



Institut für Qualitätssicherung und
Transparenz im Gesundheitswesen

Technische Dokumentation zur Basisspezifikation für Datenannahmestellen

Erfassungsjahr 2019

Erstellt im Auftrag des
Gemeinsamen Bundesausschusses

Stand: 01. Oktober 2018

Impressum

Thema:

Technische Dokumentation zur Spezifikation für Datenannahmestellen. Erfassungsjahr 2019

Auftraggeber:

Gemeinsamer Bundesausschuss

Datum der Veröffentlichung:

01. Oktober 2018

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

<https://www.iqtig.org>

Hinweis:

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

Inhaltsverzeichnis

| | |
|---|----|
| Tabellenverzeichnis..... | 9 |
| Abbildungsverzeichnis..... | 13 |
| Abkürzungsverzeichnis..... | 15 |
| Änderungsindex | 19 |
| Leseanleitung | 26 |
| 1 Einleitung..... | 28 |
| 1.1 Spezifikationsbegriff..... | 28 |
| 1.1.1 Benennungsschema für Spezifikationspakete | 29 |
| 1.1.2 Benennungsschema für Spezifikationskomponenten..... | 31 |
| 1.2 Zielsetzung der Technischen Dokumentation..... | 32 |
| 1.3 Releaseplanung | 32 |
| 1.4 Freiwillige und landesbezogene Verfahren..... | 34 |
| 1.4.1 Besonderheiten der Qualitätssicherung in Hessen | 34 |
| 1.4.2 Übermittlung von Datumsangaben..... | 37 |
| A Prozesse | 38 |
| 1 QS-Dokumentation..... | 40 |
| 1.1 Datenannahme..... | 40 |
| 1.1.1 Datenübertragung und Authentifizierung..... | 40 |
| 1.1.2 Fristen für die Datenannahme und -weiterleitung | 41 |
| 1.2 Datenprüfung..... | 41 |
| 1.2.1 Paket-Prüfungen (nur LQS/LKG)..... | 42 |
| 1.2.2 Dateiprüfungen | 42 |
| 1.3 Pseudonymisierung der LE-Daten..... | 44 |
| 1.4 Weitere Datenverarbeitungen | 47 |
| 1.5 XML-Verschlüsselung | 49 |
| 1.6 Datenübermittlung..... | 50 |
| 1.6.1 Dateibenennung (LQS/LKG) | 50 |
| 1.6.2 Ausgangsprüfung..... | 50 |
| 1.6.3 Datenweiterleitung | 50 |
| 1.7 Rückprotokollierung durch die Datenannahmestellen..... | 51 |

| | | |
|-------|--|----|
| 1.7.1 | Empfangsbestätigung..... | 51 |
| 1.7.2 | Miniprotokoll..... | 52 |
| 1.7.3 | Datenflussprotokoll..... | 52 |
| 1.7.4 | Erzeugung der Antwortdatei für die Übertragung via E-Mail (LQS/LKG)..... | 52 |
| 1.7.5 | Empfangsbestätigung durch die VST..... | 53 |
| 1.7.6 | Referenz für die HTML-Formatierung des Rückprotokolls..... | 53 |
| 1.8 | Weiterleitung der Datenflussprotokolle der BAS..... | 53 |
| 1.9 | Datenhaltung (LQS/LKG) – QSKH-Richtlinie | 54 |
| 1.10 | Zusammenfassung | 55 |
| 2 | Risikostatistik..... | 57 |
| 2.1 | Datenübermittlung..... | 57 |
| 2.2 | Datenprüfung und Rückprotokollierung | 58 |
| 2.3 | Datenweiterleitung | 60 |
| 3 | Sollstatistik | 62 |
| 3.1 | Erzeugung..... | 62 |
| 3.1.1 | Erzeugung durch die LQS/LKG..... | 63 |
| 3.1.2 | Erzeugung durch die DAS-SV | 64 |
| 3.1.3 | Erzeugung durch die KVen | 64 |
| 3.2 | Pseudonymisierung..... | 67 |
| 3.3 | Datenprüfung..... | 67 |
| 3.3.1 | Prüfung durch die LQS/LKG | 67 |
| 3.3.2 | Prüfung durch die KV und DAS-SV..... | 71 |
| 3.4 | Verschlüsselung..... | 72 |
| 3.4.1 | Transportverschlüsselung durch die LQS/LKG | 72 |
| 3.4.2 | XML-Verschlüsselung durch die KV und DAS-SV | 72 |
| 3.5 | Datenübermittlung..... | 72 |
| 3.5.1 | Datenweiterleitung bei LQS/LKG..... | 72 |
| 3.5.2 | Datenweiterleitung bei KV/DAS-SV..... | 73 |
| 3.6 | Rückprotokollierung..... | 73 |
| 3.6.1 | Rückprotokollierung durch LQS/LKG..... | 73 |
| 3.6.2 | Rückprotokollierung an KV/DAS-SV | 74 |
| 3.7 | Konformitätserklärung..... | 74 |

| | | |
|--------|--|-----|
| 4 | Auswertungen | 75 |
| 5 | Allgemeine Regelungen zur Datenübermittlung..... | 80 |
| 5.1 | Datenübermittlung..... | 80 |
| 5.1.1 | Gesicherte Datenübertragung..... | 80 |
| 5.1.2 | Abgrenzung von Test-, Probe- und Regelbetrieb | 90 |
| 5.1.3 | Allgemeine Prüfungen des XML-Dokuments | 94 |
| 5.2 | Rückprotokollierung..... | 96 |
| 5.2.1 | Funktion von Empfangsbestätigung und Datenflussprotokoll im Datenfluss..... | 96 |
| 5.2.2 | Die Rückprotokollierung..... | 98 |
| 5.3 | Datenfluss | 108 |
| 5.3.1 | Datenfluss der QS-Daten..... | 108 |
| 5.3.2 | Datenfluss der Rückprotokolle..... | 112 |
| B | Komponenten | 114 |
| 1 | QS-Filter..... | 119 |
| 1.1 | Anmerkungen zur Struktur der Spezifikationsdatenbank für QS-Filter | 119 |
| 1.2 | Grundlegende Tabellen der Datenbank..... | 120 |
| 1.2.1 | Module (Datensätze der QS-Dokumentation) | 120 |
| 1.2.2 | Struktur der Datensatzdefinitionen | 122 |
| 1.2.3 | OPS-Listen | 126 |
| 1.2.4 | ICD-Listen | 127 |
| 1.2.5 | FAB-Listen..... | 127 |
| 1.2.6 | EBM-Listen | 128 |
| 1.2.7 | Entgelt-Listen | 128 |
| 1.2.8 | Versionsverwaltung..... | 128 |
| 1.2.9 | Meta-Tabellen | 130 |
| 1.2.10 | DB-Änderungen gegenüber der Vorgängerversion..... | 130 |
| 1.3 | Der QS-Filter-Datensatz | 131 |
| 1.3.1 | Der QS-Filter-Eingangsdatensatz..... | 131 |
| 1.3.2 | Der QS-Filter-Ausgangsdatensatz..... | 138 |
| 1.4 | Der Algorithmus zur Ermittlung der Dokumentationspflicht..... | 143 |
| 1.4.1 | Einleitung und Überblick | 143 |
| 1.4.2 | Leistungsbereichsbezogene Einschlusskriterien | 144 |

| | | |
|-------|--|-----|
| 1.4.3 | Administrative Einschlusskriterien..... | 148 |
| 1.4.4 | Struktur und Syntax der Auslösebedingungen..... | 151 |
| 1.4.5 | Stufen der Dokumentationsverpflichtung | 154 |
| 1.4.6 | Fehlerprüfung..... | 155 |
| 1.5 | Sollstatistik | 157 |
| 1.5.1 | Sollstatistik des Moduls PCI gemäß Qesü-RL | 157 |
| 1.5.2 | Sollstatistik des Moduls CHE gemäß DeQS-RL | 158 |
| 2 | QS-Dokumentation..... | 160 |
| 2.1 | Anmerkungen zur Struktur der Spezifikation zur QS-Dokumentation..... | 160 |
| 2.2 | Patientenidentifizierende Daten zur Follow-up-Erhebung | 162 |
| 2.3 | Datenfeldbeschreibung..... | 163 |
| 2.3.1 | Dokumentationsmodule (Datensätze) | 164 |
| 2.3.2 | Teildatensätze | 166 |
| 2.3.3 | Datenfelder (Bogenfelder) | 172 |
| 2.3.4 | Überschriften | 182 |
| 2.3.5 | Ausfüllhinweise | 182 |
| 2.4 | Plausibilitätsprüfungen | 184 |
| 2.4.1 | Die Regeltabelle | 184 |
| 2.4.2 | Regelsyntax | 186 |
| 2.4.3 | Funktionen | 192 |
| 2.4.4 | Syntaxvariablen | 195 |
| 2.4.5 | Einzelregeln | 196 |
| 2.4.6 | Teildatensatzübergreifende Regeln | 196 |
| 2.4.7 | Feldgruppenregeln | 198 |
| 2.4.8 | Prüfung von Feldeigenschaften..... | 207 |
| 2.4.9 | Verfahren für die Evaluation von Regeln | 212 |
| 2.5 | Listen von Schlüsselkodes (OPS, ICD-10-GM) | 215 |
| 2.5.1 | OPS-Listen | 215 |
| 2.5.2 | ICD-Listen | 216 |
| 2.6 | Exportfeldbeschreibung..... | 217 |
| 2.6.1 | Exportmodule..... | 218 |
| 2.6.2 | Exportdatensatz | 219 |

| | | |
|--------|--|-----|
| 2.7 | Versionierung..... | 225 |
| 2.7.1 | Grundlegende Definitionen..... | 225 |
| 2.7.2 | Delta-Informationen zur vorhergehenden Version | 226 |
| 2.7.3 | Abgrenzung zwischen Erfassungsjahren und Datensatzformaten..... | 229 |
| 2.7.4 | Version des Exportverfahrens | 230 |
| 2.8 | Administrative Objekte | 230 |
| 2.8.1 | CSV/XML-Mapping in der Spezifikationsdatenbank (QSDOK) | 231 |
| 2.8.2 | Datenservices | 234 |
| 2.8.3 | Prüfschritte..... | 235 |
| 3 | Risikostatistik..... | 239 |
| 3.1 | Allgemeine Anmerkungen über die Struktur der Spezifikationsdatenbank zur Risikostatistik | 239 |
| 3.2 | Tabellenstruktur der Datenbank..... | 239 |
| 3.3 | Struktur der Datensatzdefinitionen | 239 |
| 3.3.1 | Datensätze..... | 239 |
| 3.3.2 | Teildatensätze | 240 |
| 3.3.3 | Felder der Teildatensätze..... | 241 |
| 3.3.4 | Felder..... | 241 |
| 3.3.5 | Basistypen | 241 |
| 3.3.6 | Schlüssel | 242 |
| 3.3.7 | Schlüsselwerte..... | 243 |
| 3.4 | ICD-Listen und OPS-Listen | 243 |
| 3.5 | Delta-Informationen zur vorhergehenden Version | 243 |
| 3.6 | Versionierung..... | 244 |
| 3.7 | Der Risikostatistik-Eingangsdatensatz | 244 |
| 3.8 | Felder der Risikostatistik | 246 |
| 3.9 | Algorithmus zur Berechnung der Risikostatistik | 247 |
| 3.10 | Struktur und Syntax der Bedingungen | 248 |
| 3.10.1 | Die Variablen der Bedingungen | 248 |
| 3.10.2 | Diagnose und Prozedurenlisten | 249 |
| 3.10.3 | Die Operatoren der Bedingungen | 249 |
| 3.11 | Fehlerprüfung..... | 250 |
| 4 | XML-Schema..... | 252 |

| | | |
|-------|---|-----|
| 4.1 | Kompositionsmodell | 252 |
| 4.2 | Schnittstellen..... | 253 |
| 4.3 | Darstellung der XML-Struktur | 256 |
| 4.4 | Aufbau der XML-Exportdatei | 257 |
| 4.4.1 | Namensräume | 257 |
| 4.4.2 | Wurzelement <root> | 258 |
| 4.4.3 | Header-Bereich | 259 |
| 4.4.4 | Body-Bereich | 272 |
| 4.5 | Aufbau XML-Dokument – Sollstatistik | 285 |
| 5 | Tools | 286 |
| 5.1 | Java-Installation | 286 |
| 5.2 | Datenprüfprogramm | 287 |
| 5.2.1 | Umfang der Prüfungen | 287 |
| 5.2.2 | Ausgangskontrolle vor Versand | 287 |
| 5.2.3 | Programmaufruf | 288 |
| 5.2.4 | Verzeichnisstruktur | 291 |
| 5.2.5 | Ausgabe | 292 |
| 5.2.6 | Grafische Oberfläche | 293 |
| 5.2.7 | Programmierschnittstelle – API | 293 |
| 5.3 | Verschlüsselungspaket | 293 |
| 5.3.1 | XPacker – XML-Verschlüsselung | 294 |
| 5.3.2 | TPacker – Transportverschlüsselung | 296 |
| 5.3.3 | TPacker und XPacker ohne externe Abhängigkeiten | 299 |
| 5.3.4 | Programmierschnittstelle – API | 299 |
| 5.3.5 | GPacker | 299 |
| 5.4 | LE-Pseudonymisierungsprogramm | 299 |
| C | Anhang | 300 |
| | Austausch von Pseudonymen im IQTIG | 300 |
| | Zeitfenster zum Austausch der Patientenpseudonyme | 300 |
| | Spezifikation des Austauschverfahrens von Patientenpseudonymen | 301 |
| | Austausch der Leistungserbringerpseudonyme beim Wechsel einer Datenannahmestelle | 302 |
| | Glossar | 304 |

Tabellenverzeichnis

| | |
|--|-----|
| Tabelle 1: Meilensteine der Releaseplanung der Spezifikation 2018 für den Regelbetrieb..... | 33 |
| Tabelle 2: Übersicht über die Prozesse QS-Dokumentation, Sollstatistik und Risikostatistik und ihre Unterprozesse | 38 |
| Tabelle 3: Pseudonymisierung der LE-Daten bei Qesü- und DeQS-Verfahren | 45 |
| Tabelle 4: Kodierung der Landesgeschäftsstellen der KVen | 47 |
| Tabelle 5: Bisher vorgesehene Datenübertragungswege | 51 |
| Tabelle 6: XML-Schemata der Datenflussprotokolle..... | 52 |
| Tabelle 7: Aufgaben der Datenannahmestellen | 55 |
| Tabelle 8: Länderkodes der Landesgeschäftsstellen..... | 58 |
| Tabelle 9: Zuständigkeit für die Erstellung der Sollstatistik | 62 |
| Tabelle 10: Sollstatistik - Quantifizierung übermittelter Vorgänge - Umsetzungsbeispiele..... | 66 |
| Tabelle 11: Berücksichtigung verschiedener Konstellationen von Entgeltarten in der Sollstatistik | 69 |
| Tabelle 12: Softwareprodukte zum Entpacken von zip-Dateien mit 256-Bit AES- Verschlüsselung | 76 |
| Tabelle 13: Dateinamensbestandteile von Auslieferungseinheiten und deren beinhaltete Dateien..... | 77 |
| Tabelle 14: Benennungselemente der Exportdateien | 84 |
| Tabelle 15: Übersicht über die Schnittstellenschemata der DAS | 95 |
| Tabelle 16: Ausfüllen der Elemente eines Validation-Items in Abhängigkeit von den Fehlerarten..... | 101 |
| Tabelle 17: Mögliche Fehlerarten in Prüfprozessen | 102 |
| Tabelle 18: Beispiele von Fehlermeldungen | 102 |
| Tabelle 19: Übersicht über die Exportverfahren..... | 111 |
| Tabelle 20: Struktur der Tabelle Modul | 121 |
| Tabelle 21: Struktur der Tabelle Ds..... | 122 |
| Tabelle 22: Struktur der Tabelle Tds | 123 |
| Tabelle 23: Struktur der Tabelle BasisTyp..... | 124 |
| Tabelle 24: Struktur der Tabelle Schluessel | 125 |
| Tabelle 25: Struktur der Tabelle SchluesselWert..... | 126 |
| Tabelle 26: Struktur der Tabelle Version | 129 |
| Tabelle 27: Ausschnitt der Tabelle Ds | 131 |
| Tabelle 28: Felder des QS-Filter-Eingangsdatensatzes nach §301..... | 132 |
| Tabelle 29: Felder des QS-Filter-Eingangsdatensatzes nach § 295 | 136 |
| Tabelle 30: Felder des QS-Filter-Ausgangsdatensatzes nach §301..... | 139 |
| Tabelle 31: Felder des QS-Filter-Ausgangsdatensatzes nach § 295..... | 141 |
| Tabelle 32: Beispielhafter Auszug einer Sollstatistik 2015 für QS-Verfahren gemäß QSKH-RL | 145 |
| Tabelle 33: Struktur der Tabelle ModulAusloeser | 146 |
| Tabelle 34: Struktur der Tabelle AdminKriterium | 150 |

| | |
|---|-----|
| Tabelle 35: Basistypen der Variablen..... | 151 |
| Tabelle 36: Präzedenz und Assoziativität der Operatoren..... | 153 |
| Tabelle 37: Stufen der Dokumentationsverpflichtung..... | 155 |
| Tabelle 38: Fehlercodes des QS-Filters..... | 155 |
| Tabelle 39: Struktur der Tabelle PseudonymVerfahren..... | 163 |
| Tabelle 40: Struktur der Tabelle Modul..... | 165 |
| Tabelle 41: Struktur der Tabelle Bogen..... | 167 |
| Tabelle 42: Inhalte der Tabelle BogenTyp..... | 168 |
| Tabelle 43: Struktur der Tabelle BogenFeld..... | 173 |
| Tabelle 44: Struktur der Tabelle Feld..... | 175 |
| Tabelle 45: Struktur der Tabelle BasisTyp..... | 176 |
| Tabelle 46: Struktur der Tabelle Schluessel..... | 177 |
| Tabelle 47: Struktur der Tabelle SchluesselWert..... | 179 |
| Tabelle 48: Schlüssel mit spezieller Sortierung (pTMamma)..... | 180 |
| Tabelle 49: Struktur der Tabelle Abschnitt..... | 182 |
| Tabelle 50: Arten von Hinweistypen..... | 183 |
| Tabelle 51: Tabelle RegelTyp..... | 184 |
| Tabelle 52: Struktur der Tabelle Regeln..... | 184 |
| Tabelle 53: Struktur der Tabelle RegelFelder..... | 185 |
| Tabelle 54: Struktur der Tabelle MehrfachRegel..... | 186 |
| Tabelle 55: Basistypen der Datenfelder in den Plausibilitätsregeln..... | 186 |
| Tabelle 56: Präzedenz und Assoziativität der Operatoren..... | 189 |
| Tabelle 57: Typen von Feldgruppen..... | 198 |
| Tabelle 58: Struktur der Tabelle FeldGruppe..... | 199 |
| Tabelle 59: Struktur der Tabelle FeldgruppeFelder..... | 201 |
| Tabelle 60: Formale Definition einer Feldgruppe..... | 202 |
| Tabelle 61: Plausibilitätsregeln der Feldgruppe NEO:OPArt in Spezifikation 2017..... | 206 |
| Tabelle 62: Definition der Feldgruppe NEO:OPArt in Spezifikation 2017..... | 206 |
| Tabelle 63: Beispiel für Wertebereichsgrenzen (Datenfeld OPDAUER)..... | 210 |
| Tabelle 64: Beispiel für Prüfungen von Wertebereichsgrenzen..... | 210 |
| Tabelle 65: Beispiele für die Evaluation von Regeln..... | 213 |
| Tabelle 66: Identitätsprüfung zwischen dokumentierten OPS-Kodes und Kodes von OPS- Listen..... | 216 |
| Tabelle 67: Struktur der Tabelle ExportModul..... | 218 |
| Tabelle 68. Struktur der Tabelle ZusatzFeld..... | 221 |
| Tabelle 69. Struktur der Tabelle Ersatzfeld..... | 222 |
| Tabelle 70. Struktur der Tabelle ErsatzFuerFeld..... | 223 |
| Tabelle 71. Struktur der Tabelle DeltaNeu..... | 227 |
| Tabelle 72: Struktur der Tabelle DeltaAttribut..... | 227 |
| Tabelle 73: Struktur der Tabelle DeltaGeloesch..... | 228 |
| Tabelle 74: Inhalt der Tabelle TabellenFeldStruktur (fkTabellenFeldStruktur = Regeln)..... | 229 |

| | |
|---|-----|
| Tabelle 75: Überblick über neben der QS-Dokumentation weitere potenzielle Workflows mit definierten Prüfschritten | 235 |
| Tabelle 76: Felder der Abfrage vPruefung | 237 |
| Tabelle 77: Struktur der Tabelle Ds | 240 |
| Tabelle 78: Struktur der Tabelle Tds..... | 240 |
| Tabelle 79: Struktur der Tabelle BasisTyp..... | 241 |
| Tabelle 80: Struktur der Tabelle Schluessel..... | 242 |
| Tabelle 81: Struktur der Tabelle SchluesselWert | 243 |
| Tabelle 82: Felder des Risikostatistik-Eingangsdatensatzes | 244 |
| Tabelle 83: Felder der Risikostatistik | 246 |
| Tabelle 84: fallbezogene Risikostatistik | 247 |
| Tabelle 85. Struktur der Tabelle AdminKriterium..... | 248 |
| Tabelle 86. Fehlercodes | 250 |
| Tabelle 87: Verwendbare Schemata und Ablageort | 253 |
| Tabelle 88: Weiche Schemavarianten für das Datenprüfprogramm | 255 |
| Tabelle 89: XML-Schemata für die Rückprotokollierung | 256 |
| Tabelle 90: Symbole in den XML-Schema-Diagrammen | 256 |
| Tabelle 91: Root-Element – Attribute | 258 |
| Tabelle 92: Kind-Elemente des Elements document | 260 |
| Tabelle 93: Kind-Elemente des Elements software | 262 |
| Tabelle 94: Kind-Elemente des Elements information_system | 262 |
| Tabelle 95: Angabe des betreffenden Datenflusses | 263 |
| Tabelle 96: Angabe des betreffenden Datenfluss-Ziels | 263 |
| Tabelle 97: Attribute des Elements header/provider | 264 |
| Tabelle 98: Attribute des Elements header/protocol | 265 |
| Tabelle 99: Attribute des Elements validation_provider | 266 |
| Tabelle 100: Attribute des Elements validation_item | 267 |
| Tabelle 101: Attribute des Elements status_document | 268 |
| Tabelle 102: Attribut des Elements status..... | 268 |
| Tabelle 103: Attribut des Elements error | 269 |
| Tabelle 104: Kind-Elemente des Elements error..... | 269 |
| Tabelle 105: feedback_key: Kind-Elemente..... | 271 |
| Tabelle 106: Leistungserbringeridentifizierende Daten im kollektiven, selektiven und stationären Bereich..... | 273 |
| Tabelle 107: Attribute des Elements cases | 275 |
| Tabelle 108: Verfahrenskennung: „pseud_procedure“ | 275 |
| Tabelle 109: Kind-Elemente des Elements case_admin | 277 |
| Tabelle 110: Kind-Element des Elements statistic | 280 |
| Tabelle 111: Attribut des Elements sent..... | 281 |
| Tabelle 112: Kind-Elemente des Elements statistic/sent..... | 281 |
| Tabelle 113: Attribute des Elements statistic | 282 |
| Tabelle 114: Kind-Elemente des Elements statistic/processed | 282 |

Tabelle 115: Schemata der Sollstatistik 285

Abbildungsverzeichnis

| | |
|--|-----|
| Abbildung 1: Beispiel für Benennung von Paketen und Komponenten für die Spezifikation 2017 bis 2018..... | 32 |
| Abbildung 2: Überblick über die Prozesse und Werkzeuge in der Datenannahmestelle | 40 |
| Abbildung 3: XML-Elemente für die Ermittlung von Fallzahlen | 66 |
| Abbildung 4: Beispiel einer Registrierungsnummer | 81 |
| Abbildung 5: Übersicht über die einzusetzenden Suffixe | 84 |
| Abbildung 6: Datenflüsse im Test-, Probe- und Regelbetrieb am Bsp. der Follow-up-Verfahren | 93 |
| Abbildung 7: Attribut "originator" im Prüfungs- und Fehlerprotokoll..... | 97 |
| Abbildung 8: Beispiel einer Empfangsbestätigung..... | 98 |
| Abbildung 9: Beziehungen zwischen <validation_item> im header und <validation_item> im body über die id | 104 |
| Abbildung 10: Aufnahme des XSLT-Pfads in das XML-Protokoll | 107 |
| Abbildung 11: HTML-Darstellung nach einer XSLT-Transformation am Beispiel einer QS-Übertragung..... | 107 |
| Abbildung 12: Übersicht der Datenflüsse direkte/indirekte PID-/Nicht-PID-Verfahren..... | 110 |
| Abbildung 13: Übersicht der Datenflüsse der Rückprotokollierung | 113 |
| Abbildung 14: Zuordnung der Version des QS-Filters zu den Behandlungsfällen: Kriterium ist das Aufnahmedatum | 130 |
| Abbildung 15: Tabellen und Relationen der Datenfeldbeschreibung..... | 163 |
| Abbildung 16: Teildatensatzstruktur des Datensatzes PCI | 168 |
| Abbildung 17: Fallbeispiel HEP in Bezug auf Dokumentation und Sollstatistik | 171 |
| Abbildung 18: Feldgruppe NEO:OPart auf dem Dokumentationsbogen (Spezifikation 2017). 205 | |
| Abbildung 19: Algorithmus zur Evaluation von Plausibilitätsregeln | 213 |
| Abbildung 20: Beziehungen der administrativen Objekte (Prüfungen)..... | 231 |
| Abbildung 21: Beziehungen der administrativen Objekte (Datenservices, Mapping-Informationen)..... | 231 |
| Abbildung 22: Ergebnis der Abfrage vExportZieleXml | 233 |
| Abbildung 23: Beispiel für XPath-Ausdrücke in der Tabelle ExportZielXml in Verbindung mit weiteren Informationen | 234 |
| Abbildung 24: Tabelle Datenservice..... | 234 |
| Abbildung 25: Verfahrensbezogene Datenservices | 235 |
| Abbildung 26: HTML-Ansicht der Prüfschritte innerhalb der QS-Dokumentation | 236 |
| Abbildung 27: Dateiordner der Schnittstellen-Schemata | 252 |
| Abbildung 28: Root-Element und Kind-Elemente header und body | 258 |
| Abbildung 29: Aufbau des Elements header..... | 259 |
| Abbildung 30: Aufbau des Elements document..... | 260 |
| Abbildung 31: Aufbau des Elements software..... | 261 |
| Abbildung 32: Aufbau des Elements information_system | 262 |

| | |
|--|-----|
| Abbildung 33: Aufbau des Elements provider | 264 |
| Abbildung 34: Aufbau des Elements header/protocol | 265 |
| Abbildung 35: Aufbau und Kind-Elemente des Elements validation_provider..... | 266 |
| Abbildung 36: Aufbau und Kind-Elemente des Elements validation_item | 267 |
| Abbildung 37: Aufbau des Elements status_document..... | 268 |
| Abbildung 38: Aufbau und Kind-Elemente des Elements status | 268 |
| Abbildung 39: Aufbau des Elements error..... | 269 |
| Abbildung 40: Aufbau und Attribute des Elements encryption(Krankenhaus) | 270 |
| Abbildung 41: Aufbau und Attribute des Elements encryption(AP-selektiv/-kollektiv) | 271 |
| Abbildung 42: Aufbau des Elements feedback_key..... | 271 |
| Abbildung 43: Aufbau des Elements body | 272 |
| Abbildung 44: Aufbau des Elements body/data_container..... | 272 |
| Abbildung 45: Aufbau des Elements care_provider – kollektiv-, selektivvertraglich | 273 |
| Abbildung 46: Aufbau des Elements care_provider – Krankenhaus..... | 273 |
| Abbildung 47: Aufbau des Elements cases | 274 |
| Abbildung 48: Aufbau des Elements case | 276 |
| Abbildung 49: Aufbau des Elements case_admin..... | 276 |
| Abbildung 50: Aufbau des Elements patient | 278 |
| Abbildung 51: Aufbau des Elements pid | 279 |
| Abbildung 52: Aufbau des Elements case_admin/protocol..... | 279 |
| Abbildung 53: Aufbau des Elements statistic..... | 280 |
| Abbildung 54: Aufbau des Elements sent | 281 |
| Abbildung 55: Aufbau des Elements processed..... | 282 |
| Abbildung 56: Integration des MDS in das Primärmodul | 283 |
| Abbildung 57: Diagramme „Bogen komplex“ und „Bogen einfach“ | 284 |
| Abbildung 58: Ausprägungen des qs_data-Elements (Erfassungsmodule) | 285 |
| Abbildung 59: Sollstatistik-Modul..... | 285 |
| Abbildung 60: Weiche Schemavariante für das DPP..... | 291 |
| Abbildung 61: Beispiel einer typischen Verzeichnisstruktur..... | 292 |
| Abbildung 62: Beispiel für eine Index.html Datei im Ordner <output>/html | 292 |
| Abbildung 63: Grafische Oberfläche des Datenprüfprogramms | 293 |
| Abbildung 64: Verschlüsselung eines XML-Elements (qs_data) | 294 |
| Abbildung 65: Darstellung der Überlappung von Datenlieferungen | 301 |

Abkürzungsverzeichnis

| Abkürzung | Bedeutung |
|------------|---|
| AES | Advanced Encryption Standard (Verschlüsselungsalgorithmus) |
| AIS | Arztinformationssystem |
| AG | Arbeitsgruppe |
| AG Qesü-RL | ArbeitsgruppeQesü-RL des G-BA |
| ASCII | American Standard Code for Information Interchange (Amerikanischer Standard-Code für den Informationsaustausch) |
| BAS | Bundesauswertungsstelle |
| BE | Bundesebene |
| BPfIV | Bundespflegesatzverordnung (Verordnung zur Regelung der Krankenhauspflegesätze) |
| BSI | Bundesamt für Sicherheit in der Informationstechnik |
| BSNR | Betriebsstättennummer |
| CSV | Comma-separated values (Dateiformat) |
| DAS | Datenannahmestelle |
| DAS-SV | Datenannahmestellen für selektivvertraglich erbrachte Leistungen |
| DB | Datenbank |
| DeQS-RL | Richtlinie (des G BA) zur datengestützten einrichtungsübergreifenden Qualitätssicherung |
| DIMDI | Deutsches Institut für Medizinische Dokumentation und Information |
| dv | direkte Verfahren |
| DMP | Disease-Management-Programm |
| DPP | Datenprüfprogramm |
| DRG | Diagnosis Related Groups (diagnosebezogene Fallgruppen) |
| DÜV | Datenübermittlungsvereinbarung |
| EBM | Einheitlicher Bewertungsmaßstab |
| eGK | elektronische Gesundheitskarte |
| FAB | Fachabteilung |
| FU | Follow-up |
| G-BA | Gemeinsamer Bundesausschuss |

| Abkürzung | Bedeutung |
|------------------|--|
| GKV | Gesetzliche Krankenversicherung |
| GOP | Gebührenordnungsposition |
| GQH | Geschäftsstelle Qualitätssicherung Hessen |
| GUID | Globally Unique Identifier |
| HTML | Hypertext Markup Language (Hypertext-Auszeichnungssprache) |
| ICD | International Statistical Classification of Diseases and Related Health Problems (Internationale statistische Klassifikation der Krankheiten und verwandter Gesundheitsprobleme) |
| ICD-10-GM | Internationale statistische Klassifikation der Krankheiten und verwandter Gesundheitsprobleme – 10. Revision – German Modification |
| ID | Identifikationsnummer |
| IK | Institutionskennzeichen |
| IKNR | Institutionskennzeichennummer |
| IQTIG | Institut für Qualität und Transparenz im Gesundheitswesen |
| IV | integrierte Versorgung |
| iv | indirekte Verfahren |
| JRE | Java Runtime Environment (Java-Laufzeit-Umgebung) |
| JVM | Java Virtual Machine, ist Teil der Java-Laufzeitumgebung |
| K | Kann-Feld |
| KBV | Kassenärztliche Bundesvereinigung |
| KH | Krankenhaus, Krankenhäuser |
| KHEntgG | Gesetz über die Entgelte für voll- und teilstationäre Krankenhausleistungen (Krankenhausentgeltgesetz) |
| KIS | Krankenhausinformationssystem |
| KV | Kassenärztliche Vereinigung |
| KVDT | Kassenärztliche Vereinigung-Datentransfer (Datenformat) |
| LANR | Lebenslange Arztnummer |
| LE | Leistungserbringer |
| LE-amb | Leistungserbringer ambulant |
| LE-stat | Leistungserbringer stationär |
| LID | Leistungserbringeridentifizierenden Daten |

| Abkürzung | Bedeutung |
|-----------|--|
| LKG | Landeskrankenhausgesellschaft(en) |
| LQS | Landesgeschäftsstelle(n) für Qualitätssicherung |
| M | Muss-Feld |
| MDS | Minimaldatensatz |
| MKU | Mechanische Kreislaufunterstützung |
| MVZ | Medizinisches Versorgungszentrum |
| NHS | Neugeborenen-Hörscreening |
| OPS | Operationen- und Prozedurenschlüssel |
| OR | ODER-Operator |
| PB | Probetrieb |
| PCI | Perkutane Koronarintervention (PCI) und Koronarangiographie |
| PID | Patientenidentifizierende Daten |
| Plan QI | planungsrelevanten Qualitätsindikatoren |
| PR | Arztpraxis |
| Qb-R | Qualitätsbericht der Krankenhäuser |
| Qesü-RL | Richtlinie (des G BA) gemäß §92 Abs. 1 Satz 2 Nr. 13 i.V.m. §137 Abs. 1 Nr. 1 SGB V über die einrichtungs- und sektorenübergreifenden Maßnahmen der Qualitätssicherung |
| QS | Qualitätssicherung |
| QSDOK | Access-Datenbank, in der die QS-Dokumentation spezifiziert wird |
| QSF | Access-Datenbank, in der der QS-Filter spezifiziert wird |
| QSKH-RL | Richtlinie (des G BA) gemäß §137 Abs. 1 SGB V i.V.m. §135a SGB V über Maßnahmen der Qualitätssicherung für nach §108 SGB V zugelassene Krankenhäuser |
| RMB | Rückmeldeberichte |
| RL | Richtlinie |
| RSA | Verfahren zur Datenverschlüsselung, entwickelt von R. Rivest, A. Shamir und L. Adleman |
| SGB | Sozialgesetzbuch |
| SGB V | Sozialgesetzbuch Fünftes Buch |
| SNK | sichere Netze der KVen |
| SWA | Softwareanbieter |

| Abkürzung | Bedeutung |
|------------------|--|
| TB | Testbetrieb |
| TDS | Teildatensatz |
| TPacker | Programm für die Transportverschlüsselung |
| URL | Uniform Resource Locator (einheitlicher Ressourcenzeiger) |
| V | Versionierung |
| VST | Vertrauensstelle |
| VST-DAS | Vertrauensstelle des G-BA in der Funktion als Datenannahmestelle |
| VST-PSN | Vertrauensstelle des G-BA in der Funktion als Pseudonymisierungsstelle |
| XML | Extensible Markup Language |
| XPacker | Verschlüsselungsprogramm |
| XSD | XML-Schema-Datei |
| XSLT | Extensible Stylesheet Language Transformation (Programmiersprache zur Transformation von XML-Dokumenten) |
| ZIP | zipper, Abkürzung für ein Format für verlustfrei komprimierte Dateien |

Änderungsindex

Änderungen der Datenbanken im Vergleich zur Vorversion lassen sich anhand der Delta-Tabellen nachvollziehen.

Kapitelübergreifende Änderungen:

- Redaktionelle Änderungen, Konkretisierungen und Optimierung von Formulierungen
- Anpassung von Jahreszahlen, Beispielen und Abbildungen
- Anpassung von Abkürzungen

Konkrete Informationen zu den inhaltlichen Änderungen sind der Spezifikationskomponente Uebersicht_Aenderungen zu entnehmen.

Kapitelspezifische Änderungen:

Es wird ein neuer Anhang aufgenommen, der den Austausch und Abgleich der Pseudonyme erläutert (Anhang C).

