Zu diesem Zweck kann das case-Element optional auch ein Element qs\_data aufnehmen, welches lediglich das Inhaltselement INFO beinhalten kann. Innerhalb des Elementes INFO befindet sich ausschließlich eine in base64-kodierte PDF-Datei. Bei der Übertragung krankenkassenspezifischer Rückmeldeberichte ist nur eine base64-kodierte PDF-Datei je Transaktion erlaubt, d. h. die Anzahl der case-Elemente ist in diesem Fall auf eins beschränkt. Darüber soll mit dieser Limitierung die Übertragung und Verarbeitung von sehr großen XML-Dateien verhindert werden.

#### 6.4.8 Versionierung der XML-Schemata

Das Schemamodell besteht im Wesentlichen aus zwei Teilstrukturen:

1. ein über die Zeit stabiler Umschlag (Datencontainer)
2. ein sich durch geänderte oder neu hinzugekommene Verfahren ändernder Inhalt

Da zu 1. eine spezifikationsübergreifende Konstanz erwartet wird – mit Versionswechseln unabhängig von der Spezifikation – und zu 2. ggf. eine jährliche mit der Spezifikation gekoppelte neue Version, soll die Versionskontrolle über zwei voneinander unabhängige Mechanismen erreicht werden:

- Für den Umschlag (Datencontainer) wird im Schema das Attribut `/root/@container_version` erzwungen und mit einem Fixwert besetzt. Eine Änderung im Schema, die nicht abwärtskompatibel ist, wäre ein Grund für einen Versionswechsel.
- Für die spezifikationsabhängige Struktur wird das Attribut `/root/@content_version` erzwungen und mit einem Fixwert besetzt. Hier gilt für den Versionswechsel im Prinzip dasselbe wie bei der „`container_version`“. Die Versionsnummer wird jedoch nur erhöht, wenn der XML-Inhalt unabhängig von der zentralen Spezifikation geändert werden muss.

### 6.5 XML-Verschlüsselung

Aufgrund der Anforderung der DeQS-RL und der oKFE-RL, mehrere Datenarten (QS-Daten, patientenidentifizierende Daten, administrative Daten, leistungserbringeridentifizierende Daten (sofern vorliegend) verschiedenen Akteuren im Gesamtprozess (Leistungserbringer, Krankenkassen, Datenannahmestelle, Vertrauensstelle, (Bundes-)Auswertungsstelle) selektiv zu übermitteln, ist die Anwendung eines alle Datensicherheitsanforderungen erfüllenden Verschlüsselungsprogramms erforderlich. An der Schnittstelle von den Krankenkassen zur DAS-KK wird die Verschlüsselung gemäß SECON nach PKCS#7 verwendet. Ab der DAS-KK werden die Daten mittels XML-Encryption verschlüsselt.

#### Verschlüsselungsalgorithmen

Für die Verschlüsselung der XML-Knoten wird die hybride Verschlüsselung nach dem W3C-Standard „XML Encryption Syntax and Processing“ verwendet.

Als Verschlüsselungsalgorithmen werden „AES128“ für die symmetrische Verschlüsselung der XML-Elemente und „RSA mit 2048-Bit“ für die asymmetrische Verschlüsselung des generierten symmetrischen Schlüssels verwendet.

### 6.6 Die Rückprotokollierung

Es existieren zwei Arten von Rückmeldungen: Die Eingangsbestätigung für den Datenabsender durch den Datenempfänger als Bestätigung des Empfangs einer Lieferung sowie das Datenflussprotokoll als Ergebnis der Gesamtverarbeitung bzw. immer dann, wenn eine weitere Verarbeitung als Folge von Fehlern nicht möglich ist.

### 6.6.1 Empfangsbestätigung

Die Empfangsbestätigung wird vom Datenempfänger zurück an den Datenabsender geschickt, sofern das angenommene Datenpaket in Bezug auf die Prüfungen des Datenempfängers verarbeitet werden konnte. Im Fall einer segmentierten Transaktion wird eine Empfangsbestätigung je Segment erstellt.

Wie Abbildung 64 zeigt, enthält die Empfangsbestätigung lediglich eine Benachrichtigung, dass die vom Datenabsender übermittelte XML-Datei bearbeitbar war und an die nachfolgende Stelle weitergeleitet wurde. Das Element `<status_document>` dient zur automatischen Auswertung der Empfangsbestätigung und enthält immer das Attribut `V="B"`. Die enthaltene Dokument-ID bezieht sich auf die ID des Segmentes und ermöglicht die Zuordnung der Empfangsbestätigung zur jeweiligen Datenlieferung:

```

1  <?xml version="1.0" encoding="iso-8859-1"?>
2  <root xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3  content_version="1.0"
4  container_version="2.0"
5  xsi:noNamespaceSchemaLocation="..\interface_KK_DAS\response_receipt_DAS_KK.xsd">
6    <header>
7      <document>
8        <id V="{55664537-5642-9056-8676-456468327446}"/>
9      </document>
10     <provider email="datenannahmestelle@test-datenannahmestelle.de" name="Test-Datenannahmestelle" registration="bu000000"
11       function="Datenannahmestelle" address="Test Adresse"/>
12     <status_document V="B"/>
13   </header>
14   <body>
15     <nachricht>
16       Ihre Dateneinsendung konnte erfolgreich eingelesen und an die Vertrauensstelle weitergeleitet werden.
17       Ein umfassendes Datenflussprotokoll über die Prüfergebnisse der Datenannahmestelle, der Vertrauensstelle
18       und der Bundesauswertungsstelle erhalten Sie von uns in den nächsten 24 Stunden.
19     </nachricht>
20   </body>
21 </root>

```

Abbildung 64: Beispiel einer Empfangsbestätigung der DAS-KK an eine Krankenkasse

Die Empfangsbestätigung wird per SFTP – via Schnittstelle KK/DAS-KK über KKS – verschickt.