Weitere spezifische Änderungen der vorliegenden technischen Dokumentation werden im Folgenden mit Bezug zur jeweiligen Version dargestellt.

| Änderung | Kapitel/Abschnitt | Version |
|---|---|----------|
| Ergänzung eines Hinweises auf die Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) | 1 (Einleitung) | 2019 V01 |
| Erweiterung bezüglich der neuen Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (Regelbetrieb (QSKH, Qesü und DeQS)) | 1.1 (Spezifikationsbegriff) | 2019 V01 |
| Ergänzung eines Hinweises auf die Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) | 1.1.1 (Benennungsschema für Spezifikationspakete) | 2019 V01 |
| Ergänzung eines Hinweises auf die Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) | 1.3 (Releaseplanung) | 2019 V01 |
| Beschreibung zum Umgang mit Minimaldatensätzen für den Leistungsbereich 15/1 für die hessische Zusatzserhebung Gynäkologie | 1.4.1 (Besonderheiten der Qualitätssicherung in Hessen) | 2019 V01 |
| Löschen des Moduls CHE_HE | 1.4.1 (Besonderheiten der Qualitätssicherung in Hessen) | 2019 V01 |
| Erweiterung der Tabelle eines bezüglich der Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) | A „Prozesse“ | 2019 V01 |

| Änderung | Kapitel/Abschnitt | Version |
|---|-------------------|-----------------|
| Ergänzung eines Hinweises auf die Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) | A 1 | 2019 V01 |
| Hinweis auf die Datenlieferfristen in den Themenspezifischen Bestimmungen der Qesü-RL bzw. der DeQS-RL festgelegt. | A 1.1.2 | 2019 V01 |
| Löschen von Hinweisen, da mit der Spezifikation 2019 alle bundesweit verpflichtenden Verfahren auf lange Überliegerverfahren umgestellt werden | A 1.2.2 | 2019 V01 |
| Anpassungen an die 2019 entfallende LE-Pseudonymisierung durch DAS bei QSKH-Modulen in Tabelle 3 | A 1.3 | 2019 V02 |
| Anpassungen an die 2019 entfallende LE-Pseudonymisierung durch DAS bei QSKH-Modulen | A 1.3 | 2019 V01 |
| Anpassung an die Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) inkl. Sonderregelungen für das Modul CHE_KV | A 1.3 | 2019 V01 |
| Anpassung der Vorgaben zur Herstellung der Eindeutigkeit der Vorgangsnummern | A 1.4 | 2019 V02 |
| Anpassung der Vorgaben zur Herstellung der Eindeutigkeit der Vorgangsnummern | A 1.4 | 2019 V01 |
| Anpassung im Abschnitt „Datenflussprotokoll für die DAS“, dahingehend dass bei Datenflussprotokollen zu Datenlieferungen für QSKH-Verfahren keine Depseudonymisierung der Leistungserbringerkennung in der DAS zu erfolgen hat. | A 1.8 | 2019 V02 |
| Ergänzungen in Tabelle 7 hinsichtlich der Herstellung der Eindeutigkeit von Vorgangsnummern, Verschlüsselung der QS-Daten sowie Verschlüsselung der leistungserbringeridentifizierenden Daten | A 1.10 | 2019 V02 |
| Anpassungen bzgl. der für das Erfassungsjahr 2019 entfallenden LE-Pseudonymisierung durch DAS | A 1.10 | 2019 V01 |
| Anpassung an die Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL); Tabelle 7: Aufgaben der Datenannahmestellen | A 1.10 | 2019 V01 |
| Löschen eines fehlerhaften Absatzes in der Einleitung zur Sollstatistik | A 2 | 2019 V01 |
| Korrektur der Prüfung auf FAELLE_GEPRUEFT (statt ANZAHLFAELLE) in der Risikostatistik | A 2.2 | 2019 V01 |

| Änderung | Kapitel/Abschnitt | Version |
|--|-------------------|-----------------|
| Anpassungen an die 2019 entfallende LE-Pseudonymisierung durch DAS bei der Risikostatistik | A 2.3 | 2019 V01 |
| Ergänzung eines Hinweises auf die Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) | A 3 | 2019 V01 |
| Ergänzung eines Hinweises, da bei dem Modul CHE_KV Leistungserbringer durch die Felder BSN-RAMBULANT und NBSNRAMBULANT gekennzeichnet werden, die Erzeugung der LE-Pseudonyme aber dennoch mit dem gleichen öffentlichen Schlüssel der auch bei IKNRKH verwendet wird, erfolgt. Dieser Schlüssel ist von den DAS LQS/LKG und DAS-KK zu verwenden. | A 3 | 2019 V01 |
| Ergänzung eines Hinweises auf die Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) inkl. Sonderregelungen für das Modul CHE_KV | A 3.1 | 2019 V01 |
| Ergänzung eines Hinweises auf die Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) inkl. Sonderregelungen für das Modul CHE_KV | A 3.1.1 | 2019 V01 |
| Verdeutlichung, dass die Struktur zur Übermittlung der Sollstatistik-Exportdateien der QS-Filter-Datenbank zu entnehmen ist | A 3.1.1 | 2019 V01 |
| QUARTAL5 entfällt in Regel 7 | A 3.1.1 | 2019 V01 |
| Ergänzung relevanter Punkte bzgl. der Übermittlung von Sollzahlen im KV-Bereich sowie beispielhafter Erläuterungen zum Nachvollziehbarkeit der IST-Zählung auf Bundesebene. | A 3.1.3 | 2019 V02 |
| Anpassungen an die 2019 entfallende LE-Pseudonymisierung durch DAS bei der QSKH-Sollstatistik | A 3.2 | 2019 V01 |
| Ergänzung eines Hinweises auf die Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) in Bezug auf die Leistungserbringerpseudonymisierung | A 3.2 | 2019 V01 |
| Anpassungen der Prüfungen der Sollstatistik | A 3.3.1 | 2019 V01 |
| Korrektur der Prüfungen der Sollstatistik (Prüfung 3 und 4 werden zu einer gemeinsamen Prüfung zusammengefasst; Korrektur der Anzahl Zeilen in der Sollstatistik, da es für QSKH insgesamt 42 Zeilen und für Qesü 5 Zeilen sind) | A 3.3.1 | 2019 V02 |

| Änderung | Kapitel/Abschnitt | Version |
|---|-------------------|-----------------|
| Streichung des Absatzes zur gem. Qesü-RL gestrichenen Ausnahmeregelung bzgl. der Verschlüsselung von Rückmeldeberichten im Krankenhausbereich sowie Aktualisierung der Beispiele zu Auslieferungseinheiten. | A 4 | 2019 V02 |
| Ergänzungen bzgl. der neuen Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) | A 4 | 2019 V01 |
| Löschen der beiden Beispiele zu Auswertungseinheiten BU0815 und BU0815-01 | A 4 | 2019 V01 |
| Aufnahme des Interfaces für die DAS-KV nach Entschlüsselung der QS-Daten (in Tabelle 15) sowie kleinere Anpassungen in der Spalte „Zweck“. | A 5.1.3 | 2019 V02 |
| Ergänzung eines Hinweises auf die Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) | A 5.3 | 2019 V01 |
| Anpassung des Abschnitts an die die Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) inkl. Sonderregelungen für das Modul CHE_KV | A 5.3.1 | 2019 V01 |
| Ergänzung einer Fußnote, da bei Verfahren gemäß Qesü-RL auch diejenigen Datensätze, die sich auf Nicht-GKV-Versicherte beziehen, über die VST weitergeleitet werden, es nach Qesü-RL aktuell aber nur Verfahren gibt, die auf GKV-Versicherte beschränkt sind | A 5.3.1 | 2019 V01 |
| Anpassung der Tabelle 19: Übersicht über die Exportverfahren an die die Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) | A 5.3.1 | 2019 V01 |
| Aktualisierung der Beschreibung der Komponente „Vorlagen_Sollstatistik“, die sich ausschließlich aus Word-Dokumenten zusammensetzt (ohne TXT-Hinweisdokument) | B „Komponenten“ | 2019 V01 |
| Erläuterung der Datenbank zu Datenserviceinformationen | B „Komponenten“ | 2019 V01 |
| Ergänzung des Moduls CHE in den Abfragen der QS-Filter-Datenbank | B 1.1 | 2019 V01 |
| Ergänzen eines neuen Attributs deqs in Tabelle Modul (QS-Filter-Datenbank) | B 1.2.1 | 2019 V01 |

| Änderung | Kapitel/Abschnitt | Version |
|---|-------------------|----------|
| Erläuterung des neuen Attributs deqs in Tabelle Modul (QS-Filter-Datenbank) | B 1.2.1 | 2019 V01 |
| Aktualisierung der Regelung zum Feld PERSONENKREIS | B 1.2.2 | 2019 V01 |
| Anpassung des Abschnitts an die die Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) | B 1.2.2 | 2019 V01 |
| Anpassung des Abschnitts „Zuordnung der QS-Filter-Version zu Behandlungsfällen“, da ab dem Erfassungsjahr 2019 alle Module auf lange Überliegerverfahren umgestellt werden | B 1.2.8 | 2019 V01 |
| Anpassung des Abschnitts an die die Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) | B 1.3.2 | 2019 V01 |
| Ergänzung des Moduls NWIF, da dies bislang im Abschnitt noch unerwähnt war | B 1.3.2 | 2019 V01 |
| Korrektur der zur Berechnung notwendigen Funktion für die Felder DRGFALL, IVFALL, DMPFALL und SONSTFALL | B 1.3.2 | 2019 V01 |
| Anpassung des Abschnitts in Bezug auf das neuen Exportmodul CHE_LKG gemäß DeQS-RL | B 1.3.2 | 2019 V01 |
| Ergänzung der QS-Filter-Leistungsbereiche CHE_LKG, CHE_KV und CHE_SV | B 1.4.2 | 2019 V01 |
| Löschen des Moduls CHE_HE | B 1.4.2 | 2019 V01 |
| Ergänzung eines Hinweises bezüglich der Zuordnung belegärztlicher Leistungen im Modul CHE | B 1.4.2 | 2019 V01 |
| Ergänzen eines Hinweises, da ab der Spezifikation 2019 alle bundesweit verpflichtenden Module auf lange Überliegerverfahren umgestellt werden | B 1.4.3 | 2019 V01 |
| Löschen von Hinweisen auf kurze Überliegerverfahren | B 1.4.3 | 2019 V01 |
| Anpassung in Bezug auf die Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL), da das Formular zur Sollstatistik der Qesü-RL um Verfahren gemäß DeQS-RL erweitert wird. Daraus ergibt sich eine Änderung der Dateinamen der Sollstatistikformulare. | B 1.5 | 2019 V01 |
| Ergänzen eines Kapitels „Sollstatistik des Moduls CHE gemäß DeQS-RL“ | B 1.5.2 | 2019 V01 |

| Änderung | Kapitel/Abschnitt | Version |
|--|-------------------|----------|
| Anpassung des Abschnitts an die die Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) | B 2 | 2019 V01 |
| Anpassung des Abschnitts an die die Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) inkl. Tabelle 39: Struktur der Tabelle PseudonymVerfahren | B 2.1 | 2019 V01 |
| Anpassung des Abschnitts an die die Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) | B 2.3.1 | 2019 V01 |
| Ergänzen eines neuen Attributs deqs in Tabelle Modul (QS-DOK-Datenbank) | B 2.3.1 | 2019 V01 |
| Erläuterung des neuen Attributs deqs in Tabelle Modul (QS-DOK-Datenbank) in Tabelle 40: Struktur der Tabelle Modul | B 2.3.1 | 2019 V01 |
| Aktualisierung, da die Schlüsselwerte des externen Fachabteilungsschlüssels aus den Datenbanken gestrichen werden | B 2.3.3 | 2019 V01 |
| Aktualisierung des Abschnitts zum Schlüssel Personenkreis (QS-Dokumentation) | B 2.3.3 | 2019 V01 |
| Aktualisierung der Funktion personenkreis-mapping | B 2.4.3 | 2019 V01 |
| Aktualisierung, da die automatische Befüllung des Datenfeldes PERSONENKREIS in FU-Modulen ebenfalls mittels Syntaxvariablen gesteuert wird. | B 2.4.4 | 2019 V01 |
| Anpassung der Beschreibung des Attributs nurPositiv bei Feldgruppen | B 2.4.7 | 2019 V01 |
| Anpassungen in Bezug auf die modulübergreifende Umstellung auf lange Überliegenderverfahren | B 2.7.3 | 2019 V01 |
| Anpassungen bzgl. der Auslagerung der Datenserviceinformationen in eine separate Datenbank | B 2.8 | 2019 V01 |
| Anpassungen bzgl. der Auslagerung der Datenserviceinformationen in eine separate Datenbank | B 2.8.2 | 2019 V01 |
| Ergänzung eines Teildatensatzes ABTLG | B 3.3.2 | 2019 V01 |
| Erweiterung des Eingangsdatensatzes um den Teildatensatzes ABTLG um ein Datenfeld FALL . AUN-ZEIT | B 3.7 | 2019 V01 |
| Ergänzung eines Feldes AUFNFACHABT in der Risikostatistik | B 3.8 | 2019 V01 |

| Änderung | Kapitel/Abschnitt | Version |
|--|--------------------|----------|
| Aktualisierungen bzgl. der Einsichtnahme der Datenannahmestellen KV in die QS-Daten, sowie Erhebung des Feedbackkeys für die Verschlüsselung von Rückmeldeberichten auch im stationären Bereich. | B 4 | 2019 V01 |
| Anpassung in Bezug auf die Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) bzw. in Bezug auf das neuen Verfahren CHE | B 4.2 | 2019 V01 |
| Ergänzung von zwei Ausprägungen des Elementes <error_type> im Text | B 4.4.3 | 2019 V01 |
| Anpassung des Element header/encryption/feedback_key in Bezug auf die Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) bzw. in Bezug auf das neuen Verfahren CHE | B 4.4.3 | 2019 V01 |
| Ergänzung des Pseudonymisierungsverfahrens CHE in der Tabelle 108: Verfahrenskennung: „pseud_procedure“ | B 4.4.4 | 2019 V01 |
| Anpassung der Hinweises „Umgang mit PID-Daten“ in Bezug auf die Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) | B 4.4.4 | 2019 V01 |
| Korrektur und Ergänzung der Einleitung in Bezug auf das Datenprüfprogramm | B 5.2 | 2019 V01 |
| Anpassung in Bezug auf die Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) | B 5.3 | 2019 V01 |
| Anpassung in Bezug auf die Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) | B 5.3.5 | 2019 V01 |
| Anpassung in Bezug auf die Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) | C Anhang „Glossar“ | 2019 V01 |

Leseanleitung

Diese technische Dokumentation orientiert sich in ihrem Aufbau an den Abläufen der Erfassung und Übermittlung der Qualitätssicherungsdaten. Ziel dieser Struktur ist es, eine nachvollziehbare und logische Sicht auf die Umsetzung und Durchführung der beschriebenen Schritte zu gewährleisten. Die Prozesse und Unterprozesse werden im Abschnitt A Prozesse beschrieben und spiegeln die reale, chronologische Abfolge wider. Jede Prozessbeschreibung berücksichtigt zudem die unterschiedlichen Komponenten, die für die Umsetzung benötigt und in Abschnitt B Komponenten detailliert beschrieben werden. Im Abschnitt C Anhang wird ein Glossar mit den wichtigen Begriffen zum Themenbereich der Spezifikation zur Verfügung gestellt.

Für eine korrekte Umsetzung der Spezifikation ist es notwendig, die Dokumentation entsprechend ihrer Anordnung von Prozessen zu Komponenten zu befolgen. Einige Bereiche, die sich ausschließlich an bestimmte Zielgruppen richten, sind entsprechend gekennzeichnet.

Legende

Die in dieser Dokumentation verwendeten Symbole heben bestimmte Aspekte bei der Umsetzung der Spezifikation hervor.



Achtung

Beschreibt Ursache, Folge und Vermeidung einer besonderen Fehlanwendung, die zu Problemen bei der Implementierung oder Ähnlichem führen kann.



Hinweis

Nützliche Informationen, Tipps oder Ratschläge zur Anwendung. Keine wesentlichen oder für das korrekte Funktionieren erforderlichen Informationen.

Beispiel:

Beispiele sind ein Hilfsmittel, um zuvor vermittelte Informationen oder konkrete Abschnitte der Anwendung zu verdeutlichen.



Zielgruppe LQS/LKG

Markiert nur für folgende Zielgruppe relevanten Bereich:

Landesgeschäftsstelle für Qualitätssicherung (LQS)

Landeskrankenhausgesellschaften (LKG)



Zielgruppe VST-DAS

Markiert nur für folgende Zielgruppe relevanten Bereich:
Vertrauensstellen (VST), Datenannahmestellen (DAS)



Zielgruppe KV

Markiert nur für folgende Zielgruppe relevanten Bereich:
Kassenärztliche Vereinigung (KV)

1 Einleitung

Die technische Dokumentation für Datenannahmestellen beschreibt die Spezifikation zur einrichtungs- und sektorenübergreifenden Qualitätssicherung gemäß §§ 136 ff. SGB V und richtet sich an die Datenannahmestellen.

Die Basisspezifikation umfasst alle Komponenten im Zusammenhang mit der Datenerfassung, d.h. von der Bestimmung einer Dokumentationspflicht bis hin zur Rückprotokollierung übermittelter Datensätze und einem Soll-Ist-Abgleich. Die vorliegende technische Dokumentation beschreibt die für die Datenannahmestellen relevanten Prozesse und Komponenten.

Regelungsbereich der Basisspezifikation sind die verschiedenen Leistungsbereiche und Verfahren gemäß der Richtlinie über Maßnahmen der Qualitätssicherung in Krankenhäusern (QSKH-RL)¹ gemäß der Richtlinie zur einrichtungs- und sektorenübergreifenden Qualitätssicherung (Qesü-RL)², sowie der Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL), die vom Gemeinsamen Bundesausschuss (G-BA) beschlossen werden. Darüber hinaus beinhaltet die Basisspezifikation im Sinne einer einheitlichen Spezifizierung außerdem Vorgaben für landesweit gültige oder freiwillige Qualitätssicherungsverfahren.

Die Basisspezifikation bezieht sich dabei auf die fallbezogene QS-Dokumentation beim Leistungserbringer – für die einrichtungsbezogene QS-Dokumentation sowie für eine Nutzung von Sozialdaten bei den Krankenkassen gemäß § 299 SGB V wird jeweils eine unabhängige Spezifikation veröffentlicht.³ Für eine spezifikationskonforme Datenannahme, -übermittlung und Rückprotokollierung sind alle Spezifikationskomponenten zu berücksichtigen.

Aus Gründen der leichteren Lesbarkeit werden in der vorliegenden technischen Dokumentation alle Verfahren gemäß Qesü-RL sowie alle Verfahren nach DeQS-RL als einrichtungs- und/oder sektorenübergreifende Verfahren bezeichnet. Den jeweiligen Richtlinien ist für jedes Verfahren zu entnehmen, ob die Bewertung einrichtungsübergreifend oder sektorenübergreifend (sektorüberschreitend, sektorgleich oder im Sinne eines sektorüberschreitenden Follow-up-Verfahrens) erfolgt.

1.1 Spezifikationsbegriff

Die Spezifikation ist die Gesamtheit aller Vorgaben, nach denen die Bestimmung der dokumentationspflichtigen Fälle, die QS-Dokumentation selbst und die Übermittlung der Daten erfolgen sollen, bezogen auf ein Erfassungsjahr. Die Zuordnung eines Falles zu einer Spezifikation richtet sich nach dem Kalenderjahr der Aufnahme in das Krankenhaus bzw. dem Eingriffsdatum bei ambulanten Fällen. Die Spezifikation beinhaltet weiterhin die Aufstellung der im Erfassungsjahr zu

¹ <https://www.g-ba.de/informationen/richtlinien/38/>

² <https://www.g-ba.de/informationen/richtlinien/72/>

³ <https://iqtig.org/datenservice>

dokumentierenden Leistungen (Sollstatistik) und die automatische Erstellung von Basisinformationen für eine spezifische Zielpopulation auf Basis vorhandener Abrechnungsdaten beim Leistungserbringer (Risikostatistik).

Um die komplexen Anforderungen an die stationäre und die sektorenübergreifende QS-Dokumentation sowie die zugehörigen Datenflüsse zu erfüllen, besteht die Spezifikation aus verschiedenen Komponenten, die je nach Anwender spezifisch zusammengestellt werden. Als Komponenten werden dabei Access-Datenbanken, technische Dokumentationen, Ausfüllhinweise und anderes bezeichnet. Jeder Anwender bekommt damit das für ihn Relevante in einem eigenen Spezifikationspaket als Download zur Verfügung gestellt. Jedes dieser Pakete kann auf diese Weise auch unabhängig von den anderen aktualisiert werden.

Damit gibt es ein Spezifikationspaket für

- den Regelbetrieb (QSKH, Qesü und DeQS)
- ggf. Sonderexporte
- ggf. Probetriebe
- ggf. Testbetriebe

Sowohl die Spezifikationspakete als auch die einzelnen Komponenten werden nach einem einheitlichen Schema benannt, das bereits im Namen übersichtlich die relevanten Informationen wie Betriebsart, Exportformat und Versionierung enthält. Dieses Schema 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 einzelnen Komponenten und ggf. eine Übersicht über die Änderungen zur vorhergehenden Version bei.

1.1.1 Benennungsschema für Spezifikationspakete

Die Benennung der Spezifikationspakete setzt sich wie folgt zusammen:

```
<Erfassungsjahr>_<Richtlinie>_<Name>_[ <DAS> ]_<Betriebsart>_<Exportformat>_V<Versionsnummer>
```

Das Erfassungsjahr gilt für alle Spezifikationspakete und -komponenten, die Daten dieses Erfassungsjahres betreffen, egal in welchem Jahr das jeweilige Paket veröffentlicht wurde.

Da seit dem Erfassungsjahr 2014 die Spezifikation als Basisspezifikation konzipiert ist und damit alle relevanten Richtlinien (QSKH, Qesü und DeQS) abgedeckt sind, muss nicht zwischen diesen unterschieden werden. Daher steht im Platzhalter <Richtlinie> immer BASIS für Basisspezifikation. Bei anderen Spezifikationen kann hier beispielsweise zwischen QSKH und Qesue differenziert werden.

Bei der Angabe <Name> kann der die Spezifikation kennzeichnende Name angegeben werden. Namen können beispielsweise wie folgt definiert werden:

- FDOK: fallbezogene QS-Dokumentation
- EDOK: einrichtungsbezogene QS-Dokumentation

- SozDat : Nutzung von Sozialdaten bei den Krankenkassen
- DBSD: Datenformat zum Bericht über den Strukturierten Dialog

Bei der optionalen Angabe [<DAS>] kann beispielsweise zwischen folgenden Kürzeln unterschieden werden:

- LKG: LQS/LKG
- KV: DAS-KV
- KK: DAS-KK

Bei der Betriebsart kann zwischen folgenden Kürzeln unterschieden werden:

- RB: Regelbetrieb
- SE: Sonderexport
- PB: Probetrieb
- TB: Testbetrieb

V<Versionsnummer>: Die Versionierung erfolgt in ganzen Zahlen, die zweistellig angegeben sind (unter 10 mit einer vorlaufenden 0, z. B. V01).

Beispiele:

Als erstes planmäßiges Release wird folgendes Paket für die fallbezogenen QS-Dokumentation veröffentlicht:

2019_BASIS_FDOK_RB_XML_V01

Im September 2018 wird folgendes Paket für die fallbezogenen QS-Dokumentation veröffentlicht:

2019_BASIS_FDOK_RB_XML_V02

Ausformuliert bezeichnet dies die Spezifikation zur fallbezogenen QS-Dokumentation für den Regelbetrieb des Erfassungsjahres 2019. Das Format für die zu exportierenden und von den beteiligten Stellen (DAS, VST) zu übermittelnden Datensätze der QS-Dokumentation ist seit 2015 XML.

Beispiele:

Neben der fallbezogenen QS-Dokumentation können beispielsweise folgende Spezifikationspakete veröffentlicht werden.

Die einrichtungsbezogene QS-Dokumentation:

2018_Qesue_EDOK_RB_XML_V01

Die Spezifikation für die Nutzung der Sozialdaten bei den Krankenkassen:

2019_Qesue_SozDat_KK_RB_XML_V01

Die Spezifikation des Datenformats zum Bericht über den Strukturierten Dialog:

2017_QSKH_DBSD_RB_CSV_V02

1.1.2 Benennungsschema für Spezifikationskomponenten

Die Benennung der Spezifikationskomponenten lehnt sich an das bei den Spezifikationspaketen verwendete Prinzip an:

```
[<Erfassungsjahr>_]<Art der Komponente>_[<Exportformat>_][V<Versionsnummer>].<Dateierweiterung>
```

„Art der Komponente“ bezieht sich auf die jeweilige Funktion und wird durch ein Kürzel angegeben.

Die in eckige Klammern gesetzten Benennungsbestandteile sind optional und werden nur dann gesetzt, wenn dieses Merkmal für die Komponenten von Relevanz ist. Das Erfassungsjahr ist für Hilfsprogramme wie z. B. den XPacker nicht relevant; bei diesen erfolgt die Versionierung mit einer Versionsnummer, die über die Jahre fortgeführt wird.

Die Bezeichnungen der einzelnen Komponenten werden in Abschnitt B auf Seite 114 erläutert.

Abbildung 1 dient der Veranschaulichung des Benennungsschemas. Die abgebildeten Daten sind beispielhaft und bilden nur einen Teil der Pakete und Komponenten der Spezifikationen 2017 und 2018 ab. Die gesamte Spezifikation 2017 ist durch einen dunkelgrünen Rahmen begrenzt. Die grünen Kästen innerhalb sind die Spezifikationspakete, die je nach Anwender zusammengestellt werden. Innerhalb der Pakete sind die Spezifikationskomponenten aufgelistet. In deren Bezeichnungen ist hier jeweils rot markiert, wenn sich etwas von Paket zu Paket geändert hat. Ausgegraut und damit nur angedeutet ist die Spezifikation 2018, die zwar bereits während des Erfassungsjahres 2017 veröffentlicht wurde, aber von der Spezifikation 2017 zu unterscheiden ist.

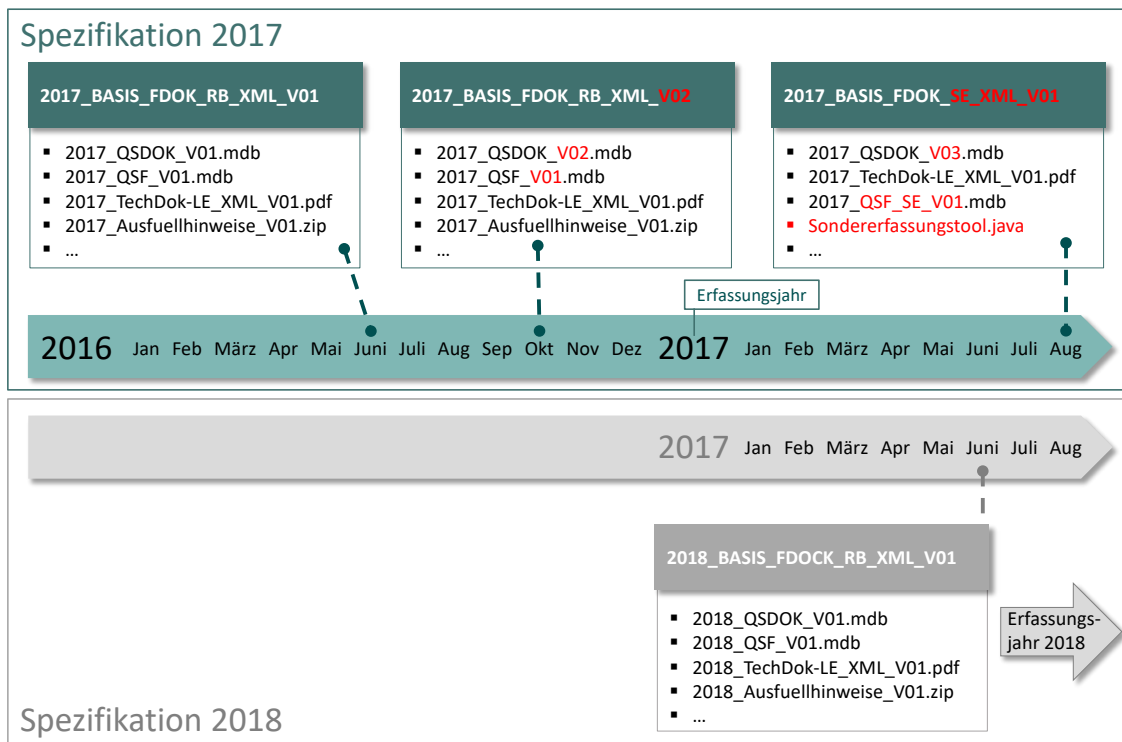


Abbildung 1: Beispiel für Benennung von Paketen und Komponenten für die Spezifikation 2017 bis 2018

1.2 Zielsetzung der Technischen Dokumentation

Diese technische Dokumentation beschreibt Export- und Übertragungsprozesse der für das Erfassungsjahr 2019 spezifizierten Module.

Sie soll eine Hilfestellung für Datenannahmestellen bieten, die Datenannahme, Verarbeitung und Weiterleitung der 2019 erhobenen Daten durchzuführen. Für die Vertrauensstelle soll eine Hilfestellung gegeben werden, die Datenannahme und die Weiterleitung durchzuführen. Die Verarbeitung bei der Vertrauensstelle erfolgt nach gesonderten Regelungen zwischen der Vertrauensstelle und deren Auftraggeber, dem G-BA und ist somit nicht Bestandteil dieser Technischen Dokumentation.

1.3 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 1 stellt eine entsprechende Übersicht für das Paket der QS-Basisspezifikation 2018 beim Leistungserbringer für den Regelbetrieb dar. Die genannten Termine sind als Zielwerte zu betrachten und basieren auf Vorgaben des G-BA, Abstimmungen mit Softwareanbietern sowie Anforderungen aus der Umsetzung. Regulär wird die erste Version der Spezifikation eines Erfassungsjahres am 30. Juni des Vorjahres veröffentlicht.

Tabelle 1: Meilensteine der Releaseplanung der Spezifikation 2018 für den Regelbetrieb

| Frist | Meilenstein | Bereitstellung | Bemerkung |
|---------------------------------|-------------------------------|---|---|
| nach Beschluss durch das Plenum | Version 2019 V01 | Veröffentlichung auf der IQTIG-Webseite (http://www.iqtig.org) | Finale Version für QS-Verfahren gem. QSKH-RL sowie für freiwillige und landesbezogene Verfahren und QS-Verfahren gem. Qesü-RL und DeQS-RL |
| 15.09.2018 | Frist für Fehlerrückmeldungen | E-Mail an verfahrensupport@iqtig.org oder Nutzung der Kommunikationsplattform | |
| 30.09.2018 | Version 2019 V02 | Veröffentlichung auf der IQTIG-Webseite (http://www.iqtig.org) | Fehlerkorrekturen |
| 20.10.2018 | Frist für Fehlerrückmeldungen | E-Mail an verfahrensupport@iqtig.org oder Nutzung der Kommunikationsplattform | |
| November 2018 | Version 2019 V03 | Veröffentlichung auf der IQTIG-Webseite (http://www.iqtig.org) | Aktualisierung der ICD- und OPS-Kodes, ggf. Aktualisierung von GOP gemäß EBM-Katalog; ggf. Fehlerkorrekturen; Frist abhängig vom Veröffentlichungszeitpunkt der endgültigen ICD- und OPS-Kataloge für das Jahr 2019 durch das DIMDI ⁴ |

Über die in der Tabelle aufgeführten Meilensteine hinaus erfolgt eine regelmäßige Abstimmung mit Softwareherstellern und weiteren Verfahrensteilnehmern (z. B. Datenannahmestellen, Vertrauensstelle) in Form von Informationstreffen, Workshops und Kommunikationsplattform. Zudem wurden neue Meilensteine in den Prozess der Systempflege integriert (z. B. die Bereitstellung von Änderungsempfehlungen und 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).

Change- und Fehlermanagement

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

- Meldung von festgestellten Fehlern (z. B. Spezifikations- und Softwarefehlern)
- Verbreitung von Änderungsvorschlägen

⁴ DIMDI: Deutsches Institut für Medizinische Dokumentation und Information

- 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, obwohl Sie eine beteiligte Institution (z. B. Softwareanbieter, Datenannahmestelle) sind, lassen Sie sich bitte bei uns registrieren.

Ihr Ansprechpartner:

Institut für Qualitätssicherung und Transparenz im Gesundheitswesen
Katharina-Heinroth-Ufer 1
10787 Berlin

Telefon: (+49) 30 58 58 26 340

Fax: (+49) 30 58 58 26 341

verfahrenssupport@iqtig.org

www.iqtig.org/

1.4 Freiwillige und landesbezogene Verfahren

Die Basisspezifikation bietet die Möglichkeit, neben den bundesweit verpflichtenden QS-Modulen auch freiwillige und/oder landesbezogene Verfahren zu spezifizieren. Die freiwilligen und landesbezogenen Verfahren zeichnen sich dadurch aus, dass sie nicht bundesweit verpflichtend sind oder in einem anderen (gesetzlichen) Rahmen als der gesetzlichen Qualitätssicherung nach §§136ff. SGB V durchgeführt werden. Das IQTIG trägt dabei keine inhaltliche Verantwortung und fungiert nicht als Datenannahmestelle. Durch die Integration dieser Module wird eine technische Vereinheitlichung erreicht.

1.4.1 Besonderheiten der Qualitätssicherung in Hessen

Im Auftrag der GQH (Geschäftsstelle Qualitätssicherung Hessen bei der Hessischen Krankenhausgesellschaft e.V.) wurden die Spezifikationen für QS-Dokumentation und QS-Filter ergänzt:

1. Über die bundesweit im Modul 15/1 verpflichtend zu dokumentierenden gynäkologischen Eingriffe hinaus werden im Bundesland Hessen alle Fälle mit der Hauptdiagnose Leiomyom des Uterus (ICD D25.-) in die Qualitätssicherung eingeschlossen. Diese Unterschiede im Umgang mit der Erhebung werden in der Spezifikation berücksichtigt.
2. Die Übermittlung von Datumsangaben⁵ in den Modulen 15/1 und 18/1 wird ermöglicht (Abschnitt 1.4.2).

⁵ Gemeint sind Datumsangaben, die im Format TT.MM.JJJJ übermittelt werden.

3. Der Modulauslöser für den Datensatz zu multiresistenten Erregern MRE_HE wird erstmalig mit der Spezifikation 2017 abgebildet.

Anpassung des Datensatzes 15/1 für die zusätzliche Erhebung gynäkologischer Operationen beim Leiomyom des Uterus

Die Spezifikation ermöglicht die zusätzliche Erhebung gynäkologischer Operationen beim Leiomyom des Uterus in Hessen. Zum Zweck der Differenzierung wurde bereits 2005 im Teildatensatz Operation (15/1:O) mit der Spezifikation 8.0 ein zusätzliches Datenfeld eingeführt, das aktuell heißt:

Eingriff im Rahmen der Zusatzerhebung Leiomyom des Uterus)⁶ [GYN-ZUSATZ]

Eine Angabe (= ja) in diesem Datenfeld ist ausschließlich in Hessen möglich. In allen anderen Bundesländern darf das Datenfeld nicht ausgefüllt werden.



Achtung

Das Datenfeld GYNZUSATZ soll nur in Erfassungssystemen hessischer Krankenhäuser sichtbar sein, um Anwender aus Krankenhäusern anderer Länder vor einer Fehldokumentation zu bewahren.

Wurde für den Eingriff mindestens eine bundesweit verpflichtende Prozedur (definiert in der OPS-Liste GYN_OPS der Spezifikation für QS-Filter) erbracht, so muss das Datenfeld GYNZUSATZ leer bleiben.

Wurden für den Eingriff ausschließlich hessenweit verpflichtende Prozeduren (definiert in der OPS-Liste GYN_OPS_HESSEN der Spezifikation für QS-Filter) erbracht, so muss das Datenfeld GYNZUSATZ mit einer 1 (= ja) dokumentiert werden.

Übermittlung von 15/1er-Datensätzen an die Bundesebene

Die Datenstelle der GQH leitet Teildatensätze 15/1:O nicht weiter, wenn sie ausschließlich im Rahmen der hessischen Zusatzerhebung beim Leiomyom des Uterus dokumentiert worden sind:

Ein Datensatz 15/1 wird von der GQH an die Bundesebene übermittelt, wenn für mindestens einen Teildatensatz 15/1:O die Bedingung GYNZUSATZ = LEER zutrifft. In diesem Fall werden der Teildatensatz 15/1:B sowie alle Teildatensätze 15/1:O, für die diese Bedingung zutrifft, übertragen.

Datensätze 15/1, bei denen für alle Teildatensätze 15/1:O die Bedingung GYNZUSATZ = 1 zutrifft, werden nicht übermittelt.

Beispiele:

Folgende Beispiele zeigen für drei Fälle, welche Teildatensätze (TDS) einer Dokumentation an die Bundesebene weitergeleitet werden.

⁶ Das Datenfeld hat den Schlüssel JLeer.

Fall 1

| Dokumentierte Teildatensätze | GYNZUSATZ = 1 | TDS an IQTIG weiterleiten? |
|------------------------------|---------------|----------------------------|
| 15/1 : B | - | ja |
| 15/1 : O (1. Eingriff) | ja | nein |
| 15/1 : O (2. Eingriff) | nein | ja |

Fall 2

| Dokumentierte Teildatensätze | GYNZUSATZ = 1 | TDS an IQTIG weiterleiten? |
|------------------------------|---------------|----------------------------|
| 15/1 : B | - | nein |
| 15/1 : O (1. Eingriff) | ja | nein |
| 15/1 : O (2. Eingriff) | ja | nein |

Fall 3

| | GYNZUSATZ = 1 | TDS an IQTIG weiterleiten? |
|------------------------|---------------|----------------------------|
| 15/1 : B | - | ja |
| 15/1 : O (1. Eingriff) | nein | ja |

**Achtung**

Die Datenstelle der GQH hat dafür Sorge zu tragen, dass Minimaldatensätze zum Modul 15/1 nur an die Bundesauswertungsstelle weitergeleitet werden, wenn der Datensatz gemäß QS-Filter-Algorithmus GYN bundeweit verpflichtend ist.

Handelt es sich jedoch ausschließlich um einen Minimaldatensatz des zusätzlichen Leistungsbereichs 15/1 für die hessische Zusatzerhebung Gynäkologie gemäß QS-Filter-Algorithmus GYNHESSEN ist dieser nicht an die Bundesauswertungsstelle zu übermitteln.

Ist-Bescheinigung

Die GQH bescheinigt die Datensätze den hessischen Krankenhäusern getrennt nach den QS-Filter-Leistungsbereichen:

- 15/1 B (= gelieferte Datensätze, die an die Bundesebene weitergeleitet werden)
- 15/1 L (= gelieferte Datensätze, die nicht an die Bundesebene weitergeleitet werden)

Beispiel:

Die Fälle 1 und 3 des letzten Beispiels gehören zum QS-Filter-Leistungsbereich 15/1, der Fall 2 zum QS-Filter-Leistungsbereich 15/1 H.

1.4.2 Übermittlung von Datumsangaben

Die in den Exportdatensätzen als parametrierbare Ersatzfelder definierten Datumsangaben sind der Datenbank für QS-Dokumentation zu entnehmen.

Die Festlegung, ob parametrierbare Exportfelder auch wirklich vom Krankenhaus an die Datenstelle übermittelt werden, wird – in Abstimmung mit den zuständigen Landesdatenschutzbeauftragten – von der jeweils zuständigen Landesebene getroffen.⁷

⁷ Dies gilt generell für alle parametrierbaren Exportfelder.

A Prozesse

Im Folgenden werden die einzelnen Prozesse und Unterprozesse der Spezifikation beschrieben. In Tabelle 2 werden diese Prozesse in Abhängigkeit von den Richtlinien (QSKH, Qesü und DeQS), den betroffenen Datenannahmestellen (DAS) und ihrer Relevanz für die jeweiligen Zielgruppen aufgeführt. Die einzelnen Abschnittsnummern verweisen auf die entsprechenden Abschnitte, in denen sie genauer erläutert werden.

Tabelle 2: Übersicht über die Prozesse QS-Dokumentation, Sollstatistik und Risikostatistik und ihre Unterprozesse

| Zielgruppe Richtlinie Prozesse | LQS/LKG | | KV | DAS-SV |
|--------------------------------------|---------|-----------|-----------|-----------|
| | QSKH | Qesü/DeQS | Qesü/DeQS | Qesü/DeQS |
| QS-Dokumentation | | | | |
| Datenannahme | A 1.1 | A 1.1 | A 1.1 | A 1.1 |
| Datenprüfung | A 1.2 | A 1.2 | A 1.2 | A 1.2 |
| LE-Pseudonymisierung | A 1.3 | A 1.3 | A 1.3 | A 1.3 |
| Verschlüsselung | A 1.5 | A 1.5 | A 1.5 | A 1.5 |
| Datenübermittlung | A 1.6 | A 1.6 | A 1.6 | A 1.6 |
| Rückprotokollierung | A 1.7 | A 1.7 | A 1.7 | A 1.7 |
| Sollstatistik | | | | |
| Erzeugung | A 3.1.1 | A 3.1.1 | A 3.1.3 | A 3.1.2 |
| Pseudonymisierung | A 3.2 | A 3.2 | A 3.2 | A 3.2 |
| Datenprüfung | A 3.3.1 | A 3.3.1 | A 3.3.2 | A 3.3.2 |
| Verschlüsselung | A 3.4.1 | A 3.4.1 | A 3.4.2 | A 3.4.2 |
| Datenübermittlung | A 3.5.1 | A 3.5.1 | A 3.5.2 | A 3.5.2 |
| Rückprotokollierung | A 3.6.1 | A 3.6.1 | A 3.6.2 | A 3.6.2 |
| Konformitätserklärung | A 3.7 | A 3.7 | - | A 3.7 |
| Risikostatistik | | | | |
| Datenübermittlung | A 2.1 | - | - | - |
| Datenprüfung und Rückprotokollierung | A 2.2 | - | - | - |
| Datenweiterleitung | A 2.3 | - | - | - |

| Zielgruppe | LQS/LKG | | KV | DAS-SV |
|---------------------|------------|------|-----------|-----------|
| | Richtlinie | QSKH | Qesü/DeQS | Qesü/DeQS |
| Prozesse | | | | |
| Auswertungen | | | | |
| Rückmeldeberichte | - | A 4 | A 4 | A 4 |

1 QS-Dokumentation

Die Exportdateien werden beim Leistungserbringer erstellt und dann an die zuständige Datenannahmestelle (DAS) weitergeleitet. Datenannahmestellen nehmen die Datenlieferungen im Rahmen der Qesü-RL, der DeQS-RL bzw. QSKH-RL entgegen (Abschnitt A 5.3).

In diesem Kapitel werden die Aufgaben und Werkzeuge der Datenannahmestelle in Bezug auf die Annahme, Verarbeitung und Weiterleitung der QS-Dokumentation beschrieben (Abbildung 2).

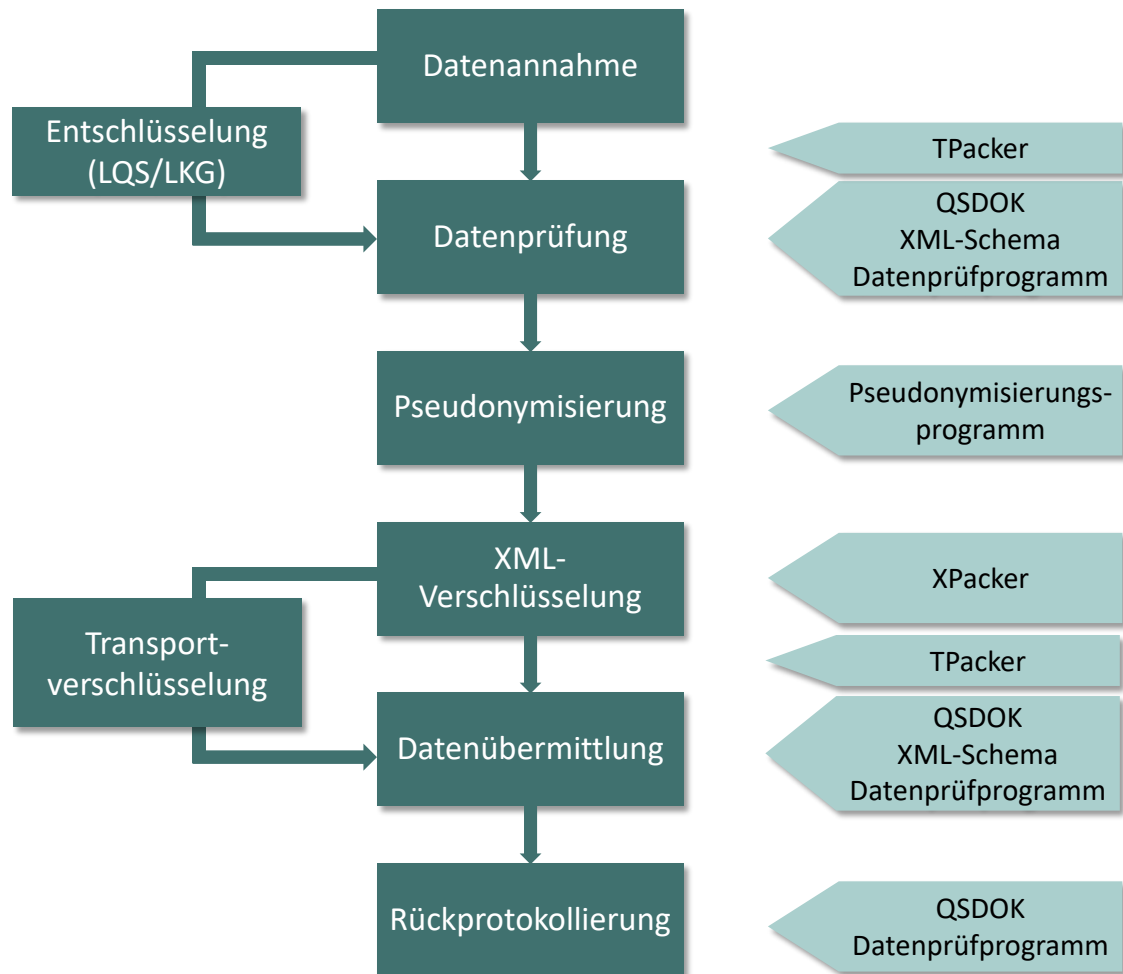


Abbildung 2: Überblick über die Prozesse und Werkzeuge in der Datenannahmestelle

1.1 Datenannahme

Im Folgenden werden die Prozesse bei der Datenannahme (Gesicherte Datenübertragung, Authentifizierung der Datenabsender, Registrierungen, Frist der Datenannahme) beschrieben.

1.1.1 Datenübertragung und Authentifizierung

Der Datenabsender muss überprüft werden. Nur die Leistungserbringer, welche sich zuvor bei der Datenannahmestelle registriert haben oder sich mittels Anmeldung authentifizieren, dürfen

Daten abgeben. Auf welchen Übertragungswegen eine DAS Daten entgegennehmen kann, ist den allgemeinen Regelungen der Datenübermittlung zu entnehmen (Abschnitt A 5.1.1).

Für die Datenübermittlung und Authentifizierung der Datenlieferanten ist in der Regel eine Registrierung erforderlich. Dies gilt für die Krankenhausdokumentationssysteme für die selektivtätigen Arztpraxen, Datenannahmestellen und Vertrauensstelle (Abschnitt A „Registrierungen“).

1.1.2 Fristen für die Datenannahme und -weiterleitung

Die Datenlieferfristen sind für die Leistungsbereiche der externen stationären Qualitätssicherung in der **QSKH-RL** festgelegt. Die Daten eines Erfassungsjahres werden jeweils bis Mitternacht am 28.02. des nachfolgenden Jahres durch eine DAS angenommen. Fällt der 28. Februar auf ein Wochenende oder einen Feiertag, gilt der nächste Werktag. Die BAS hält ihren Datenservice jeweils bis zum 15.03. offen, um die Verarbeitung der fristgerechten Lieferungen zu garantieren. Die Datenübermittlung erfolgt quartalsweise. Daten für Verfahren mit planungsrelevanten Qualitätsindikatoren sind ebenfalls quartalsweise zu liefern. Dies wird durch die **Richtlinie Planungsrelevante Qualitätsindikatoren** vorgegeben. Für die sektoren- und/oder einrichtungsübergreifenden Verfahren werden die Datenlieferfristen in den Themenspezifischen Bestimmungen der **Qesü-RL** bzw. der **DeQS-RL** festgelegt.