### 6.6.2 Datenflussprotokoll

Die vorgenommenen Prüfungen werden in den dafür vorgesehenen Bereichen im XML-Code des übermittelten Dokuments auf Header- und auf Datensatzebene protokolliert. Das Protokoll des Dokuments wird damit von jeder entgegennehmenden Stelle um deren Prüfergebnisse ergänzt.

Da das Datenflussprotokoll vom ursprünglichen XML-Dokument abgeleitet wird, müssen die QS-Daten und die PID vor Übermittlung an die Krankenkasse entfernt werden. Das übriggebliebene XML enthält innerhalb der ursprünglichen Struktur des Dokuments die bis dahin protokollierten Prüfungen und die sich daraus ergebenden Statusmeldungen der Datensätze und des Dokuments. Der Aufbau des Protokolls ist im Abschnitt 6.6.4 detailliert beschrieben.

Zur Visualisierung des Datenflussprotokolls kann dieses mithilfe einer Template-Definition z. B. nach HTML transformiert werden.

Datenflussprotokolle werden von der DAS-KK oder der VST nur in den Fällen erstellt, in denen eine Weiterleitung der Datenlieferungen nicht möglich war. In diesem Fall wird keine zusätzliche

Empfangsbestätigung an den Absender verschickt, er erhält stattdessen das Datenflussprotokoll.

**Miniprotokoll**

Die Erstellung eines vollständigen Datenflussprotokolls ist nur möglich, wenn die ursprüngliche XML-Datei lesbar ist und nach Entfernung der PID und der QS-Daten schemakonform bleibt. Andernfalls ist ein reduziertes Protokoll (Miniprotokoll) zu erstellen, das die ID (GUID) des Dokuments (sofern diese zur Verfügung stand und lesbar war, sonst wird diese nicht angegeben) und die konkrete Fehlermeldung enthält. Die Struktur dazu ist Abschnitt 6.4.7 zu entnehmen.

**6.6.3 Fehlermeldungen**

Eine Fehlermeldung ist das Ergebnis einer negativ verlaufenen Prüfung. Neben den Plausibilitätsprüfungen werden in das Datenflussprotokoll auch die Fehlermeldungen von Prüfungen auf administrativer Ebene aufgenommen. Die Struktur der Plausibilitätsregeln bzw. der Prüfungen auf administrativer Ebene werden in Abschnitt 2.3.2 bzw. Abschnitt 2.3.3 beschrieben.

Sollte keine administrative Prüfung definiert sein, kann in begründeten Ausnahmefällen statt einer standardisierten Fehlermeldung eine vorläufige Fehlermeldung benutzt werden.

Fehlermeldungen sind Ausdruck des `<error_message>`-Elements innerhalb des `<error>`-Elements. Im Folgenden werden Aufbau und Attribute (Tabelle 57) sowie Kind-Elemente (Tabelle 58) des `<error>`-Elements beschrieben:

*Tabelle 57: Aufbau und Attribute des Elements `<error>`*

| Name       | Typ               | Use      | Beschreibung   |
|------------|-------------------|----------|--|
| originator | enum_organisation | optional | Mögliche Werte (u. a.): (Bundes-)Auswertungsstelle, Datenannahmestelle, Vertrauensstelle, Krankenkasse |

*Tabelle 58: Kind-Elemente des `<error>`-Elements*

| Kind-Element                   | Relevant für Prüfungsart                          | Beschreibung  |
|--------------------------------|---|---|
| <code>&lt;rule_id&gt;</code>   | Administrative Prüfungen, Plausibilitätsprüfungen | ID der Fehlermeldung  |
| <code>&lt;rule_type&gt;</code> | Plausibilitätsprüfungen                           | Werte H (=hart), D (=weich, eigentlich hart) oder W (= weich). Bei Fehlern auf Headerebene oder bei Fehlermeldungen als Ergebnis administrativer Prüfungen wird dieses Element nicht angegeben. |
| <code>&lt;list&gt;</code>      | Plausibilitätsprüfungen                           | Pfad zu den an der Plausibilitätsregel beteiligten Attributen (als Trennzeichen wird das „ “ Zeichen verwendet). Bei  |



| Kind-Element    | Relevant für Prüfungsart                          | Beschreibung   |
|-----------------|---|--|
|                 |   | Fehlern auf Headerebene oder bei Fehlermeldungen als Ergebnis administrativer Prüfungen wird dieses Element nicht angegeben. |
| <error_message> | Administrative Prüfungen, Plausibilitätsprüfungen | Fehlermeldung  |

Die Abbildung 65 stellt ein <error>-Objekt exemplarisch dar:

```
<protocol>
  <status_case V="ERROR"/>
  <validation_item V="Spezifikation" id="1">
    <status V="ERROR">
      <error originator="Krankenkasse">
        <rule_id V="12355"/>
        <rule_type V="H"/>
        <list V="versicherter/@sterbedatum|fall_301[1]/prozedur/@datum"/>
        <error_message V="Das Sterbedatum liegt vor dem Operationsdatum"/>
      </error>
    </status>
  </validation_item>
</protocol>
```

Abbildung 65: Beispiel eines <error>-Elements

Dem Wert im Attribut `originator` ist zu entnehmen, dass der strenge Fehler (`<rule_type V="H" />`) mit der beispielhaften ID 12355 (`<rule_id V="12355" />`) durch die Krankenkasse verursacht wurde. Die in der Fehlermeldung (`<error_message>`) erwähnten Datenfeld-Attribute werden im Element `<list>` als Pfad referenziert.