Die Exportfristen pro Exportmodul sind außerdem der Tabelle `Exportmodul` der Datenbank zur QS-Dokumentation zu entnehmen.

1.2 Datenprüfung

Unabhängig von der Übertragungsart wird der Inhalt einer Datenlieferung nach der Annahme geprüft. Zunächst sind allgemeine Prüfungen durchzuführen, die das gesamte Dokument betreffen und von allen DAS durchgeführt werden müssen.

Eine zielgruppenspezifische Referenzliste der durchzuführenden Prüfungen befindet sich innerhalb der administrativen Objekte der Spezifikationsdatenbank QSDOK (Abschnitt B 2.8.3). Die Liste umfasst die Prüfschritte, und kann von den DAS als Vorlage für die Implementierung eines Datenservices genutzt werden. Jeder Prüfschritt in der Datenbank umfasst u.a. folgende Elemente:

- Bezeichnung der Prüfung
- Beschreibung der Prüfung
- Zielgruppe (LQS/LKG, KV, DAS-SV)
- Optionale oder obligatorische Prüfung
- Bedingung, die zu einer Regelverletzung führt
- Protokollierungsebene (fall- oder dokumentbezogen)
- ID der Fehlermeldung
- Ins Protokoll aufzunehmende Fehlermeldung
- Einstufung des Fehlers (WARNING oder ERROR)

Für alle Prüfungen gilt, dass sie im Fehlerfall entweder das geprüfte Item (Dokument oder Datensatz) ausschließen oder mit einem entsprechenden Hinweis versehen in die Weiterverarbeitung gegeben werden (Abschnitt A 5.1.3).



1.2.1 Paket-Prüfungen (nur LQS/LKG)

Dieser Abschnitt beschreibt die notwendigen Prüfungen beim Versand per E-Mail. In diesem Fall sind die Dateien für den Transport vorab vom LE durch die Transportverschlüsselung abzusichern. Diese auf die Art gesicherten Pakete tragen die Dateiendung AES.

Wenn eine Datenlieferung an eine DAS erfolgt, ist das Datenpaket vor dem Öffnen zunächst zu überprüfen und ggf. die Annahme abzulehnen.

Formale Prüfungen

Wenn der Datenabsender ein autorisierter Leistungserbringer ist, ist in einem zweiten Prozess die Datenlieferung genauer zu analysieren:

- Wenn keine Datei oder mehr als eine Datei enthalten ist, ist die Datenlieferung abzuweisen. Nur bei exakt einer Datendatei ist die Datenlieferung anzunehmen.
- Sind die XML-Datei und das Paket korrekt benannt worden?

Diese beiden formalen Prüfungen können nicht vom Datenprüfprogramm durchgeführt werden, daher sind individuelle Implementierungen in der DAS erforderlich.

Nachdem die Paketlieferung im vorangegangenen Schritt verifiziert worden ist, wird nun die im Paket enthaltene Datei geprüft. Die Datei muss dazu aus dem Paket durch einen Entpackvorgang bereitgestellt werden. Informationen über die Transport-Entschlüsselung können dem Abschnitt über die Entschlüsselung entnommen werden (Abschnitt B 5.3.2).

1.2.2 Dateiprüfungen

Mit der enthaltenen XML-Datei werden weitere Prüfungen durchgeführt. Im Folgenden werden beispielhafte Prüfungen beschrieben. Die vollständige Liste der Prüfungen ist der Spezifikationsdatenbank zu entnehmen.

Schemakonformer Dateiaufbau

Ein XML-Dokument hat einem vorgegebenen formalen Aufbau zu entsprechen. Das „Grundgerüst“ jedes XML-Dokuments muss immer gleichbleibend sein, damit identische Informationen immer an gleich bezeichneten Stellen wiederzufinden sind. Für jede Übertragungsschnittstelle (LE/DAS, DAS/VST, VST/BAS) gibt es ein spezifisches XML-Schema, gegen welches das übertragene Dokument zu validieren ist. Die jeweiligen Schemadateien sind im Spezifikationspaket enthalten.

Welches Schema für eine DAS vorgesehen ist, kann dem Abschnitt XML-Schemata (Abschnitt B 4.2) entnommen werden. Als Hilfsmittel kann dabei das Datenprüfprogramm verwendet werden: Dieses Programm kennt den notwendigen Aufbau einer XML-Datendatei für DAS und vergleicht diesen Aufbau mit dem Aufbau der zu prüfenden Datenlieferung.

Prüfung der administrativen Daten

In diesem Rahmen werden die administrativen Daten auf Korrektheit überprüft. Sie definieren die Datenlieferung und sind getrennt von den eigentlichen QS-Daten zu prüfen. Beispielhafte Prüfungen:

- Stimmt die IKNR bzw. die BSNRAMBULANT im Dokument mit der Angabe in der Registrierung überein?
- Wurde die Datei ggf. früher schon einmal übermittelt? Dazu wird die ID (GUID), die innerhalb der Datei im entsprechenden Tag genannt wird, gegen eine Datenbank geprüft, die administrative Daten enthält. Dateien mit gleichem Namen dürfen nicht mehrfach übermittelt werden.
- Fällt die Datei in die Zuständigkeit der betreffenden DAS?
- Stimmt der Dateiname mit dem ID-Element (GUID) überein?
- Ist der Status des Dokuments „OK“ oder wurde das Dokument beispielsweise bereits auf Status „ERROR“ (Fehler) gesetzt?
- Hat sich der Leistungserbringer, der durch die Registrierung identifiziert wurde, auch in der Datei korrekt verankert?

Datenprüfung – zusätzliche Schritte

LQS und LKG sowie KV dürfen gemäß den G-BA-Richtlinien die QS-Daten einsehen. Daher verschlüsseln Krankenhäuser als Leistungserbringer die QS-Daten, die an die LQS/LKG bzw. KV gesendet werden, innerhalb des XML-Codes mit dem öffentlichen Schlüssel der zuständigen Datenannahmestelle.

Die DAS-SV – darf die QS-Daten nicht einsehen. Die QS-Daten sind daher in diesen Fällen von den Leistungserbringern im XML-Code mit dem öffentlichen Schlüssel der BAS zu verschlüsseln, sodass nur die BAS die QS-Daten mit ihrem privaten Schlüssel entschlüsseln kann.

Voraussetzung für die Prüfung der QS-Daten (<qs_data>-Elemente) ist dementsprechend eine vorausgegangene XML-Entschlüsselung der QS-Daten mit dem privaten Schlüssel der zuständigen DAS. Informationen über das XML-Verschlüsselungsverfahren können dem Abschnitt über die Verschlüsselung entnommen werden (Abschnitt B 5.3.1).

Für die Prüfung der QS-Daten werden die Regeln aus der Spezifikationsdatenbank (QSDOK) herangezogen (Abschnitt B 2.4).

Die einfachste Lösung für die Prüfung auf Plausibilität liegt in der Nutzung eines Datenprüfprogramms, das auf der Basis von XSLT die aus der Spezifikationsdatenbank (QSDOK) ausgeleiteten Plausibilitätsregeln im XML-Dokument prüft und Verstöße entsprechend im XML-Code dokumentiert.

Umfang der Prüfung mit dem Datenprüfprogramm ist das Einhalten eines vorgegebenen Schemas, das nach der Entschlüsselung der QS-Daten erwartet wird. Das jeweils heranzuziehende Schema ist der Dokumentation im Abschnitt Schemata zu entnehmen (Abschnitt B 4.2). Die Beschreibung für die Verwendung des Datenprüfprogramms kann der Dokumentation zum Datenprüfprogramm entnommen werden (Abschnitt B 5.2).

**Achtung**
Das richtige Schema anwenden

Für die Validierung der QS-Daten mit dem Datenprüfprogramm ist die „weiche“ Variante der XML-Schemata zu verwenden. Bei der Verwendung der „harten“ Variante kann das Datenprüfprogramm ggf. nicht mehr die ausführlicheren Plausibilitätsprüfungen durchführen und standardisierte Fehlermeldungen ausgeben, wenn Werteverstöße in der XML-Datei vorliegen.

**Hinweis**

Mit der Spezifikation 2019 werden alle bundesweit verpflichtenden Verfahren auf lange Überliegerverfahren umgestellt.

**Hinweis**

Möchten DAS weitergehende Prüfungen durchführen, die nicht in der Spezifikation enthalten sind, ist dieses an die BAS zu melden, welche die Aufnahme in das Spezifikationsregelwerk vornimmt. Dies gilt auch für neu zu spezifizierenden oder zu ändernde Fehlermeldungen.

1.3 Pseudonymisierung der LE-Daten

Zu den Aufgaben der DAS gehört für Verfahren gemäß Qesü-RL und 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 [IKNRKH], Betriebsstättennummer bei Praxen/MVZ [BSNRAMBULANT]) durch ein Pseudonym zu ersetzen.

Das Pseudonym des Leistungserbringers (LE-Pseudonym) wird unter Entfernung der IKNRKH/BSNRAMBULANT des Leistungserbringers in den XML-Code an vorgesehener Stelle eingetragen. DAS müssen in der Lage sein, den Leistungserbringer zu depseudonymisieren. Zudem muss das Pseudonymisierungsverfahren konstant sein, um Datenzusammenführungen mit Vorjahren zu erlauben.

Ab dem Erfassungsjahr 2019 entfällt die LE-Pseudonymisierung durch die DAS für Verfahren nach QSKH-RL.

Bei der Pseudonymisierung sind folgende richtlinienspezifische Regelungen zu berücksichtigen:

- Gemäß QSKH-RL entfällt die LE-Pseudonymisierung durch die DAS
- Gemäß Qesü-RL und DeQS-RL ist die LE-Pseudonymisierung verfahrensspezifisch. Welche Module ein bestimmtes Pseudonymisierungsverfahren bilden, ist der Spezifikationsdatenbank (QSDOK), Tabelle „Modul“, Spalte „fkPseudonymVerfahren“ zu entnehmen.

Für die systemweit einheitliche Pseudonymisierung gemäß Qesü-RL und DeQS-RL muss das Pseudonymisierungsprogramm (PSP) verwendet werden (Abschnitt B 5.4). Welche Elemente durch welche DAS zu pseudonymisieren sind, wird in der folgenden Tabelle aufgeführt.

Tabelle 3: Pseudonymisierung der LE-Daten bei Qesü- und DeQS-Verfahren

| Datenannahmestelle | Beschreibung |
|--------------------|---|
| LQS/LKG | <p><care_provider>/<IKNR> ist durch <care_provider>/<pseudonym> zu ersetzen. Für die Pseudonymisierung wird nur die IKNR verwendet.</p> <hr/> <p>Qesü-RL/DeQS-RL</p> <hr/> <p>Die Pseudonymisierung ist verfahrensspezifisch. Pseudonymisierung mittels des systemweit einheitlichen Verfahrens (PSP).</p> <hr/> <p>Spezielle Regelungen für CHE_KV:</p> <ul style="list-style-type: none"> ▪ Es ist das XML-Schema für die Datenübermittlung von vertragsärztlichen Daten zu verwenden. ▪ <care_provider>/<BSNRAMBULANT> ist durch <care_provider>/<pseudonym> zu ersetzen und <care_provider>/<NBSNRAMBULANT> ist durch <care_provider>/<pseudonym_nbsnr> zu ersetzen. ▪ Für die Pseudonymisierung wird die BSNRAMBULANT⁸ und ggf. die NBSNRAMBULANT verwendet. ▪ Die BSNRAMBULANT⁹, LANR¹⁰ und NBSNRAMBULANT¹¹ müssen vor der Weiterleitung entfernt werden. ▪ Die Pseudonymisierung erfolgt mittels des systemweit einheitlichen Verfahrens (PSP). Es ist auch hierfür das Schlüsselpaar für stationäre Datensätze zu verwenden. |
| KV | <ul style="list-style-type: none"> ▪ <care_provider>/<BSNRAMBULANT> ist durch <care_provider>/<pseudonym> zu ersetzen und <care_provider>/<NBSNRAMBULANT> ist durch <care_provider>/<pseudonym_nbsnr> zu ersetzen. ▪ Für die Pseudonymisierung wird die BSNRAMBULANT¹² und ggf. die NBSNRAMBULANT verwendet. ▪ Die Pseudonymisierung ist verfahrensspezifisch durchzuführen. |

⁸ Betriebsstättennummer (ambulant)

⁹ Betriebsstättennummer (ambulant)

¹⁰ Lebenslange Arztnummer

¹¹ Nebenbetriebsstättennummer (ambulant)

¹² Betriebsstättennummer (ambulant)

| Datenannahmestelle | Beschreibung |
|--------------------|--|
| | <ul style="list-style-type: none"> ▪ Die BSNRAMBULANT¹³, LANR¹⁴ und NBSNRAMBULANT¹⁵ müssen vor der Weiterleitung entfernt werden. ▪ Die Pseudonymisierung erfolgt mittels des systemweit einheitlichen Verfahrens (PSP). |
| DAS-SV | <ul style="list-style-type: none"> ▪ Für die Pseudonymisierung wird nur die BSNRAMBULANT verwendet. ▪ Die Pseudonymisierung ist verfahrensbezogen durchzuführen. ▪ Die LANR und NBSNRAMBULANT müssen vor der Weiterleitung entfernt werden. ▪ Die DAS-SV trägt zusätzlich die Registriernummer der Datenannahmestelle, welche ursprünglich das Pseudonym des Leistungserbringers erstellt hat, in das Attribut <pseudonym> / @registration. ▪ Die DAS-SV trägt zusätzlich die KV-Region in das Attribut <pseudonym> / @kvregion. Die KV-Region ist aus den ersten zwei Stellen der BSNRAMBULANT abzuleiten. ▪ Mittels des systemweit einheitlichen Verfahrens (PSP). |



Hinweis für DAS LQS/LKG

Bei dem Modul CHE_KV werden Leistungserbringer durch die Felder BSNRAMBULANT und NBSNRAMBULANT gekennzeichnet. Die Erzeugung der LE-Pseudonyme erfolgt aber dennoch mit dem gleichen öffentlichen Schlüssel der auch bei IKNRKH verwendet wird.

Hinweis für DAS-KK

Beim Modul CHE_KV ist für die LE-Pseudonymisierung der öffentliche Schlüssel der jeweiligen DAS LQS/LKG zu verwenden.



Für eine landesbezogene Auswertung der selektivvertraglichen Leistungen gemäß Qesü-RL bzw. DeQS-RL ist eine eindeutige Zuordnung zu den Bundesländern erforderlich. Hierfür ist eine von der KBV bereits spezifizierte Kodierung sehr gut geeignet (KV-Region). Der entsprechende Code (Tabelle 4, 2. Spalte, „Kode“) ist von der DAS-SV in die LE-Daten nach der Pseudonymisierung aufzunehmen. Dies lässt sich aus den ersten zwei Stellen der BSNRAMBULANT ableiten:

¹³ Betriebsstättennummer (ambulant)

¹⁴ Lebenslange Arztnummer

¹⁵ Nebenbetriebsstättennummer (ambulant)

Tabelle 4: Kodierung der Landesgeschäftsstellen der KVen

| KV-Region | Kode | BSNR (Stelle 1 und 2) |
|------------------------|------|--------------------------------|
| 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 |
| 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 |

**Hinweis**

Ob die Pseudonymisierung erfolgreich durchgeführt wurde, kann auch mit den entsprechenden Schemata unter `/interface_DAS/interface_<DAS>_psn.xsd` geprüft werden.

1.4 Weitere Datenverarbeitungen

Nach der Verarbeitung von XML-Dateien sind noch folgende Änderungen vorzunehmen:

- Aktualisierung der Absenderinformationen
- Gewährleistung der Eindeutigkeit der Vorgangsnummern (Datensatz-IDs)

Aktualisierung der Absenderinformationen

Die Absenderinformationen (`/header/provider`) müssen durch eigene Informationen (z. B. Funktion, Registriernummer) ersetzt werden.

Eindeutigkeit der Vorgangsnummern

Bei der Datenübermittlung muss sichergestellt sein, dass die Vorgangsnummern (/case/case_admin/id) der einzelnen Behandlungsfälle auch bei der BAS eindeutig sind. Für den ambulanten Sektor wird dieses durch die Verwendung von GUIDs erreicht.

Im stationären Bereich wird auf Ebene der DAS die Eindeutigkeit durch die Registrierung von QS-Systemen erreicht, in denen die Vorgangsnummern z. B. als laufende Nummern vergeben werden. Da die Registrierung des Leistungserbringers bei der Weiterleitung an die BAS durch die Registrierungsinformationen der DAS ersetzt wird, muss jede DAS die Eindeutigkeit der Vorgangsnummern innerhalb ihres Zuständigkeitsbereichs sicherstellen. Das Leistungserbringerpseudonym reicht dazu nicht aus, weil es innerhalb eines Krankenhauses mehrere registrierte QS-Systeme geben kann, unter deren Verwendung die fallbezogene QS-Dokumentation für ein Modul stattfindet

Darüber hinaus ist zu beachten, dass gemäß Qesü-RL sowie DeQS-RL ab dem Erfassungsjahr 2019 auch die Berichte und Auswertungen an die Krankenhäuser (stationärer Bereich) von der BAS so zu verschlüsseln sind, dass die zuständigen Datenannahmestellen, welche die Berichte und Auswertungen an die Leistungserbringer weiterleiten, keine unmittelbare Einsicht in diese Berichte und Auswertungen erhalten. Hierdurch soll sichergestellt werden, dass außer den betroffenen Leistungserbringern keine Dritten Befugnis zur Einsicht haben. Gemäß Richtlinie können Datenannahmestellen jedoch unter Wahrung des Datenschutzes auf Anfrage und mit Einwilligung des Leistungserbringers Einsicht in den jeweiligen Rückmeldebericht nehmen, um Fragen beantworten zu können und um die Leistungserbringer bei Fragen zu beraten.

Auch bei der Durchführung von Strukturierten Dialogen gem. QSKH-RL sowie Stellungnahmeverfahren gem. plan. QI-RL, bei denen die BAS mit dem Leistungserbringer in Kontakt tritt, werden die originären Vorgangsnummern der Leistungserbringer in Bezug auf auffällige Vorgänge benötigt.

Die Verschlüsselung der Berichte und Auswertungen berücksichtigend, muss aus Sicht der Leistungserbringer sichergestellt werden, dass die originär im QS-System der Leistungserbringer erstellte und an die Datenannahmestelle übermittelte Vorgangsnummer in der BAS zugänglich ist. Dies wird dadurch erreicht, dass im Ergebnis einer von der Datenannahmestelle veränderten Vorgangsnummer, die originäre Vorgangsnummer des Leistungserbringers als Suffix, unter Verwendung des Trennzeichens # extrahiert werden kann.

Um die Eindeutigkeit der Vorgangsnummer für die Bundesebene herzustellen, kommen verschiedene Verfahren in Frage:

- Umsetzungstabelle: Wenn die originäre Vorgangsnummer des Leistungserbringers nach dem Trennzeichen # als Suffix beinhaltet, kann die Vorgangsnummern im Zusammenhang mit der Registriernummer und der Modulkenung in einer Datenbanktabelle aufgenommen und auf eine in Kombination mit der Registriernummer der Datenannahmestelle eindeutige Kennung umgesetzt werden.
- Formvorschrift: Ein mit dem Zeichen # getrenntes, vorlaufendes Anfügen der Registriernummer sowie ggf. weiterer Kennungen an die Vorgangsnummer erzeugt ebenfalls eine eindeutige Vorgangsnummer.

Wichtig ist, dass innerhalb der Vorgangsnummern einer DAS eine Kollision ausgeschlossen wird. Wegen der sogenannten Überliegerverfahren muss zudem eine jahresübergreifende Identifizierung sichergestellt werden. Nur dann sind die Vorgangsnummern über die Zuordnung zur Registriernummer der DAS auf Bundesebene für die BAS eindeutig.



Hinweise und Umsetzungsempfehlung

- Um der gem. Qesü-RL bzw. DeQS-RL vorgegebenen, verfahrensspezifischen Leistungserbringerpseudonymisierung Rechnung zu tragen, sollte als Präfix vor dem Trennzeichen nicht alleinig die Registriernummer des Dokumentationssystems verwendet werden.¹⁶
- Wenn ein Leistungserbringer das gleiche Dokumentationssystem für die Dokumentation zweier Verfahren verwendet, welche jeweils einer verfahrensspezifischen Leistungserbringerpseudonymisierung unterliegen (z. B. PCI_LKG und NWIF), so ist zu vermeiden, dass über das Präfix einer von der DAS vergebenen Vorgangsnummer die unterschiedlichen Leistungserbringerpseudonyme zusammengeführt werden könnten.¹⁷
- Wenn ein Leistungserbringer Vorgänge zu dem gleichen Modul mithilfe von unterschiedlichen QS-Systemen erfasst, kann dies mit einem erhöhten Suchaufwand bei der Fallidentifizierung, z. B. Fälle aus Listen mit auffälligen Vorgängen, einhergehen.

1.5 XML-Verschlüsselung

Vor der Weiterleitung des verarbeiteten Dokuments sind zum einen die Leistungserbringerkennungen (`/care_provider`) und zum anderen die QS-Daten (`/qs_data`) von allen DAS mit dem öffentlichen Schlüssel der BAS zu verschlüsseln. Um zwei unterschiedliche Verarbeitungsprozesse bei den Datenannahmestellen und der Bundesauswertungsstelle zu vermeiden, wird diese Verschlüsselung auch bei Modulen ohne PID eingehalten, obwohl diese Daten von der Datenannahmestelle direkt an die Bundesauswertungsstelle geliefert werden.



Hinweis

Ob die XML-Verschlüsselung erfolgreich durchgeführt wurde, kann auch mit den entsprechenden Schemata für die Ausgangskontrolle geprüft werden.

¹⁶ Für die Verwertung von Datenflussprotokollen beim Leistungserbringer ist die in der BAS fehlende Information der Registriernummer des Dokumentationssystems insofern unerheblich, als dass diese bereits im Dateinamen der Transaktionsdatei, welche von der Datenannahmestelle an die Leistungserbringer übermittelt wird, enthalten ist, vgl. Abschnitt 1.8 Weiterleitung der Datenflussprotokolle der BAS bzw. 1.7.4 Erzeugung der Antwortdatei für die Übertragung via E-Mail (LQS/LKG).

¹⁷ Aus diesem Grunde besteht auch nicht die Möglichkeit die Anforderung an die von der DAS zu erstellende Vorgangsnummer derart zu definieren, dass neben der originären Vorgangsnummer des Leistungserbringers auch die Registriernummer des Dokumentationssystems in der BAS zugänglich ist.

Das Verschlüsselungsverfahren und die konkreten XML-Elemente, auf denen die Verschlüsselung durchzuführen ist, können dem Abschnitt „Verschlüsselung“ entnommen werden (Abschnitt B 5.3.1). Die Notwendigkeit zu dieser Verschlüsselung ergibt sich aus Datenschutzgründen für Dokumente, die über die Vertrauensstelle einer Pseudonymisierung der patientenidentifizierenden Daten (PID) zugeführt werden sollen.

1.6 Datenübermittlung

Im folgenden Abschnitt werden die abschließend zu unternehmenden Arbeitsschritte (Dateibenennung (LQS/LKG), Ausgangsprüfung und Datenweiterleitung) beschrieben.

LQS
LKG

1.6.1 Dateibenennung (LQS/LKG)

Die Daten werden als XML-Datei an die VST weitergeleitet. Die Exportdatei muss nach einem festgelegten Schema benannt werden (Abschnitt A „Benennung der Exportdateien“).

Beispiel:

QS-Daten einer DAS

47d16341-9e27-4e75-a27e-b791fbbd2dc8_Q_DAS.xml

1.6.2 Ausgangsprüfung

Vor der Weiterleitung der Daten an die nächste Stelle ist eine Ausgangskontrolle durchzuführen. Zu diesem Zweck werden die entsprechenden XML-Schemata (interface_DAS_VST/, interface_DAS_BAS/ Abschnitt B 4.2) als Bestandteil der Spezifikation bereitgestellt. Diese Prüfung kann auch mithilfe des Datenprüfprogramms durchgeführt werden.

Beispiel:

Ausgangskontrolle DAS/VST

Vor der Weiterleitung der Exportdatei an die VST kann das Schema interface_DAS_VST/interface_DAS_VST.xsd verwendet werden.

LQS
LKG

1.6.3 Datenweiterleitung

Für die Übermittlung der Daten über nicht gesicherte Übertragungswege (E-Mail) muss die Exportdatei in eine Transaktionsdatei¹⁸ umgewandelt werden.

Das IQTIG stellt ein Verschlüsselungsprogramm bereit, mit dem eine verfahrenskonforme Transportverschlüsselung durchgeführt werden kann. Das Einbinden der Funktionen des Verschlüsselungsprogramms erfolgt über einen Befehlszeilenaufwurf mit Parametern. Das Verschlüsselungsprogramm übernimmt auch die Dateibenennung der Transaktionsdatei mithilfe von übergebenen Parametern (Abschnitt B 5.3.2).

¹⁸ Verpackte und symmetrisch verschlüsselte AES-Datei.

Der Leistungserbringer erhält bei der Registrierung seiner Dokumentationssysteme von der zuständigen DAS einen Verschlüsselungskode, der bei der Erzeugung der Archivdatei verwendet werden muss. Die Transaktionsdatei wird nach einem vordefinierten Schema benannt (Abschnitt A „Übermittlung der Daten im Datenfluss“):

Beispiel:

Transaktionsdatei

T-BU1234a-2016_01_19_160945.zip.aes

T-BU1234a-2016_01_23_114113_045.zip.aes (millisekundengenau)

Sollte der Transportweg (z. B. eine Webschnittstelle) sicher sein (z. B. durch eine SSL-Verschlüsselung) ist eine Transportverschlüsselung nicht durchzuführen (Tabelle 5).

Tabelle 5: Bisher vorgesehene Datenübertragungswege

| Daten-absender | Daten-empfänger | Übertragungsweg | Transportverschlüsselung mit dem TPACKER |
|----------------|-----------------|------------------|--|
| LQS/LKG | VST | E-Mail-Verfahren | Erforderlich |
| | | Rest-Service | Nicht erforderlich, da SSL-verschlüsselt |
| | BAS | E-Mail-Verfahren | Erforderlich |
| KV | VST | Rest-Service | Nicht erforderlich, da SSL-verschlüsselt |
| DAS-SV | VST | Rest-Service | Nicht erforderlich, da SSL-verschlüsselt |

1.7 Rückprotokollierung durch die Datenannahmestellen

Im folgenden Abschnitt wird beschrieben, welche Protokolle zu einer Datenlieferung die DAS erstellt oder nach einer erfolgreichen Datenweiterleitung erhält.

1.7.1 Empfangsbestätigung

Nach erfolgreicher Eingangsverarbeitung und Weiterleitung durch die DAS erhält der LE von der DAS auf dem Eingangskanal eine Empfangsbestätigung, die den Erhalt der Exportdatei bestätigt. Ist eine Datenlieferung nicht bearbeitbar, wird anstelle einer Empfangsbestätigung eine reduzierte Form des Datenflussprotokolls, ein sog. „Miniprotokoll“ an den LE übermittelt. Wenn die Exportdatei bearbeitbar ist, aber das Dokument bei der Prüfung wegen harter Fehler den Status `ERROR` (`status_document=ERROR`) erhält, wird ebenfalls keine Empfangsbestätigung, sondern ein Datenflussprotokoll an den LE gesendet. Diese Fälle kommen vor, wenn

- die administrativen Daten fehlerhaft sind (`status_document=ERROR`),
oder wenn
- alle QS-Datensätzen fehlerhaft sind (`status_case=ERROR`)

Die genaue Struktur ist dem Abschnitt A 5.2.2 zu entnehmen.

1.7.2 Miniprotokoll

Die Erstellung eines vollständigen Datenflussprotokolls durch die DAS ist nur möglich, wenn die ursprüngliche XML-Datei lesbar ist und nach Entfernung der PID und der QS-Daten schemakonform bleibt. Andernfalls ist durch die annehmende DAS an dieser Stelle ein reduziertes Datenflussprotokoll („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. Ein Miniprotokoll ist z. B. zu erstellen, wenn die Exportdatei nicht wohlgeformt ist.

1.7.3 Datenflussprotokoll

Wenn nach einer vollständigen Prüfung der XML-Datei der Status des Dokuments auf `ERROR` gesetzt wurde, wird die Export-Datei nicht weitergeleitet. Stattdessen wird durch die DAS ein Datenflussprotokoll erstellt, indem die Exportdatei von allen QS-Daten (Element `<qs_data>`) und patientenidentifizierenden Daten (Element `<patient>`) befreit wird.



1.7.4 Erzeugung der Antwortdatei für die Übertragung via E-Mail (LQS/LKG)

Das Datenflussprotokoll der DAS wird nach dem gleichen Verfahren wie die Transaktionsdatei in ein mit AES verschlüsseltes ZIP-Archiv gepackt und nach einem vordefinierten Schema benannt (Abschnitt A „Übermittlung der Daten im Datenfluss“)

Beispiel:

Benennung eines Datenflussprotokolls

Antwortdatei: A-BW1234a-2016_09_23_094051.zip.aes
auf die Transaktionsdatei: T-BW1234a-2016_09_23_094051.zip.aes

Nach welchen XML-Schemata die Rückprotokolle aufgebaut sind, wird in der folgenden Tabelle aufgeführt.

Tabelle 6: XML-Schemata der Datenflussprotokolle

| Schnittstelle | Schema | Beschreibung |
|-------------------|----------------------|-----------------------------|
| LE/DAS | response_DAS_LE.xsd | Datenflussprotokoll der DAS |
| interface_LE_DAS | response_receipt.xsd | Empfangsbestätigung der DAS |
| DAS/VST | response_VST_DAS.xsd | Datenflussprotokoll der VST |
| interface_DAS_VST | response_receipt.xsd | Empfangsbestätigung der VST |
| DAS/BAS | response_BAS_DAS.xsd | Datenflussprotokoll der BAS |
| interface_DAS_BAS | | |



Hinweis

Die Struktur der Datenflussprotokolle sowie der Empfangsbestätigungen ist in allen Schnittstellen identisch.

Weitere Informationen zur Erzeugung und Struktur der Rückprotokolle sind dem Abschnitt A 5.2 zu entnehmen.

1.7.5 Empfangsbestätigung durch die VST

Wenn die DAS eine QS-Datendatei verarbeitet und an die Vertrauensstelle weitergeleitet hat, so wird die VST über den Eingangskanal eine Empfangsbestätigung an die DAS senden. Sollte diese nicht binnen 24 Stunden bei der DAS eingehen, ist die VST zu informieren. Eine wiederholte Datenübermittlung ohne vorherige Abstimmung mit der VST ist zu vermeiden.

Helpdesk der VST: <http://www.vertrauensstelle-gba.de/kontakt.html>

1.7.6 Referenz für die HTML-Formatierung des Rückprotokolls

Das Institut nach §137a SGB V stellt ein XSLT-Skript für die Umwandlung des Rückprotokolls in ein lesbares HTML-Format zur Verfügung. Um für den Leistungserbringer möglichst einfach nutzbar zu sein, müssen die DAS im Rückprotokoll einen entsprechenden Verweis eintragen (Abschnitt A „Prüfungsprozess und Ergebnisprotokollierung“).

1.8 Weiterleitung der Datenflussprotokolle der BAS

Die Bundesauswertungsstelle hat im Zusammenhang mit Rückprotokollen eine Sonderrolle, da sie am Ende der Verarbeitungskette steht. Sie erzeugt standardmäßig zwei Protokolle:

- Die VST erhält im Erfolgsfall eine Empfangsbestätigung
- DAS erhalten ein Datenflussprotokoll

Empfangsbestätigung für die VST

Es kommt zu Abweichungen dieses Vorgehens, wenn die Datei nicht schemakonform ist oder die Registrierung der DAS bei der VST der BAS nicht bekannt ist. In diesen Fällen wird nur ein Datenflussprotokoll an die Vertrauensstelle geschickt.

Datenflussprotokoll für die DAS

- Die Datenflussprotokolle werden transportverschlüsselt und an die von der BAS registrierten E-Mail-Adressen gesendet (AES-Datei).
- Da das Datenflussprotokoll symmetrisch verschlüsselt ist, muss die DAS die AES-Datei symmetrisch mit dem in der Registrierung vergebenen Passwort entschlüsseln (Abschnitt B 5.3.2).
- Das im Datenflussprotokoll enthaltene LE-Pseudonym bei Verfahren gem. Qesü-RL bzw. DeQS-RL muss vor der Weiterleitung an den LE depseudonymisiert werden (IKNRKH bzw. BSNRAMBULANT).
- Ggf. im Datenfluss vom LE an die BAS veränderte Vorgangsnummern (betrifft lediglich Krankenhaus-Bereich) müssen für die LE zurückgesetzt werden.
- Die Weiterleitung ist nur vorzunehmen, wenn der Verursacher eines Fehlers nicht die DAS ist.

Dies ist aus dem Attribut (`validation_item/status/error/@originator`) zu entnehmen. Sollte die DAS der Verursacher des Fehlers auf Dokumentenebene sein, ist das Dokument zu korrigieren und erneut zu übermitteln. Die Voraussetzung für eine erneute Übermittlung ist die Stornierung der vorherigen Transaktion (Übermittlung der Exportdatei) bei der BAS.



Achtung
Transaktion ohne Quittierung

Sollte das Protokoll nicht binnen 24 Stunden bei der DAS eingehen, ist die VST zu informieren. Eine wiederholte Datenübermittlung ohne Abstimmung mit der VST ist zu vermeiden.

Besonderer Fehlerfall

Aufgrund von fehlerhaften, automatisierten Verarbeitungen in der DAS, VST oder BAS kann der Zustand entstehen, dass beim Leistungserbringer bereits ein Datenflussprotokoll zu einer GUID vorliegt, deren involvierte Vorgänge jedoch erneut verarbeitet werden müssen.

Das präferierte Vorgehen ist ein erneuter Versand aller betroffenen Datensätze durch den Leistungserbringer, welcher telefonisch von der zuständigen DAS an den betroffenen Leistungserbringer initiiert werden muss. Für den erneuten Versand ist beim Leistungserbringer ein Export aller betroffenen Vorgangsnummern einschließlich der Erhöhung der zugehörigen Versionsnummer durchzuführen. Das im Zuge des Exports erstellte XML-Dokument erhält eine neue und damit unverbrauchte GUID. Somit können alle Datensätze im Datenfluss über alle folgenden Instanzen (DAS, VST, BAS) einschließlich der zugehörigen Rückprotokollierung mittels des Datenflussprotokolls bis hin zum Leistungserbringer regulär verarbeitet werden.

Soll jedoch alternativ ein durch alle Instanzen (einschließlich Rückprotokollierung mittels Datenflussprotokoll bis hin zum Leistungserbringer) bereits verarbeitetes Dokument und somit eine bereits verarbeitete GUID, erneut verarbeitet werden, so dass kein erneuter Versand der betroffenen Vorgangsnummern direkt beim Leistungserbringer (unter einer neuen GUID) veranlasst wird, muss die Software des Leistungserbringers in der Lage sein, zu einer GUID mehr als ein Datenflussprotokoll aufnehmen zu können. Analog zur wiederholten Verarbeitung einer GUID für den Fall, dass im Attribut „`originator`“ ein von Leistungserbringer oder Softwareanbieter abweichender Wert steht, kann das Zurücksetzen eines Dokuments per E-Mail über die Adresse verfahrensupport@iqtig.org vereinbart werden.



1.9 Datenhaltung (LQS/LKG) – QSKH-Richtlinie

Die DAS, die Daten von Krankenhäusern im Rahmen der QSKH-RL entgegennehmen, sind gemäß QSKH-RL für die leistungserbringerspezifischen Auswertungen und die Durchführung des strukturierten Dialogs zuständig.

Wichtig ist dafür die Speicherung der Daten auf dem Weg vom Leistungserbringer zu nachfolgenden Einrichtungen, also vor dem Versand an die VST bzw. der BAS, da auf dem Rückweg mit

dem Datenflussprotokoll die Daten selbst nicht mehr in den versendeten Dokumenten enthalten sind.



Hinweis

Da der DAS nachgelagerte Prüfungen bei der Vertrauensstelle und bei der Bundesauswertungsstelle stattfinden, ist auf dem Rückweg von der Datenannahmestelle jeweils der Status eines Datensatzes `<status_case>` mitzulesen und im Datenpool der Datenannahmestelle zu vermerken.

1.10 Zusammenfassung

Folgende Tabelle stellt einen Überblick über die Aufgabenbereiche der Datenannahmestellen dar.

Tabelle 7: Aufgaben der Datenannahmestellen

| Prozesse | KV | DAS-SV | LQS | Anmerkungen/Eigenschaften |
|---|----|--------|-----|---|
| Datenannahme | x | x | x | Gesicherte Datenübertragung; unterschiedliche Übertragungswege möglich, Registrierung (LQS/LKG, DAS-SV), Authentifizierung, Beachtung der Fristen |
| Entpacken der AES-Dateien | | | x | Durch die LQS/LKG |
| Eingangskontrolle | x | x | x | Mittels XML-Schema (mit oder ohne DPP) |
| Prüfung der Admin-Daten | x | x | x | Mittels DPP, administrative Objekte in der QSDOK-Datenbank |
| Entschlüsselung der QS-Daten | | | x | Mittels XPacker (qs_data) |
| Prüfung der QS-Daten | | | x | Mittels Plausibilitätsregeln der QSDOK-DB, DPP |
| LE-Pseudonymisierung | x | x | x | QSKH: Die LE-Pseudonymisierung wird durch das IQTIG durchgeführt; Qesü-RL/DeQS-RL: verfahrensspezifisch mittels Pseudonymisierungsprogramm |
| Eintragung der KV-Region | | x | | Durch die DAS-SV |
| Anpassung der Absenderinformationen | x | x | x | Provider-Tag im header |
| Herstellung der Eindeutigkeit von Vorgangsnummern auf Bundesebene | | | x | id-Tag im case_admin |

| Prozesse | KV | DAS-SV | LQS | Anmerkungen/Eigenschaften |
|--|----|--------|-----|---|
| Verschlüsselung der QS-Daten | x | x | x | Mittels XPacker (qs_data) |
| Verschlüsselung der LE-Kennungen bzw. deren Pseudonyme | x | x | x | Mittels XPacker (care_provider) |
| Ausgangskontrolle (Schema) | x | x | x | Mittels Schemata für die Schnittstelle zwischen DAS/VST |
| Empfangsbestätigung | x | x | x | Benachrichtigung über Erhalt und Weiterleitung der Daten |
| Datenflussprotokoll im Fehlerfall | x | x | x | Benachrichtigung über fehlerhafte Datenlieferung (Dokumentstatus ERROR) Mittels Schema, DPP, QSDOK-DB (administrative Objekte) |
| Datenversand an die VST per Web-Service | x | x | | Nach Registrierung bei der VST |
| Datenversand per E-Mail | | x | x | Voraussetzung: Registrierung der LE bzw. ihrer einzelnen Dokumentations-systeme bei der DAS |
| Versand an die BAS | | | x | Indirekte Verfahren |
| Erhalt der Datenflussprotokolle von der BAS | x | x | x | E-Mail-Verfahren |
| Entpacken der Datenflussprotokolle | x | x | x | Mittels TPacker |
| Entpseudonymisierung der LE-Pseudonyme | x | x | x | Qesü-RL/DeQS-RL: Mittels Pseudonymisierungsprogramm |

2 Risikostatistik

Zur Verringerung des Dokumentationsaufwands werden im Leistungsbereich Pflege: Dekubitusprophylaxe seit dem Erfassungsjahr 2013 händische Dokumentationen nur noch für Patienten mit Dekubitus durchgeführt. Ergänzend hierzu sind für eine sachgerechte Risikoadjustierung jedoch Basisinformationen für die gesamte Zielpopulation des Leistungsbereichs erforderlich. Die notwendigen Informationen müssen vom Leistungserbringer automatisiert auf Basis vorhandener Abrechnungsdaten erstellt und in einer Risikostatistik zusammengefasst werden.

Nachfolgend werden die einzelnen Prozesse und Unterprozesse der Risikostatistik beschrieben.

2.1 Datenübermittlung

Die Risikostatistik für das Erfassungsjahr 2019 ist spätestens bis zum 15. Februar des der Datenerhebung nachfolgenden Jahres (einschließlich) als komprimierter und verschlüsselter E-Mail-Anhang an die Datenannahmestelle der zuständigen Landesgeschäftsstelle Qualitätssicherung zu übermitteln. Fällt der 15. Februar auf ein Wochenende oder einen Feiertag, gilt entsprechend der gesetzlichen Regelungen der nächste Werktag.

Die beiden Exportdateien `RISIKOBASIS_<Erfassungsjahr>.txt` und `RISIKOSTATISTIK_<Erfassungsjahr>.txt` werden mithilfe eines pkzip 2.04g-kompatiblen Archivierungsprogramms komprimiert und unter dem Namen `RS_<Erfassungsjahr>_<IKNRKH>_<STANDORT>.zip` gespeichert.

Wenn das Krankenhaus nur einen Standort hat, dann ist im Namen „00“ als Standortnummer anzugeben. Bei mehreren Standorten wird jeweils die entsprechende Standortnummer aus den Daten übernommen.

Die Verschlüsselung des ZIP-Archivs erfolgt mit der Verschlüsselungssoftware GnuPG¹⁹ Version 1.2.1 (GNU Privacy Guard). Dasselbe Public-Key-Verschlüsselungsverfahren wird bereits von den deutschen Krankenhäusern im Rahmen der Datenübermittlung nach § 21 KHEntG an die DRG-Datenstelle (<http://www.g-drg.de>) eingesetzt.

Für die Verschlüsselung wird der öffentliche Schlüssel der jeweiligen Datenannahmestelle verwendet, an welche die Risikostatistik übermittelt wird. Die E-Mail-Adressen der auf Landesebene zuständigen Stelle erfolgt in einem separaten Merkblatt, das auf der Seite www.iqtig.org zusammen mit den öffentlichen Schlüsseln bereitgestellt wird.