Referenzierung der Datenfelder

Die vom Fehlerfall betroffenen Datenfelder werden als Pfadangabe mittels XPath beschrieben. Der Pfadausdruck geht dabei vom Enkelelement des zugehörigen `<case>`-Elements als aktuellem Kontextknoten aus und verwendet nur einfache Element- und Attributknoten, die jeweils mit „/“ getrennt sind. Die immer vorhandenen Containerelemente `<qs_data>` und `<BSP>`, `<PCI>` etc. (Modulkennungs-Element) werden also nicht explizit angegeben.

Tabelle 59: Verwendete XPath-Syntax

| Syntax                  | Semantik  | Beispiel  |
|-------------------------|---|---|
| <code>element[n]</code> | <b>Elementknoten</b><br>n-tes Kindelement (beginnend mit [1]) mit dem angegebenen Namen. Wenn nur ein Kindelement dieses Namens vorhanden ist, kann die Angabe „[n]“ entfallen. | <b>fall_301[2]</b><br>Bezeichnet das zweite <fall_301>-Element unterhalb des Modulkennungs-Elements |

| Syntax    | Semantik   | Beispiel  |
|-----------|--|---|
| /         | <b>Trennzeichen</b><br>Trennt zwei Knoten im Pfadausdruck. Der rechte Knoten ist ein Kindelement des linken Knotens. | <b>fall_301[2]/hauptdiagnose</b><br>Bezeichnet das (einzige) <hauptdiagnose>-Element unterhalb des zweiten <fall_301>-Elements unterhalb des Modulkennungs-Elements |
| @attribut | <b>Attributknoten</b><br>Bezeichnet ein Attribut des vorherstehenden Elementknotens                                  | <b>fall_301[2]/hauptdiagnose/@icd</b><br>Bezeichnet das Attribut „icd“ des <hauptdiagnose>-Elements   |

#### 6.6.4 Prüfergebnisse und Ergebnisprotokollierung

Für die Protokollierung der Prüfungen und deren Ergebnisse gibt es auf Dokumentenebene im Header und auf Fallebene im <case\_admin> das Element <protocol>.

Auf Dokumentenebene sind alle Prüfungen zu dokumentieren, einschließlich der Prüfungen, die ausschließlich die Datensätze betreffen. Die prüfende Einrichtung trägt sich jeweils als <validation\_provider> in die entsprechende Auflistung ein und dokumentiert dann ihre durchgeführten Prüfungen in der Auflistung <validation\_item>.

Prüfungen, die – wie z. B. die Schemakonformität – das Dokument insgesamt betreffen, sind ausschließlich im Headerbereich einzutragen.

Prüfungen, die – wie zum Beispiel die Prüfung auf Plausibilitätsregeln – auf Fallebene erfolgen, müssen folgendermaßen protokolliert werden:

Die Auswirkung des Prüfungsergebnisses in Bezug auf das gesamte Dokument muss im <header> eingetragen werden.

Das Ergebnis der Fallprüfung muss in <case\_admin> eingetragen werden.

Alle Ergebnisse einer Prüfung, die auf Fallebene erfolgt, müssen mit einer gemeinsamen, dokumentweit eindeutigen ID im Attribut ID des Elements <validation\_item> eingetragen werden. Dadurch ist es möglich, über die ID eines Prüfungsergebnisses, die man auf Fallebene findet, auf Dokumentenebene den <validation\_provider> eindeutig zu identifizieren (siehe Abbildung 66):

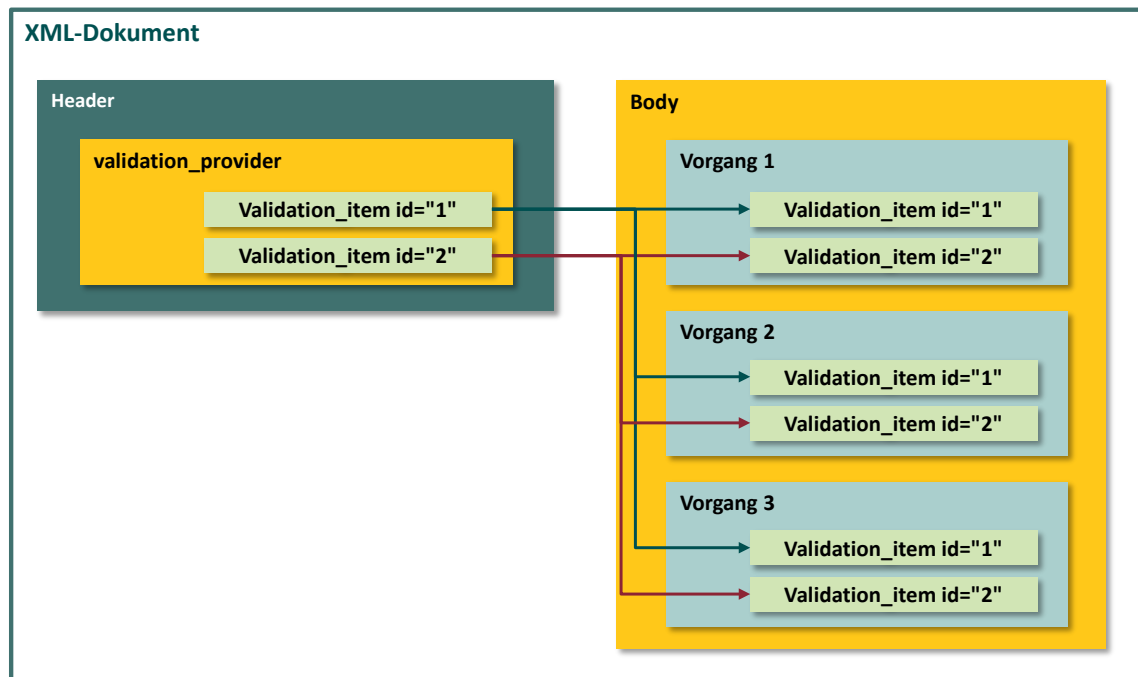


Abbildung 66: Beziehungen zwischen `<validation_item>` im header und `<validation_item>` im body über die `id`

Zur Veranschaulichung dieser Konstruktion soll im Folgenden eine Analogie aus dem relationalen Datenbank-Modell bemüht werden: die Dokumentenebene kann als Master-Tabelle, die Fallebene als Detail-Tabelle bezeichnet werden. Letztere enthält die zum Master gehörenden Detail-Datensätze, auf die über das Attribut „ID“ referenziert werden kann.

### Prüfungsergebnisse

Prinzipiell wird als Ergebnis jeder Prüfung eine der folgenden Aussagen über das geprüfte Objekt getroffen:

- Keine Auffälligkeiten
- Auffälligkeiten, die einer Weiterverarbeitung nicht im Weg stehen
- Auffälligkeiten bzw. Fehler, die eine Weiterverarbeitung des Objekts ausschließen.

In der datentechnischen Übersetzung wird dieses durch

- OK
- WARNING
- ERROR

ausgedrückt, die das Ergebnis der Prüfung im Attribut „V“ des Elements `<status>` im Element `<validation_item>` zusammenfassen.

Darüber hinaus gibt es die Möglichkeit, eine beliebige Anzahl von `<error>` Elementen mit einer `<error_message>` im `<status>` Element unterzubringen.

## Beziehung zwischen Vorgangsebene und Dokumentenebene

Es gibt zwei Kategorien geprüfter Objekte, die zueinander in einer hierarchischen Beziehung stehen:

- Erste Hierarchieebene: das gesamte Dokument
- Zweite Hierarchieebene: der Fall

Jedes dieser Objekte hat einen Status in Bezug auf die Weiterverarbeitung, der sich aus den Ergebnissen der durchgeführten Prüfungen ergibt.

Auf Dokumentenebene ist dieser Status im Unterelement `<status_document>` von `<protocol>` im Attribut „V“ abgelegt.

Auf Fallebene ist dieser Status ebenfalls in einem Attribut „V“ eines Unterelements von `<protocol>` abgelegt, welches hier aber `<status_case>` benannt wird.

In Bezug auf die Weiterverarbeitung gibt es folgende Regeln:

1. Ein „ERROR“ als Ergebnis einer der Prüfungen verhindert die Weiterverarbeitung des geprüften Objekts.
2. Eine oder mehrere Auffälligkeiten („WARNING“, „ERROR“) auf Fallebene bedeuten ein „WARNING“ in dem korrespondierenden Eintrag auf Dokumentenebene.
3. Wenn bei einer fallbezogenen Validierung in allen Fällen auf Status ERROR erkannt wird, muss auch für das Dokument abweichend von der Regel unter 2. der korrespondierende Eintrag auf Dokumentenebene auf Status ERROR gesetzt werden.
4. Der Status (`<status_case>`/`<status_document>`) eines Objekts kann nicht „besser“ sein als sein schlechtestes Prüfergebnis.



### Hinweis

Entspricht das Attribut `regelErrorIsWarning` aus der Tabelle DB-SDAT.ModulErfassungsjahr dem Wert `true`, dann werden Regeln mit der Strenge `hard` für dieses Modul und Erfassungsjahr weich interpretiert. In den XML-Datenflussprotokollen markiert der `rule_type` „D“ die entsprechenden Regelverletzungen. `regelErrorIsWarning` hat im Gegensatz zu Regeln aus der Tabelle DB-SDAT.Regel keinen Einfluss auf administrative Prüfungen aus DB-SDAT.Pruefung.

---

## Szenarien

Aus diesen Regeln abgeleitet, soll der Status jedes geprüften Objekts nach jeder Prüfung entsprechend dem Prüfergebnis aktualisiert werden. Daraus ergibt sich folgender Aktualisierungs- und Protokollierungsplan:

### Vor der Prüfung und Protokollierung:

- (1) → Feststellen der höchsten ID in Bezug auf vorhandene `<validation_item>`-Elemente.
- (2) → Festgestellte ID um 1 inkrementieren und als ID der anstehenden Prüfung festlegen.