Das ZIP-Archiv `RS_<Erfassungsjahr>_<IKNRKH>_<STANDORT>.zip` ist mit dem öffentlichen Schlüssel der zuständigen Landesgeschäftsstelle zu verschlüsseln und unter folgendem Namen zu speichern:

```
RS_<Erfassungsjahr>_<IKNRKH>_<STANDORT>_<Ländercode>.zip.gpg
```

Die einzutragenden Ländercodes sind in der folgenden Tabelle aufgeführt.

¹⁹ Diese frei verfügbare Open-Source-Software unterliegt der GNU General Public License.

Tabelle 8: Ländercodes der Landesgeschäftsstellen

| Ländercode | Landesgeschäftsstelle |
|------------|------------------------|
| BA | Bayern |
| BB | Brandenburg |
| BE | Berlin |
| BW | Baden-Württemberg |
| HB | Bremen |
| HE | Hessen |
| HH | Hamburg |
| MV | Mecklenburg-Vorpommern |
| NI | Niedersachsen |
| NW | Nordrhein-Westfalen |
| RP | Rheinland-Pfalz |
| SH | Schleswig-Holstein |
| SL | Saarland |
| SN | Sachsen |
| ST | Sachsen-Anhalt |
| TH | Thüringen |

2.2 Datenprüfung und Rückprotokollierung



Form der Rückprotokollierung

Für die Rückprotokollierung zur Risikostatistik ist in der Spezifikation bisher kein standardisiertes Format vorgegeben. Die Landesgeschäftsstellen werden die Bestätigung der erfolgreichen Datenlieferung bzw. von Fehlermeldungen in geeigneter Form sicherstellen.

Die Risikostatistik muss vollständig und plausibel sein. Unplausible oder unvollständige Risikostatistiken werden von den Datenannahmestellen als nicht geliefert angesehen. Für nicht akzeptierte Risikostatistiken wird ein Fehlerprotokoll erstellt und per E-Mail an das Krankenhaus übermittelt.

Auf Seiten der Datenannahmestellen sind folgende Prüfungen vorgesehen:

Technische Vorgaben der Risikostatistik

Die Dateien RISIKOBASIS_<Erfassungsjahr>.txt und RISIKOSTATISTIK_<Erfassungsjahr>.txt müssen alle Daten gemäß der spezifizierten Felder und ihrer Eigenschaften (Feldnamen, Datentyp, Feldlänge) enthalten.

Inhaltliche Prüfung der RISIKOBASIS

1. Felder IKNRKH und STANDORT

Der Inhalt der beiden Felder soll im Dateinamen der verschlüsselten Datei (RS_<Erfassungsjahr>_<IKNRKH>_<STANDORT>...) enthalten sein. Geprüft werden kann auch anhand der in jeder Landesgeschäftsstelle vorliegenden Liste der IKNR (ggf. zusätzlich STANDORT), ob die Risikostatistik von einem Krankenhaus aus dem jeweiligen Zuständigkeitsbereich stammt.

2. Feld RSJAHR

Für die Risikostatistik zum Erfassungsjahr 2019 muss hier der Wert 2019 stehen.

3. Feld DOKABSCHLDAT

Da nach den Vorgaben der Risikostatistik alle vollstationären Fälle in die Statistik einzubeziehen sind, die bis spätestens 31.12.2019 entlassen wurden, kann formal einheitlich nur mit einem Datum ab dem 01.01.2020 eine Prüfung erfolgen.

4. Feld FAELLE_GEPRUEFT

In diesem Feld ist die Anzahl der geprüften Behandlungsfälle einzutragen. Die Risikostatistik ist nur von Krankenhäusern bzw. Standorten abzugeben, die mindestens einen Behandlungsfall gemäß Spezifikation haben. Daher soll ein Warnhinweis gegeben werden, wenn die angegebene Zahl gleich 0 ist.

Inhaltliche Prüfung der RISIKOSTATISTIK

1. Felder IKNRKH und STANDORT

Da die Prüfung des Inhalts der beiden Felder bei RISIKOBASIS beschrieben ist und dort sinnvollerweise durchgeführt wird, ist nur noch zu überprüfen, ob die Inhalte in RISIKOBASIS mit denjenigen in jeder Zeile von RISIKOSTATISTIK übereinstimmen.

$$\text{RISIKOBASIS.IKNRKH} = \text{RISIKOSTATISTIK.IKNRKH}$$

$$\text{RISIKOBASIS.STANDORT} = \text{RISIKOSTATISTIK.STANDORT}$$

2. Die Felder DEK2, DEK3, DEK4, DEKnnb, DIABETES, MOBILITAET, INFEKTION, DEMENZ, INKONTINENZ, UNTERGEWICHT, ADIPOSITAS und SONSTERKRANKUNG können ausschließlich den Wert 1 oder 0 annehmen.

3. Die Anzahl an Datensätzen in der Risikostatistik muss dem Wert des Feldes FAELLE_GEPRUEFT in der RISIKOBASIS entsprechen.

4. Jeder Fall darf nur eine Ausprägung pro Gruppe haben, d. h. pro Gruppe ist in einem Datensatz nur einmal der Wert 1 zu vergeben, während die anderen Felder derselben Gruppe den Wert 0 annehmen müssen. Innerhalb einer Gruppe können auch alle Felder den Wert 0 haben.

Folgende Gruppen sind definiert:

- DEK2, DEK3, DEK4
- DEKnnb
- DIABETES
- MOBILITAET
- INFEKTION
- DEMENZ

- INKONTINENZ
- UNTERGEWICHT
- ADIPOSITAS
- SONSTERKRANKUNG

2.3 Datenweiterleitung

Gegenstand dieses Kapitels ist die Erzeugung und Weiterleitung der Risikostatistik-Daten von der Landesebene an die Bundesebene.

Das Verfahren orientiert sich eng an dem in der „Technischen Dokumentation für Leistungserbringer zur Basisspezifikation“ beschriebenen Prozess zwischen Krankenhaus und LQS:

Die Risikostatistik-Lieferung von der Landesebene an die Bundesebene enthält ebenfalls die beiden Exportdateien `RISIKOBASIS_<Erfassungsjahr>.TXT` und `RISIKOSTATISTIK_<Erfassungsjahr>.TXT`.

Die Struktur der Exportdateien (`RISIKOBASIS_<Erfassungsjahr>.txt` und `RISIKOSTATISTIK_<Erfassungsjahr>.txt`) ist gleich.

Die Exportdateien besitzen die gleichen Exportfelder (identische Feldnamen) wie die Exportdateien der Spezifikation für die Risikostatistik.

Der wesentliche Unterschied in der Erzeugung der Exportdateien in den Krankenhäusern und bei den Landesgeschäftsstellen besteht darin, dass die Exportdateien an die Bundesebene möglichst die Datensätze von allen Krankenhäusern des Zuständigkeitsbereiches umfassen sollen. Ergänzend ist insbesondere im Zusammenhang mit Korrekturlieferungen aber auch die Übermittlung für einzelne Krankenhäuser bzw. Standorte möglich.

Bei der Übermittlung der Exportdateien ist entsprechend der Spezifikation keine Versionskontrolle vorgesehen. Falls für die einzelnen Krankenhäuser bzw. Standorte mehrere Risikostatistiken an die BAS gesendet werden, gilt daher stets die letzte, fehlerfrei übermittelte Lieferung als Basis für die Auswertungen.

Datenübermittlung

Die Risikostatistiken sind als komprimierter und verschlüsselter E-Mail-Anhang an die Bundesebene bis spätestens zum 15. März zu übermitteln.

Die beiden Exportdateien `RISIKOBASIS_<Erfassungsjahr>.txt` und `RISIKOSTATISTIK_<Erfassungsjahr>.TXT` werden mit Hilfe eines Archivierungsprogramms komprimiert und unter folgendem Namen gespeichert `RS_<Erfassungsjahr>_< Ländercode >.zip`.

Die Verschlüsselung des ZIP-Archivs erfolgt per PGP-Verfahren mit dem öffentlichen Schlüssel der Bundesebene. Die komprimierte und verschlüsselte Exportdatei hat den Dateinamen:

`RS_<Erfassungsjahr>_<Ländercode>.zip.gpg`

Beispiel: `RS_<Erfassungsjahr>_HH.zip.gpg`

Die verschlüsselten Risikostatistiken werden als Transaktionsdatei per E-Mail an die zuständige E-Mail-Adresse der Bundesebene übermittelt.

Die Plausibilitätsregeln (Abschnitt A 2.2) sind auch bei der Weiterleitung der Risikostatistikdaten einzuhalten. Nachdem die eingehenden Risikostatistiken auf Plausibilität geprüft wurden, erhalten die Landesgeschäftsstellen für jede eingegangene Transaktionsdatei eine Antwortdatei.

3 Sollstatistik

Mit der Sollstatistik soll nach Abschluss eines Erfassungsjahres und unabhängig von den tatsächlich dokumentierten QS-Fällen ermittelt werden, wie viele Fälle im Krankenhaus bzw. im ambulanten Bereich nach den definierten Bedingungen dokumentationspflichtig waren. Sie bildet damit eine wichtige Grundlage zur Bestimmung der Vollständigkeit und Vollzähligkeit. Die rechtliche Grundlage zur Ermittlung der Sollstatistik wird durch die §§ 23-24 der „Richtlinie über Maßnahmen der Qualitätssicherung in Krankenhäusern – QSKH-RL“²⁰, bzw. § 15 der „Richtlinie zur einrichtungs- und sektorenübergreifenden Qualitätssicherung – Qesü-RL“²¹ bzw. der Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) definiert. Mit der vorliegenden Spezifikation können Systeme entwickelt werden, die die korrekten Angaben ermitteln und dadurch die Grundlage für die Konformitätserklärung und die Vollzähligkeit der QS-Dokumentationen des Leistungserbringers liefern.

Nachfolgend werden die einzelnen Prozesse und Unterprozesse der Sollstatistik beschrieben. In der Tabelle werden diese Prozesse in Abhängigkeit von den Richtlinien (QSKH, Qesü und DeQS), den betroffenen Datenannahmestellen (DAS) und ihrer Relevanz für die jeweiligen Zielgruppen dargestellt. Die einzelnen Abschnittsnummern verweisen auf die entsprechenden Abschnitte, in denen sie genauer erläutert werden.

Spezifikationskomponenten sind neben dieser technischen Dokumentation die Vorlagen zur Sollstatistik und die Datenbank für QS-Filter-Software (Abschnitt B Komponenten).

3.1 Erzeugung

Die Sollstatistik muss nur für stationär bzw. ambulant erbrachte Leistungen am Krankenhaus und für selektivvertraglich erbrachte Leistungen vom Leistungserbringer erstellt werden. Die Sollstatistik wird in diesen beiden Fällen zusätzlich zu den QS-Daten berechnet und an die Datenannahmestelle übermittelt (Tabelle 9). Für kollektivvertraglich erbrachte Leistungen niedergelassener Praxen/MVZ übernehmen die KVen die Ermittlung der Sollzahlen auf Basis der Abrechnungsdaten und übermitteln das Ergebnis dieser Prüfung an die Bundesauswertungsstelle.

Tabelle 9: Zuständigkeit für die Erstellung der Sollstatistik

| Datenannahmestelle | Erstellung | Konformitätserklärung |
|--------------------|---------------------------------|--|
| LQS/LKG | Erfolgt durch die Krankenhäuser | Die LE bestätigen die Sollzahlen mit einer Konformitätserklärung |
| KV | Erstellung erfolgt durch die KV | Eine Konformitätserklärung ist nicht erforderlich |

²⁰ <https://www.g-ba.de/informationen/richtlinien/38/>

²¹ <https://www.g-ba.de/informationen/richtlinien/72/>

| Datenannahmestelle | Erstellung | Konformitätserklärung |
|--------------------|---------------------------------|--|
| DAS-SV | Erstellung erfolgt durch die LE | Die LE bestätigen die Sollzahlen mit einer Konformitätserklärung |

Die Daten der Sollstatistik werden der Bundesauswertungsstelle zur Berichterstellung übermittelt.

Im Modul CHE_KV gemäß DeQS-RL werden belegärztliche Leistungen abgebildet. Die zuständige KV erstellt eine Aufstellung aus der die Zahl der zu dokumentierenden Datensätze (Soll) hervorgeht. Anders als im Verfahren PCI übermittelt die KV diese

- im csv-Format
- an die jeweilige Datenannahmestelle.

Zusätzlich zur Sollstatistik in elektronischer Form muss eine Konformitätserklärung an die Datenannahmestelle übermittelt werden. Für die kollektivvertraglich erbrachten Leistungen von Praxen/MVZ entfällt die Konformitätserklärung durch den Leistungserbringer, weil diese Daten von der zuständigen KV zur Verfügung gestellt werden.

3.1.1 Erzeugung durch die LQS/LKG

Gegenstand dieses Kapitels ist die Erzeugung der Sollstatistik-Daten für die Übermittlung von der Landesebene an die Bundesebene.

Das Verfahren orientiert sich eng an dem in der „Technischen Dokumentation für Leistungserbringer zur Basisspezifikation“ beschriebenen Prozess im Krankenhaus:

- Die Struktur der Exportdateien (SOLLBASIS_<RL>_<Erfassungsjahr>.txt und SOLLMODUL_<RL>_<Erfassungsjahr>.txt) ist gleich.
- Die Exportdateien besitzen die gleichen Exportfelder (identische Feldnamen) wie die Exportdateien der Spezifikation für QS-Filter-Software.

Der wesentliche Unterschied in der Erzeugung der Exportdateien in den Krankenhäusern und bei der Landesebene besteht darin, dass die Exportdateien an die Bundesebene möglichst die Datensätze von allen Krankenhäusern des Zuständigkeitsbereichs umfassen sollen. Darüber hinaus ist aber die Pseudonymisierung der Leistungserbringer zu beachten.



Achtung **Struktur zur Übermittlung der Exportdateien**

Die Struktur zur Übermittlung der Exportdateien (SOLLBASIS_<RL>_<Erfassungsjahr>.txt sowie SOLLMODUL_<RL>_<Erfassungsjahr>.txt) an die Bundesebene sind der Spezifikationsdatenbank für QS-Filter-Software zu entnehmen.

Nähere Angaben zur Pseudonymisierung der *IKNRKH* sind im Abschnitt A 1.3 zu finden.

3.1.2 Erzeugung durch die DAS-SV

Die Struktur der XML-Datei (header- und body-Bereich) ist identisch zur Struktur der QS-Dokumentation (Abschnitt 4.4). Auf folgende Besonderheiten in der Sollstatistik ist zu achten:

- Alle Datensätze (Sollzahlen) sind in eine XML-Datei zu exportieren und der BAS zu übermitteln. In das Header-Element `data_flow` ist „Sollstatistik“ einzutragen.
- PID-Daten `<patient>` sind nicht Bestandteil des Datencontainers.
- In das Element `<care_provider>` ist das LE-Pseudonym einzutragen `<pseudonym>`.
- Die Sollzahlen sind wie ein QS-Modul zu betrachten und in das Element `<qs_data>` zu integrieren.
- Für jedes LE-Pseudonym ist ein weiteres `<data_container>`-Element zu erstellen.
- Der Datentyp des `<qs_data>`-Elements ist mit folgender Ausprägung (Datentyp) zu versehen `soll_sv_type`.
- In das Element `DATENSAETZE_MODUL` ist die Soll-Zahl für das Modul `PCI_SV` einzutragen.

Beispiel:

```
<qs_data module="PCI_SV" xsi:type="soll_sv_type">
<B>
<felder>
<DATENSAETZE_MODUL V="73" />
</felder>
</B>
</qs_data>
```

Die QS-Filter-Datenbank beinhaltet analog hierzu die Definitionen der Felder (Datensatz gemäß Tabelle `Ds: SOLL_Qesue_SV`). Die einzelnen Exportfelder sind in der Tabelle `TdsField` definiert und können über das Merkmal `fktDs` gefiltert werden. Die zugehörigen Module sind der Abfrage `SOLL_Qesue_SV` zu entnehmen.

3.1.3 Erzeugung durch die KVen

Für kollektivvertraglich erbrachte Leistungen niedergelassener Praxen/MVZ übernehmen die KVen die Ermittlung der Sollzahlen auf Basis der Abrechnungsdaten und übermitteln das Ergebnis dieser Prüfung an die Bundesauswertungsstelle. Dafür sind die Auslösekriterien (Selektionskriterien) aus der QS-Filter-Datenbank anzuwenden. Details dazu befinden sich in Abschnitt B 1.4.

Die Struktur der XML-Datei (header- und body-Bereich) ist identisch zur Struktur der QS-Dokumentation (Abschnitt 4.4).

Auf folgende Besonderheiten in der Sollstatistik ist zu achten:

- Alle Datensätze (Sollzahlen) sind in eine XML-Datei zu exportieren und der BAS zu übermitteln.
- In das Header-Element `<data_flow>` ist „Sollstatistik“ einzutragen.
- In das Header-Element `<specification>` (`root/header/document/software/specification/@V`) ist die Versionskennung der zu Grunde liegenden Spezifikation einzutragen. Hierüber erfolgt die Zuordnung zum Erfassungsjahr und somit der Abgleich mit den IST-Zahlen der übermittelten Datensätze. Beispielsweise ist für die Übermittlung der Sollzahlen betreffend das Erfassungsjahr 2019 eine gültige 2019er Versionskennung anzugeben, selbst wenn die Übermittlung erst im Februar 2020 erfolgt.
- PID-Daten `<patient>` sind nicht Bestandteil des Datencontainers.
- In das Element `<care_provider>` ist das LE-Pseudonym einzutragen `<pseudonym>`.
- Die Sollzahlen sind wie ein QS-Modul zu betrachten und in das Element `<qs_data>` zu integrieren.
- Für jedes LE-Pseudonym ist ein weiteres `<data_container>`-Element zu erstellen.
- Der Datentyp des `<qs_data>`-Elements ist mit folgender Ausprägung (Datentyp) zu versehen `soll_kv_type`.
- Jedes `<case>`-Element muss eine transaktionsweit eindeutige Vorgangsnummer erhalten (`case/case_admin/id/@V`), d. h. gleiche Vorgangsnummern in unterschiedlichen `<data_container>`-Elementen sind nicht zulässig.
- In das Element `DATENSAETZE_MODUL` ist die Sollzahl für das Modul `PCI_KV` einzutragen.

Beispiel:

```
<qs_data module="PCI_KV" xsi:type="soll_kv_type">
<B>
<felder>
<DATENSAETZE_MODUL V="73"/>
</felder>
</B>
</qs_data>
```

Die QS-Filter-Datenbank beinhaltet analog hierzu die Definitionen der Felder (Datensatz gemäß Tabelle `Ds: SOLL_Qesue_KV`). Die einzelnen Exportfelder sind in der Tabelle `TdsField` definiert und können über das Merkmal `fktDs` gefiltert werden. Die zugehörigen Module sind der Abfrage `SOLL_Qesue_KV` zu entnehmen.

Für das neue Modul `CHE_KV` wird die Sollstatistik in elektronischer Form im csv-Format erstellt und an die jeweilige Datenannahmestelle übermittelt. Die Sollstatistik muss daher getrennt von der Sollstatistik für `PCI_KV` exportiert werden.

Zur Kontrolle der Vollzähligkeit ist die Anzahl der dokumentierten Datensätze (Ist-Zahlen) der Leistungserbringer auf der Basis des Statistik-Elements im Datenflussprotokoll zu ermitteln, das

bei den einzelnen Datenlieferungen von der BAS an die DAS zurückgesendet wird. Die Gesamtzahl der dokumentierten Datensätze (Ist-Zahlen) für den einzelnen Leistungserbringer ergibt sich dabei aus der Summe von <create> abzüglich <delete> für alle <cases>-Elemente der verschiedenen Lieferungen im Erfassungszeitraum:

```
- <statistic>
  - <sent count="31">
    <create V="0" />
    <update V="31" />
    <delete V="0" />
    <corrupted V="0" />
  </sent>
  - <processed count="31">
    <create V="31" />
    <update V="0" />
    <delete V="0" />
    <corrupted V="0" />
  </processed>
</statistic>
</cases>
```

Abbildung 3: XML-Elemente für die Ermittlung von Fallzahlen

Zur Verdeutlichung der Zählweise übermittelter Datensätze für die Ermittlung der IST-Zahlen sei ausgehend von folgender Tabelle folgende relevante Aspekte festgehalten.

Tabelle 10: Sollstatistik - Quantifizierung übermittelter Vorgänge - Umsetzungsbeispiele

| Vorgang case_id | Version | action | case_status | result_ action | Zählung | gültige Version |
|--------------------|---------|--------|-------------|-------------------|---------|--------------------|
| 1 | 1 | create | OK | create | ja | 2 |
| 1 | 2 | update | OK | update | | |
| 2 | 1 | create | OK | create | ja | 2 |
| 2 | 2 | update | OK | update | | |
| 2 | 3 | update | ERROR | corrupted | | |
| 3 | 1 | create | OK | create | nein | |
| 3 | 2 | update | OK | delete | | |
| 4 | 1 | create | ERROR | corrupted | Nein | |
| 5 | 2 | update | OK | create | ja | 2 |

Bei der Verarbeitung von Datenflussprotokollen ist zu beachten, dass in einer Transaktion, welche auf Dokumentenebene mit dem Status ERROR klassifiziert wurde, der Status jedes einzel-

nen in diesem Dokument enthaltenen Vorgangs (<case>-Elemente) innerhalb dieser Transaktion ebenfalls als `ERROR` und somit als `corrupted` anzusehen ist. Wenn eine XML-Datei mit QS-Daten schemakonform die BAS erreicht, jedoch aus technisch administrativen Gründen keiner inhaltlichen Verarbeitung zugeführt werden kann, so wird eine solche XML-Datei mit dem Dokumentenstatus `ERROR` unter Wegfall von QS- und ggf. PID-Daten in ein zugehöriges Datenflussprotokoll mit dem Dokumentenstatus `ERROR` überführt, ohne dass hierbei der Status jedes einzelnen Vorgangs verändert wird. Näheres über diesen Zusammenhang ist insbesondere im Absatz „Beziehungen Vorgangsebene/Dokumentenebene“ im Abschnitt 5.2.2 Die Rückprotokollierung beschrieben.



3.2 Pseudonymisierung

Die leistungserbringeridentifizierenden Daten sind für Verfahren nach Qesü-RL und DeQS-RL analog zum Prozess der QS-Dokumentation zu pseudonymisieren (Abschnitt A 1.3).

LQS/LKG Besonderheit

Da die Sollstatistik für Verfahren gemäß QSKH-RL nicht nur Daten zu direkten, sondern auch zu indirekten Verfahren enthält, werden diese nicht direkt vom Leistungserbringer an die Bundesauswertungsstelle geschickt, sondern zunächst von der Datenannahmestelle gesammelt und anschließend als Gesamtpaket an die Bundesauswertungsstelle gesendet. Bei den Daten zu den Modulen gemäß QSKH-RL ist ab dem Erfassungsjahr 2019 auf die LE-Pseudonymisierung zu verzichten, da diese durch die BAS vorgenommen wird.

3.3 Datenprüfung

3.3.1 Prüfung durch die LQS/LKG

Unplausible oder unvollständige Sollstatistiken werden von den Sollstatistik-Datenannahmestellen als nicht geliefert angesehen. Für nicht akzeptierte Sollstatistiken wird ein Fehlerprotokoll erstellt und per E-Mail an das Krankenhaus übermittelt.

Die inhaltliche Prüfung des Teildatensatzes SOLLBASIS umfasst folgende Bedingungen, die erfüllt sein müssen:

1. Felder `IKNRKH` und `STANDORT`

Der Inhalt dieser beiden Felder muss im Dateinamen der verschlüsselten Datei (`SOLL_<Erfassungsjahr>_<IKNRKH>_<STANDORT>...`) enthalten sein. Geprüft werden kann auch anhand der in jeder Landesgeschäftsstelle vorliegenden Liste der `IKNR` (ggf. zusätzlich der Standortnummer), ob die Sollstatistikmeldung von einem Krankenhaus aus dem jeweiligen Zuständigkeitsbereich stammt.

2. Feld `SOLLJAHR`

An dieser Stelle ist immer nur die Angabe des jeweiligen Erfassungsjahres richtig.

`SOLLJAHR = 2019`

3. Um die Jahreszuordnung unter Berücksichtigung von möglichen Fallzusammenführungen eindeutig abzugrenzen, soll die Sollstatistik frühestens 30 Tage nach dem 31.12.2019 erstellt werden, da nach den Vorgaben der QS nur Fälle aus 2019 zu erfassen sind, die spätestens

am 31.12.2019 entlassen werden. Für das Feld DOKABSCHLDAT wird formal einheitlich daher ein Datum ab dem 01.02.2020 erwartet.

DOKABSCHLDAT > 31.12.2019

4. Feld FREIGABEDATUM

Das Datum der Freigabe kann nicht vor dem Datum des Dokumentationsabschlusses liegen.

FREIGABEDATUM >= DOKABSCHLDAT

5. Nur QSKH: Es können maximal alle geprüften Fälle als Fälle für die Risikostatistik relevant sein.

Summe (FAELLE_GEPRUEFT+FAELLE_GEPRUEFT_VJ) >=FAELLE_GEPRUEFT_RS

6. Nur QSKH: Es können maximal alle geprüften Fälle als nicht dokumentationspflichtig bezeichnet werden (sog. Nullmeldung).

Summe(FAELLE_GEPRUEFT+FAELLE_GEPRUEFT_VJ) >= FAELLE_NDOKPFLICHT

7. Nur QSKH: Da ein dokumentationspflichtiger Fall mindestens einen Datensatz ausgelöst haben muss, aber auch mehr als einen haben kann - darf die Zahl der DATENSAETZE_B nicht niedriger sein als die Zahl der dokumentationspflichtigen Fälle. Der Unterschied zwischen Fälle und Datensätze ist dabei zu beachten.

Summe (FAELLE_GEPRUEFT+FAELLE_GEPRUEFT_VJ)
minus FAELLE_NDOKPFLICHT) <= DATENSAETZE_B

Die inhaltliche Prüfung des Teildatensatzes SOLLMODUL umfasst folgende Bedingungen:

1. Felder IKNRKH und STANDORT

Da die Prüfung des Inhalts der beiden Felder bei SOLLBASIS beschrieben ist, muss hier nur noch überprüft werden, ob die Inhalte in SOLLBASIS mit denjenigen in jeder Zeile von SOLLMODUL übereinstimmen.

SOLLMODUL.IKNRKH = SOLLBASIS.IKNRKH
SOLLMODUL.STANDORT = SOLLBASIS.STANDORT

2. Feld DOKVERPFLICHT

QSKH: Unter Berücksichtigung von DOKVERPFLICHT= [B] sind 32 Zeilen mit Moduleinträgen erforderlich. Außerdem sind mit DOKVERPFLICHT= [Z] weitere 10 Zeilen erforderlich, sodass für alle bundesverpflichtenden Angaben insgesamt 42 Zeilen verbindlich erwartet werden.

Qesü und DeQS: Unter Berücksichtigung von DOKVERPFLICHT= [B] sind 5 Zeilen mit einem Moduleintrag erforderlich.

3. Feld AUFNJAHR

Dieser Wert muss das aktuelle Erfassungsjahr oder das jeweilige Vorjahr sein.

AUFNJAHR IN (2018 ; 2019)

4. Felder MODUL und DOKVERPFLICHT und AUFNJAHR

Eine Kombination der Felder MODUL-DOKVERPFLICHT-AUFNJAHR darf nicht doppelt vorkommen.

Beispiel (QSKH):

Die Sollstatistik darf keine zwei Datensätze für das Modul LTX mit DOKVERPFLICHT = B und AUFNJAHR = 2019 enthalten.

5. nur QSKH: Feld DATENSAETZE_MODUL

Gemäß QS-Filter stellen die Datensätze mit DOKVERPFLICHT=Z eine Untermenge der gezählten Datensätze des jeweiligen Moduls dar. Es kann daher die Prüfung der Gesamtzahl der angegebenen Datensätze in allen Modulen nicht durch eine einfache Addition erfolgen, da hierbei die o.a. Zählmodule doppelt gezählt würden. Es gilt also formal, dass die Summe der auf Bundesebene verpflichtend zu dokumentierenden Datensätze (B) nur aus den Einträgen im Feld DATENSAETZE_MODUL in SOLLMODUL mit der DOKVERPFLICHT=B ermittelt werden. Nur auf dieser Basis kann die resultierende Zahl auf Gleichheit gegen den Eintrag der Anzahl der Datensätze B in SOLLBASIS geprüft werden. Dieses Vorgehen gilt analog für Datensätze mit DOKVERPFLICHT=L.

SUMME (DATENSAETZE_MODUL mit DOKVERPFLICHT=[B])
= SOLLBASIS.DATENSAETZE_B

6. Felder DS_DRG und DS_IV und DS_DMP und DS_SONST je Zeile in SOLLMODUL

Hier ist die Situation komplex. Die Abrechnungsvorschriften für die Entgeltberechnung bzw. die Vorgaben des QS-Filters für die Zählung der Entgeltarten führen dazu, dass auf der Ebene eines Behandlungsfalles eine oder mehrere Zählung(en) in den Feldern DS_DRG, DS_IV bzw. DS_DMP möglich sind. Da es sich bei einem DMP_Entgelt um ein Zusatzentgelt handelt, kann diese Angabe auf Fallebene allein nicht vorkommen. Wenn der QS-Filter eine Zählung in DS_SONST vornimmt, ist jedoch in keinem anderen der drei Felder ein Zähleintrag zulässig, auch nicht DS_DMP, obwohl dieses rein formal als Zusatzentgelt durchaus mit einer Abrechnung des Pflegesatzes kombiniert sein könnte. Folgende Kombinationen auf Einzelfallebene sind vorstellbar:

Tabelle 11: Berücksichtigung verschiedener Konstellationen von Entgeltarten in der Sollstatistik

| Konstellation | DS_DRG | DS_IV | DS_DMP | DS_SONST | Erläuterung |
|---------------|--------|-------|--------|----------|---|
| 1 | 1 | 0 | 0 | 0 | Ein Fall mit reiner DRG-Abrechnung |
| 2 | 0 | 1 | 0 | 0 | Ein Fall mit reiner Abrechnung als integrierte Versorgung |
| 3 | 0 | 0 | 1 | 0 | Ein Fall mit Abrechnung DMP und einer anderen Entgeltart |
| 4 | 0 | 0 | 0 | 1 | Ein Fall mit sonstiger Abrechnung (z. B. Pflegesatz in der Psychiatrie) Technisch: NICHT (DS_DRG und/oder DS_IV und/oder DS_DMP) |

| Konstellation | DS_DRG | DS_IV | DS_DMP | DS_SONST | Erläuterung |
|---------------|--------|-------|--------|----------|--|
| 5 | 1 | 1 | 0 | 0 | Fall mit Kombination der Abrechnung von DRG und integrierter Versorgung |
| 6 | 1 | 0 | 1 | 0 | Fall mit Kombination der Abrechnung von DRG und DMP |
| 7 | 0 | 1 | 1 | 0 | Fall mit Kombination der Abrechnung von integrierter Versorgung und DMP |
| 8 | 1 | 1 | 1 | 0 | Fall mit Kombination der Abrechnung von DRG und integrierte Versorgung und DMP |

```
DATENSAETZE_MODUL <= SUMME ( DS_DRG , DS_IV , DS_DMP , DS_SONST )
```

Nur bei Auftreten bestimmter Konstellationen kann eine konkrete Prüfung der Plausibilität erfolgen. Abfragetechnisch lassen sich diese Konstellationen so formulieren: in mindestens 2 von 3 Feldern (DS_DRG, DS_IV, DS_DMP) steht eine [0]. Dann resultiert daraus, dass die Summe über alle vier Felder gleich sein muss mit der Angabe der Anzahl der Datensätze im Modul.

```
WENN [0] zweimal in (DS_DRG,DS_IV, DS_DMP)
DANN DATENSAETZE_MODUL= SUMME ( DS_DRG,DS_IV,DS_DMP,DS_SONST)
```

7. Felder QUARTAL1 bis QUARTAL4 je Zeile in SOLLMODUL

Alle Fälle lassen sich auf Basis des Entlassungsdatums genau einem Quartal zuordnen, sodass die Summe der einzelnen Quartale der Gesamtzahl dokumentationspflichtiger Datensätze des jeweiligen Moduls entsprechen muss.

```
DATENSAETZE_MODUL = SUMME(QUARTAL1, QUARTAL2,
QUARTAL3, QUARTAL4)
```

Eine Abweichung von dieser Regel führt nur zu einer Ablehnung der Sollstatistik, weil die Unstimmigkeit auf eine fehlerhafte Abgrenzung der Quartale bzw. des Erfassungsjahres hindeutet.

8. nur QSKH: Feld DS_GKV

Beim Feld DS_GKV ist durch den Hinweis im Attribut `Feld.hinweis` in Verbindung mit dem Attribut `Modul.pid` der Tabelle `Modul` sowie durch die Mustervorlagen zur Sollstatistik eindeutig vorgegeben, bei welchen Modulen die Angabe vorzusehen ist. Für die Sollstatistik gilt daher:

```
WENN MODUL in
('09/1', '09/2', '09/3', '09/4', '09/5', '09/6', '16/1', 'NEO',
'HEP', 'HEP_WE', 'HEP_IMP', 'KEP', 'KEP_WE', 'KEP_IMP')
DANN DS_GKV <> LEER.
```

Ist bei den einzelnen Modulen eine Angabe in diesem Feld erforderlich, so kann die Zahl maximal den Wert von DATENSAETZE_MODUL erreichen.

```
WENN DS_GKV <> LEER DANN DS_GKV <= DATENSAETZE_MODUL
```

9. nur QSKH: Zählleistungsbereiche

Die Zählleistungsbereiche (DOKVERPFLICHT=[Z]) erfassen spezifische Teilmengen der korrespondierenden Module (DOKVERPFLICHT=[B]). Bei den Modulen HEP, KEP und HTXM gilt dabei, dass die Summe der zugehörigen Zählleistungsbereiche größer als die Anzahl an Datensätzen [B] sein kann, wenn während eines Aufenthalts Eingriffe in mehreren Zählleistungsbereichen vorgenommen wurden. Für diese Module gelten daher folgende Regeln:

```
WENN MODUL in ('HEP') DANN
DATENSAETZE_MODUL <= SUMME (DATENSAETZE_MODUL('HEP_IMP')
+ DATENSAETZE_MODUL ('HEP_WE'))
```

```
WENN MODUL in ('KEP') DANN
DATENSAETZE_MODUL <= SUMME (DATENSAETZE_MODUL('KEP_IMP')
+ DATENSAETZE_MODUL ('KEP_WE'))
```

```
WENN MODUL in ('HTXM') DANN
DATENSAETZE_MODUL <= SUMME (DATENSAETZE_MODUL('HTXM_TX')
+ DATENSAETZE_MODUL ('HTXM_MKU'))
```

Bei HTXM kann diese Prüfung, jeweils bezogen auf das Feld AUFNJAHR, angewendet werden. Für die beiden Zählleistungsbereiche zur Herzchirurgie sind entsprechende Regeln nicht anwendbar, da diese insgesamt nur einen Teil der Eingriffe erfassen.



Hinweis zu landesverpflichtenden Verfahren

Da die Vorlage zur Sollstatistik für QS-Verfahren gemäß QSKH-RL zusätzlich Module verschiedener Bundesländer enthält, können diese auch in den Daten übermittelt werden. Die Daten der nicht zutreffenden Bundesländer sind in diesem Fall mit [0] zu befüllen.

3.3.2 Prüfung durch die KV und DAS-SV

Die Datei-Prüfungen sind identisch zu den Prüfungen der QS-Daten (Abschnitt B 2.8.3). Die Prüfung vor dem Versand beschränkt sich allerdings auf die administrativen Daten (Elemente des Datencontainers) und auf die Schemavalidierung. Folgende Unterschiede sind allerdings zu berücksichtigen:

- Jeder Fehler in der XML-Datei führt zur Ablehnung des gesamten Dokuments.
- Stornierung oder Lieferung einzelner korrigierten Datensätze ist bei der Sollstatistik nicht erlaubt.
- In den Bundesdatenpool wird immer nur die letzte von der DAS übermittelte Datei (Transaktion) übernommen.

3.4 Verschlüsselung

3.4.1 Transportverschlüsselung durch die LQS/LKG

Die Verschlüsselung der Sollstatistik-Dateien erfolgt per PGP-Verfahren mit dem öffentlichen Schlüssel der Bundesauswertungsstelle.

3.4.2 XML-Verschlüsselung durch die KV und DAS-SV

Vor der Weiterleitung der erstellten XML-Datei sind zum einen die LE-Pseudonyme (/care_provider) und zum anderen die QS-Daten (/qs_data) mit dem öffentlichen Schlüssel der BAS zu verschlüsseln. Um zwei unterschiedliche Verarbeitungsprozesse bei den Datenannahmestellen und der Bundesauswertungsstelle zu vermeiden, wird diese Verschlüsselung auch bei der Sollstatistik eingehalten, obwohl diese Daten von der Datenannahmestelle direkt an die Bundesauswertungsstelle geliefert werden.



Hinweis

Ob die XML-Verschlüsselung erfolgreich durchgeführt wurde, kann auch mit den entsprechenden Schemata für die Ausgangskontrolle geprüft werden.

3.5 Datenübermittlung

Im folgenden Abschnitt werden die abschließend zu unternehmenden Arbeitsschritte (Dateibenennung (LQS/LKG), Ausgangsprüfung und Datenweiterleitung) beschrieben.

3.5.1 Datenweiterleitung bei LQS/LKG

Die elektronischen Sollstatistiken sind als komprimierter und verschlüsselter E-Mail-Anhang **spätestens bis zum**

- **15.03. des Folgejahres** bei der QSKH-Sollstatistik
- **31.03. des Folgejahres** bei der Qesü/DeQS-Sollstatistik
- an die Bundesebene zu übermitteln – eine frühzeitige Erstlieferung mit den fristgerecht bis zum 15.02. eingegangenen Daten ist erwünscht.

Die beiden Exportdateien SOLLBASIS_<RL>_<Erfassungsjahr>.TXT und SOLLMODUL_<RL><Erfassungsjahr>.TXT werden mithilfe eines Archivierungsprogramms komprimiert und unter folgendem Namen gespeichert:

SOLL_<Erfassungsjahr>_<Länderkürzel>_<DAS>_<RL>.zip

Beispiel: SOLL_2019_BB_LQS_QSKH.zip

Als DAS sind LQS (Landesgeschäftsstelle) und LKG (Landeskrankenhausgesellschaft) zulässig.

Die Verschlüsselung des ZIP-Archivs erfolgt per PGP-Verfahren mit dem öffentlichen Schlüssel der Bundesauswertungsstelle. Die komprimierte und verschlüsselte Exportdatei hat den Dateinamen:

SOLL_<Erfassungsjahr>_<Länderkürzel>_<DAS>_<RL>.zip.gpg

Beispiel: SOLL_2019_BB_LQS_<RL>.zip.gpg

3.5.2 Datenweiterleitung bei KV/DAS-SV

Für die Vollzähligkeitsprüfung übermittelt die KV/DAS-SV die ermittelte Anzahl der verfahrensspezifisch (z. B. PCI) abgerechneten Fälle (Soll) **spätestens bis zum 31.03. des Folgejahres** an die Bundesauswertungsstelle. Vor der Weiterleitung der Daten an die nächste Stelle ist eine Ausgangskontrolle durchzuführen. Zu diesem Zweck wird das entsprechende XML-Schema (interface_DAS_BAS_soll.xsd, siehe Abschnitt B 4.2) als Bestandteil der Spezifikation bereitgestellt.

Sollen die Daten als XML-Datei und nicht als Stream an die (BAS) weitergeleitet werden, muss die Exportdatei nach einem festgelegten Schema benannt werden (Abschnitt A „Benennung der Exportdateien“).

Beispiel:

SOLL-Daten einer DAS

```
47d16341-9e27-4e75-a27e-b791fbbd2dc8_s_DAS.xml
```

Für die Übermittlung der Daten über nicht gesicherte Übertragungswege (E-Mail) muss die Exportdatei in eine Transaktionsdatei²² umgewandelt werden.

Die DAS erhält bei der Registrierung von der BAS einen Verschlüsselungskode, der bei der Erzeugung der Archivdatei verwendet werden muss. Die Transaktionsdatei wird nach einem vordefinierten Schema benannt (Abschnitt A „Übermittlung der Daten im Datenfluss“).

Beispiel:

Transaktionsdatei

```
T-BU1234a-2018_01_19_160945.zip.aes
```

```
T-BU1234a-2018_01_23_114113_045.zip.aes (millisekundengenau)
```

3.6 Rückprotokollierung

3.6.1 Rückprotokollierung durch LQS/LKG

Für die Rückprotokollierung zur Sollstatistik ist in der Spezifikation bisher kein standardisiertes Format vorgegeben. Die Landesstellen werden die Bestätigung der erfolgreichen Datenlieferung bzw. von Fehlermeldungen in geeigneter Form sicherstellen.

²² Verpackte und symmetrisch verschlüsselte AES-Datei.

3.6.2 Rückprotokollierung an KV/DAS-SV

Das Datenflussprotokoll, das dieselbe Struktur wie bei der QS-Dokumentation besitzt, wird durch die Bundesauswertungsstelle erstellt, transportverschlüsselt und an die von der BAS registrierten E-Mail-Adressen gesendet (AES-Datei). Da das Datenflussprotokoll symmetrisch verschlüsselt ist, muss die DAS die AES-Datei symmetrisch mit dem in der Registrierung vergebenen Passwort entschlüsseln (Abschnitt B 5.3.2).

Auf folgende Unterschiede zur Protokollierung bei der QS-Dokumentation ist besonders zu achten:

- Die harten Fehler in den Datensätzen führen automatisch zur Ablehnung des gesamten Dokuments.
- Datenlieferungen mit `status_document = "ERROR"` sind dann zu korrigieren und mit einer neuen GUID erneut an die BAS zu übermitteln.
- Falls mehrere Sollstatistiken an die BAS gesendet werden, gilt stets die letzte Lieferung.

3.7 Konformitätserklärung

Die Konformitätsbescheinigung der Krankenhäuser und selektivvertraglich erbrachter Leistungen wird zusammen mit der Sollstatistik an die Datenannahmestellen übermittelt. Dabei handelt es sich um ein von der BAS spezifiziertes Formular, welches der Leistungserbringer mit den Sollzahlen füllt und deren Richtigkeit von der Geschäftsführung/dem Praxisinhaber durch Unterschrift bestätigt werden. Die Konformitätserklärung wird von den zuständigen Datenannahmestellen angenommen.