**Protokollierung der fallbezogenen Prüfung:**

Nachdem die fallbezogene Prüfung erfolgt ist, ist dies auf der Fallebene und der Dokumentenebene folgendermaßen zu protokollieren:

Protokollierung auf Fallebene:

(3) → <validation\_item> der Liste hinzufügen, dabei die unter (2) ermittelte ID verwenden.

(4) → <status\_case> des Falls auslesen.

(5) → Ergebnis der Prüfung mit dem Status des Falls vergleichen. In den Fällen, in denen das Ergebnis der Prüfung schlechter ist als der aktuelle Status des Falls den Status mit dem Ergebnis der Prüfung aktualisieren.

(6) → Falls ein Ergebnis der Prüfung schlechter ist als „OK“, muss dieses als dokumentbezogenes Ergebnis „WARNING“ vermerkt werden.

Protokollierung auf Dokumentenebene:

(7) → <validation\_item> mit dem unter (4) ermittelten Prüfungsergebnis der fallbasierten Prüfung unterhalb des Elements <validation\_provider> eintragen. Falls <validation\_provider> für die eigene Einrichtung noch nicht besteht, muss er angelegt werden.

(8) → <status\_document> auslesen.

(9) → Das unter (6) ermittelte Gesamtergebnis der Prüfung muss mit dem Status des Dokuments verglichen werden. Falls das Ergebnis der Prüfung schlechter ist als der aktuelle Status des Dokuments, muss dessen Status mit dem Ergebnis der Prüfung aktualisiert werden.

**Protokollierung der dokumentenbezogenen Prüfung:**

Nachdem die dokumentenbezogene Prüfung erfolgt ist, ist dies auf der Dokumentenebene folgendermaßen zu protokollieren:

(10) → <validation\_item> mit dem Prüfungsergebnis unterhalb des Elements <validation\_provider> eintragen. Falls <validation\_provider> für die eigene Einrichtung noch nicht besteht, muss er angelegt werden.

(11) → <status\_document> auslesen.

(12) → Das Ergebnis der Prüfung mit dem Status des Dokuments vergleichen und in dem Fall, in dem das Ergebnis der Prüfung schlechter ist als der aktuelle Status des Dokuments, dessen Status mit dem Ergebnis der Prüfung aktualisieren.

**6.7 Verfahrensbezogene Spezifikationen**

Die Spezifikationsdatenbank enthält alle notwendigen Informationen für den Export der Daten für die einzelnen Verfahren. Zur besseren Übersicht werden gemeinsam mit der allgemeinen Spezifikation die spezifischen Daten je Verfahren und Erfassungsjahr in einer XML-Datei zur Verfügung gestellt, die den Krankenkassen eine weitgehend automatisierte Verarbeitung ermöglicht. Der Aufbau dieser Datei ist im Folgenden beschrieben.

### 6.7.1 Allgemeines

Am Beginn der Datei werden die Version der zugehörigen Spezifikation und das Erfassungsjahr angegeben. Außerdem wird hier festgelegt, für welchen Zeitraum Daten zu exportieren sind. Diese Angabe entspricht der `validity_period` im Header-Bereich der Exportdatei (siehe Abschnitt 6.4.3).

```
<qs-verfahren·erfassungsjahr="2016"·version="SozDat_RB_V01">
  <!-- Zeitraum, aus dem Daten geliefert werden -->
  <validity·start="2015-01-01"·end="2017-12-31"/>
```

Abbildung 67: Allgemeine Angaben

Einige Elemente der Datei sind gesondert versioniert. In diesem Fall enthalten sie die Attribute `version` und `release`. Das Attribut `version` gibt die aktuelle Versionsnummer der Komponente an, die unabhängig von anderen Komponenten hochgezählt wird. Das Attribut `release` gibt an, in welcher Version der Spezifikation diese Version der Komponente eingeführt wurde. Entspricht der Wert dieses Attributs der Versionsangabe im Element `<qs-verfahren>` handelt es sich um eine aktualisierte Version und die Verarbeitungsroutinen der Krankenkassen müssen entsprechend angepasst werden.

```
·version="1"·release="SozDat_RB_V01">
```

Abbildung 68: Versionierung



### Hinweis

Die Sonderzeichen „<“ und „>“ sind innerhalb der verwendeten Pseudocodes (Filter, Plausibilitätsregeln) durch die XML-Entitäten „&lt;“ bzw. „&gt;“ ersetzt. Dies muss beim Auswerten der Pseudocodes berücksichtigt werden.

## 6.7.2 Modul

Hier stehen Basisinformationen zum verwendeten Modul.

```
<!-- Basisinformationen zum Modul -->
<modul name="PCI"
bezeichnung="Perkutane Koronarintervention und Koronarangiographie"/>
```

Abbildung 69: Modulbeschreibung

## 6.7.3 Selektionsfilter

Der Patientenfilter sowie der Leistungs- und Medikationsfilter sind sowohl als Pseudocode als auch in Textform angegeben.

```
<!-- Verwendete QS-Filter, diese und die folgende Versionsnummern
<qs_filter>
<filter name="patientenfilter" version="1" release="SozDat_RB_V01">
<code>
Admin@erfassungsjahr-Stamm@gebjahr.&gt;= 18

UND

(
(
source(301)@quelle='301'
UND
```