Die Konformitätserklärungen werden nicht an die BAS weitergeleitet, sondern werden in der Datenannahmestelle archiviert. Die Mustervorlagen (Formulare) zum Erfassungsjahr sind unter www.iqtig.org veröffentlicht.

4 Auswertungen

In diesem Abschnitt werden Regelungen zur Übermittlung der Auswertungen bzw. Berichte von der BAS an die DAS bzw. die LE, ggf. über die DAS beschrieben.

Entsprechend den Vorgaben der Richtlinien des G-BA (insbesondere Qesü-RL, DeQS-RL, plan. QI-RL, QSD-RL, QKSH-RL) werden Jahres- und Quartalsberichte (Rückmeldeberichte, Zwischenberichte, Länderauswertungen, etc.) von der BAS einerseits für die Landesstellen sowie andererseits für die Leistungserbringer erstellt.

Übertragungsweg

Berichte werden grundsätzlich über das Extranet des IQTIG bereitgestellt und können nach Benachrichtigung vom Adressaten aktiv abgeholt werden (Pull-Verfahren). Für die Nutzung des Extranets ist eine Registrierung einschließlich einer Vertraulichkeitserklärung erforderlich. Unter Verwendung des Extranets können zielgerichtet und empfängerspezifisch Pakete (respektive Dokumente) seitens des IQTIG zur Abholung bereitgestellt werden. Diese Bereitstellung erfolgt stets in Form Einzeldateien und/oder Auslieferungseinheiten.

Bei Berichten zu direkten QS-Verfahren erfolgt die Abholung der leistungserbringerspezifischen Einzeldateien und/oder Auslieferungseinheiten vom IQTIG durch die Leistungserbringer selbst. In allen anderen Fällen erfolgt die Abholung der Auslieferungseinheiten durch die jeweils zuständigen Datenannahmestellen auf Landesebene.

Auslieferungseinheiten

Eine einzelne Auslieferungseinheit wird für einen konkreten Adressaten erstellt und beinhaltet in der Regel mehrere Dateien, ggf. auch mit unterschiedlichen Dateitypen (zip, pdf, csv). Auslieferungseinheiten können ineinander verschachtelt sein, so dass eine Auslieferungseinheit für eine Datenannahmestelle auf Landesebene sämtliche leistungserbringerspezifischen Auslieferungseinheiten ihres Zuständigkeitsbereiches beinhalten kann.

Passwort (Feedbackkey) zur Verschlüsselung von Auslieferungseinheiten (Qesü-RL und DeQS-RL)

Auslieferungseinheiten werden stets in Form von Archivdateien im ZIP-Format bereitgestellt. Um der Maßgabe des § 18 Abs. 1 Satz 2 Teil 1 Qesü-RL sowie der DeQS-RL zu entsprechen, wird das ZIP-Archiv einer leistungserbringerspezifischen Auslieferungseinheit, welche einen Zwischen- bzw. Rückmeldebericht für Verfahren gem. Qesü-RL und DeQS-RL beinhaltet, verschlüsselt. Dafür wird die Verschlüsselung WinZIP AES 256 Bit (Spezifikation siehe: http://www.winzip.de/aes_info.htm) unter Verwendung eines auf LE-Ebene gewählten Passwortes (Feedbackkey) genutzt.

Der für die Verschlüsselung von Zwischen- bzw. Rückmeldeberichten für Verfahren gem. Qesü-RL und DeQS-RL verwendete Feedbackkey kann von den Leistungserbringern selbst gewählt oder durch die jeweilige QS-Dokumentationssoftware unterstützend vergeben und verwaltet werden. Die Übermittlung des Feedbackkeys erfolgt grundsätzlich im Rahmen einer jeden Da-

tenübermittlung von QS-Daten. Das Feedbackkey wird im Header (s. `Feedback_key`, Abschnitt B 4.4.3) der QS-Datenlieferungen platziert und beim Leistungserbringer mit dem öffentlichen Schlüssel der BAS verschlüsselt.

Für den Fall, dass ein Feedbackkey nicht übermittelt, verlegt oder anderweitig nicht zugänglich ist, besteht für Leistungserbringer die Möglichkeit, in einem durch das IQTIG vorgegebenen Zeitraum, einen Feedbackkey für die Verschlüsselung der Rückmelde- bzw. Zwischenberichte unter folgender Website an die BAS zu übermitteln:

<https://anmeldung.iqtig.org/feedbackkey.php>

Liegt vom jeweiligen Leistungserbringer kein zugehöriger Feedbackkey vor, darf die entsprechende leistungserbringerspezifische Auslieferungseinheit nicht unverschlüsselt bereitgestellt werden.



Hinweis

Bei der Übermittlung von Daten vom LE muss sichergestellt werden, dass in jeder Datenlieferung ein (mit dem öffentlichen Schlüssel der BAS XML-) verschlüsselter Feedbackkey enthalten ist. Für die **Verschlüsselung der Berichte** der Leistungserbringer wird **stets der zuletzt an die BAS übermittelte Feedbackkey** verwendet. Sollte von der Möglichkeit der gesonderten Übermittlung eines Feedbackkeys für Rückmelde- bzw. Zwischenberichte Gebrauch gemacht werden, gilt diese Regelung ebenfalls.

Die folgende Tabelle enthält eine exemplarische Auflistung von Softwareprodukten, mithilfe derer die kennwortgeschützten Archive mit der 256-Bit AES-Verschlüsselung unter Verwendung des Feedbackkeys geöffnet und deren Inhalte weiterverwendet werden können.

Tabelle 12: Softwareprodukte zum Entpacken von zip-Dateien mit 256-Bit AES-Verschlüsselung

| Betriebssystem | Software | Weblink | Anmerkung |
|-----------------------------|----------------|---|------------------------------|
| MS Windows / Mac OS / Linux | 7-zip | http://www.7-zip.de/download.html | Lizenz: GNU LGPL |
| Mac OS | The Unarchiver | https://theunarchiver.com/ | |
| MS Windows / Mac OS | Zipeg | https://zipeg.de.softonic.com/ | New BSD License |
| MS Windows / Linux | PeaZip | http://www.peazip.org/ | read-only Lizenz: GNU FDL |
| MS Windows | Bandizip | https://www.bandisoft.com/bandizip/de/ | |

Namenskonventionen

Für die eindeutige Kennzeichnung der Auslieferungseinheiten sowie deren beinhaltete Dateien werden diese von der BAS nach den folgenden Konventionen benannt:

Berichtsdokumente

<Richtlinie>_<Auswertungsmodul>_<Erfassungsjahr[-Quartal]>_<Produkttyp>_<Auswertungseinheit-[Standort]>_<Erzeugungsdatum>.[Dateityp]

Auslieferungseinheiten

<Richtlinie>_<Auswertungstyp>[-<Auswertungsmodul>]_<Erfassungsjahr[-Quartal]>_<Produkttyp>_<Empfängererkennung>_<Erzeugungsdatum>.zip

Tabelle 13: Dateinamensbestandteile von Auslieferungseinheiten und deren beinhaltete Dateien

| Dateinamensbestandteil | Ausprägung (exemplarisch) | Anmerkung |
|---------------------------|---------------------------|--|
| Richtlinie | QESUE | Berichte gem. Qesü-RL |
| | DEQS | Berichte gem. DeQS-RL |
| | PlanQI | Berichte gem. plan. QI-RL |
| | QSD | Berichte gem. QSD-RL |
| | QSKH | Berichte gem. QSKH-RL |
| Auswertungsmodul/-typ | Quartalsbericht | Falls mehrere Verfahren, z. B. für Berichte gem. plan. QI-RL |
| | Quartalsbericht-PCI | Adressat DAS, z. B. Zwischenberichte gem. Qesü-RL |
| | Jahresbericht-PCI | Adressat DAS, z. B. Rückmeldeberichte gem. Qesü-RL |
| | PCI | |
| | NWI | |
| | 09n1 | |
| | CHE | |
| Erfassungsjahr [-Quartal] | 2019 | Erfassungsjahr |
| | 2018-Q1 | Erfassungsjahr mit Quartalsangabe |
| Produkttyp | LEAW | Adressat: LE LeistungsErbringer-Auswertung |
| | AV | Auffällige Vorgänge |
| | ERG | ERG ebnislisten |
| | LEAE | Adressat: LE |

| Dateinamensbestandteil | Ausprägung (exemplarisch) | Anmerkung |
|--------------------------------|---------------------------|---|
| | | LeistungsErbringer-Auslieferungseinheit |
| | LSAE | Adressat: DAS LandesStelle-Auslieferungseinheit |
| | LEPMAP | Adressat: DAS LeistungsErbringer-Pseudonym-MAPPING (Langpseudonym, Kurzpseudonym, Bemerkung) |
| | LSAE-D | Adressat: DAS LandesStelle-Auslieferungseinheit-Deltalieferung; vgl. Erläuterung Deltalieferung |
| | AEDELTA | Adressat: DAS Auslieferungseinheiten-DELTA; vgl. Erläuterung Deltalieferung |
| Auswertungseinheit [-Standort] | 260123456 | IKNR bei QS-Verfahren ohne LE-PSN, z. B. direkte Verfahren |
| | 260123456-01 | IKNR mit Standortangabe |
| | TL#KV#A123BCDEFG4H | Kurzpseudonym eines LE im Zuständigkeitsbereich der KV TestLand |
| | TL#LKG#A123BCDEFG4H | Kurzpseudonym eines LE im Zuständigkeitsbereich der LQS/LKG TestLand |
| | TL#LKG#A123BCDEFG4H-01 | Kurzpseudonym mit Standortangabe eines LE im Zuständigkeitsbereich der LQS/LKG TestLand |
| Erzeugungsdatum | 2020-05-16 | Format YYYY-MM-DD |

Dateinamensbestandteile werden durch Underscores/Unterstriche („_“) voneinander getrennt. Innerhalb eines Dateinamensbestandteils erfolgt eine Trennung optionaler Bestandteile durch einen Bindestrich („-“), z. B. TL0815-01 oder 2019-Q2.

Erläuterung Kurzpseudonym (Qesü-RL/DeQS-RL)

Datensätze, die im Rahmen von QS-Verfahren gem. der Qesü-RL und der DeQS-RL übermittelt werden, enthalten Leistungserbringer-Pseudonyme mit der Länge von 354 Zeichen (Langpseu-

donym). Für die Verwendung der Pseudonyme in Berichten und Dateinamen wird eine Kurzversion des Pseudonyms während der Datenannahme im IQTIG erstellt. Die Zuordnung von Lang- und Kurzpsudonymen wird im IQTIG hinterlegt und ist dauerhaft stabil. Darüber hinaus kann das Kurzpsudonym für die Übermittlung eines Feedbackkeys über die Website des IQTIG verwendet werden. In den Auslieferungseinheiten für die Landesstellen wird eine entsprechende Mapping-Tabelle (LEPMAP) bereitgestellt. Diese beinhalten auch Hinweise über ggf. fehlende Feedbackkeys.

Erläuterung Deltalieferung

Im Rahmen der QS-Verfahren gem. Qesü-RL und gem. DeQS-RL werden Leistungserbringer-Auslieferungseinheiten (LEAE) verschlüsselt. Die dafür verwendeten Feedbackkeys werden teilweise verspätet, d.h. nach der initialen Auslieferung beim IQTIG hinterlegt bzw. aktualisiert. In Ausnahmefällen könnten diese für eine Repaktetierung der LEAE für Folgebereitstellungen verwendet werden. Das Delta-Paket erhält dabei nur die neu verschlüsselten LEAE und wird daher in einem gesonderten Paket (LSAE-D) mit einer erläuternden Textdatei (AEDELTA) ausgeliefert. LEAE werden nur in Ausnahmefällen neu verschlüsselt. Wenn zum Zeitpunkt der initialen Auslieferung kein Feedbackkey vorlag, konnte die LEAE ursprünglich nicht verschlüsselt werden. Wurde nach der initialen Auslieferung ein Feedbackkey aktualisiert, wurde die LEAE ursprünglich mit einem inzwischen ungültigen Feedbackkey verschlüsselt. Das Delta-Paket enthält somit ausschließlich LEAE, welche aus den vorgenannten Gründen neu verschlüsselt wurden.

Beispiele Auswertungen für Verfahren gem. plan. QI-RL:

- PlanQI_Quartalsbericht_2019-Q1_LEAE_BW1005-01_2019-05-17.zip
- PlanQI_Quartalsbericht_2019-Q1_LSAE_BW_2019-05-17.zip
- PlanQI_Jahresbericht_2019_LEAE_BW1005-01_2019-01-30.zip
- PlanQI_Jahresbericht_2019_LSAE_BW_2019-01-30.zip

Beispiele Auswertungen für Verfahren gem. Qesü-RL:

- QESUE_Quartalsbericht-PCI_2019-Q1_LEAE_BW#KV#T427YQXYOU0R_2019-09-17.zip
- QESUE_Quartalsbericht-PCI_2019-Q1_LEPMAP_BW-KV_2019-09-17.csv
- QESUE_Quartalsbericht-PCI_2019-Q1_LSAE_BW-KV_2019-09-17.zip

Delta-Lieferung:

- QESUE_Quartalsbericht-PCI_2019-Q1_LSAE-D_BW-LKG_2019-10-10.zip
 - QESUE_Quartalsbericht-PCI_2019-Q1_AEDELTA_BW-LKG_2019-10-10.txt
 - QESUE_Jahresbericht-QSWI_2019_LEAE_BW#LKG#P429YOMYOU0L-00_2020-10-03.zip
 - QESUE_Jahresbericht-QSWI_2019_LEPMAP_BW-LKG_2020-10-03.csv
 - QESUE_Jahresbericht-QSWI_2019_LSAE_BW-LKG_2020-10-03.zip
-

5 Allgemeine Regelungen zur Datenübermittlung

In diesem Kapitel werden allgemeingültige Regelungen für die Datenflüsse, Datenübermittlung, Datenprüfung und Rückprotokollierung beschrieben.

5.1 Datenübermittlung

In diesem Abschnitt werden allgemeine Regelungen in Bezug auf die Datenübertragung, Registrierung und Verschlüsselung beschrieben.

5.1.1 Gesicherte Datenübertragung

Die Datenübertragung darf ausschließlich über gesicherte Übertragungskanäle erfolgen. Im Folgenden werden die verschiedenen Übertragungswege beschrieben.

Übertragungswege

Eine an der Datenübertragung beteiligte Einrichtung kann auf zwei verschiedenen Übertragungswegen Daten entgegennehmen:

- Verschlüsselung und Authentifizierung des Übertragungswegs:
Die Daten werden auf Übertragungswegen versandt, die eine Authentifizierung des Absenders und eine Transportverschlüsselung automatisch implizieren, wie beispielsweise KV-Connect und KV-SafeNet. Eine zusätzliche Registrierung des Absenders ist nicht nötig, da dieser so bereits eindeutig identifiziert werden kann.
- Verschlüsselung und Authentifizierung des Datenpakets:
Die Auswahl des Übertragungswegs ist frei. In der Regel kommen Datenträger und E-Mail-Übertragungen zum Einsatz. Die Transportverschlüsselung und der E-Mail-Versand sind in dieser Spezifikation geregelt. Die Authentifizierung und Verschlüsselung wird über den vorgelagerten Prozess einer Registrierung ermöglicht.

Registrierungen

Registrierung eines Dokumentationssystems (LQS/LKG)

Die Registrierung des Dokumentationssystems eines Leistungserbringers bei der DAS ist Voraussetzung für die Datenübermittlung. Die Registrierung unterstützt die sichere Übertragung von Daten per E-Mail und stellt sicher, dass Vorgangsnummern in Verbindung mit der Registrierungsnummer eindeutig sind.

Registrierung bei einer Datenannahmestelle

Im stationären Bereich bekommen registrierte Dokumentationssysteme von der Datenannahmestelle eine Registrierungsnummer zugewiesen, die folgendermaßen aufgebaut ist:

<Registrierungsnummer> = <Länderkode><Registrierkode>

Jedes Dokumentationssystem des stationären Leistungserbringers ist auf diese Weise bundesweit identifizierbar. Beim Wechsel des Dokumentationssystems eines Leistungserbringers ist zu beachten, dass eine neue Registrierungsnummer zu beantragen ist.



Abbildung 4: Beispiel einer Registrierungsnummer

Die Adressen der einzelnen DAS werden auf der IQTIG-Homepage aufgeführt. Die dort angegebenen Informationen über die Datenservices sind eine Generierung aus den entsprechenden Tabellen in den administrativen Objekten (Abschnitt B 2.8.2).

Die Frage der Registrierung im ambulanten Bereich ist bisher noch nicht abschließend geklärt.

Registrierung bei der Bundesauswertungsstelle (direkte Verfahren)

Für die direkten Verfahren (bundesbezogene Verfahren) muss ein Dokumentationssystem bei der Bundesauswertungsstelle registriert werden.

Die Registrierungsnummer ist analog zur Registrierungsnummer eines Bundeslandes aufgebaut. Der Ländercode wird durch BQ ersetzt. Der Registriercode ist fünfstellig und besteht aus Ziffern und Kleinbuchstaben. Hier gilt wie bei den direkten Verfahren, dass nur Dokumentationssysteme, die bei der Bundesauswertungsstelle registriert sind, Datensätze übermitteln dürfen.

Beispiel:

Registrierungsnummer: bq1011a

Adresse und Ansprechpartner für die Registrierung im Direktverfahren werden auf der Homepage <https://www.iqtig.org/> aufgeführt.

Registrierung von Datenannahmestellen bei der Bundesauswertungsstelle

DAS, die in den indirekten Verfahren Daten an die Vertrauensstelle oder die Bundesauswertungsstelle weiterleiten, müssen bei der Bundesauswertungsstelle registriert sein.

Registrierung von Datenannahmestellen bei der Vertrauensstelle

DAS, die in den indirekten Verfahren Modulen mit PID an die BAS weiterleiten, müssen zudem neben der Registrierung bei der BAS auch bei der Vertrauensstelle registriert sein. Hier ist darauf zu achten, dass die von der VST vergebene Registrierungsnummer der Registriernummer der DAS bei der BAS entspricht. Für die Transportverschlüsselung wird ein anderer geheimer Schlüssel vereinbart.

Registrierung von Softwareanbietern beim IQTIG für Testzwecke

Softwareanbieter, die an Testbetrieben teilnehmen wollen, müssen sich bei den Stellen registrieren, die den Test-Datenservice bereitstellen.

Eindeutige Kennzeichnung der XML-Exportdateien

Jede Exportdatei wird durch eine universell eindeutige ID (GUID) von der QS-Software gekennzeichnet.

Ein Globally Unique Identifier (GUID) ist eine global eindeutige Zahl mit 128 Bit, die eine Implementierung des Universally Unique Identifier Standards (UUID) darstellt.

GUIDs haben das Format XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX, wobei jedes X für ein Zeichen aus dem Hexadezimalsystem steht und damit eine Ziffer 0–9 oder ein Buchstabe A–F sein kann.

Erläuterung zur GUID:

- Die GUID wird im Exportprozess von der QS-Software einer bestimmten Exportdatei zugewiesen.
- Das registrierte Dokumentationssystem ordnet jeder an eine Datenannahmestelle übermittelten Exportdatei eine eindeutige GUID zu.
- Diese GUID wird im XML-Code des Dokuments als ID gesetzt. Sie muss bei dateibasierten Übertragungsverfahren in der Dateibenennung verwendet werden.
- Eine GUID wird durch eine gelungene Transaktion zwischen den jeweils beteiligten Übertragungspartnern verbraucht. Jede Datenannahmestelle muss dafür sorgen, dass ein eindeutig über die GUID identifizierbares Dokument nur einmal denselben Verarbeitungsschritt durchlaufen kann. Anderenfalls ist die Verarbeitung mit einem entsprechenden Fehlerprotokoll abzulehnen.



Achtung

Aufgrund von fehlerhaften, automatisierten Verarbeitungen in der DAS, VST oder BAS kann der Ausnahmezustand entstehen, dass ein bereits verarbeitetes Dokument nebst Rückprotokollierung bis hin zum Leistungserbringer erneut verarbeitet werden muss.

Für den Fall, dass ein erneuter Datenfluss von QS-Daten ab der zuständigen DAS erfolgt, muss auch die Dokumentationssoftware beim Leistungserbringer über die Möglichkeit verfügen, mehr als ein Datenflussprotokoll zu einer GUID aufnehmen können.

Identifizierung von Datensätzen

Die Vorgangsnummer (auch Datensatz-ID oder ID genannt) kennzeichnet in eindeutiger Weise jeden dokumentierten Vorgang eines registrierten Dokumentationssystems.

Im einfachsten Fall könnten die Vorgangsnummern jeweils um 1 inkrementiert werden, wenn ein neuer Datensatz angelegt wird. Wenn während eines stationären Aufenthalts z. B. zwei QS-

Dokumentationen eines Leistungsfalls angelegt werden, so müssen auch unterschiedliche Vorgangsnummern vergeben werden. Die Vorgangsnummer ist daher allein ein Merkmal des QS-Dokumentationssystems, um einen Datensatz innerhalb des registrierten Dokumentationssystems eindeutig identifizieren zu können. Aus diesem Grund wäre es falsch, die Patientenidentifikationsnummer oder die Fallnummer aus dem KIS/AIS zu verwenden bzw. zu pseudonymisieren.

Eine Vorgangsnummer darf keine Rückschlüsse auf Personen ermöglichen. In der Vorgangsnummer darf z. B. nicht das Geburtsdatum enthalten sein.

Die QS-Dokumentationssoftware verwaltet jahresübergreifend die Vorgangsnummern der QS-Dokumentationen. Sie soll dem Leistungserbringer eine Zuordnung der Vorgangsnummern zu seinen internen Fall- oder Patientennummern (vgl. nicht übermitteltes Datenfeld IDNRPAT) ermöglichen. Zum Zweck der Datenvalidierung und der Qualitätsverbesserung muss es dem Leistungserbringer möglich sein, über die Vorgangsnummer Zugang zur Fall- bzw. Patienten-Akte zu bekommen.

Annahme oder Ablehnung unterschiedlicher Versionen eines Datensatzes

Bei der Datenannahmestelle eingehende Datensätze werden anhand der Kombination aus Registrierungsnummer und Vorgangsnummer als ein Vorgang identifiziert. Der für einen bestimmten Vorgang gespeicherte Datensatz kann nur durch eine neuere Version (mit höherer Versionsnummer) überschrieben werden.²³

Unterschiedliche Versionen eines Datensatzes müssen demselben Primärmodul²⁴ zugeordnet sein. Ein Datensatz mit einer Vorgangsnummer aus derselben Registrierung, die bereits unter einem anderen Modul eingeschickt wurde, wird abgelehnt.

Ein Minimaldatensatz darf nur für Primärmodule angelegt werden.

Beispiel:

Kein Minimaldatensatz für Sekundärmodule

Das Modul HTXFU ist ein Sekundärmodul zum Primärmodul HTXM. Statt HTXFU darf also nicht der Minimaldatensatz (MDS) übermittelt werden.

Benennung der Exportdateien

Die Daten werden als XML-Datei an die VST gesendet. Die Exportdatei muss nach dem folgenden Schema benannt werden:

Syntax: <GUID>_<Inhaltskennung><Protokolltyp>_<Rolle Absender>.xml

²³ Gegebenenfalls ist der geänderte Datensatz mit einer neuen Versionsnummer zu übermitteln.

²⁴ Jeder Datensatz ist einem Primärmodul zugeordnet. Auch dem Minimaldatensatz (MDS) ist ein Primärmodul zugeordnet (Bogenfeld ZUQSMODUL).

Tabelle 14: Benennungselemente der Exportdateien

| Element | Bedeutung |
|----------------|---|
| GUID | Die verwendete GUID ist die im Dokument verwendete ID des Datenpakets (Abschnitt A „Eindeutige Kennzeichnung der XML-Exportdateien“). |
| Inhaltskennung | Q → QS-Daten S → Sollstatistik R → Routinedaten |
| Protokolltyp | T → Transaktionsprotokoll bzw. Empfangsbestätigung D → Datenflussprotokoll |
| Rolle Absender | LE → Leistungserbringer DAS → Datenannahmestelle VST → Vertrauensstelle BAS → Bundesauswertungsstelle |

Beispiele:

47d16341-9e27-4e75-a27e-b791fbbd2dc8_Q_LE.xml

(QS-Daten eines Leistungserbringers)

47d16341-9e27-4e75-a27e-b791fbbd2dc8_QD_DAS.xml

(QS-Daten-Datenflussprotokoll einer DAS)

47d16341-9e27-4e75-a27e-b791fbbd2dc8_QT_DAS.xml

(QS-Daten- Empfangsbestätigung einer DAS)

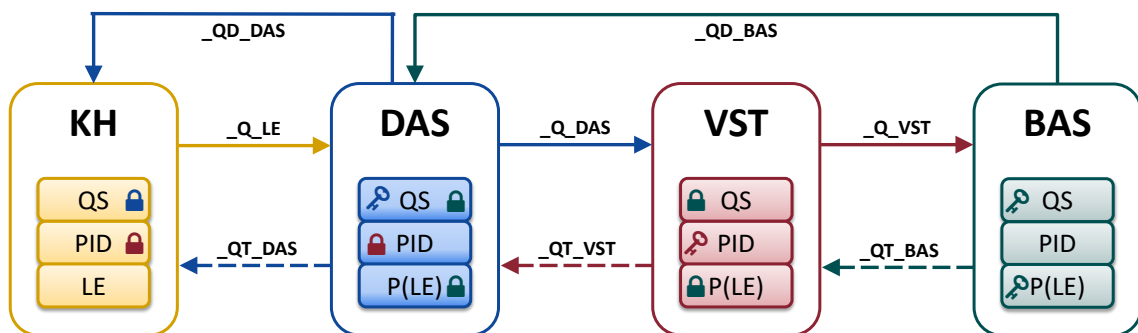


Abbildung 5: Übersicht über die einzusetzenden Suffixe

In den direkten Verfahren nimmt die Bundesauswertungsstelle (BAS) gegenüber den Leistungserbringern die Rolle einer Datenannahmestelle ein. Deswegen werden in den direkten Verfahren von der Bundesauswertungsstelle die Datenflussprotokolle ebenfalls mit dem Suffix **_QD_DAS** geschickt.

**Hinweis**

Da die endgültige Rückprotokollierung in den direkten Verfahren nur von der BAS durchgeführt wird, wird keine zusätzliche Empfangsbestätigung an die LE gesendet.

Die Verschlüsselung

Für die Verschlüsselung der QS-Daten wurde der W3C-Standard „XML-Encryption“ eingesetzt. XML-Encryption bietet eine adäquate Lösung, um komplexe Anforderungen an die Sicherheit des Datenaustausches zu erfüllen. Mit XML-Encryption können unterschiedliche Dokumentenabschnitte (XML-Knoten) für unterschiedliche Datenempfänger mit unterschiedlichen, öffentlichen Schlüsseln verschlüsselt werden, ohne diese Dokumentabschnitte, die logisch miteinander verbunden sind, in unterschiedlichen Dokumenten an die jeweiligen Datenempfänger zu senden. Auf diese Weise bleibt der Zusammenhang der Daten erhalten. Dennoch ist jeder verschlüsselte Dokumentenabschnitt nur für den Besitzer des jeweiligen Schlüssels lesbar.

Des Weiteren werden bestimmte Elemente (Tags) innerhalb der XML-Datei gezippt bzw. komprimiert, damit die Dateigröße in einem überschaubaren Rahmen bleibt.

Abschließend wird für die Datenübermittlung noch eine Transportverschlüsselung angewendet, die die gesamte Datei verschlüsselt und vor unberechtigten Zugriffen schützt.

Verschlüsselungsverfahren**XML-Encryption-Spezifikation**

Zusätzlich zur Verschlüsselung von einzelnen XML-Knoten und deren Unterelementen in einem XML-Dokument, definiert die XML-Encryption-Spezifikation weitere Möglichkeiten, wie XML-Dokumente ver- und entschlüsselt werden können:

- Verschlüsselung des gesamten XML-Dokuments
- Verschlüsselung des Inhalts eines XML-Elements
- Verschlüsselung für mehrere Empfänger
- Verschlüsselung eines einzelnen Elements und seiner Unterelemente

Verschlüsselungsalgorithmen**Symmetrische Verschlüsselung**

Die symmetrische Verschlüsselung wird mit einem einzigen Schlüssel durchgeführt, d.h. zur Ver- und Entschlüsselung wird derselbe Schlüssel von Sender und Empfänger verwendet.

Der verwendete Algorithmus für die symmetrische Verschlüsselung: AES128
Quelle: <http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf>

Asymmetrische Verschlüsselung

Verschlüsselung mit asymmetrischen Schlüsseln, d.h. einem Schlüsselpaar (öffentlicher und geheimer Schlüssel). Die Daten werden mit dem frei verfügbaren, öffentlichen Schlüssel („Public Key“) verschlüsselt und können nur mit dem privaten Schlüssel („Private Key“) entschlüsselt werden. Diese Verschlüsselung ist zeitaufwendiger, aber auch sicherer, da kein geheimer Schlüssel übertragen werden muss.

Der verwendete Algorithmus für die asymmetrische Verschlüsselung: RSA mit 2048-Bit.
Quelle: <http://www.w3.org/TR/xmlenc-core/>

Hybride Verschlüsselung

Das grundlegende Szenario dieser Verschlüsselung besteht aus folgenden Einzelschritten:

- Ein zufälliger symmetrischer Schlüssel wird erzeugt.
- Mit diesem Schlüssel wird ein XML-Element (z. B. QS-Daten) verschlüsselt.
- Der Schlüssel wird nun mit dem „Public Key“ des Empfängers (z. B. BAS) verschlüsselt.
- Der mit dem „Public Key“ verschlüsselte symmetrische Schlüssel wird dem Empfänger zusammen mit den verschlüsselten Daten übergeben.
- Der Empfänger entschlüsselt den chiffrierten Schlüssel mit seinem „Private Key“ und erhält so den symmetrischen Schlüssel, mit dem die Daten verschlüsselt wurden.
- Mit diesem symmetrischen Schlüssel entschlüsselt der Empfänger die verschlüsselten Daten.

Für die Verschlüsselung der XML-Knoten wird die hybride Verschlüsselung nach dem W3C-Standard „XML Encryption Syntax and Processing“ verwendet, da diese Verschlüsselung die Vorteile der symmetrischen und asymmetrischen Verschlüsselungsverfahren – nämlich die Schnelligkeit und die Sicherheit – kombiniert.

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.

Übermittlung der Daten im Datenfluss

Die folgenden Aspekte der Datenübermittlung werden spezifiziert:

- Datenpaket, innere Struktur
- Datenpaket, äußere Struktur

Während die innere Struktur immer eingehalten werden muss, ist die äußere Struktur nur dann einzuhalten, wenn als Übertragungsweg der Versand per E-Mail oder auf einem Datenträger gewählt wird.

Die äußere Struktur dient allein einem sicheren Übertragungsprozess. Dieser Übertragungsprozess ist für den Austausch von Dateien (z. B. per E-Mail) spezifiziert. Die innere Struktur des Datenpakets muss eingehalten werden und es müssen datenschutzrechtlich unbedenkliche Übertragungsverfahren gewählt werden. Eine Abweichung von der Übertragung mittels E-Mail soll im Konsens zwischen den Übertragungspartnern getroffen werden, wenn die Unbedenklichkeit der Übertragung sichergestellt ist (Forderung der Richtlinie).

Die innere Struktur jedes Datenpakets stellt alle notwendigen Metainformationen bereit, um dieses eindeutig zuzuordnen. Die Unbedenklichkeit der Übertragung muss nachgewiesen werden. Die innere Struktur wird durch ein XML-Schema (Übertragungsschema) definiert.

Zur äußeren Struktur gehören Festlegungen zu Dateibenennung, Transportverschlüsselung, Archivierung und Archivbenennung.

Ausgangvalidierung gegen das Übertragungsschema

Als letzte Maßnahme vor der Weiterleitung des Dokuments muss die innere Struktur des Dokuments gegen das Übertragungsschema auf Gültigkeit geprüft werden.

Die Vorteile der Ausgangvalidierung:

- Sicherstellung der Datenintegrität nach Verarbeitung der Daten
- frühe Feststellung von Fehlerquellen in der eigenen Datenverarbeitung
- Entlastung des nachfolgenden Datenservices von nicht validen Daten
- Vermeidung des Versands von Daten, die gegen den Datenschutz verstoßen

Durch diese Prüfung wird sichergestellt, dass die richtigen Bereiche des XML-Kodes verschlüsselt sind und ausgeschlossen ist, dass kritische Daten versehentlich unverschlüsselt die nächste Stelle im Datenfluss erreichen. Sie schließt ebenfalls von vornherein aus, dass Daten an den nachfolgenden Datenservice übermittelt werden, die dieser nicht verarbeiten kann.

Das an einer Übertragungsstelle gültige Schema, kann der Dokumentation über die Schemafamilie entnommen werden.

Beispiel:

Der Leistungserbringer (kollektivvertraglich) verwendet das Schema zur Schnittstelle LE-DAS:

```
interface_LE_DAS\interface_LE_KV.xsd
```

Die Datenannahmestelle verwendet das Schema zur Schnittstelle DAS-VST:

```
interface_DAS_VST\interface_DAS_VST.xsd
```

Die Validierung kann über zahlreiche frei verfügbare Tools erfolgen.²⁵ Für diese Validierung kann auch das Datenprüfprogramm des IQTIG verwendet werden.

Erzeugung der Transaktionsdatei für die Übertragung via E-Mail

Für die Übermittlung der Daten über nicht gesicherte Übertragungswege, wie z. B. E-Mail, muss die Exportdatei in eine Transaktionsdatei umgewandelt werden. Hierzu ist das symmetrische Verschlüsselungsverfahren „AES 128“ vorgeschrieben.

Das IQTIG stellt ein Verschlüsselungsprogramm bereit, mit dem eine verfahrenskonforme Transportverschlüsselung durchgeführt werden kann. Das Einbinden der Funktionen des Verschlüsselungsprogramms erfolgt über einen Befehlszeilenaufruf mit Parametern. Das Verschlüsselungsprogramm übernimmt auch die Dateibenennung der Transportdatei mithilfe von übergebenen Parametern.

Der Leistungserbringer erhält bei der Registrierung seiner Dokumentationssysteme von der zuständigen Datenannahmestelle einen Verschlüsselungskode, der bei der Erzeugung der Archivdatei verwendet werden muss. Die Transaktionsdatei wird wie folgt benannt:

```
T-<Registrierungsnummer>-<Zeitstempel im Format  
YYYY_MM_tt_hhmmss>[_<drei weitere numerische Stellen>].zip.aes
```

²⁵ <http://www.w3.org/XML/Schema>.

T-NI1234A-2012_09_19_160945.zip.aes

T-BW1234a-2012_09_23_114113_045.zip.aes (millisekundengenau)

Die Registrierungsnummer ist die Grundlage der symmetrischen Transportverschlüsselung mittels des bei der Registrierung ausgetauschten geheimen Schlüssels. Die Transportentschlüsselung kann ausschließlich über die Zuordnung der Registrierungsnummer zu dem ausgetauschten geheimen Schlüssel erfolgen. Der Zeitstempel ist auf die Registrierungsnummer einer eindeutigen Kennzeichnung einer Transaktion bezogen. Die Transaktionsbestätigung erfolgt über diese Kennzeichnung (siehe nächster Abschnitt).

Die drei weiteren numerischen Stellen sind optional und stellen im Prinzip Millisekunden dar. Sie sind aber nur als Unterscheidungsmerkmal notwendig, wenn innerhalb einer Sekunde mehr als eine Transaktionsdatei erstellt werden soll. Wenn diese Option angewendet wird, sollen alle drei Stellen gesetzt und ggf. mit „0“ aufgefüllt sein.

Erzeugung der Antwortdatei für die Übertragung via E-Mail

Die Empfangsbestätigung – und das Datenflussprotokoll werden nach dem gleichen Verfahren wie die Transaktionsdatei in ein mit AES verschlüsseltes ZIP-Archiv gepackt und wie folgt benannt:

```
A-<Registrierungsnummer>-<Zeitstempel im Format  
YYYY_MM_tt_hhmmss>[_<drei weitere numerische Stellen>].zip.aes
```

Der verwendete Zeitstempel entspricht beim Datenflussprotokoll dem Zeitstempel der Datei, mit der die Daten an die Datenannahmestelle versandt wurden. Damit ist eine einfache, eindeutige Zuordnung zur Transaktion möglich.

Beispiel:

Die Benennung der Antwortdatei

```
Antwortdatei: A-BW1234a-2016_09_23_094051.zip.aes  
auf die Transaktionsdatei: T-BW1234a-2016_09_23_094051.zip.aes
```

Im PID-Datenfluss ist der Name der Transaktionsdatei, mit dem die DAS das Datenpaket an die VST-PSN weitergeleitet hat, für die BAS nicht mehr verfügbar. Die Zuordnung des Datenflussprotokolls kann an dieser Schnittstelle nur noch über die GUID der Exportdatei (Abschnitt A „Eindeutige Kennzeichnung der XML-Exportdateien“) erfolgen, da die Transaktionskennzeichnung über Registrierung und Zeitstempel immer nur zwischen zwei Übertragungspartnern gültig ist. Hier wird für die Übertragung jeweils ein neuer Zeitstempel erzeugt.

Identifizierung und Authentifizierung des Einsenders

Es existieren einige sichere Datenübertragungssysteme zwischen bestimmten Einrichtungen. Bei diesen Übertragungen muss der Datenlieferant (Provider) die Daten nicht explizit für den

Transport verschlüsseln, da es sich um einen gesicherten Transportweg handelt. Die Autorisierung der Datenlieferung wird bereits beim Aufbau der Verbindung vorgenommen und muss nicht erneut bei jeder Lieferung geschehen.

Der vorliegende Abschnitt beschreibt die notwendigen Prüfungen auf zusätzlich gesicherten Transportwegen wie beim Versand per E-Mail, auf anderen Wegen durch das Internet oder über Datenträger. In diesen Fällen sind die Dateien für den Transport durch eine Transportverschlüsselung abzusichern. Diese gesicherten „Pakete“ tragen die Datei-Endung `.zip.aes`.

Wenn eine Datenlieferung erfolgt, sollte das entsprechende Datenpaket vor dem Öffnen zunächst überprüft und unter Umständen abgelehnt werden.

Prüfungen des E-Mail-Versands

Gültigkeit des Absenders

Die Prüfung der Gültigkeit des Absenders ist aufgrund der Möglichkeit, beliebige Absender in eine E-Mail einzusetzen, kein Kriterium für die Authentizität eines Absenders. Diese Prüfung kann aber ein Hilfsmittel sein, um eingehende Spam-E-Mails von vornherein von der Verarbeitung auszuschließen. Da es keine Seltenheit ist, dass Datenpakete von einer anderen Absenderadresse eingeschickt werden als von der, die mit der Registrierung mitgeteilt wurde, wird diese Prüfung nicht verbindlich empfohlen. Im Fall eines Ausschlusses unbekannter Absender von der Verarbeitung muss zudem mit einem erheblichen Mehraufwand auf der Seite des Telefonsupports gerechnet werden. Die Abwägung zwischen Öffnung gegenüber Spam und einem erhöhten Supportaufwand liegt im Ermessen der jeweiligen Datenannahmestelle.

Prüfungen des Anhangs

Für einen gültigen Datenversand muss eine `zip.aes`-Datei der E-Mail angehängt sein, die der im Anhang näher beschriebenen Namenskonvention entspricht. Aus dem Namen wird die Registrierungsnummer des Absenders extrahiert. Mithilfe der Registrierungsnummer wird in der Registrierungsdatenbank der individuell für diese Registrierung ausgetauschte Schlüssel herausgesucht und mit diesem ein Entschlüsselungsversuch nach den Vorgaben der Transportverschlüsselung durchgeführt.

Wenn die Entschlüsselung erfolgreich war, ist von der Authentizität des Absenders auszugehen, dessen Identität sich über die Registrierung ergibt.

Sicherheit der Datenübertragung per E-Mail

Das verwendete Verfahren hat sich seit über 10 Jahren bewährt:

- Das Übertragungsverfahren per E-Mail sieht eine vorhergehende Registrierung des Leistungserbringers/Absenders vor.
- Im Registrierungsvorgang werden über einen sicheren Kanal die Identität authentifiziert, eine Registrierungsnummer und ein geheimer symmetrischer Schlüssel ausgetauscht, der nur den beiden Seiten bekannt ist.
- Das zu versendende Datenpaket wird vom Absender mit dem geheimen Schlüssel verschlüsselt. Zurzeit wird ein AES-Verschlüsselungsverfahren mit der Stärke 128 Bit angewendet.

- Das verschlüsselte Datenpaket erhält in der Benennung an vorgeschriebener Stelle die Registrierungsnummer.
- Anhand der Registrierungsnummer bezieht die Datenannahmestelle aus der Registrierungsdatenbank den Schlüssel und die Identität des Absenders und kann mit dem Schlüssel das Datenpaket entschlüsseln.
- Eine erfolgreiche Entschlüsselung authentifiziert dabei die Identität des Absenders des Datenpakets, da nur dieser neben der Datenannahmestelle den geheimen Schlüssel kennt.

Anschließend muss das Entschlüsselungsergebnis genauer betrachtet werden:

- Ist nur eine einzige Datendatei in dem Transportarchiv enthalten?
- Stimmt die Kennzeichnung der Herkunft des Dokuments mit der erwarteten Herkunft überein?
- Stimmt die Kennzeichnung im Dateinamen in Bezug auf den Inhalt des Dokuments mit den zulässigen Inhalten der bedienten Schnittstelle überein?

Alle Verstöße gegen die vorstehend dargelegten Regeln führen zu einer Ablehnung der Einsendung.

5.1.2 Abgrenzung von Test-, Probe- und Regelbetrieb

Im Folgenden wird beschrieben, welche Testmöglichkeiten es im Datenfluss gibt, wie diese genutzt werden können und sollen und wie sich diese voneinander und vom Produktivbetrieb abgrenzen lassen.

Definition Test- und Echtdaten

Es wird zwischen Test- und Echtdaten unterschieden.

Als **Testdaten** werden QS-Daten bezeichnet, die der Datenstruktur der Spezifikation, aber nicht Datensätzen von realen Personen entsprechen.

Als **Echtdaten** werden solche QS-Daten bezeichnet, die der Datenstruktur der Spezifikation und Datensätzen von realen Personen entsprechen, sodass die besonderen Regeln des Datenschutzes auf sie zutreffen.

Datenziele

Alle auswertenden Einrichtungen (DAS, BAS) stellen drei Datenziele zur Verfügung. Diese Datenziele werden als

- Testdatenpool
- Probedatenpool
- Echtdatenpool

bezeichnet.

Die Datensicherheit bemisst sich in allen Datenpools an den Maßstäben, die für personenbezogene Daten gelten. Das bedeutet, dass prinzipiell alle drei Datenziele geeignet sind Echtdaten entgegenzunehmen, weil die Sicherheit gewährleistet wird.

Das Datenziel wird im Headerbereich des XML-Dokuments festgelegt (/document/data_target).

Testdatenpool

Der Testdatenpool ist sowohl für generierte Testdaten als auch für Echtdaten, die zu Testzwecken versendet werden, bestimmt. Für Echtdaten gilt aber, dass diese auch zu Testzwecken ausschließlich über Produktivinstanzen (siehe unten) geschickt werden dürfen.

Für Daten, die an den Testdatenpool geschickt werden, gibt es keine Erhaltungsregel. Ein Testdatenpool kann ohne vorherige Absprache geleert werden. Die Daten werden spätestens nach einem Jahr gelöscht.

Der Testdatenpool wird nicht inhaltlich ausgewertet. Er soll Testungen von Datenbankoperationen des automatisierten Datenservices und die Simulation echter Rückprotokollierungen ermöglichen.

Probedatenpool

Der Probedatenpool ist für Echtdaten bestimmt, die in Sonderfällen für Probeauswertungen genutzt werden. Diese Daten werden nach Abschluss der Auswertung oder spätestens nach einem Jahr gelöscht.

Dieser Datenpool ist nur für Machbarkeitsprüfungen, Probetriebe und Sonderexporte relevant.

Echtdatenpool

Der Echtdatenpool ist für Echtdaten bestimmt, die dem Bundesdatenpool zugeführt werden. Diese Daten werden entsprechend der Regeln für den Bundesdatenpool gepflegt. Alle Dokumentationen für den Regelbetrieb werden dem Echtdatenpool zugeführt.

Datenservices

Als ein Datenservice wird eine automatisierte Datenverarbeitung unter einem von außen erreichbaren Endpunkt (z. B. E-Mail-Adresse oder URL) verstanden.

Für jede Instanz eines Datenservices stehen folgende Eigenschaften fest:

- wer der Anbieter des Datenservices ist
- welchen Endpunkt er bedient (E-Mail-Adresse, URL)
- welche Übertragungsarten er bedient (E-Mail, Restservice, KV-Connect etc.)
- welcher öffentliche Schlüssel zu verwenden ist
- welche Bundesländer/Regionen bedient werden
- welche Einsender-Gruppen bedient werden (LE-stat, LE-amb, SWA, DAS, KK)
- welche Spezifikation bedient wird (Spezifikationskennung z. B. 2017_BASIS_RB_XML)
- welche Module bedient werden
- welche Datenziele bedient werden (Testdatenpool, Probedatenpool, Echtdatenpool)
- an welche nachfolgenden Datenservices die verarbeiteten Daten weitergereicht werden
- ob Datensicherheit garantiert wird

Es wird bei den Datenservices grundsätzlich zwischen Testinstanzen und Produktivinstanzen unterschieden. Diese Unterscheidung hat wesentliche Auswirkungen auf bestimmte Eigenschaften eines Datenservices.

Produktivdatenservices

Produktivdatenservices sind Datenservices, mit denen alle Datenpools erreicht werden können und deren Betreiber für die Datensicherheit garantieren muss.

Wegen der Datensicherheit dürfen Daten aus einem Produktivdatenservice auch nur an einen anderen registrierten Produktivdatenservice oder einen registrierten Leistungserbringer weitergeleitet werden.

Eine Testung kann auch über Produktivdatenservices erfolgen. Dabei ist unerheblich ob Echt- oder Testdaten verwendet werden. Ausschlaggebend ist, dass in dem Datenpaket, welches zur Testung verwendet werden soll, als Datenziel „Testdatenpool“ eingetragen ist.

Testdatenservices

Ein Testdatenservice ist ein Datenservice, der von der betreibenden Einrichtung vorgehalten wird, um Daten zu Testzwecken anzunehmen und gleichzeitig um die Verarbeitung der Daten durch den Datenservice zu testen.

Ein Testdatenservice beruht auf Spezifikationen des Probe- oder Regelbetriebs. In Ausnahmefällen kann auch eine Spezifikation für den Testbetrieb die Grundlage bilden, wenn es keine entsprechenden Spezifikationen für die zu testenden Aspekte gibt.

Die angenommenen Daten dürfen unabhängig von der Ausweisung des Datenziels durch den Absender nie in einen Echt oder Probedatenpool, sondern immer nur einem Testdatenpool zugeführt werden.

Anders als die garantierte Datensicherheit des Testdatenpools (siehe oben), wird die Datensicherheit des Übertragungsweges über Testdatenservices nicht garantiert. Deswegen dürfen an Testdatenservices immer nur Testdaten (also generierte Daten ohne Bezug zu realen Personen) geliefert werden.

Es wird empfohlen Testinstanzen von DAS und VST ab zwei Monaten nach Veröffentlichung der ersten Spezifikation eines Erfassungsjahres (Release V01) bereitzustellen. Im Rahmen aller weiteren Releases für ein Erfassungsjahr wird die Bereitstellung der jeweiligen Testinstanz nach einer Woche empfohlen.

Datenservices und Testdatenservices im Datenfluss

Datenpakete dürfen von Testinstanzen nur an nachgeschaltete Testinstanzen anderer Einrichtungen weitergereicht werden. Damit wird verhindert, dass versehentlich Testdaten in Echt Datenpools aufgenommen werden.

Datenpakete dürfen von Produktivinstanzen nur an nachgeschaltete Produktivinstanzen anderer Einrichtungen weitergeleitet werden. Damit wird verhindert, dass mögliche schutzbedürftige personenbezogene Daten in Testinstanzen ohne Datenschutzgarantie verarbeitet werden.

Daraus ergeben sich die folgenden Datenflüsse:

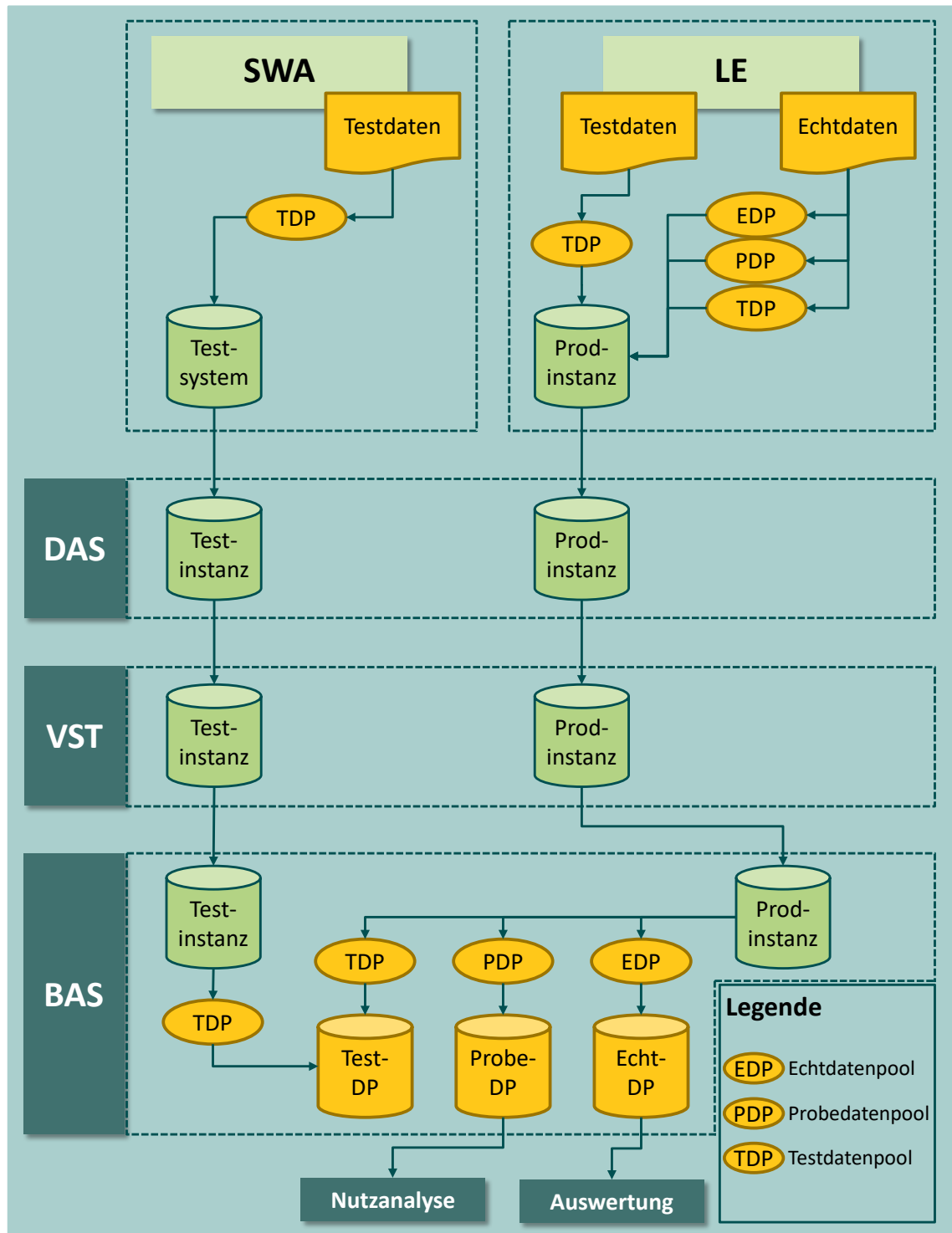


Abbildung 6: Datenflüsse im Test-, Probe- und Regelbetrieb am Bsp. der Follow-up-Verfahren

Testbetrieb mit Testsystemen und Testungen mit Produktivinstanzen

Testbetriebe finden auf der Grundlage einer Spezifikation für den Probetrieb oder den Regelbetrieb mit Testinstanzen statt.



Achtung

Beim Testbetrieb mit Testsystemen dürfen keine Echtdateien mit Personenbezug verwendet werden, da die strengen Datensicherheitsrichtlinien nur in den Produktivsystemen garantiert werden können.

Für Testinstanzen gelten unter Umständen andere Registrierungen als für Produktivsysteme (Abschnitt A „Registrierungen“). Diese Frage ist bilateral zwischen den Datenlieferanten und der Datenannahmestelle zu klären.

In Ausnahmen kann als Grundlage für einen Testbetrieb auch eine eigene Spezifikation erstellt werden (Betriebsart TB, siehe Einleitung Abschnitt 1.1), wenn z. B. Änderungen in Exportverfahren unverbindlich getestet werden sollen.

Testungen mit Produktivinstanzen können jederzeit mit Test- und mit Echtdateien durchgeführt werden. Dazu muss zwingend das Datenziel „Testdatenpool“ angegeben werden.

Probetrieb

Probetriebe erfordern eine eigene Spezifikation.

Regelbetrieb

Der Regelbetrieb erfordert eine eigene Spezifikation.

5.1.3 Allgemeine Prüfungen des XML-Dokuments

Unabhängig von der Übertragungsart wird nach der Annahme einer Datenlieferung der gelieferte Inhalt genauer geprüft.

Zunächst sind ganz allgemeine Prüfungen durchzuführen, die das gesamte Dokument betreffen und von allen Datenannahmestellen durchgeführt werden müssen. Erst nach erfolgreicher Prüfung (Status = OK oder WARNING) kommt auf einen Teil der Datenannahmestellen mit der Entschlüsselung der QS-Daten und deren Plausibilitätsprüfung eine weitere Aufgabe zu.

Für alle Prüfungen gilt, dass sie im Fehlerfall entweder das geprüfte Item (Dokument oder Datensatz) ausschließen oder mit dem Vermerk eines Hinweises in die Weiterverarbeitung geben. Zur Einstufung eines Fehlers gibt es dementsprechend zwei Stufen:

- WARNING steht für einen Hinweis
- ERROR steht für einen Ausschluss von der Verarbeitung

Welche Einstufung vorzunehmen ist, ist spezifiziert und kann der Fehlertabelle in der Spezifikationsdatenbank (QSDOK) entnommen werden.

Ist das Dokument von einem ausschließenden Fehler betroffen, führt das zu einer Ablehnung des gesamten Dokuments. Sind einzelne Datensätze von einem ausschließenden Fehler betroffen, wird der fehlerhafte Datensatz als fehlerhaft markiert, aber im Dokument belassen.

Sind alle Datensätze von einem ausschließenden Fehler betroffen, wird das gesamte Dokument als fehlerhaft markiert. Zum Aufbau des Fehlerprotokolls siehe auch Abschnitt A 5.2.2.



Hinweis

Zusätzliche Prüfungen, die nicht in der Spezifikation enthalten sind, dürfen durchgeführt werden. Sie dürfen allerdings nie zu der Konsequenz einer Ablehnung führen. Falls ein Bedarf an Erweiterungen besteht, müssen diese über das Institut nach § 137a SGB V zur Abstimmung in die Spezifikation eingebracht werden.

Prüfung des schemakonformen Dokumentaufbaus:

Grundsätzlich hat ein XML-Dokument einen formalen Aufbau zu gewährleisten. Das „Grundgerüst“ jedes XML-Dokuments muss immer gleichbleibend sein, sodass die verschiedenen Informationen auch an gleich bezeichneten Stellen wiederzufinden sind. Für jede Übertragungsschnittstelle gibt es jeweils ein zutreffendes XML-Schema, gegen welches das übertragene Dokument zu validieren ist. Die jeweiligen Schema-Dateien sind im Spezifikationspaket enthalten.

Tabelle 15: Übersicht über die Schnittstellenschemata der DAS

| Schnittstelle | Schema | Verfahren | Zweck |
|----------------------------|----------------------|----------------|--|
| LE/DAS interface_LE_DAS | interface_LE_KV | PID-Verfahren | Für die Validierung der von den LE an die KV übermittelten Daten |
| | interface_LE_LQS_iv | IV-Verfahren | Für die Validierung der von den LE an die LQS übermittelten Daten |
| | interface_LE_LQS_pid | PID-Verfahren | |
| | interface_LE_SV | PID-Verfahren | Für die Validierung der von den LE an die DAS-SV übermittelten Daten |
| DAS interface_DAS | interface_LQS_iv | IV-Verfahren | Nach der QS-Entschlüsselung |
| | interface_LQS_pid | PID-Verfahren | |
| | interface_KV_pid | PID-Verfahren | |
| | interface_LQS_psn | Alle Verfahren | Nach der LE-Pseudonymisierung, sofern bei der DAS durchzuführen |
| | interface_KV_psn | PID-Verfahren | |
| | interface_SV_psn | PID-Verfahren | |

| Schnittstelle | Schema | Verfahren | Zweck |
|------------------------------|----------------------------|---------------|---|
| DAS/VST interface_DAS_VST | interface_DAS_VST | PID-Verfahren | Datenmodell der an die VST ausgehenden Daten mit XML-verschlüsselter Leistungserbringererkennung (care_provider) und XML-verschlüsselten QS-Daten (qs_data) |
| DAS/BAS interface_DAS_BAS | inter- face_DAS_BAS.xsd | IV-Verfahren | Für die Validierung der von den DAS an die BAS übermittelten Daten |

Zu beachten ist ebenfalls, dass nur die standardisierten Fehlermeldungen zu verwenden sind. Sollte der Bedarf nach weiteren Fehlermeldungen festgestellt werden, ist dieses über das Institut nach § 137a SGB V zu melden, welche die Übernahme in das Spezifikationsregelwerk organisiert.

Zur Entlastung der Leistungserbringer soll beim Eintragen der Fehlermeldungen unbedingt auf ggf. vermerkte, vom LE abweichende Verursacher geachtet werden. In der Fehlermeldung steht in diesem Fall ein Attribut `/error@originator` (Abschnitt A 5.2.1).

Beispiel:

Verursacher=Datenannahmestelle

Wenn z. B. in der Bundesauswertungsstelle die Entschlüsselung der `<qs_data>`-Elemente nicht durchgeführt werden kann, ist dieses bei indirekten Verfahren ein Fehler der Datenannahmestelle (LQS/LKG) und kann nur dort korrigiert werden.

Beim Eintragen der Fehlermeldungen müssen die im Abschnitt Rückprotokollierung beschriebenen allgemeinen Regeln beachtet werden (Abschnitt A 5.2.2).

5.2 Rückprotokollierung

In diesem Kapitel wird die Rückprotokollierung in Bezug auf die Funktion, den Aufbau und die Erstellung beschrieben.

5.2.1 Funktion von Empfangsbestätigung und Datenflussprotokoll im Datenfluss

Empfangsbestätigung

Die Empfangsbestätigung wird nach Erhalt und abschließender erfolgreicher Eingangsverarbeitung und Weiterleitung eines Dokuments über den Eingangskanal an den Absender übermittelt. Sie bestätigt dem Absender den Übergang der Verantwortung für das Dokument an den Aussteller.

Eine Empfangsbestätigung ist nur für den Absender bestimmt und wird nicht weitergeleitet. Bei Ausbleiben ist von einer fehlgeschlagenen Übermittlung auszugehen. Es gibt zurzeit keine verbindliche Vereinbarung, in welchem zeitlichen Rahmen eine Empfangsbestätigung erwartet werden kann. Angestrebt werden soll allerdings eine Echtzeitverarbeitung, sodass allein die Verarbeitungsdauer eines Dokuments die Verzögerung einer Empfangsbestätigung bedingt.

Datenflussprotokoll

Ein Datenflussprotokoll wird erstellt, wenn das Dokument keine weitere Verarbeitung mehr erlaubt. Das ist dann der Fall, wenn das Dokument durch einen der vorgesehenen Prüfungs- und Verarbeitungsschritte den Status `ERROR` erhält oder wenn das Dokument in der Bundesauswertungsstelle vollständig und erfolgreich verarbeitet wurde und den Status `WARNING` oder `OK` trägt.

Das Datenflussprotokoll dokumentiert alle an dem Dokument durchgeführten Prüfungen und deren Ergebnisse.

Ein Datenflussprotokoll wird in der Regel bis zum Leistungserbringer zurückübermittelt. Aus diesem Grund muss die Datenannahmestelle nach der Prüfung des erhaltenen Datenflussprotokolls den Leistungserbringer depseudonymisieren.

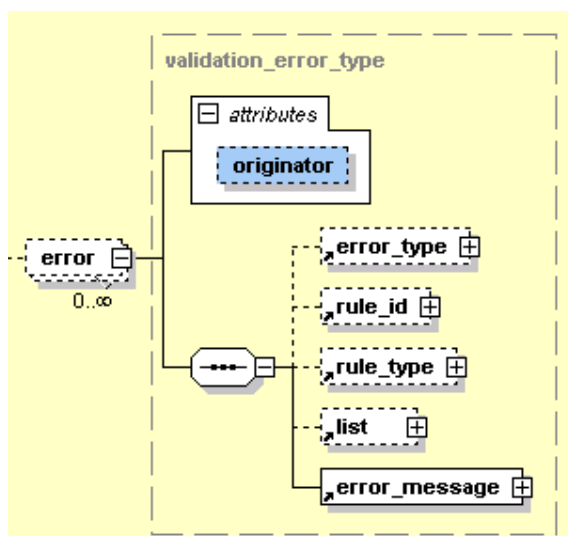


Abbildung 7: Attribut „originator“ im Prüfungs- und Fehlerprotokoll

Die im Protokoll im Attribut „originator“ ggf. enthaltenen Hinweise auf den Verursacher eines Fehlers müssen von der Datenannahmestelle so ausgewertet werden, dass Verarbeitungsfehler, die weder auf Fehler des Leistungserbringers noch auf die Software des Leistungserbringers zurückzuführen sind, nicht an den Leistungserbringer weitergeleitet werden. Dieses ist daran erkennbar, dass im Attribut „originator“ ein von Leistungserbringer oder Softwareanbieter abweichender Wert steht. In einem solchen Fall soll verhindert werden, dass das Datenflussprotokoll an den Leistungserbringer verschickt wird.

Statt der Weiterleitung sollte die fehlerhafte Verarbeitung des auslösenden Beteiligten (Datenannahmestelle, Vertrauensstelle, Bundesauswertungsstelle) korrigiert werden und ein korrekt verarbeitetes Dokument, dessen Stand in Abstimmung mit der Bundesauswertungsstelle über

die ID zurückgesetzt wurde, erneut in den Datenfluss eingebracht werden. Für das Zurücksetzen eines Dokuments steht kein automatisierter Prozess zur Verfügung. Dieses kann über die E-Mail-Adresse verfahrensupport@iqtig.org vereinbart werden.

Erzeugung der Protokolle

Die Datenflussprotokolle werden durch eine Reduktion der erhaltenen QS-Export-Daten erstellt. Dabei werden die Elemente `<patient>` und `<qs_data>` aus der Datei entfernt. Es verbleiben Header, Protokoll und Admin-Daten in der Datei.

Für das Datenflussprotokoll wird das Attribut `feedback_range` des Elements `<protocol>` auf `dataflow` gesetzt.

Für diesen Verarbeitungsschritt kann das Datenprüfprogramm eingesetzt werden.

5.2.2 Die Rückprotokollierung

Die Empfangsbestätigung

Die Empfangsbestätigung wird vom Datenempfänger über den Eingangskanal an den Datensender zurückgeschickt, sofern das angenommene Datenpaket in Bezug auf die Prüfungen verarbeitbar ist und weitergeleitet werden kann.

Beispiel:

Wie die folgende Abbildung zeigt, enthält die Empfangsbestätigung lediglich eine Benachrichtigung, dass die vom Leistungserbringer übermittelte XML-Datei verarbeitbar war und an die nachfolgende Stelle weitergeleitet wurde:

```
<?xml version="1.0" encoding="UTF-8"?>
<root xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
content_version="1.0" container_version="2.0"
xsi:noNamespaceSchemaLocation="../../../../interface_LE_DAS/response_receipt.xsd"
xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
  <header>
    <document>
      <id V="{55664537-5642-9056-8676-456468327446}"/>
    </document>
    <provider email="datenannahmestelle@test-datenannahmestelle.de"
name="Test-Datenannahmestelle" registration="bu00000"
function="Datenannahmestelle" address="Test Adresse"/>
  </header>
  <body>
    <nachricht>
      Ihre Dateneinsendung konnte erfolgreich eingelesen und an die Vertrauensstelle weitergeleitet werden.
      Ein umfassendes Datenflussprotokoll über die Prüfergebnisse der Datenannahmestelle, der Vertrauensstelle
      und der Bundesauswertungsstelle erhalten Sie von uns in den nächsten 24 Stunden.
    </nachricht>
  </body>
</root>
```

Abbildung 8: Beispiel einer Empfangsbestätigung

Die Empfangsbestätigung soll ohne Verschlüsselung an den Sender geschickt werden. Für eine weitere Vereinfachung der Benachrichtigung können die Datenannahmestellen den Inhalt der spezifizierten Benachrichtigung z. B. in die E-Mail übernehmen (betreff/body). Die vollständige Benachrichtigung lautet:

Ihre Dateneinsendung konnte erfolgreich eingelesen und an die <nachfolgende Stelle (Vertrauensstelle oder Bundesauswertungsstelle)> weitergeleitet werden.

Ein umfassendes Datenflussprotokoll über die Prüfergebnisse der Datenannahmestelle, der Vertrauensstelle und der Bundesauswertungsstelle erhalten Sie von uns innerhalb der nächsten 24 Stunden.

Die Benennung der unverschlüsselten Datei leitet sich aus dem Benennungsschema ab (Abschnitt A „Benennung der Exportdateien“).

Das Datenflussprotokoll

Die vorgenommenen Prüfungen werden in den dafür vorgesehenen Bereichen im XML-Code des übermittelten Dokuments protokolliert. Das Protokoll des Dokuments wächst damit mit jeder Prüfung an.

Nachdem alle Prüfungen der datenentgegennehmenden Stelle abgeschlossen sind, wird für die Rückprotokollierung der Prüfungsergebnisse eine Kopie des Dokuments von allen QS-Daten (Element `<qs_data>`) und patientenidentifizierende Daten (Element `<patient>`) befreit. 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 weiter unten im Abschnitt A „Prüfungsprozess und Ergebnisprotokollierung“ detailliert beschrieben.

Das XML-Protokoll kann von der datenentgegennehmenden Stelle oder von der QS-Software mithilfe einer Template-Definition z. B. nach HTML transformiert werden. Den zuvor beschriebenen Plausibilitätsprüfungen schließen sich auf Ebene der Vertrauensstelle und auf Bundesebene weitergehende Prüfungen an. Diese führen zu einem weiteren Anwachsen des Prüfprotokolls und der Datenqualität.

Bezogen auf einen bestimmten Datensatz ist es erst nach der letzten abgeschlossenen Prüfung auf Bundesebene möglich, eine Aussage darüber zu treffen, ob sich dieser Datensatz für die Aufnahme in den Bundesdatenpool eignet oder nicht.

Um dem Leistungserbringer das konkrete Ergebnis seiner Datenlieferung in Bezug auf den Bundesdatenpool mitteilen zu können, wird auch das bis zum Schluss weitergeführte Dokument von QS- und patientenidentifizierenden Daten befreit und als Datenflussprotokoll an die datenentgegennehmende Stelle versandt, die dann die Aufgabe hat, dieses an die – nur am Pseudonym erkennbaren – Leistungserbringer zu übermitteln.

Die Datenflussprotokolle werden jedoch auch von der Datenannahmestelle oder der Vertrauensstelle für den Fall erstellt, dass eine Weiterleitung der QS-Exporte nicht möglich war. In diesem Fall wird keine Eingangsbestätigung an den Absender verschickt. Um das Dokument als Datenflussprotokoll kenntlich zu machen, muss das Attribut `root/header/protocol/@feedback_range` auf den Wert `dataflow` gesetzt werden.

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 Fehlermeldungen

Für jeden Fehler wird wenigstens das Element `<error_message>` ausgefüllt. Andere Elemente bleiben bei einzelnen Fehlerarten leer. Tabelle 16 gibt einen Überblick darüber, unter welchen Bedingungen in den Feldern der Fehlerdatei Angaben erforderlich sind.

Die Bogenliste `<list>` umfasst einen oder mehrere Namen von Teildatensätzen, welche einen Bezug zu einer Regel haben. Entscheidend für den Bogenbezug sind die in der Tabelle `Regeln` formulierten Regeln, nicht die für den Exportdatensatz umformulierten Regeln.

Die Bogenfeldliste umfasst einen oder mehrere Namen von Bogenfeldern, welche einen Bezug zum Fehler haben. Bei der Fehlerart WERT enthält die Liste nur ein Element. Der Bogenfeldname umfasst auch den Namen des zugehörigen Teildatensatzes²⁶ (Beispiele: `O[2]/FLDO-SISKORO`, `B/AUFNDATUM`).

Für jede Regel gibt es eine Liste von Bogenfeldern, identifiziert über die Feldnamen der Regeln. Damit die Liste nicht durch Parsen ermittelt werden muss, wird sie auch über die Tabelle `RegelFelder` zur Verfügung gestellt. Über die Regelnummer können die Teildatensätze, welche Bezug zu einer Regel haben, durch folgende Abfrage identifiziert werden:

```
SELECT DISTINCT Bogen.name FROM (Modul INNER JOIN (Feld INNER JOIN
(Bogen INNER JOIN BogenFeld ON Bogen.idBogen = BogenFeld.fkBogen)
ON Feld.idFeld = BogenFeld.fkFeld) ON Modul.idModul = Bogen.fkModul)
INNER JOIN RegelFelder ON BogenFeld.idBogenFeld = RegelFelder.fkBogenFeld
WHERE RegelFelder.fkRegeln=<Regelnummer>;
```

Für die `Regelnummer <rule_id>` ist die entsprechende Nummer (Attribut `idRegeln`) der Tabelle `Regeln` anzugeben.

Bei Teildatensätzen, welche mehrfach angelegt werden können, muss die Nummer des betreffenden Teildatensatzes in eckigen Klammern angehängt werden (z. B. `P[1]`, `P[2]` usw.). Mit „Nummer des betreffenden Teildatensatzes“ ist die Position des Teildatensatzes im XML.

Beispiel:

Modul 16/1

Die Regel 2044 hat den Bezug zu den Teildatensätzen M und K des Moduls 16/1. Da der Teildatensatz K mehrfach angelegt werden kann, muss in der Bogenliste zusätzlich zu den betroffenen Feldern auch der betreffende Teildatensatz angegeben werden, z. B.:

²⁶ Der Bezug zum Modul kann entfallen, da dieses über die Vorgangsnummer identifiziert werden kann.

```

<error>
<error_type V="REGEL" />
<rule_id V="2044" />
<rule_type V="H" />
<list V="K[2]/LNRMEHRLING|M/ANZMEHRLINGE" />
<error_message V="Anzahl der Mehrlinge ist kleiner als ..." />
</error>

```

Tabelle 16: Ausfüllen der Elemente eines Validation-Items in Abhängigkeit von den Fehlerarten

| Feld (csv) | Fehlerart | Regelnr | Regeltyp | Liste | Meldung |
|---------------|--------------|-----------|-------------|--------------------------------|-----------------|
| Element (xml) | <error_type> | <rule_id> | <rule_type> | <list> | <error_message> |
| | STEUER | - | - | - | ja |
| | EXPORT | Ja | - | <Bogen> | ja |
| | DOPPELT | Ja | - | - | ja |
| | TDS | Ja | - | <Bogenliste> | ja |
| | WERT | Ja | - | <Bogenfeldliste> ²⁷ | ja |
| | REGEL | Ja | ja | <Bogenliste> | ja |

Beispiel:

Beispiel eines Protokolls

```

<protocol>
  <status_case V="ERROR" />
  <validation_item V="Spezifikation" c_date="2014-06-25T08:53:34" id="1">
    <status V="ERROR">
      <error>
        <error_type V="WERT" />
        <rule_id V="1001022" />
        <rule_type V="H" />
        <list V="B/DokAbschlDat" />
        <error_message V="Das Datenfeld 21/3:B:DokAbschlDat muss einen gültigen Wert enthalten." />
      </error>
      <error>
        <error_type V="WERT" />
        <rule_id V="1001022" />
        <rule_type V="H" />
        <list V="PROZ/OPDATUM" />
        <error_message V="Das Datenfeld 21/3:PROZ:OPDATUM ##34;Datum der Prozedur##34; muss einen gültigen Wert enthalten." />
      </error>
      <error>
        <error_type V="REGEL" />
        <rule_id V="7466" />
        <rule_type V="W" />
        <list V="PCI/OPSCHLUEPTCA" />
        <error_message V="Hinweis: Keine QS-Filter-PTCA-Einschlussprozedur für Datensatz 21/3 dokumentiert" />
      </error>
    </status>
  </validation_item>
</protocol>

```

²⁷ In der Regel wird hier nur ein Bogenfeld aufgeführt. Ausnahme ist, wenn Kombinationsfelder geprüft werden: ENTLDIAG | ENTLDIAGVERS u.a.

Um die Fehleranalyse zu vereinfachen, wurden die potenziellen Fehler in Kategorien (XML-Element `<error_type>`) unterteilt, die bestimmte Prüfprozesse (XML-Element `<validation_item>`) durchlaufen. Das Ergebnis ist folgende Tabelle:

Tabelle 17: Mögliche Fehlerarten in Prüfprozessen

| Prüfprozess | Fehlerart | | | | | |
|---|-----------|-------|---------|--------|------|-----|
| | | Regel | Doppelt | Export | Wert | TDS |
| Plausibilitätsprüfung QS gegen Spezifikation: „Spezifikation“ | | | | | | |
| PID_Pseudonym | Kollision | | | | Wert | |
| Schema-Konformität: „Schema“ | | | | | | |
| Entschlüsselungsprozess: „Dechiffrierung“ | PID | QS | LE | | | |
| LE_Pseudonym | | | | | Wert | |
| sonstige Prüfung | | | | | | |

Die konkreten Fehlermeldungen sind in der Spezifikationsdatenbank QSDOK hinterlegt:

- Tabelle `Regeln.meldung`: enthält spezifische Fehlermeldungen bei entsprechenden Regelverletzungen
- Tabelle `Fehlermeldung.meldung`: enthält standardisierte und allgemeingültige Fehlermeldungen. Folgende Tabelle zeigt Beispiele bei bestimmten Fehlerfällen.

Tabelle 18: Beispiele von Fehlermeldungen

| Fehlerart | Standardisierte Meldung | Beschreibung |
|--|---|---|
| Standardisierung der Meldungen bei Bestätigungsstatus mit Fehlerart DOPPELT. | <i>Es wurde bereits ein Datensatz mit derselben Registrier- und Vorgangsnummer und derselben oder einer höheren Versionsnummer übermittelt.</i> | Bei feldbezogenen Fehlern sind die standardisierten Fehlermeldungen der Plausibilitätsregeln zu verwenden. |
| Standardisierung der Meldungen bei Bestätigungsstatus mit Fehlerart TDS. | Erforderlicher Teildatensatz <code><Bogen.name></code> („ <code><Bogen.bezeichnung></code> “) existiert nicht. | Wenn ein obligatorischer Teildatensatz (Attribut <code>Bogen.fkBogenZahl + oder 1</code> ist, oder ein zu einem Kindteildatensatz zugehöriger Mutterteildatensatz) eines Vorgangs in den Exportdateien einer Transaktion nicht vorkommt |

| Fehlerart | Standardisierte Meldung | Beschreibung |
|-----------|--|---|
| | <i>Die Angaben im Datensatz erfordern einen Teildatensatz <Bogen.name> ("<i><Bogen.bezeichnung></i>"). Dieser fehlt.</i> | Wenn die Existenzbedingung eines Kindteildatensatzes im zugehörigen Mutterteildatensatz erfüllt ist, aber kein Kindteildatensatz vorhanden ist (Abschnitt B 2.3.2). |
| | <i>Die Angaben im Datensatz lassen keinen weiteren Teildatensatz <Bogen.name> ("<i><Bogen.bezeichnung></i>") zu, obwohl ein solcher übermittelt wurde.</i> | Wenn die Existenzbedingung eines Kindteildatensatzes im zugehörigen Mutterteildatensatz nicht erfüllt ist, aber trotzdem ein Kindteildatensatz existiert (Abschnitt B 2.3.2). |

Die Fehlermeldungen, die nicht von der Tabelle Regel in der Access-Datenbank abgedeckt sind, sind in der Tabelle Fehlermeldung hinterlegt. Diese sollen eine Standardisierung von Fehlermeldungen und klaren Bedeutungen unterstützen.

Prüfungsprozess und Ergebnisprotokollierung

Ausgangspunkt ist eine prinzipiell offene Anzahl von Prüfungen. Welche Prüfungen konkret durchgeführt werden, ist abhängig vom Datenfluss. 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. Eine prüfende Einrichtung trägt sich als <validation_provider> in die entsprechende Auflistung ein und dokumentiert dann ihre durchgeführten Prüfungen in der Auflistung <validation_item> (Ausnahme: Prüft der Leistungserbringer, sind in keinem Fall die Daten des Leistungserbringers einzutragen. In diesem Fall wird der Softwareanbieter als <validation_provider> eingetragen.

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:

- Das Ergebnis in Bezug auf das gesamte Dokument muss im <header> eingetragen werden.
- Das Ergebnis der Fallprüfung muss in <case_admin> eingetragen werden, sofern der Status dieser Prüfung nicht OK ist (siehe auch unten).
- 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.

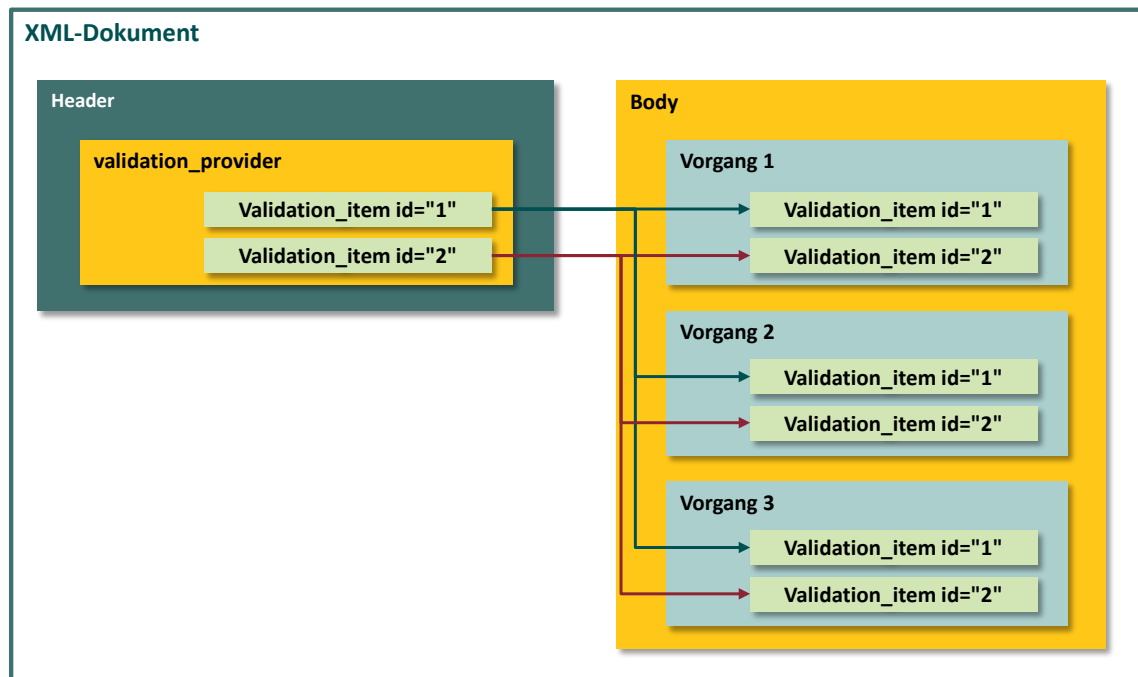


Abbildung 9: Beziehungen zwischen `<validation_item>` im header und `<validation_item>` im body über die id

Zur Veranschaulichung dieser Konstruktion soll im Folgenden eine Analogie aus dem relationalen DB-Modell bemüht werden. So kann die Dokumentenebene als Master-Tabelle und die Fallebene als Detail-Tabelle bezeichnet werden. Letztere enthält die zum Master gehörenden Detail-Datensätze, auf die über das Attribut „ID“ referenziert werden kann (Abbildung 9).

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.

Im Fall einer Auffälligkeit muss wenigstens eine standardisierte Fehlermeldung im `<status>` Element der von der Prüfung betroffenen Ebene (Vorgang oder Dokument) untergebracht werden (Abschnitt B 2.8.3).

Beziehungen Vorgangsebene/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 auf die Ergebnisse der durchgeführten Prüfungen bezieht.

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:

Ein `ERROR` in einer der Prüfungen verhindert die Weiterverarbeitung des geprüften Objekts. Eine oder mehrere Auffälligkeiten (`WARNING`, `ERROR`) auf Fallebene bedeuten ein `WARNING` in dem korrespondierenden Eintrag auf Dokumentenebene. 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 `ERROR` gesetzt werden. Der Status (`<status_case>/<status_document>`) eines Objekts kann nicht „besser“ sein als sein schlechtestes Prüfergebnis.

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

Positive Prüfungen werden auf Fallebene nicht protokolliert. Wenn eine Prüfung auf Fallebene keine Auffälligkeit feststellt, wird dieses Ergebnis nicht dokumentiert. Das Ergebnis OK ist implizit anzunehmen, wenn kein Fehler protokolliert wurde.

Auf Fallebene wird nur dann protokolliert, wenn bei der Prüfung eine Auffälligkeit festgestellt wurde. Falls eine Auffälligkeit festgestellt wird, sind die Schritte 3 bis 6 abzuarbeiten.

- (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, bei denen das Ergebnis der Prüfung schlechter ist als der aktuelle Status des Falls, wird der Status mit dem Ergebnis der Prüfung aktualisiert.
- (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.



Achtung

Rückschlüsse auf den Leistungserbringer

Der Leistungserbringer/die Krankenkasse darf aus datenschutzrechtlichen Gründen nicht als Validation-Provider im Dokument auftauchen. Wenn das Datenprüfprogramm beim Leistungserbringer ausgeführt wird, ist als Validation-Provider der Softwarehersteller des QS-Programms zu verwenden, damit keine Rückschlüsse auf den Leistungserbringer gezogen werden können. Um dieses auch über das Schema abzusichern, ist im Attribut `/validation_provider@function` der Bezeichner „Leistungserbringer“ nicht zulässig.

Rückprotokoll – Bereitstellung eines XSLT für die Transformation

Für alle Leistungserbringer/Krankenkassen, die keine Möglichkeit haben, das Datenflussprotokoll in die QS-Dokumentationssoftware zu importieren und in geeigneter Form darzustellen, stellt das IQTIG ein XSLT-Skript zur Verfügung, das die Darstellung der XML-Protokolle in Browsern in vereinfachter HTML-Darstellung ermöglicht.

Lokale Transformation (Empfehlung)

Die einfachste und sicherste Variante ist das Transformieren vom Browser selbst. Dafür soll das XML-Protokoll im Browser (z. B. Internet-Explorer, Firefox) geöffnet werden.