Abbildung 70: Filter

## 6.7.4 Kodelisten

Innerhalb des Pseudocodes der Filter werden Kodelisten referenziert. Diese Listen sind hier mit ihren einzelnen Codes aufgeführt.

```
<!-- Listen von Codes, die in den QS-Filtern verwendet werden -->
<codelisten>
<liste name="PCI OPS_INDEX" version="1" release="SozDat_RB_V01">
<code wert="12750"/>
<code wert="12751"/>
<code wert="12752"/>
<code wert="12753"/>
<code wert="12754"/>
<code wert="12755"/>
```

Abbildung 71: Kodelisten

### 6.7.5 Wertebereiche

Die im Pseudocode referenzierten Wertebereiche sind mit ihren Minimum- und Maximumwerten angegeben.

```
<!-- Wertebereiche, die in den QS-Filtern verwendet werden (meist Zeiträume) -->
<wertebereiche>
  <wertebereich name="PCI_INDEXJAHR" min="2016-01-01" max="2016-12-31" version="1" release="SozDat_RB_V01"/>
  <wertebereich name="PCI_ZEITFILTER" min="2015-01-01" max="2017-12-31" version="1" release="SozDat_RB_V01"/>
</wertebereiche>
```

Abbildung 72: Wertebereiche

### 6.7.6 Plausibilitätsregeln

Die Angaben zu den Plausibilitätsregeln sind analog zur Tabelle RegelDatenfeld, die als Abfrage in der Spezifikationsdatenbank enthalten ist (siehe Abschnitt 2.3.2). Zusätzlich zu den einzelnen Regeln ist auch die Liste <plausibilitätsregeln> versioniert, um bei Aufnahme bzw. Wegfall einzelner Regeln eine Änderung der gesamten Liste kenntlich machen zu können.

```
<!-- Plausibilitätsregeln für die Datenprüfung. Nur harte Regeln müssen bei den Krankenkassen umgesetzt werden. -->
<plausibilitaetsregeln version="1" release="SozDat_RB_V01">
  <regel id="2" bedingung="ops.NICHTIN:PCI_OPS_INDEX:UND:ops.NICHTIN:PCI_OPS_KOMPL".meldung="Prozedur-is
    <attribut name="ops".kennung="295k.OPS.Operationsschlüssel.7/7.1.1@ops".element="root.body.case.qs_data.PCI.f
    <attribut name="ops".kennung="295s.Erbrachte Leistungen-/Einzelfallrechnung.OPS.Operationsschlüssel.7/7.2.1.Ope
    <attribut name="ops".kennung="301.Entlassungsanzeige.FAB.Operation.Prozedurenschlüssel@ops".element="root.b
    <attribut name="ops".kennung="kh_ambo.Ambulante Operation.PRZ.Prozedur.Prozedurenschlüssel@ops".element="r
  </regel>
```

Abbildung 73: Plausibilitätsregeln

Die Plausibilitätsregeln werden zusätzlich auch als Excel-Datei zur Verfügung gestellt.

### 6.7.7 Lieferfristen

Die Tabelle Lieferfrist der Spezifikationsdatenbank enthält die Angaben zu den Zeiträumen, in denen Datenlieferungen für jedes einzelne Modul und Erfassungsjahr zu exportieren sind. Dabei ist für jedes Element <lieferung> ein gesonderter Export durchzuführen. Dabei gibt das Element <erstlieferung> jeweils den Zeitraum für die reguläre Datenlieferung an. Sollte es dabei zu Fehlern kommen, kann der im Element <nachlieferung> angegebene Zeitraum für Korrekturlieferungen genutzt werden.

```
<!-- Zeiträume für Datenlieferungen -->
<lieferfristen version="1" release="SozDat_RB_V01">
  <lieferung quartal="22016">
    <erstlieferung von="2016-06-01" bis="2016-06-14"/>
    <nachlieferung von="2016-06-15" bis="2016-06-30"/>
  </lieferung>
  <lieferung quartal="32016">
    <erstlieferung von="2016-09-01" bis="2016-09-14"/>
    <nachlieferung von="2016-09-15" bis="2016-09-30"/>
  </lieferung>
```

Abbildung 74: Lieferfristen



### 6.7.8 Verwendete Datenfelder

Dieser Abschnitt enthält die Angaben zu den Datenfeldern, die im Modul verwendet werden. Die einzelnen Datenfelder sind über ihre Eingangskennung spezifiziert.

```
<!--Verwendete Datenfelder-->
<datenfelder>
  <datenfeld:kennung="301.Entlassungsanzeige.FKT.IK-des-Absenders@nummer"/>
  <datenfeld:kennung="301.Aufnahmesatz.AUF.Aufnahmegrund@aufgrund"/>
  <datenfeld:kennung="301.Aufnahmesatz.AUF.Aufnahmetag@aufndatum"/>
  <datenfeld:kennung="301.Entlassungsanzeige.ETL.Tag-der-Entlassung/Verlegung@entldatum"/>
  <datenfeld:kennung="301.Entlassungsanzeige.ETL.Entlassungs-/Verlegungsgrund@entgrund"/>
  <datenfeld:kennung="301.Entlassungsanzeige.FAB.Operationstag@datum"/>
  <datenfeld:kennung="301.Entlassungsanzeige.FAR.Operation-Prozedurenschlüssel@ons"/>
</datenfelder>
```