Die Voraussetzung für die fehlerfreie Umwandlung ist,

- das lokale Ablegen des dazugehörigen XSLT-Skripts
- der entsprechende Link zum XSLT-Skript muss in das XML-Protokoll unmittelbar nach der ersten Zeile eingetragen werden:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<?xml-stylesheet type="text/xsl" href="protocol.xsl"?>
<root xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemalocation="urn:gba:sqg ../../../../interface_LE/2017_BAS_dv_1.0_Export.xsd"
  xmlns="urn:gba:sqg" container_version="2.0" content_version="1.0">
  <header>
  </header>
  <body>
  </body>
</root>
```

Abbildung 10: Aufnahme des XSLT-Pfads in das XML-Protokoll

Der Eintrag der Referenz (siehe Pfeil) im Rückprotokoll erfolgt durch die DAS.

In der HTML-Darstellung wird der Inhalt des Protokolls besser lesbar und durch die Kategorisierung der Prüfergebnisse unter Verwendung einer Ampelanzeige

- Rot für fehlerhafte Datensätze (ERRORS)
- Gelb für Datensätze mit Hinweisen (WARNINGS)
- Grün für fehlerfreie Datensätze (OK)

übersichtlicher gestaltet.

| Dokument-Status: ERROR | | | |
|---|---------------------|--|-------------------|
| | | | » Details ein/aus |
| Datenannahmestelle BAQ (DAS10000) DAS, 80331 München | | tel: 089 000000 0 fax: (+49)089/000 00-00 datenannahmestelle@das.de | |
| Validierung 1 | 2016-05-22T15:14:48 | Spezifikation | OK |
| Vertrauensstelle VST-PSN (bu20000) | | tel: - fax: - vst-psn@vertrauensstelle-gba.de | |
| Validierung 2 | 2016-05-22T16:15:13 | PID_Pseudonym | WARNING |
| Bundesauswertestelle IQTIG (BU10000) Katharina-Heinroth-Ufer 1; 10787 Berlin | | tel: (+49)030/585826-0 fax: (+49)030/585826-999 verfahrensupport@iqtig.org | |
| Validierung 3 | 2016-05-21T17:41:14 | Spezifikation | ERROR |

Abbildung 11: HTML-Darstellung nach einer XSLT-Transformation am Beispiel einer QS-Übertragung

Ausschließlich für die Erfassungssoftware im Krankenhaus gilt, dass diese in der Lage sein muss, die Fehlermeldungen und Warnungen der datenentgegennehmenden Stelle einzulesen und den Anwender durch eine möglichst komfortable Nachbearbeitungsfunktion für die betreffenden Datensätze zu unterstützen.

5.3 Datenfluss

Die Qualitätssicherung der Leistungserbringung gemäß §§ 136 ff. SGB V erfordert unterschiedliche Datenflüsse. Die Zuordnung von Datenpaketen zu einem Datenfluss ist vom Abrechnungskontext, von den zu erhebenden Modulen sowie davon abhängig, ob patientenidentifizierende Daten (PID) erhoben, exportiert und pseudonymisiert werden müssen. Unter den Leistungsbereichen der QSKH-RL gibt es sowohl Daten mit als auch ohne PID. Verfahren gem. Qesü-RL und Verfahren gemäß DeQS-RL beinhalten immer PID (Übersicht über die Verfahren, siehe Tabelle 19). In einigen Verfahren ist auch die Durchführung einer Pseudonymisierung von leistungserbringeridentifizierenden Daten bei der Zuordnung zu beachten.

5.3.1 Datenfluss der QS-Daten

Module ohne PID (ohne FU²⁸):

- **indirekte Verfahren (iv):** Datensätze zu Leistungsbereichen gem. QSKH-RL (stationär erbrachte Leistungen) werden vom Leistungserbringer über die auf Landesebene als Datenannahmestellen (DAS) beauftragten Stellen an die Bundesauswertungsstelle (BAS) übermittelt.
- **direkte Verfahren (dv):** Die Daten werden vom Leistungserbringer direkt an die BAS als zuständige DAS gesandt.

Module mit PID (mit FU²⁹):

Datensätze zu Leistungsbereichen gem. Qesü-, DeQS- oder QSKH-RL (stationär sowie ambulant erbrachte Leistungen bzw. Leistungen verschiedener Abrechnungskontexte) werden über die zuständigen DAS und die VST an die BAS übermittelt. Die zuständige(n) DAS

- für kollektivvertraglich erbrachte Leistungen (außer CHE_KV) sind die Kassenärztlichen Vereinigungen (KV),
- für selektivvertraglich erbrachte Leistungen ist die Datenannahmestelle (DAS-SV),
- für ambulant oder stationär erbrachte Leistungen durch die Krankenhäuser und für kollektivvertraglich erbrachte Leistungen im Modul CHE_KV sind die Landesgeschäftsstellen für Qualitätssicherung (LQS) bzw. Landeskrankenhausesellschaften (LKG).

²⁸ „Ohne FU“ meint hier, dass **kein** Follow-up mittels PID-Verknüpfung stattfindet. Die ebenfalls als Follow-up-Module bezeichneten Module der Leistungsbereiche der Transplantationen fallen hinsichtlich des Datenflusses unter die hier aufgeführte Kategorie „Module ohne PID“ (Unterkategorie „direkte Verfahren“).

²⁹ Analog zu Fußnote 27 bedeutet „mit FU“ an dieser Stelle ausschließlich ein Follow-up mittels PID-Verknüpfung. Die Follow-up-Module der Leistungsbereiche der Transplantationen sind somit **nicht** in dieser Kategorie inbegriffen. Sie fallen folglich **nicht** in die hier aufgeführte Kategorie „Module mit PID“.

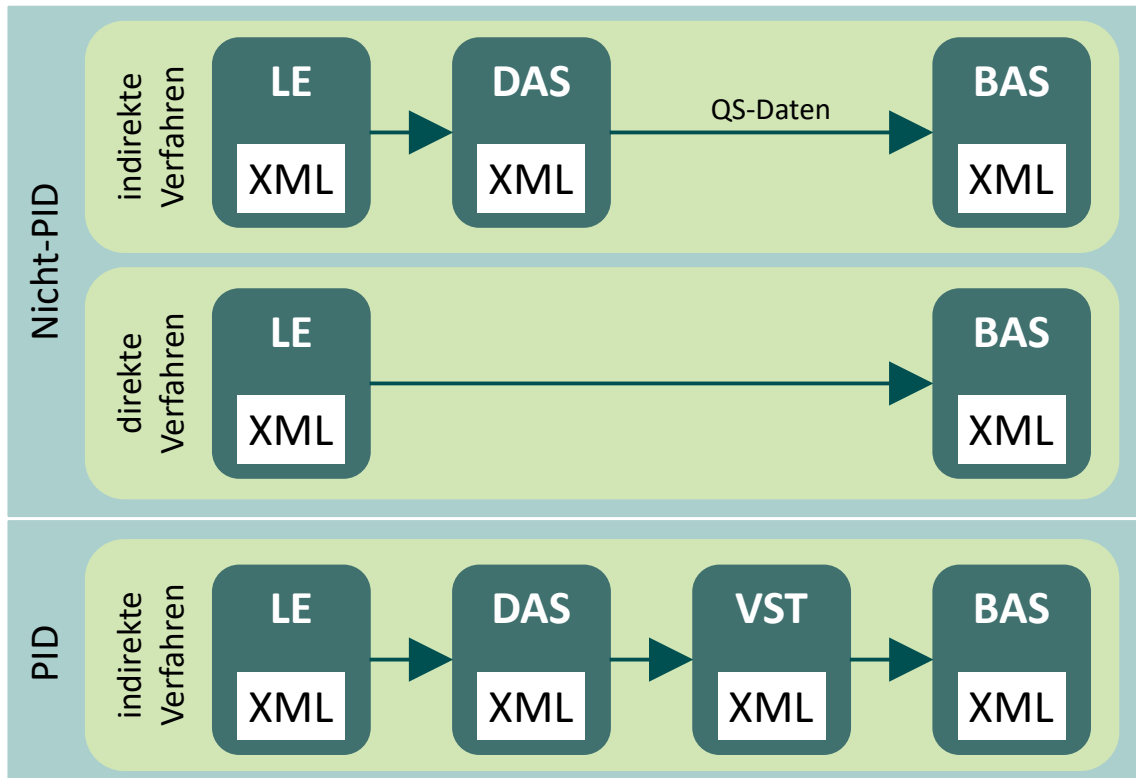


Hinweise

- Belegärztlich erbrachte Leistungen werden sowohl hinsichtlich der Dokumentationsverantwortung als auch hinsichtlich der Auswertung im Rahmen der QSKH-RL dem Krankenhaus zugeschlagen
- Belegärztlich erbrachte Leistungen werden im Rahmen der Qesü-RL und der DeQS-RL dem Belegarzt (als kollektivvertraglich erbrachte Leistung) zugeschlagen. Leistungen, die durch Krankenhausärzte im Rahmen einer Ermächtigung zur Teilnahme an der vertragsärztlichen Versorgung der Versicherten nach §116 SGB V erbracht werden, sind als kollektivvertraglich erbrachte Leistungen zu werten.
- Gemäß DeQS-RL gilt im Verfahren Cholezystektomie die Besonderheit, dass die Erhebung und Weiterleitung der belegärztlichen Daten an die Datenannahmestellen nach Teil 1 § 9 Absatz 1 Satz 3 (LQS/LKG) erfolgen soll.

Alle Module, die PID enthalten, werden durch die DAS nach der Plausibilitätsprüfung und ggf. der Pseudonymisierung der leistungserbringeridentifizierenden Daten an die Vertrauensstelle als Pseudonymisierungsstelle (VST-PSN) gesendet, die die PID pseudonymisiert. Bei Verfahren gemäß Qesü-RL und nach DeQS-RL werden auch diejenigen Datensätze, die sich auf Nicht-GKV-Versicherte beziehen und somit kein PID-Element enthalten, über die Vertrauensstelle weitergeleitet.³⁰ In diese Kategorie von PID-Modulen fallen alle Leistungen aus dem Bereich der Qesü-RL, aus dem Bereich der DeQS-RL sowie die sogenannten Follow-up-Module (mit PID) aus dem Bereich der QSKH-RL (Tabelle 19).

Die folgende Abbildung gibt einen Überblick über die verschiedenen Datenflüsse:



³⁰ Derzeit gibt es derzeit gemäß Qesü-RL nur Verfahren, die sich auf GKV-Versicherte beziehen.

Abbildung 12: Übersicht der Datenflüsse direkte/indirekte PID-/Nicht-PID-Verfahren

Aus technischer Sicht sind die drei Exportverfahren gleich: Mit den Daten eines oder mehrerer abgeschlossener Module wird vom Dokumentationssystem eine XML-Exportdatei erzeugt, in eine Transaktionsdatei komprimiert und als verschlüsselter E-Mail-Anhang oder über einen sonstigen sicheren Übertragungsweg (z. B. KV-Connect) an die zuständige DAS übermittelt.

Der Leistungserbringer muss beim Export darauf achten, dass nur diejenigen Module in einem Dokument zusammengefasst werden, die demselben Datenfluss zugeordnet sind. Welche Module zu einem Datenfluss gehören, kann Tabelle 19 entnommen werden. Diese Zuordnung kann auch der Abfrage ExportmoduleGemeinsam der Spezifikationsdatenbank QSDOK entnommen werden.

Für die in der Tabellenübersicht als „zusammengefasst exportierbare Module“ aufgeführten Module kann es abweichend aufgrund von Absprachen zwischen DAS und Dienstleistern regional festgelegte Aufteilungserfordernisse geben. Die Aufteilung ergibt sich aus einer Überlagerung der hier dargestellten Matrix mit der Matrix der Datenservices, in denen ggf. modulspezifisch unterschiedliche DAS aufgeführt werden (Abschnitt B 2.8.2). Module, die an unterschiedliche Datenservices geschickt werden sollen, dürfen also auch dann nicht in einem Dokument verschickt werden, wenn sie in der untenstehenden Tabelle als „zusammengefasst exportierbare Module“ aufgeführt werden.

Die DAS müssen Dokumente zurückweisen, wenn Datensätze, die nicht in ihren Zuständigkeitsbereich fallen, oder Mischlieferungen von Modulen unterschiedlicher Datenflüsse enthalten sind.

Im Folgenden werden Beispiele zu Datenlieferungen, die nicht von einer DAS auf Landesebene akzeptiert werden dürfen, dargestellt.

Wegen der abweichenden Regelung zur Pseudonymisierung der LE-Daten in der Qesü-RL und in der DeQS-RL (verfahrensbezogene Pseudonymisierung), sind die Qesü-bzw. die DeQS-Verfahren in getrennten XML-Dateien zu exportieren. So sind Mischlieferungen von QSKH-Verfahren und Qesü-Verfahren (z. B. HEP und PCI) und DeQS-Verfahren (z. B. CHE) als auch Mischlieferungen von unterschiedlichen Qesü-Verfahren oder unterschiedlichen DeQS-Verfahren von den Datenannahmestellen abzulehnen.

Beispiele für abzulehnende Datenlieferungen:

- Dateien aus einem direkten Verfahren werden an eine DAS auf Landesebene geschickt; zuständige DAS in direkten Verfahren ist jedoch die Bundesauswertungsstelle.
- Dateien aus Nicht-PID- und PID-Modulen werden in einer Datenlieferung verschickt: Sie sind von der DAS abzulehnen, weil sie sich hinsichtlich der Verarbeitungswege unterscheiden, obwohl sie zu Beginn des Datenflusses von derselben DAS entgegengenommen werden.
- Dateien aus PID-Modulen (PCI, KEP), deren Grundlage aber unterschiedliche Richtlinien (QSKH/Qesü/DeQS) sind, werden in einer Datenlieferung verschickt: Sie sind von der DAS abzulehnen, weil sie sich hinsichtlich der LID (mit und ohne LE-Pseudonymisierung) unterscheiden, obwohl die Datenübermittlung sich auf denselben Datenfluss bezieht.

- Dateien aus PID-Modulen (PCI_LKG, NWIF) werden in einer Datenlieferung verschickt: Sie sind von der DAS abzulehnen, weil sie gemäß Qesü-RL und DeQS-RL einer verfahrensspezifischen und somit unterschiedlichen LE-Pseudonymisierung unterliegen. Eine Aufspaltung dieser Daten in mehr als ein Data-Container innerhalb einer Datei ist nicht zulässig, um die Zusammenführbarkeit unterschiedlicher Pseudonyme in der BAS zu verhindern.

Tabelle 19: Übersicht über die Exportverfahren

| Richtlinie | Exportverfahren | zusammengefasst exportierbare Module | LE Art | Datenannahmestellen |
|--------------------|--|--|---|---------------------|
| QSKH | Direkte Verfahren (Nicht-PID dv) | HCH, HTXM, HTXFU, LLS, LLSFU, LTX, LTXFU, LUTX, LUTXFU, NLS, NLSFU, PNTX, PNTXFU | Krankenhaus | BAS |
| | Indirekte Verfahren (Nicht-PID iv) | 10/2, 15/1, 18/1, 17/1, PNEU, DEK | Krankenhaus | LQS |
| | Indirekte Verfahren (PID iv) QSKH ohne LE-Pseudonymisierung ³¹ , Qesü mit LE-Pseudonymisierung und DeQS mit LE-Pseudonymisierung | HEP, KEP, 09/1, 09/2, 09/3, 09/4, 09/5, 09/6, 16/1, NEO | Krankenhaus | LQS |
| QESÜ ³² | | PCI_KV | Arztpraxis/MVZ ³³ (kollektivvertraglich) | KV |

³¹ Bei Verfahren gemäß QSKH-RL erfolgt ebenfalls eine LE-Pseudonymisierung, jedoch wird diese erst durch die Bundesauswertungsstelle (IQTIG) durchgeführt.

³² Die Exportmodule unterschiedlicher Verfahren im Zuständigkeitsbereich der Qesü-RL dürfen nicht gemeinsam exportiert werden, weil hier die Pseudonymisierung der Leistungserbringer und der Patienten getrennt nach Verfahren erfolgen muss. Hierbei ist zwischen Exportmodulen und Verfahren zu differenzieren: Die Exportmodule PCI_KV, PCI_SV und PCI_LKG gehören zu demselben Verfahren (PCI), sodass über diese Exportmodule hinweg dieselbe Pseudonymisierung von Leistungserbringern und Patienten erfolgt.

³³ Die Arztpraxis/MVZ ist hierbei nicht als Ort der Leistungserbringung zu verstehen, sondern als zuständige/verantwortliche Einrichtung.

| Richtlinie | Exportverfahren | zusammengefasst exportierbare Module | LE Art | Datenannahmestellen |
|------------|-----------------|--------------------------------------|--|---------------------|
| DeQS | | | <ul style="list-style-type: none"> ▪ ambulant³⁴ ▪ stationär³⁵ | |
| | | PCI_SV | Arztpraxis/MVZ (selektivvertraglich) ³⁶ <ul style="list-style-type: none"> ▪ ambulant | DAS-SV |
| | | PCI_LKG | Krankenhaus <ul style="list-style-type: none"> ▪ vorstationär³⁷ ▪ stationär³⁸ ▪ ambulant³⁹ | LQS/LKG |
| | | NWIF | Krankenhaus <ul style="list-style-type: none"> ▪ stationär⁴⁰ | LQS/LKG |
| | | CHE_LKG | Krankenhaus stationär ⁴¹ | LQS/LKG |
| | | CHE_KV | Belegarzt (kollektivvertraglich) stationär ⁴² | LQS/LKG |
| | | CHE_SV | Arztpraxis/MVZ (selektivvertraglich) | DAS-SV |

5.3.2 Datenfluss der Rückprotokolle

Mit Ausnahme der direkten Verfahren ohne PID verläuft der Datenfluss mit einer Empfangsbestätigung vom jeweiligen Empfänger (Abschnitt A 5.2.2) zum jeweiligen Absender.

Mit dem Abschluss der Datenverarbeitung in der BAS erfolgt eine weitere Rückprotokollierung – das sog. Datenflussprotokoll – durch die BAS über die DAS (indirekte Verfahren mit und ohne PID) bzw. direkt (direkte Verfahren) zum Leistungserbringer (siehe Abbildung 13). Die Vertrauensstelle (VST-PSN) wird in diesem Datenfluss übersprungen.

³⁴ § 295 SGB V, § 116 SGB V

³⁵ § 121 Abs. 3 SGB V

³⁶ §§ 73b und c SGB V

³⁷ § 115a SGB V

³⁸ § 301 SGB V; § 121 Abs. 5 SGB V; §§ 140a ff. SGB V

³⁹ § 115b, § 116b, § 117, § 118 und § 119 SGB V

⁴⁰ § 301 SGB V; § 121 Abs. 5 SGB V; §§ 140a ff. SGB V

⁴¹ § 301 SGB V; § 121 Abs. 5 SGB V; §§ 140a ff. SGB V

⁴² § 121 Abs. 3 SGB V

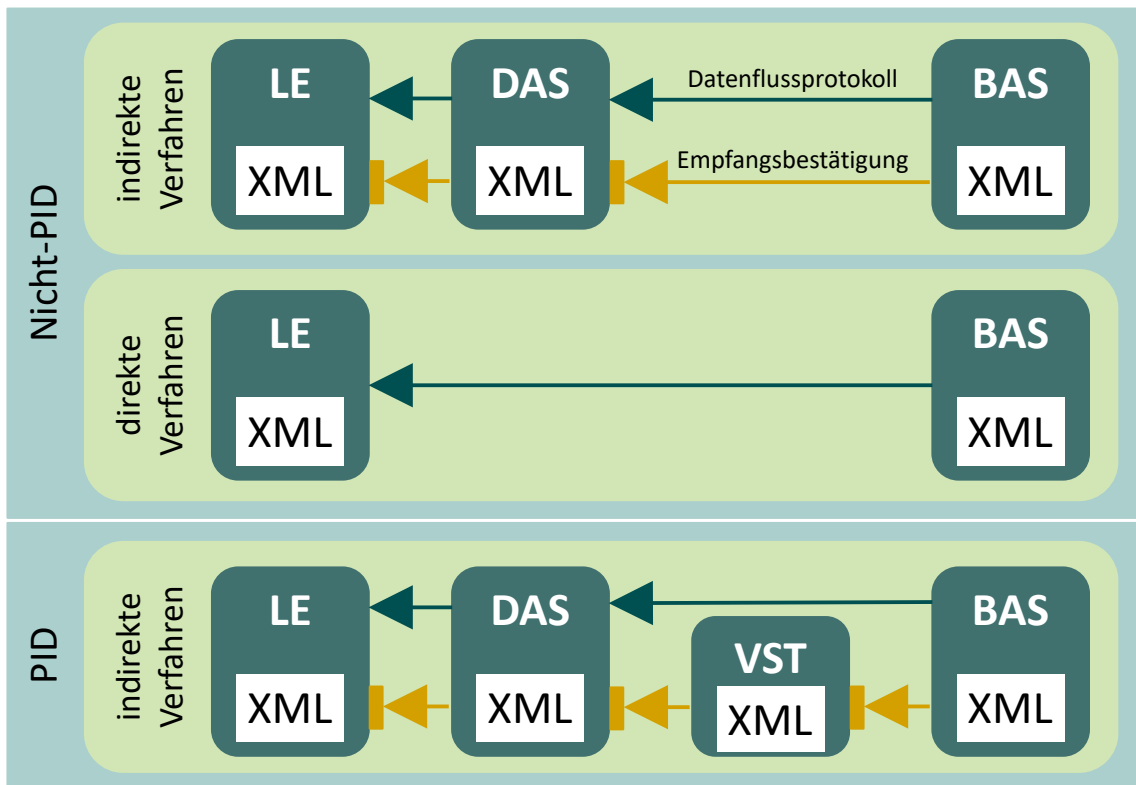


Abbildung 13: Übersicht der Datenflüsse der Rückprotokollierung

B Komponenten

In diesem Kapitel werden die einzelnen Komponenten der Spezifikation beschrieben. Ein Spezifikationspaket bildet die Gesamtheit seiner Spezifikationskomponenten ab.

Spezifikationskomponenten

Ein Spezifikationspaket kann sich aus folgenden Komponenten zusammensetzen:

- **TechDok** – bezeichnet alle technischen Dokumentationen; diese geben detaillierte Erläuterungen zur Funktionsweise und Verwendung der einzelnen Komponenten. Da es verschiedene spezifisch auf eine jeweilige Zielgruppe hin verfasste TechDoks gibt, wird die Zielgruppe gleich im Kürzel vermerkt.
 - **TechDok_LE** – Technische Dokumentation für Leistungserbringer
 - **TechDok_DAS** – Technische Dokumentation für die Datenannahmestelle
- **QSDOK** – bezeichnet die Access-Datenbank, in der die QS-Dokumentation spezifiziert wird.
- **QSF** – bezeichnet die Access-Datenbank, in der der QS-Filter spezifiziert wird.
- **Schema** – auf der Komponentenebene ist dies eine ZIP-Datei, die die Versionierung und vollständige Bezeichnung enthält. Sie enthält einzelne XML-Schemata, die festlegen, in welcher Struktur XML-Daten an Schnittstellen im Datenfluss vorliegen müssen.
- **Precheck** – auf der Komponentenebene ist dies eine ZIP-Datei, die Schemata für alle administrativen Daten enthält, die ab dem Export dieser Spezifikation gültig sind. Diese Schemata sind ausschließlich für die Verarbeitung der Datenservices der DAS, VST und BAS relevant.
- **Ausfuellhinweise** – auf der Komponentenebene ist dies eine ZIP-Datei, die die Versionierung und vollständige Bezeichnung enthält. Sie enthält einzelne HTML-Dateien für jedes Modul, die mit den Kürzeln der einzelnen Module benannt sind.
- **Anwenderinformationen** – auf der Komponentenebene ist dies eine ZIP-Datei, die die Versionierung und vollständige Bezeichnung enthält. Die ZIP-Datei enthält einzelne HTML-Dateien für jedes im QS-Filter definiertes Modul, die mit den Kürzeln der einzelnen Modulauslöser benannt sind.
- **Dokuboegen** – auf der Komponentenebene ist dies eine ZIP-Datei, die die Versionierung und vollständige Bezeichnung enthält. Sie beinhaltet die Dokumentationsbögen als einzelne PDF-Dateien für jedes Modul, die mit den Kürzeln der einzelnen Module benannt sind. Die Dokumentationsbögen bilden einige wichtige Datenbankinhalte⁴³ ab.
- **Vorlagen_Sollstatistik** – auf der Komponentenebene ist dies eine ZIP-Datei, die die Versionierung und vollständige Bezeichnung enthält. Sie enthält WORD-Dateien – das Formular zur Methodischen Sollstatistik für Krankenhäuser, das Formular zur Vorlage bei den Budgetverhandlungen für Krankenhäuser und das Formular zur Sollstatistik für selektivvertraglich erbrachte Fälle durch niedergelassene Leistungserbringer.

⁴³ Die Papierform ist hier nur als eine Abbildung des Eingabeformulars oder der Eingabemaske zu verstehen. Verbindlich sind daher nur die Inhalte der Datenbank zur QS-Dokumentation.

- **Anwendungsfaelle_mds** – ist eine PDF-Datei mit Erläuterungen zum Minimaldatensatz. In diesem Dokument werden Beispiele für Ausnahmesituationen beschrieben, in denen ein Behandlungsfall durch den QS-Filter als dokumentationspflichtig ausgelöst wird, obwohl eine abschlussfähige Dokumentation nicht möglich ist.
- **Risikostatistik** – bezeichnet die Access-Datenbank, in der die Risikostatistik spezifiziert wird.
- **Anwenderinformation_DEK_RS** – auf der Komponentenebene ist dies eine ZIP-Datei, die die Versionierung und vollständige Bezeichnung enthält. Die ZIP-Datei enthält eine HTML-Datei.
- **ÜbersichtAenderungen** – ist eine PDF-Datei mit der Übersicht über die Änderungen in der Spezifikation zu allen Vorversionen des Erfassungsjahres und zur letzten Version des Vorjahres. Hier werden die Änderungen übersichtlich zusammengefasst und erläutert.
- **erginformationen** – auf der Komponentenebene ist dies eine ZIP-Datei, die die Versionierung und vollständige Bezeichnung enthält. Die ZIP-Datei beinhaltet Java-Code für ein Programm und eine Testklasse mit Testfällen.
- **Protocol** – bezeichnet das Stylesheet zur Darstellung des Rückprotokolls (XSLT).
- **Komponentenliste** – bezeichnet die Auflistung aller im Spezifikationspaket enthaltenen Komponenten mit Bezug zu Versionsnummer und Veröffentlichungsdatum (CSV).

Hilfsprogramme werden ggf. ebenfalls als Komponenten in ein Spezifikationspaket aufgenommen. Hier sind zur Zeit der XPack, der TPack, das Datenprüfprogramm und das Pseudonymisierungsprogramm zu nennen. Bei der Angabe der Betriebsart und des Exportformats gelten die gleichen Abkürzungen wie bei den Spezifikationspaketen. Diese Angaben erfolgen aber nur dann, wenn sich die Komponenten durch diese Merkmale unterscheiden.

- **Verschlüsselungsprogramme** – ist auf Komponentenebene eine ZIP-Datei, die die Versionierung der Komponente trägt. Enthalten sind der X- und der TPack sowie die zum Veröffentlichungszeitpunkt aktuellen öffentlichen Schlüssel der Datenservices im Datenfluss.
- **Datenprüfprogramm** – ist auf Komponentenebene eine ZIP-Datei, welche die Bestandteile des Datenprüfprogramms enthält.
- **PSP** – ist auf Komponentenebene eine ZIP-Datei, die neben dem Pseudonymisierungsprogramm ein Informationsmerkblatt beinhaltet.

V<Versionsnummer>: Die Versionierung erfolgt in ganzen Zahlen, die zweistellig angegeben sind (unter 10 mit einer vorstehenden 0, z. B. V01).

Spezifikationsdatenbanken

Als Access-Datenbanken zur Verfügung gestellte Spezifikationskomponenten dienen der (automatisierten) Erstellung von Software für QS-Filter, QS-Dokumentation sowie für die Soll- und Risikostatistik. Folgende Spezifikationskomponenten werden als Access-Datenbanken (MS Access 2007-2013) zur Verfügung gestellt:

Datenbank zur QS-Dokumentation (QSDOK)

Die Datenbank zur QS-Dokumentation dient der Spezifikation von Datenerhebung und -erfassung unter Berücksichtigung von Plausibilitätsprüfungen und zu exportierenden Datenfeldern.

Einige wichtige Datenbankinhalte werden über die Dokumentationsbögen (Dokubögen) abgebildet.

Folgende Spezifikationskomponenten ergänzen die Datenbank um Hinweise und Informationen:

- Die Ausfüllhinweise (Ausfuellhinweise) dienen als Hilfestellung bei der Dokumentation durch den Anwender. Die Namen der HTML-Dateien für einzelne Datenfelder sind in der Datenbank (`BogenFeld.ahinweis`) hinterlegt.
- Ergänzende Informationen (erginformationen) erläutern die in der Datenbank definierte Syntaxfunktion gewichtsPerzentile.

Die Datenbank zur QS-Dokumentation ist in Kapitel B 2 erläutert.

Datenbank zum QS-Filter (QSF)

Die Datenbank zur QS-Dokumentation dient der Spezifikation zur Auslösung von Dokumentationsmodulen und der Sollstatistik. Die Auslösebedingungen pro Modul werden als Übersicht in Form der Anwenderinformationen (Anwenderinformationen) zur Verfügung gestellt.

Die Datenbank zum QS-Filter ist in Kapitel B 1 erläutert.

Datenbank zur Risikostatistik

Die Datenbank zur Risikostatistik ist in Kapitel B 3 erläutert.

Datenbank zu Datenserviceinformationen

Seit der Spezifikation 2019 werden relevante Datenserviceinformationen (Abschnitt 2.8.2) in einer separaten Datenbank gepflegt. Die Datenbank zu Datenserviceinformationen ist keine dem Spezifikationspaket zugehörige Komponente, verfügt jedoch über dieselbe Verbindlichkeit. Da sie eine eigenständige Spezifikationsdatenbank darstellt kann sie außerhalb des Releasezyklus angepasst werden.

Tabellenstruktur der Datenbanken

Die Tabellen und deren Spalten (Attribute) unterliegen einem einheitlichen Namensschema. Erlaubte Zeichen sind die Buchstaben a–z, A–Z und die Ziffern 0–9. Umlaute und Sonderzeichen werden nicht verwendet. Das erste Zeichen eines Namens darf keine Ziffer sein.

Ein Tabellenname beginnt immer mit einem Großbuchstaben und ein Attributname mit einem Kleinbuchstaben. Wenn ein Name aus mehreren Teilen (z. B. Substantiven) besteht, so beginnt jeder nachfolgende Namensteil mit einem Großbuchstaben.

`BasisTyp` (Tabelle)

`idBasisTyp` (Spalte)

Für jede Tabelle ist in der Spezifikation ein Primärschlüssel definiert, der nach folgendem Schema aufgebaut ist:

`id<TabellenName>`

Der Ausdruck in spitzen Klammern ist ein Platzhalter für den Namen der Tabelle. Die meisten Tabellen haben einen einfachen Primärschlüssel vom Typ `AUTOINCREMENT`. Zusätzlich enthalten derartige Tabellen mindestens ein identifizierendes Attribut⁴⁴, welches durch Setzen eines weiteren, eindeutigen Indexes (bestehend aus einem oder mehreren Attributen) definiert ist.

Beispiele:

Identifizierendes Attribut: Attribut `name` in Tabelle `BasisTyp`

Identifizierende Attributkombination: Attribute `code` und `fkSchluessel` in Tabelle `SchluesselWert`

Es gibt auch Tabellen, deren einziger eindeutiger Schlüssel der Primärschlüssel ist. Ein Beispiel ist die Tabelle `MussKann` mit dem Primärschlüssel `idMussKann` vom Typ `TEXT(1)` (entspricht `VARCHAR(1)`). Diese Tabellen sind als einfache „Nachschlagtabellen“ zu interpretieren. Im Fall der Tabelle `MussKann` soll im entsprechenden Fremdschlüsselfeld der verknüpften Detailtabelle durch das Datenbankschema gewährleistet werden, dass nur ein `M` oder `K` eingegeben werden darf.

Die Namen von Fremdschlüsseln sind analog zum Namen der Primärschlüssel aufgebaut:

`fk<FremdTabellenName>`

Die Namensgebung von Primär- und Fremdschlüsseln vereinfacht den Aufbau von komplexeren Abfragen, welche sich über mehrere Entitäten erstrecken (Inklusionsverknüpfungen, Joins).

Die Fremdschlüsselattribute (Namen beginnen mit `fk`) wurden als Datenbankattribute zum Nachschlagen eingerichtet. Zum Beispiel wird beim Fremdschlüsselattribut `fkModul` in der Tabelle `Tds` nicht mehr der Primärschlüssel des jeweiligen Moduls, sondern der Name des Moduls angezeigt.



Hinweis

Diese Änderung betrifft nur die Anzeige, nicht jedoch die Struktur der Datenbank.

Sind zwei Tabellen mehrfach durch Schlüssel-Fremdschlüssel-Beziehungen miteinander verknüpft, so kann der Name eines Fremdschlüssels auch folgendermaßen aufgebaut sein:

`fk<FremdTabellenName><Rolle>`

`<Rolle>` ist der Platzhalter für eine zusätzliche Qualifizierung der Relation.

N-M-Beziehungen werden wie üblich über Verknüpfungstabellen realisiert. In der Spezifikation haben Verknüpfungstabellen gewöhnlich keinen Primärschlüssel⁴⁵, jedoch einen eindeutigen Schlüssel, der über die Fremdschlüsselfelder definiert ist.

Folgende Attribute treten in vielen Tabellen auf:

⁴⁴ Oder sie enthalten eine identifizierende Attributkombination, die einen eindeutigen Schlüssel definiert.

⁴⁵ Hier: Primärschlüssel im Sinne der Access-Definition eines Primärschlüssels. Streng genommen wird über die beiden Fremdschlüssel ein neuer Primärschlüssel definiert.

- `name` ist in der Regel als technischer Name zu verstehen. Zum Beispiel wird `Feld.name` als Variablenname in den Plausibilitätsregeln verwendet.
- `bezeichnung` ist eine kurze Beschreibung. Zum Beispiel ist `TdsFeld.bezeichnung` der Text, welcher ein Feld auf einem Eingabeformular beschreibt.
- `bedingung` enthält einen logischen Ausdruck. Prominentester Vertreter dieses Attributtyps ist das Attribut `bedingung` in der Tabelle `ModulAusloeser`.

1 QS-Filter

Der QS-Filter definiert, unter welchen Bedingungen ein Modul ausgelöst wird. In der Datenbank zum QS-Filter und den dazugehörigen Anwenderinformationen sind die Informationen hierzu hinterlegt. Die QS-Filter-Software entscheidet für jeden Fall im Krankenhaus, welche Module der externen Qualitätssicherung dokumentationspflichtig sind.

1.1 Anmerkungen zur Struktur der Spezifikationsdatenbank für QS-Filter

Die QS-Filter-Spezifikation ist in einer relationalen Datenbank abgelegt. Zurzeit wird sie ausschließlich als Access-Datenbank (MS Access 2007-2013) zur Verfügung gestellt. Der Name der QS-Filter-Spezifikation richtet sich nach folgendem Schema:

<Erfassungsjahr>_BASIS_QSF_V<Versionsnummer>.mdb

<Versionsnummer> bezeichnet die 2-stellige Versionsnummer (z. B. 01).

Beispiel:

Im Erfassungsjahr 2019 ist die QS-Filter-Spezifikation 2019_BASIS_QSF_V01.mdb⁴⁶ gültig. Die Kennung 2019 gilt für das Erfassungsjahr 2019.

Weiterführende Erläuterungen zum Benennungsschema für Spezifikationskomponenten sind der Einleitung in Abschnitt 1.1.2 zu entnehmen.

Die Tabellenstruktur der Spezifikationsdatenbank wird in Abschnitt B Komponenten beschrieben.

Abfragen der Datenbank

Die Abfragen der Access-Datenbank geben einen vereinfachenden Überblick über die Inhalte der Spezifikation:

- QS-Filter-Leistungsbereiche
- Diese Abfrage zeigt Name, Textdefinition, Dokumentationsverpflichtung und QS-Datensatz für alle Leistungsbereiche an.
- Datensatz QS-Filter
- Hier wird die Beschreibung der Struktur des QS-Filter-Eingangsdatensatzes (Teildatensätze z. B. FALL, PROZ, DIAG) sowie des Ausgabedatensatzes (Erweiterung um die Teildatensätze QSMODUL und FEHLER) angezeigt.
- ICD-Listen
- Die ICD-Listen geben die Einschluss- und Ausschlussdiagnosen für jedes Modul an.
- OPS-Listen
- Die OPS-Listen geben die Einschluss- und Ausschlussprozeduren für jedes Modul an.

⁴⁶ Die Versionsnummer der gültigen Spezifikation (z. B. V01, V02, usw.) ist dem zuletzt veröffentlichten Update zu entnehmen.

- Auslösebedingungen
- Diese Abfrage liefert einen Überblick über die in der Spezifikation enthaltenen Auslösebedingungen der QS-Filter-Leistungsbereiche.
- Auslösebedingungen_sektorenübergreifend
- Diese Abfrage liefert einen Überblick über die in der Spezifikation enthaltenen sektorenübergreifenden Auslösebedingungen der QS-Filter-Leistungsbereiche. Ab der Spezifikation 2016 wird hier das Modul PCI angezeigt. Ab der Spezifikation 2017 wird zudem das Modul NWIF angezeigt.
- KonfigurationDelta
- Hier wird ein Überblick der Konfiguration zur Erstellung des Deltas gegeben.
- Schlüsselkodes
- Diese Abfrage liefert eine Übersicht der Schlüssel und der zugehörigen Codes.
- SOLL_Qesue_KV
Diese Abfrage liefert einen Überblick über den Datensatz SOLL_Qesue_KV
- SOLL_DeQS_KV
Diese Abfrage liefert einen Überblick über den Datensatz SOLL_DeQS_KV
- SOLL_Qesue_DeQS_LKG
Diese Abfrage liefert einen Überblick über den Datensatz SOLL_Qesue_DeQS_LKG
- SOLL_Qesue_DeQS_SV
Diese Abfrage liefert einen Überblick über den Datensatz SOLL_Qesue_DeQS_SV
- SOLL_QSKH
- Diese Abfrage liefert einen Überblick über den Datensatz SOLL_Qesue_QSKH

1.2 Grundlegende Tabellen der Datenbank

Dieser Abschnitt beinhaltet die Darstellung der grundlegenden Tabellen der Spezifikationsdatenbank mit ihren zugehörigen Datensätzen ausgehend von den jeweiligen Modulen.

1.2.1 Module (Datensätze der QS-Dokumentation)

In der Tabelle Modul sind Referenzen auf die Module hinterlegt, deren Dokumentationspflicht durch den QS-Filter ausgelöst werden kann.

Hinter jedem Modul verbirgt sich ein Datensatz der korrespondierenden Spezifikation für QS-Dokumentationssoftware.⁴⁷ Diese Datensätze bilden in der Spezifikation für QS-Dokumentation den auszulösenden Dokumentationsbogen der Software ab. Ausgenommen sind z. B. die Zählleistungsbereiche, die der getrennten Darstellung von Kindgruppen eines (Eltern-)Moduls in der Sollstatistik dienen, jedoch keinen eigenen Datensatz auslösen (Abschnitt B 1.4.2). Auch die Exportmodule eines Moduls lösen keinen entsprechenden QS-Datensatz aus.

⁴⁷ Landesweit verpflichtende Module haben keinen Datensatz in der Spezifikation für QS-Dokumentationssoftware, mit Ausnahme von Modul 15/1 in Hessen.

Beispiel:

Der Modulauslöser `PCIKORO` löst den QS-Datensatz `PCI` aus. Der Datensatz wird als Exportmodul `PCI_LKG`, `PCI_KV` oder `PCI_SV` exportiert.

Die Module der Spezifikation für QS-Dokumentationssoftware sind in der gleichnamigen Tabelle definiert. Eine Zuordnung ist über das Attribut `Modul.name` möglich. Die Datensätze der QS-Dokumentation werden in Abschnitt B 2.3.1 beschrieben.

Tabelle 20: Struktur der Tabelle `Modul`

| Feldname | Datentyp | Bemerkung |
|-------------------------------|----------|--|
| <code>idModul</code> | INTEGER | Primärschlüssel |
| <code>name</code> | TEXT | Technischer Name des Moduls (Identifizierendes Attribut) |
| <code>bezeichnung</code> | TEXT | Beschreibender Text für den QS-Datensatz |
| <code>fkSchluesselWert</code> | INTEGER | Verweis auf denjenigen Code des Schlüssels <code>Modul</code> (Tabelle <code>SchluesselWert</code>), welcher dem betreffenden Datensatz zugeordnet ist. Der Code wird in den Dateien <code>SOLLMODUL_QSKH_<Erfassungsjahr>.txt</code> und <code>SOLLMODUL_Qesue_<Erfassungsjahr>.txt</code> der elektronischen Sollstatistik verwendet. |
| <code>fkModulParent</code> | INTEGER | Verweis auf das Eltern-Modul eines Zählleistungsbereichs |
| <code>zaehlLb</code> | BOOLEAN | Wenn WAHR, existiert kein entsprechender QS-Datensatz, es handelt sich um einen „Zählleistungsbereich“ zur separaten Darstellung in der Sollstatistik. |
| <code>ausloeseModul</code> | BOOLEAN | Wenn WAHR, existiert ein entsprechender QS-Datensatz, der ausgelöst werden kann. Dieses Modul kann gleich dem Exportmodul sein, z. B. <code>HCH</code> . |
| <code>exportModul</code> | BOOLEAN | Wenn WAHR, handelt es sich um ein Exportmodul, z. B. <code>PCI_LKG</code> . Das Exportmodul kann ungleich dem im Modulauslöser referenzierten Modul sein, z. B. <code>PCI</code> . |
| <code>direkt</code> | BOOLEAN | handelt es sich um ein direktes Datenexportverfahren? |
| <code>indirekt</code> | BOOLEAN | handelt es sich um ein indirektes Datenexportverfahren? |
| <code>pid</code> | BOOLEAN | handelt es sich um ein Modul zur Follow-up-Erhebung? ⁴⁸ |

⁴⁸ Gemeint ist eine Follow-up-Erhebung mittels PID-Verfahren.

| Feldname | Datentyp | Bemerkung |
|----------|----------|---|
| qskh | BOOLEAN | handelt es sich um ein Modul der QSKH-Richtlinie? |
| qesue | BOOLEAN | handelt es sich um ein Modul der Qesü-Richtlinie? |
| deqs | BOOLEAN | handelt es sich um ein Modul der DeQS-Richtlinie? |

In der QS-Filter-Spezifikation fehlen die Sekundärmodule (z. B. MDS oder HTXFU, NLSFU etc.) der Spezifikation für QS-Dokumentationssoftware, da sie nicht durch QS-Filter-Kriterien ausgelöst werden.

1.2.2 Struktur der Datensatzdefinitionen

Die Module werden über den definierten Modulauslöser als dokumentationspflichtig erkannt, indem die hinterlegte Bedingung mit den im KIS/AIS gespeicherten Daten geprüft wird. Ist die Bedingung erfüllt, wird das Modul ausgelöst. Da die in der Bedingung enthaltenen Felder im KIS/AIS vorliegen müssen, ist der Eingangs- und Ausgangsdatsatz gemäß §301 SGB V und §295 SGB V in der Spezifikationsdatenbank hinterlegt. Die definierten Felder der Datensätze und der Filterbedingungen sind analog zur Spezifikation für QS-Dokumentationssoftware aufgebaut.

In den nachfolgenden Abschnitten dieses Unterkapitels wird die grundlegende Struktur der Tabellen der Datensatzdefinitionen beschrieben und die Definitionen der Datenfelder erläutert.

Datensätze

Jeder Datensatz besteht aus Teildatensätzen, welche ausgehend von einem Basisteildatensatz hierarchisch angeordnet sind. Folgende Datensätze sind definiert:

- QS-Filter-Datensätze (umfasst den Eingangs- und Ausgangsdatsatz)
- QS-Filter-Datensatz für Sollstatistik für Verfahren gemäß QSKH-RL
- QS-Filter-Datensatz zur Sollstatistik für Verfahren gemäß Qesü-RL oder gemäß DeQS-RL mit Krankenhausabrechnung
- QS-Filter-Datensatz zur Sollstatistik für Verfahren gemäß Qesü-RL oder gemäß DeQS-RL für selektivvertragliche Leistungen im ambulanten Bereich
- QS-Filter-Datensatz zur Sollstatistik für Verfahren gemäß Qesü-RL oder gemäß DeQS-RL für kollektivvertragliche Leistungen (Erstellung durch KV)

Für jeden dieser Datensätze wird in der Tabelle Ds ein Eintrag angelegt.

Tabelle 21: Struktur der Tabelle Ds

| Feldname | Datentyp | Bemerkung |
|-------------|----------|---|
| idDs | INTEGER | Primärschlüssel |
| name | TEXT | Technischer Name des Datensatzes (Identifizierendes Attribut) |
| bezeichnung | TEXT | Beschreibender Text |

Teildatensätze

Die Definition von Teildatensätzen befindet sich in der Tabelle `Tds` der Datenbank (Tabelle 22). Jeder Teildatensatz ist eindeutig durch seinen Namen (z. B. `FALL`, `PROZ` oder `DIAG`) charakterisiert.

Tabelle 22: Struktur der Tabelle `Tds`

| Feldname | Datentyp | Bemerkung |
|-----------------------------|----------|---|
| <code>idTds</code> | INTEGER | Primärschlüssel |
| <code>Name</code> | TEXT | Technischer Name des Teildatensatzes (Identifizierendes Attribut) |
| <code>bezeichnung</code> | TEXT | Beschreibender Text |
| <code>fkTds</code> | INTEGER | Optionaler Fremdschlüssel zu einem Mutterteildatensatz |
| <code>fkDs</code> | INTEGER | Bezug des Teildatensatzes zum übergeordneten Datensatz in der Tabelle <code>Ds</code> , z. B. QS-Filter-Datensatz oder QS-Filter-Datensatz für Sollstatistik |
| <code>fkRelation-Typ</code> | TEXT(1) | Relationstyp, bezieht sich auf die Relation zum Mutterteildatensatz: * Eine beliebige Anzahl von Teildatensätzen darf angelegt werden! ? Höchstens ein Teildatensatz darf angelegt werden! + Mindestens ein Teildatensatz muss angelegt werden! 1 Genau ein Teildatensatz muss angelegt werden! |

In der Tabelle `Tds` ist eine Hierarchie der Teildatensätze definiert. Der Ausgangspunkt („root“) für die Teildatensatzhierarchie eines Datensatzes (z. B. QS-Filter-Datensatz) ist immer der Basisteildatensatz (daraus folgt: Teildatensatz⁴⁹ `fkTds` = NULL in der Tabelle `Tds`). Über die in den restlichen Teildatensätzen des Moduls definierten Bezüge zu den Mutterteildatensätzen und Relationstypen lässt sich ein Hierarchiebaum der Teildatensätze aufbauen.

Jeder Datensatz besteht aus

- genau einem Basisteildatensatz
- ggf. einem oder mehreren weiteren Teildatensätzen (= Kindteildatensätze)

⁴⁹ Es darf nur eine Definition eines Basisteildatensatzes existieren.

Beispiele:

Der QS-Filter-Datensatz DATENSATZ_301 besitzt den Basisteildatensatz FALL und die Kindteildatensätze DIAG, PROZ, QSMODUL, FEHLER, ENTGELT und ABTLG.

Die QS-Filter-Datensätze für Sollstatistik (SOLL_QSKH, SOLL_Qesue_DeQS_LKG, SOLL_Qesue_KV, SOLL_DeQS_KV und SOLL_Qesue_DeQS_SV) besitzen jeweils den Basisteildatensatz SOLLBASIS und den Kindteildatensatz SOLLMODUL.

Felder der Teildatensätze

Die Felder eines Teildatensatzes sind in der Tabelle TdsField definiert. Jedes Feld eines Teildatensatzes (kurz TDS-Feld) ist eindeutig charakterisiert durch die Zugehörigkeit zu einem Teildatensatz und zum referenzierten Feld. Jedes Feld darf also nur einmal in einem Teildatensatz verwendet werden. Listenfelder erfordern einen Wert > 1 beim Attribut elemente.

Felder

Ein Feld wird eindeutig über seinen technischen Namen definiert. Jedes Feld hat einen Basistyp (z. B. SCHLUESSEL, NUMSCHLUESSEL, ZAHL). Schlüsselfelder erfordern zusätzlich einen Schlüssel.

Basistypen

Das Hauptmerkmal eines Basistyps ist der technische Typ eines Eingabefeldes (z. B. Zeichenkette, numerischer Typ, Datum usw.). Weiteres Charakteristikum ist die Beschreibung des Eingabeformats. Die Basistypen sind Voraussetzung für die Beschreibung einer formalen Regelsyntax. Das identifizierende Merkmal eines Basistyps ist sein technischer Name (Attribut name).

Tabelle 23: Struktur der Tabelle BasisTyp

| Feldname | Datentyp | Bemerkung |
|-----------------|----------|--|
| idBasisTyp | INTEGER | Primärschlüssel |
| name | TEXT | Technischer Name (muss eindeutig sein) |
| bezeichnung | TEXT | Beschreibender Text |
| formatAnweisung | TEXT | Regulärer Ausdruck für die Formatprüfung |

**Hinweis**

- In Zeichenketten (Basistyp TEXT) sind alle Zeichen des ASCII-Formats mit einem Kode >= 32 erlaubt. Ausgenommen sind das Semikolon, die doppelten Anführungsstriche und Hochkommata.
- Es gibt zwei Arten von Schlüsseln: numerische und nichtnumerische.
- Das Komma trennt die Nachkommastellen, Vorzeichen + und – sind erlaubt.
- Das Datumstrennzeichen ist der Punkt.

Schlüssel

Identifizierendes Merkmal eines Schlüssels ist sein technischer Name. Die meisten Schlüsselcodes sind in der Tabelle `SchluesseWert` definiert. Externe Schlüsselkataloge (z. B. OPS, EBM oder ICD10) sind von den entsprechenden Anbietern zu beziehen.

Tabelle 24: Struktur der Tabelle `Schluesse`

| Feldname | Datentyp | Bemerkung |
|----------------------------|----------|---|
| <code>idSchluesse</code> | INTEGER | Primärschlüssel |
| <code>name</code> | TEXT | Technischer Name (muss eindeutig sein) |
| <code>bezeichnung</code> | TEXT | Beschreibender Text |
| <code>extern</code> | BOOLEAN | Zeigt an, ob der Schlüssel in der Tabelle <code>Schluesse</code> oder in einer externen Tabelle gespeichert ist. |
| <code>externVerweis</code> | TEXT | Verweis auf externe Quelle des Schlüsselkataloges |
| <code>zahl</code> | BOOLEAN | Wenn WAHR, wird das Attribut <code>code</code> der zugehörigen Schlüsselwerte als ganze Zahl interpretiert, ansonsten als Zeichenkette. |

Schlüsselcodes können auf zwei Arten kodiert werden. Wenn das Attribut `zahl = WAHR`, so werden die Codes als ganze Zahl interpretiert. Ansonsten werden sie als Zeichenketten angesehen. In der Syntax der Auslösebedingungen werden die letztgenannten Codes in einfache Hochkommata gesetzt.

Beispiel:

Attribut `zahl` bei Schlüsselfeldern

Felder des Basistyps `NUMSCHLUESSEL` haben das Attribut `zahl = TRUE`.

Felder des Basistyps `SCHLUESSEL` haben das Attribut `zahl = FALSE`. Es handelt sich um alphanumerische Schlüssel, die Buchstaben, Ziffern oder Sonderzeichen verwenden (z. B. `ypN0`). Hierbei kann es sich auch um Werte handeln, die lediglich Ziffern verwenden, aber mit führender Null beginnen (z. B. `01`).

Externe Schlüsselkataloge

Externe Schlüsselkataloge sind über das Attribut `extern` deklariert. Externe Schlüsselkataloge werden nicht vom IQTIG bereitgestellt und daher auch nicht verantwortet.

Hinweise zu den Bezugsquellen sind in der Spalte `externVerweis` zu finden (z. B. <http://www.dimdi.de>). Ein Verweis auf eine Bezugsquelle kann unabhängig vom Attribut `extern` angegeben werden (siehe Schlüssel `EntlGrund`).

**Hinweis**

Das Feld `PERSONENKREIS` referenziert zwei externe Quellen, da dieses Feld sowohl die Schlüsselwerte nach § 301 als auch nach § 295 (KVD-Datensatzbeschreibung) abbildet. Es gelten die folgenden externen Verweise:

- besonderer Personenkreis gemäß Anlage 2 zur Vereinbarung nach § 301 Abs. 3 SGB V (Syntaxvariable `DATENSATZ_301 : FALL : PERSONENKREIS`)
externer Verweis: § 301-Vereinbarung: <http://www.dkgev.de>
- besondere Personengruppe gemäß der Datensatzbeschreibung KVDT (Kassenärztliche Vereinigung-Datentransfer) (Syntaxvariable `DATENSATZ_295 : FALL : PERSONENKREIS`)
externer Verweis: Datensatzbeschreibung KVDT: <http://www.kbv.de>

**Achtung**

Der Softwareanbieter hat dafür Sorge zu tragen, dass die aktuellen externen Schlüsselkataloge in der Software verwendet werden.

Schlüsselwerte

Identifizierendes Merkmal ist hier eine Kombination der Spalten `fkSchluessel` und `code`. Das bedeutet, dass jeder Schlüsselcode innerhalb eines Schlüssels nur einmal vorkommen darf.

Tabelle 25: Struktur der Tabelle *SchlüsselWert*

| Feldname | Datentyp | Bemerkung |
|-------------------------------|----------|--|
| <code>idSchluesselWert</code> | INTEGER | Primärschlüssel |
| <code>fkSchluessel</code> | INTEGER | Fremdschlüssel zur Tabelle Schlüssel |
| <code>code</code> | INTEGER | 0, 1, 2 ... |
| <code>bezeichnung</code> | TEXT | Textliche Definition des Schlüsselwertes |

**Hinweis**

Die Schlüsselwerte lassen sich am einfachsten über die Abfrage `Schlüsselkodes` ermitteln.

1.2.3 OPS-Listen

Die Struktur der OPS-Listen (Tabellen `OPSListe` und `OPSWert`) entspricht den gleichnamigen Tabellen der Spezifikation für QS-Dokumentation (Abschnitt B 2.5.1).

**Achtung**

In dieser Datenbank definierte Codes der OPS-Listen sind nicht noch einmal in der Datenbank für QS-Dokumentationssoftware hinterlegt. Für die Erstellung von Plausibilitätsprüfungen für QS-Dokumentationssoftware sind die hier hinterlegten OPS-Listen verbindlich.

1.2.4 ICD-Listen

Die Struktur der ICD-Listen (Tabellen `ICDListe` und `ICDWert`) entspricht den gleichnamigen Tabellen der Spezifikation für QS-Dokumentation (Abschnitt B 2.5.2).

**Achtung**

In dieser Datenbank definierte Codes der ICD-Listen sind nicht noch einmal in der Datenbank für QS-Dokumentationssoftware hinterlegt. Für die Erstellung von Plausibilitätsprüfungen für QS-Dokumentationssoftware sind die hier hinterlegten ICD-Listen verbindlich.

Umgang mit Kennzeichen zur Diagnosesicherheit

Das Zusatzkennzeichen zur Diagnosesicherheit ist in den ICD-Listen der Spezifikationsdatenbanken nicht enthalten. Das Zusatzkennzeichen ist für ambulante Fälle dennoch zu dokumentieren.

Bei der Auslösung von ambulanten Fällen ist die Auslösung und die Erstellung der Sollstatistik nur mithilfe von gesicherten Diagnosen zulässig.

Beispiel PCI:

Die **Auslösung** eines Falls soll nicht stattfinden, wenn der ICD-Kodes Z52.7 nicht mit A (Ausschluss), V (Verdacht auf) oder Z (Zustand nach) gekennzeichnet ist.