Abbildung 75: Verwendete Datenfelder

## 6.8 Releaseplanung

### 6.8.1 Definition von Begriffen

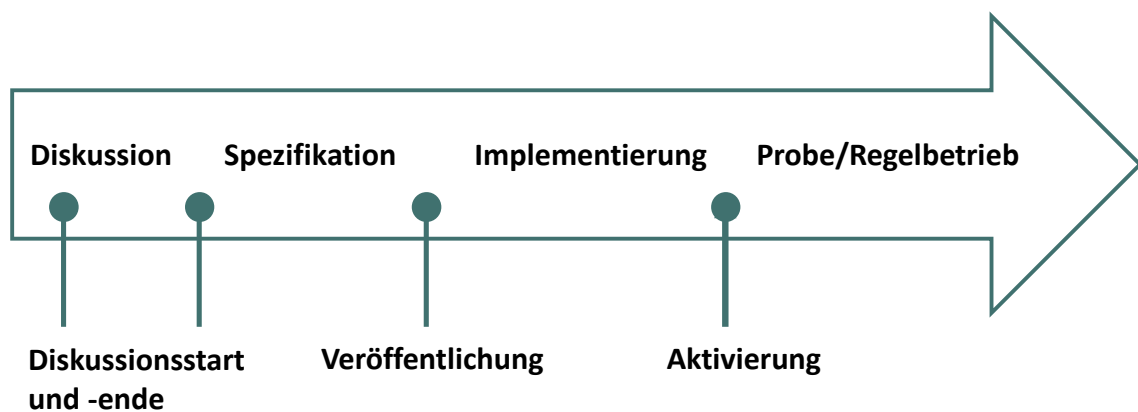


Abbildung 76: Begriffe Releaseplanung

#### Meilensteine

1. Diskussionsstarttermin  
Datum, an dem eine Diskussion/Umfrage begonnen wird (im Anschluss an die Bekanntgabe des Vorschlags/Änderungswunsches)
2. Diskussionsendtermin  
Datum, an dem eine Diskussion/Umfrage beendet wird
3. Veröffentlichungstermin  
Datum, an dem in Folge eines G-BA-Beschlusses eine Spezifikation vom IQTIG veröffentlicht wird und somit von allen Beteiligten mit der Umsetzung begonnen werden kann
4. Testbeginn  
Datum, ab dem alle Beteiligten die neue Spezifikation auf Testsystemen einsetzen

## 5. Aktivierungsdatum

Datum, an welchem eine Spezifikation gemäß den Vorgaben des G-BA in den Probe-/Echtbetrieb geht und Gültigkeit erlangt.

## Typen von Veröffentlichungen

- **Release**  
Grundsätzlich neue Funktionalität. Abstimmung erfolgt mit Teilnehmern der AG Sozialdaten und dem G-BA.
- **Service Release Korrekturen/kleine Erweiterungen zu Releases.** Abstimmung erfolgt mit Teilnehmern der AG-Sozialdaten und dem G-BA.
- **Patch**  
Aufgrund eines Notfalls (Fehler) kurzfristig erzeugtes Reparatur-Paket. Abstimmung erfolgt mit dem G-BA.
- **SpecialPatch**  
Aufgrund eines Notfalls (Fehler) kurzfristig erzeugtes Reparatur-Paket. Abstimmung erfolgt nur mit direkt betroffenen Teilnehmern und dem G-BA. Ein SpecialPatch kann nun eingesetzt werden, wenn dies keine Auswirkung auf den laufenden Gesamtprozess hat.

## 6.8.2 Informationen zu Beginn des Veröffentlichungsprozesses

### Erforderliche Angaben

- Von der Änderung betroffene Beteiligte
- Art der Änderung oder Dringlichkeit
- Detaillierte Beschreibung der vorgesehenen Änderung(en)
- Begründung für die Änderung(en)
- Zeitpunkte, zu denen
  - die Diskussionsfrist beginnt und endet (Diskussionsspanne)
  - die vorgeschlagene Änderung endgültig veröffentlicht werden soll (Veröffentlichungsdatum)
  - die Änderung verbindlich werden soll (Aktivierungsdatum)

### Arten von Änderungen und die sich daraus ergebenden Fristen

Es werden zwei Arten von Veröffentlichungen unterschieden: Normale, fristgebundene Veröffentlichungen und dringende Notfall-Veröffentlichungen.

| Änderungsart | Dauer Diskussionsfrist in Wochen | Voraussichtliche Veröffentlichung nach Ende der Diskussionsdauer in Wochen | Richtwerte zur Aktivierung nach endgültiger Veröffentlichung <sup>15</sup> |
|--------------|----------------------------------|--|--|
| Release      |                                  | 2–6 Wochen   | 3 Quartale   |

<sup>15</sup> Die Frist beginnt jeweils mit Ende des Quartals, in dem die Veröffentlichung erfolgt. Je nach Art und Umfang der Änderungen können im Einzelfall auch längere Umsetzungsfristen vereinbart werden.

| Änderungsart    | Dauer Diskussionsfrist in Wochen | Voraussichtliche Veröffentlichung nach Ende der Diskussionsdauer in Wochen | Richtwerte zur Aktivierung nach endgültiger Veröffentlichung <sup>15</sup> |
|-----------------|----------------------------------|--|--|
| Service Release | 2 Wochen                         | 1–4 Wochen   | 1 Quartal  |
| Patch           | 1 Woche                          | Je nach Absprache  | Je nach Absprache  |
| SpecialPatch    | Je nach Absprache                | Je nach Absprache  | Je nach Absprache  |

Es wird empfohlen gegebenenfalls erforderliche Testinstanzen mit Inkrafttreten eines Spezifikationsreleases durch die betreffenden Stellen bereitzustellen. Fällt der Termin des Inkrafttretens nicht auf den Beginn eines Lieferzeitraumes, besteht somit ein zeitlicher Vorlauf für den Testbetrieb im Vergleich zum Produktivbetrieb.

### 6.8.3 Entscheidungsverfahren

Grundsätzlich erfolgt die letztendliche Entscheidung über die Umsetzung einer Änderung technischer Aspekte durch das Institut nach § 137a SGB V, da es die Verantwortung für die Spezifizierung und damit auch das Funktionieren des gesamten Verfahrens trägt. Veröffentlicht die Institution als Folge der Diskussion eine modifizierte Änderung, erfolgt keine erneute Diskussion.

### 6.8.4 Praktische Umsetzung

Die Aktivierungszeitpunkte sind je nach Richtlinie langfristig festgelegt. Daraus ergibt sich der Prozess der Einordnung von Änderungen entsprechend der Änderungsart. Die vorgesehene Änderung wird vom IQTIG in einem Vorschlag begründet und einer Änderungsart (z. B. Service Release) zugeordnet. Diese Zuordnung unterliegt einem Vorbehalt, der innerhalb der Diskussionsfrist geltend gemacht werden kann.

Die Veröffentlichung erfolgt u. a. über die IQTIG-Website <http://www.iqtig.org>. Der GKV-SV und die DAS-KK werden zusätzlich per E-Mail informiert.

Um den Krankenkassen ausreichend Zeit für ggf. erforderliche Anpassungen zu geben, erfolgt für strukturelle Änderungen an der Spezifikation die Veröffentlichung der Spezifikation bis zum 30. Juni auf der Internetseite des Instituts nach § 137a SGB V. Eine Ergänzung bzw. Anpassung von Kodelisten (z. B. OPS-/ICD-Kodes) ist bis zum 30. November möglich.

### 6.8.5 Termine für die Releaseplanung

Um Planungssicherheit zu gewährleisten und angemessen auf Fehler reagieren zu können, werden die Termine zur Veröffentlichung von Spezifikationspaketen (Versionen) eines Erfassungsjahres und zu Rückmeldefristen in der Technischen Dokumentation angekündigt. Tabelle 60 stellt eine entsprechende Übersicht für das Paket der Spezifikation dar. Die genannten Termine sind als Zielwerte zu betrachten und basieren auf Vorgaben des G-BA, Abstimmungen mit Krankenkassen sowie Anforderungen aus der Umsetzung. Regulär wird die erste Version der Spezifikation eines Erfassungsjahres am 30. Juni des Vorjahres veröffentlicht.

Tabelle 60: Meilensteine der Releaseplanung der Spezifikation 2024 für den Regelbetrieb

| Frist         | Meilenstein      | Bereitstellung  | Bemerkung  |
|---------------|------------------|---|--|
| 30.06.2023    | Version 2024 V01 | Veröffentlichung auf der IQTIG-Webseite ( <a href="http://www.iqtig.org">http://www.iqtig.org</a> ) | Finale Version<br>Frist abhängig vom Plenumsbeschluss des G-BA   |
| November 2023 | Version 2024 V02 | Veröffentlichung auf der IQTIG-Webseite ( <a href="http://www.iqtig.org">http://www.iqtig.org</a> ) | Aktualisierung der ICD- und OPS-Kodes, ggf. Aktualisierung von GOP gemäß EBM-Katalog; ggf. Fehlerkorrekturen;<br>Frist abhängig vom Veröffentlichungszeitpunkt der endgültigen ICD- und OPS-Kataloge für das Jahr 2024 durch das BfArM <sup>16</sup> |

Über die in der Tabelle aufgeführten Meilensteine hinaus erfolgt eine regelmäßige Abstimmung mit Krankenkassen und weiteren Beteiligten (z. B. Vertrauensstelle). Zudem wurden neue Meilensteine in den Prozess der Systempflege integriert (z. B. Alphaversionen) und Festlegungen getroffen, die die Qualität der Spezifikation erhöhen und die Richtlinienkonformität sicherstellen (z. B. werden wesentliche Änderungen nur im Rahmen finaler Versionen berücksichtigt).

### 6.8.6 Fehler- und Changemanagement

Das IQTIG empfiehlt für die Optimierung der Zusammenarbeit mit den beteiligten Stellen die folgenden Aktivitäten:

- Meldung von festgestellten Fehlern
- Verbreitung von Änderungsvorschlägen
- Abstimmung von Terminen und Umsetzungen im Rahmen der Releaseplanung
- Erfahrungsaustausch, um eine möglichst einheitliche Vorgehensweise zu ermöglichen
- Abstimmung der Spezifikationsänderungen

Vorschläge, Fehlermeldungen und Diskussionspunkte können per E-Mail an den Verfahrenssupport oder über die Kommunikationsplattform mitgeteilt werden.

Sollten Sie keine Zugangsdaten zur Kommunikationsplattform haben, lassen Sie sich bitte bei uns registrieren.

#### Ihr Ansprechpartner:

Herr Klaus Rittmeier

Telefon: (+4930) 58 58 26-340

Telefax: (+4930) 58 58 26-341

[verfahrenssupport@iqtig.org](mailto:verfahrenssupport@iqtig.org)

<sup>16</sup> BfArM: Bundesinstitut für Arzneimittel und Medizinprodukte