```
format(DIAG; '[a-zA-Z][0-9]{2}(\.[0-9]{1,2})?([#\+*\+!])?([Gg])?([RLBr1b])?') = WAHR UND DIAG KEINSIN
PCI_ICD_EX
```

1.2.5 FAB-Listen

Jede FAB-Liste ist charakterisiert durch ihren Namen (Attribut `name` in Tabelle `FABListe`), welcher nach folgendem Schema gebildet wird:

```
{<TEXT>_}FAB{_<TEXT>}
```

Hinter `<TEXT>` verbirgt sich ein frei wählbarer Name (Erlaubte Zeichen: A-Z, a-z, 0-9, _, Umlaute sind nicht erlaubt). Die `{}`-Ausdrücke sind optional.

Beispiel:

```
FAB_GEB          Geburtshilfliche Fachabteilungen
```

1.2.6 EBM-Listen

Jede EBM-Liste ist charakterisiert durch ihren Namen (Attribut `name` in Tabelle `EBMListe`), welcher nach folgendem Schema gebildet wird:

```
{<TEXT>_}EBM{_<TEXT>}
```

Hinter `<TEXT>` verbirgt sich ein frei wählbarer Name (Erlaubte Zeichen: A–Z, a–z, 0–9, Umlaute sind nicht erlaubt). Die `{}`-Ausdrücke sind optional.

Beispiel:

PCI_KORO_EBM Einschlussgebührenordnungspositionen Koronarangiographie gemäß EBM-Katalog

String-Vergleich bei EBM-Kodes

EBM-Ziffern können auf Landesebene um weitere Zeichen ergänzt werden. Das Feld `Gebührenordnungsnummer gemäß EBM-Katalog [EBM]` hat daher die Feldlänge 7.

In der Spezifikationsdatenbank für QS-Filtersoftware sind die EBM-Kodes in Tabelle `EBMWert` fünfstellig – ohne ergänzenden Zeichen – definiert. Für die Prüfung, ob zwei Codes identisch sind, genügt kein einfacher Stringvergleich. Stattdessen wird ein Stringvergleich der Normcodes⁵⁰ durchgeführt, um die Übereinstimmung zwischen dem dokumentierten Code und dem einer EBM-Liste zu ermitteln.

1.2.7 Entgelt-Listen

Jede Entgelt-Liste ist charakterisiert durch ihren Namen (Attribut `name` in Tabelle `EntgeltListe`), welcher nach folgendem Schema gebildet wird:

```
<Fallart>{_<TEXT>}ENTGELT{_<TEXT>}
```

Hinter `<TEXT>` verbirgt sich ein frei wählbarer Name (Erlaubte Zeichen: A–Z, a–z, 0–9, Umlaute sind nicht erlaubt). Die `{}`-Ausdrücke sind optional.

Beispiel:

STAT_BELEG_ENTGELT belegärztliche Fälle (FALLART = stationär; gemäß [ENTGELTARTSTAT])

1.2.8 Versionsverwaltung

Jede Spezifikationsdatenbank hat eine Version. Die Versionsinformation ist in der Tabelle `Version` der Eintrag, welcher den Attributwert `gueltig = WAHR` besitzt.

⁵⁰ Jeder EBM-Kode lässt sich entweder als Code mit ergänzenden Zeichen (Normcode + ergänzende Zeichen) oder als Code ohne ergänzende Zeichen (Normcode) darstellen.

Die wichtigsten Eigenschaften einer Version sind der Versionsname (Attribut `name`) und der Gültigkeitszeitraum (Attribute `ab` und `bis`). Der Gültigkeitszeitraum einer Version ist in der Regel ein Erfassungsjahr (z. B. Aufnahme zwischen dem 01.01.2019 und dem 31.12.2019).

Versionen können den Status `in Entwicklung` oder `final` haben. Diese Zustände werden in der Nachschlagetabelle `VersStatus` verwaltet. Das Attribut `gueltig` zeigt die gültige Version der Datenbank an. Nur eine einzige Version darf als gültig markiert sein. Darüber hinaus verwaltet die Tabelle `Version` die Historie der Versionen⁵¹: Welche Vorgängerversion vorher gültig war, kann über das Attribut `fkVersion` ermittelt werden.

Tabelle 26: Struktur der Tabelle `Version`

| Feldname | Datentyp | Bemerkung |
|---------------------------|----------|---|
| <code>idVersion</code> | INTEGER | Primärschlüssel |
| <code>name</code> | TEXT | Technischer Name der Version (Identifizierendes Attribut) |
| <code>bezeichnung</code> | TEXT | Beschreibender Text |
| <code>ab</code> | DATUM | Anfang des Gültigkeitszeitraumes |
| <code>bis</code> | DATUM | Ende des Gültigkeitszeitraumes |
| <code>pub</code> | DATUM | Datum der Publikation |
| <code>gueltig</code> | BOOLEAN | gültige Version (nur ein Eintrag darf als gültig markiert sein) |
| <code>fkVersion</code> | INTEGER | Bezug zur Vorgängerversion |
| <code>fkVersStatus</code> | CHAR | Bezug zum Status einer Version (Tabelle <code>VersStatus</code>): E = in Entwicklung F = finale Version S = Service Release zur finalen Spezifikation U = Update der finalen Spezifikation |

Das Attribut `Modul.fkVersion` wird verwendet, um die aktuelle Version eines Moduls kenntlich zu machen. Freiwillige Module werden im Rahmen der inhaltlichen Systempflege⁵² nicht berücksichtigt. Jahreszahlen in Regeln werden für diese Module weiterhin angepasst. Änderungen aufgrund von modulübergreifenden Anpassungen, z. B. Umbenennung technischer Feldnamen oder Ergänzung von Schlüsselwerten sind nicht auszuschließen.

⁵¹ Die Inhalte der Vorversionen sind nicht Teil der aktuellen Spezifikationsdatenbank.

⁵² Die Aktualisierung der ICD- und OPS-Kodes erfolgt auch für nicht verpflichtende Module weiterhin im Rahmen des DIMDI-Umstiegs.

Zuordnung der QS-Filter-Version zu Behandlungsfällen

Die QS-Filter-Software eines Erfassungsjahres wird für Behandlungsfälle verwendet, deren Aufnahme- oder Behandlungsdatum bei stationärer Behandlung bzw. deren Behandlungsdatum bei ambulanten Eingriffen in den definierten Gültigkeitszeitraum fällt. Hierbei wird die QS-Filter-Software im Folgejahr auch noch für Patienten benutzt, welche nach dem in der Datenbank definierten Gültigkeitszeitraum (also nach dem 31.12.) entlassen worden sind.

Abbildung 14 stellt dar, für welche stationären Behandlungsfälle welche Version der QS-Filter-Software verwendet wird. Exemplarische Behandlungsfälle sind durch Querbalken visualisiert, welche den Behandlungsabschnitt vom Aufnahme- bis zum Entlassungsdatum des Krankenhausaufenthalts abdecken. In Abhängigkeit vom Aufnahme- oder Behandlungsdatum wird die passende Version des QS-Filters verwendet. Für ambulante Fälle gilt das Datum der Behandlung für die Zuordnung zur korrekten QS-Filter-Software-Version.

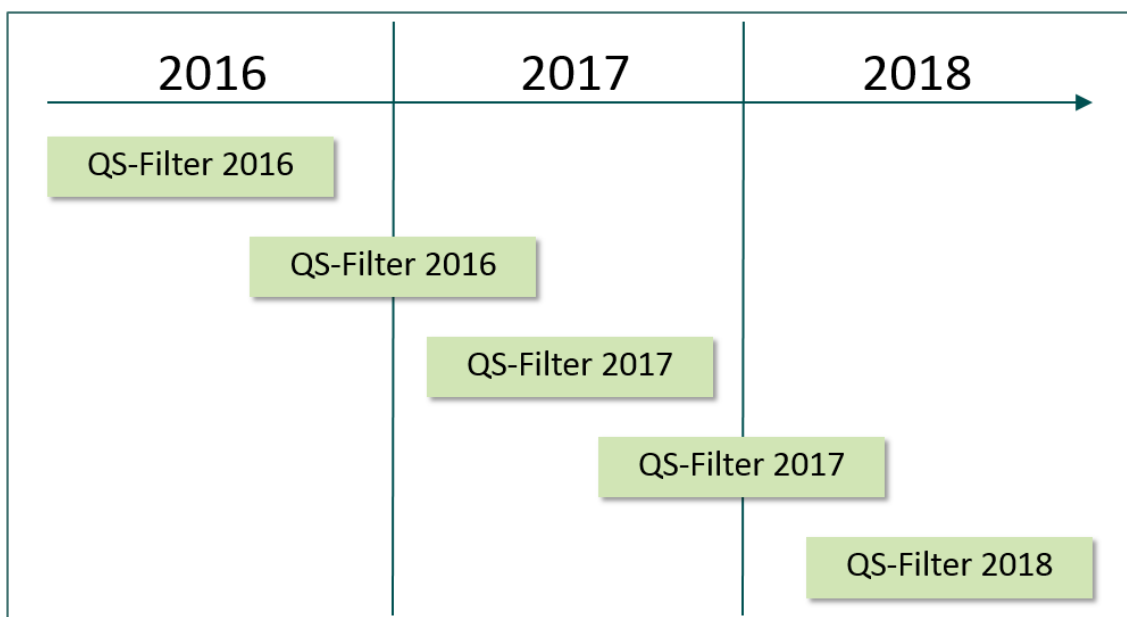


Abbildung 14: Zuordnung der Version des QS-Filters zu den Behandlungsfällen: Kriterium ist das Aufnahme- oder Behandlungsdatum

1.2.9 Meta-Tabellen

In den Tabellen `TabellenStruktur` und `TabellenFeldStruktur` werden die Tabellen und ihre Attribute aufgelistet. Der Inhalt wird automatisch generiert (Abschnitt B 2.7.2).

1.2.10 DB-Änderungen gegenüber der Vorgängerversion

Die Tabellen `DeltaNeu`, `DeltaGeloescht` und `DeltaAttribut` zeigen die Änderungen zur letzten gültigen Spezifikation des Vorjahres und zur Vorversion an (Abschnitt B 2.7.2).

Beispiel:

Die Datenbank 2019_BASIS_QSF_V01 stellt die Änderungen im Vergleich zur letzten gültigen Spezifikationsdatenbank des Vorjahres 2018_BASIS_QSF_V04 dar.

Die Datenbank 2019_BASIS_QSF_V02 stellt zusätzlich die Änderungen im Vergleich zur Vorversion 2019_BASIS_QSF_V01 dar.

1.3 Der QS-Filter-Datensatz

Der QS-Filter-Datensatz umfasst den QS-Filter-Eingangsdatensatz und den QS-Filter-Ausgangsdatensatz. Verbindlich für ein Erfassungsjahr ist die Datenfeldbeschreibung, welche in der gültigen Spezifikationsdatenbank über die Abfrage Datensatz QS-Filter definiert ist. Hier werden ab der Spezifikation 2016 die Datensätze DATENSATZ_301 und DATENSATZ_295 dargestellt. Beide bilden jeweils den entsprechenden QS-Filter-Eingangs- und QS-Filter-Ausgangsdatensatz ab.

Tabelle 27: Ausschnitt der Tabelle Ds

| idDs | name | bezeichnung |
|------|----------------|---|
| 1 | DATEN-SATZ_301 | Datenübermittlung nach §301 Abs. 3 SGB V, ergänzt um die Festlegungen der Vereinbarung nach §120 Abs. 3 SGB V |
| 3 | DATEN-SATZ_295 | Datensatzbeschreibung KVDT [KBV_ITA_VGEX_Datensatzbeschreibung_KVDT], Kassenärztliche Bundesvereinigung |

**Achtung****Datenverlust durch Fehlbezug und falsche Falleinschränkung**

Die Bezeichnungen der Datensätze sowie deren Felder nehmen Bezug auf §301 SGB V und §295 SGB V. Diese Gesetzesgrundlagen beziehen sich lediglich auf den Kontext der Gesetzlichen Krankenversicherung. Der QS-Filter-Datensatz hingegen bezieht sich prinzipiell auf alle behandelten Patienten/Fälle.

1.3.1 Der QS-Filter-Eingangsdatensatz

Der QS-Filter-Eingangsdatensatz ist je nach Richtlinien- und Abrechnungskontext zu wählen:

DATENSATZ_301 enthält den QS-Filter-Eingangsdatensatz nach §301 Abs. 3 SGB V

Die für den QS-Filter maßgeblichen administrativen und medizinischen Datenfelder zur Identifikation dokumentationspflichtiger Fälle im Rahmen der QSKH-RL, im Rahmen der Qesü-RL sowie im Rahmen der DeQS-RL für durch ein Krankenhaus erbrachte Fälle finden sich im Aufnahmedatensatz bzw. in der Entlassungsanzeige des §301-Datenformats⁵³. Daher ist der QS-Filter-Eingangsdatensatz nach §301 Abs. 3 SGB V (DATENSATZ_301) so beschaffen, dass fast alle seine

⁵³ Datenübermittlung nach §301 Abs. 3 SGB V, ergänzt um die Festlegungen der Vereinbarung nach §120 Abs. 3 SGB V

Datenfelder aus der Struktur des §301-Datensatzes abgeleitet werden können. Einzige Ausnahmen sind die Felder PATALTER, ALTERINTAGEN, FALLART (lassen sich aus Inhalten des §301-Datensatzes berechnen) und STANDORT (Bestandteil des Datensatzes nach §21 KHEntgG).

Der QS-Filter-Eingangsdatensatz nach §301 Abs. 3 SGB V besteht aus folgenden Teilen:

- Behandlungsfall (Teildatensatz FALL)
- Prozedurangaben des Behandlungsfalls (Teildatensatz PROZ, wiederholbar)
- Diagnoseangaben des Behandlungsfalls (Teildatensatz DIAG, wiederholbar)
- Entgeltangaben des Behandlungsfalls (Teildatensatz ENTGELT, wiederholbar)
- Fachabteilungsangaben des Behandlungsfalls (Teildatensatz ABTLG, wiederholbar)

Die hier definierten Teildatensätze finden sich im §301-Datensatz bzw. dem DRG-Datenformat nach §21 KHEntgG⁵⁴ wieder. Der QS-Filter-Eingangsdatensatz basiert zum größten Teil auf einer Teilmenge der Inhalte dieser Datensätze.

Tabelle 28: Felder des QS-Filter-Eingangsdatensatzes nach §301

| Feld | Beschreibung | M/K | Zeichenlänge | BasisTyp |
|-------------------------------|---|-----|--------------|----------------------------|
| Behandlungsfall (FALL) | | | | |
| FALLNUMMER | Fallnummer | M | 15 | TEXT |
| FALLART | Fallart 1: ENTGELTARTAMB <> LEER UND ENTGELTARTSTAT = LEER 2: ENTGELTARTSTAT <> LEER UND ENTGELTARTAMB = LEER | M | 1 | NUMSCHLUESSEL (Fallart) |
| AUFNGRUND | Aufnahmegrund gem. §301 SGB V | K | 2 | SCHLUESSEL (AufnGrund) |
| AUFNDATUM | Datum der Aufnahme Krankenhaus | K | 10 | DATUM |
| ENTLDATUM | Datum der Entlassung Krankenhaus | K | 10 | DATUM |
| GEBDATUM | Geburtsdatum | M | 10 | DATUM |
| PATALTER | Alter in Jahren am Auf- nahmetag | M | 3 | GANZEZAHL |

⁵⁴ Vereinbarung über die Übermittlung von DRG-Daten nach §21 KHEntgG. Stand: 30. November 2012. Siegburg: Institut für das Entgeltsystem im Krankenhaus (InEK gGmbH). Online verfügbar unter http://www.gdrg.de/cms/Datenlieferung_gem._21_KHEntgG/Dokumente_zur_Datenlieferung/Datensatzbeschreibung.

| Feld | Beschreibung | M/K | Zeichenlänge | BasisTyp |
|--|--|-----|--------------|-------------------------------|
| ENTLGRUND | Entlassungsgrund §301 | K | 2 | SCHLUESSEL (EntlGrund) |
| IKNRVERANLKH | IK des veranlassenden Krankenhauses | K | 9 | TEXT |
| AUFNZEIT | Aufnahmeuhrzeit | K | 5 | UHRZEIT |
| ALTERINTAGEN | Alter in Tagen am Aufnahme-tag | K | 5 | GANZEZAHL |
| GEWICHT | Aufnahmegewicht in Gramm | K | 6 | GANZEZAHL |
| DAUBEAT | Beatmungsstunden | K | 4 | GANZEZAHL |
| ENTLZEIT | Entlassungs-/ Verlegungsuhrzeit | K | 5 | UHRZEIT |
| STANDORT | entlassender Standort | M | 2 | TEXT |
| KASSEIKNR | Institutionskennzeichen der Krankenkasse der Versichertenkarte | K | 9 | TEXT |
| VERSI- CHERTENIDNEU | eGK-Versichertennummer | K | 10 | TEXT |
| PERSONEN- KREIS | besonderer Personenkreis | K | 2 | SCHLUESSEL (Personenkreis) |
| Prozedurangaben (wiederholbar) (PROZ) | | | | |
| FALLNUMMER | Fallnummer | M | 15 | TEXT |
| OPDATUM | Datum der Operation | M | 10 | DATUM |
| OPS | Prozedurschlüssel (OPS-Kode) | M | 13 | SCHLUESSEL (OPS) |
| Diagnoseangaben (wiederholbar) (DIAG) | | | | |
| FALLNUMMER | Fallnummer | M | 15 | TEXT |
| ICD | Diagnoseschlüssel (ICD-Kode) | M | 9 | SCHLUESSEL (ICD) |
| DIAGART | Diagnoseart | M | 2 | SCHLUESSEL (DiagArt) |
| SEKDIAG | Sekundär-Diagnoseschlüssel (Sekundär-Kode) | K | 9 | SCHLUESSEL (ICD) |

| Feld | Beschreibung | M/K | Zeichenlänge | BasisTyp |
|---|--|-----|--------------|--|
| Entgeltangaben (wiederholbar) (ENTGELT) | | | | |
| FALLNUMMER | Fallnummer | M | 15 | TEXT |
| ENTGELTARTSTAT | Entgeltschlüssel stationär gem. § 301 SGB V (1.–3. Stelle) | K | 3 | NUMSCHLUESSEL (EntgeltSchluessel-Stat) |
| ENTGELTARTAMB | Entgeltschlüssel ambulant gem. § 301 SGB V | K | 8 | NUMSCHLUESSEL (EntgeltSchluessel-Amb) |
| EBMDATUM | Leistungstag (Entgelt) | K | 10 | DATUM |
| Fachabteilungsangaben (wiederholbar) (ABTLG) | | | | |
| FALLNUMMER | Fallnummer | M | 15 | TEXT |
| FACHABT | Fachabteilung | M | 4 | SCHLUESSEL (Fachabt) |
| FABAUFNDATUM | Datum der Aufnahme in die Fachabteilung | M | 10 | DATUM |
| FABAUFNZEIT | Uhrzeit der Aufnahme in die Fachabteilung | M | 5 | UHRZEIT |
| FABENTLDATEM | Datum der Entlassung/Verlegung aus der Fachabteilung | M | 10 | DATUM |
| FABENTLZEIT | Uhrzeit der Entlassung/Verlegung aus der Fachabteilung | M | 5 | UHRZEIT |

In den Datenfeldern ENTGELTARTSTAT und ENTGELTARTAMB sind die Codes zu dokumentieren, welche im Schlüssel Entgeltarten des Datenübermittlungsverfahrens nach §301 Abs. 3 SGB V definiert sind⁵⁵. Die nachfolgend aufgelisteten Codes sind für die Zuordnung der Versorgungsarten relevant:

- Kode 70⁵⁶: DRG-Fallpauschale nach §7 Nr. 1 KHEntgG (§ 1 Abs. 1 Satz 1 KVPV/FPV)
- Kode 61⁵⁷: Entgelt für integrierte Versorgung nach § 140c SGB V
- Kode 65⁵⁸: Zusatzentgelt für DMP
- Sonstiger Fall⁵⁹: Fall, der weder ein DRG-Fall noch ein IV-Fall noch ein DMP-Fall ist (z. B. ein

⁵⁵ Aktuelle Version abrufbar im Download-Bereich der DKG-Homepage (<http://www.dkgev.de>)

⁵⁶ Versorgungsart DRGFALL (Datenfeld des QS-Filter-Ausgangsdatensatz)

⁵⁷ Versorgungsart IVFALL (Datenfeld des QS-Filter-Ausgangsdatensatz)

⁵⁸ Versorgungsart DMPFALL (Datenfeld des QS-Filter-Ausgangsdatensatz)

⁵⁹ Versorgungsart SONSTFALL (Datenfeld des QS-Filter-Ausgangsdatensatz)

ambulanter Behandlungsfall oder ein Fall, der auf Basis der BPfIV nach Tagessätzen abgerechnet wird).

Die Details der Zuordnung sind in Abschnitt B 1.3.2 beschrieben.

Seit der Spezifikation 2016 sind für den externen Schlüssel `EntgeltSchluesselStat` dreistellige Schlüsselwerte hinterlegt. Diese dienen im Rahmen der Spezifikation der Identifikation belegärztlicher Fälle.

Primär und Sekundär Diagnosen

Mit Einführung des Moduls `NWIF` in der Spezifikation 2017 wurde das Feld `SEKDIAG` (bzw. die Syntaxvariablen `SEKDIAG` und `HSEKDIAG`) in den Eingangsdatensatz aufgenommen. Für das Modul `NWIF` sind Einschlussdiagnosen nach Primär- und Sekundär Diagnosen im Sinne der Kreuz-Stern-Notation zu differenzieren. Die Zusammengehörigkeit bestimmter Sekundär Diagnosen zu bestimmten Primär Diagnosen ist hierbei zu berücksichtigen. Dies gilt lediglich für das Modul `NWIF`. In allen Modulen außer `NWIF` soll nicht zwischen Primär- und Sekundär Diagnosen differenziert werden. Die dort in den Auslösekriterien referenzierten Felder `DIAG` bzw. `HDIAG` beinhalten weiterhin sowohl Primär- als auch Sekundär Diagnosen.

DATENSATZ_295 enthält den QS-Filter-Eingangsdatensatz nach §295 SGB V

Der QS-Filter-Eingangsdatensatz nach §295 SGB V bezieht sich auf Fälle, die kollektivvertraglich durch einen niedergelassenen Leistungserbringer (Arztpraxis, MVZ, auch Eingriffe durch Belegärzte oder durch Krankenhausärzte im Rahmen einer KV-Ermächtigung)⁶⁰ erbracht werden und ist so beschaffen, dass fast alle seine Datenfelder aus der Struktur der KVDT-Datensatzbeschreibung für den Einsatz von IT-Systemen in der Arztpraxis der Kassenärztlichen Bundesvereinigung gemäß § 295 SGB V abgeleitet werden können⁶¹. Einzige Ausnahme sind die Felder `PATALTER` (lässt sich aus Inhalten des KVDT-Datensatzes berechnen) und `BEHANDLFALLID` (kann zur Verknüpfung der Teildatensätze vom System vergeben werden oder auf andere Felder des KVDT-Datensatzes Bezug nehmen, sofern diese eine eindeutige Zuordnung erlauben).

Der QS-Filter-Eingangsdatensatz nach §295 SGB V besteht aus folgenden Teilen:

- Behandlungsfall (Teildatensatz `FALL`)
- Prozedurangaben des Behandlungsfalls (Teildatensatz `PROZ`, wiederholbar)
- Diagnoseangaben des Behandlungsfalls (Teildatensatz `DIAG`, wiederholbar)
- Angaben zu Gebührenordnungspositionen des Behandlungsfalls (Teildatensatz `EBM`, wiederholbar)

⁶⁰ Ambulante Leistungen nach §295 SGB V, §116 SGB V, stationäre Fälle nach §121 Abs. 3 SGB.

⁶¹ Zeichenart und Feldlänge der im Folgenden beschriebenen Datenfelder können vom KVDT-Datensatz abweichen, da hier die in der Spezifikation definierten Datentypen verwendet werden. Das Format `DATUM` wird beispielsweise über 10 Zeichen (`TT.MM.JJJJ`) abgebildet statt über 8 Zeichen (`JJJJMMTT`).

Tabelle 29: Felder des QS-Filter-Eingangsdatensatzes nach § 295

| Feld | Beschreibung | M/K | Zeichenlänge | BasisTyp |
|--|---|-----|--------------|--|
| Behandlungsfall (FALL) | | | | |
| SATZART | Satzart | M | 4 | NUMSCHLUESSEL (Satzart) |
| BEHANDLFALLID | Behandlungsfall-ID | M | 15 | TEXT |
| PATNUMMER | Patientennummer | K | 15 | TEXT |
| SCHEINID | Schein-ID | K | 60 | TEXT |
| AUFNDATUM | Aufnahmedatum | K | 10 | DATUM |
| ENTLDATEM | Entlassungsdatum | K | 10 | DATUM |
| PATALTER | Alter in Jahren am Behandlungstag | M | 3 | GANZEZAHL |
| BSNRAMBULANT | Betriebs- (BSNR) oder Nebenbetriebsstättennummer (NBSNR) (ambulant) | M | 9 | SCHLUESSEL (BSNRRambulant) |
| LANR | Lebenslange Arztnummer (LANR) | M | 9 | SCHLUESSEL (LANRRambulant) |
| KASSEIKNR | Kostenträgernummer | M | 9 | TEXT |
| VERSICHERTENIDNEU | eGK-Versichertennummer | K | 10 | TEXT |
| PERSONENKREIS | besonderer Personenkreis | K | 2 | SCHLUESSEL (Personenkreis) |
| Prozedurangaben (wiederholbar) (PROZ) | | | | |
| BEHANDLFALLID | Behandlungsfall-ID | M | 15 | TEXT |
| OPDATUM | OP-Datum | K | 10 | DATUM |
| OPS | OP-Schlüssel | K | 13 | SCHLUESSEL (OPS) |
| OPSLOK | Seitenlokalisierung OPS | K | 1 | SCHLUESSEL (OPSSeitenLokalisierung) |
| Diagnoseangaben (wiederholbar) (DIAG) | | | | |
| BEHANDLFALLID | Behandlungsfall-ID | M | 15 | TEXT |

| Feld | Beschreibung | M/K | Zeichenlänge | BasisTyp |
|--|----------------------------------|-----|--------------|-------------------------------------|
| ICD | ICD-Kode | M | 9 | SCHLUESSEL (ICD) |
| DIAGSICHERHEIT | Diagnosensicherheit | M | 1 | SCHLUESSEL (DiagSicherheit) |
| DIAGLOK | Seitenlokalisierung | K | 1 | SCHLUESSEL (ICDSeitenLokalisierung) |
| Entgeltangaben (wiederholbar) (EBM) | | | | |
| BEHANDLFALLID | Behandlungsfall-ID | M | 15 | TEXT |
| EBMDATUM | Leistungstag | M | 10 | DATUM |
| EBM | Gebührennummer gemäß EBM-Katalog | M | 7 | SCHLUESSEL (EBM) |

Die Gebührenordnungspositionen bzw. Gebührennummern sind nach dem gültigen einheitlichen Bewertungsmaßstab in der jeweils aktuellen Fassung zu kodieren. Für jede dokumentierte Gebührenordnungsposition ist ein neuer Teildatensatz EBM anzulegen.

DATENSATZ_301 und DATENSATZ_295 definieren den QS-Filter bei selektivvertraglicher Leistungserbringung und unbekannter Abrechnungsart

Je nach vorliegender Abrechnungsart ist der entsprechende Datensatz zu wählen. Sofern die Abrechnungsart noch nicht bekannt ist, ein durch einen niedergelassenen Leistungserbringer selektivvertraglich erbrachter Fall vorliegt⁶² oder es sich um eine Leistung handelt, die nicht im Rahmen der gesetzlichen Krankenversicherung abgerechnet wird, bilden alle Felder beider Eingangsdatensätze (DATENSATZ_301 und DATENSATZ_295) zusammen einen allgemein gültigen Eingangsdatensatz. In diesem Fall wird die Muss-Regelung aller im jeweils anderen Eingangsdatensatz nicht vorkommenden oder dort nicht als obligat aufgeführten Felder aufgehoben, sodass dann eine Kann-Regelung gilt.

Beispiel:

Verschiedene Fallkonstellationen zur Muss-/Kann-Feld-Regelung im allgemein gültigen Eingangsdatensatz

FALLNUMMER ist sowohl ein Feld des Datensatzes DATENSATZ_301 als auch des Datensatzes DATENSATZ_295, es ist in beiden Datensätzen ein Muss-Feld, d.h. es ist auch ein Muss-Feld, wenn es sich um einen durch einen niedergelassenen Leistungserbringer selektivvertraglich erbrachten Fall handelt oder wenn die Abrechnungsart (noch) nicht vorliegt.

⁶² Da die Abrechnungsgrundlagen für selektivvertraglich erbrachte ambulante Leistungen nicht bundesweit eindeutig definiert sind, wird der QS-Filter-Datensatz für diese Fälle möglichst breit spezifiziert.

AUFNDATUM ist sowohl ein Feld des Datensatzes DATENSATZ_301 als auch des Datensatzes DATENSATZ_295, es ist nach §301 ein Muss-Feld und nach §295 ein Kann-Feld, d.h. es ist ein Kann-Feld, wenn es sich um einen durch einen niedergelassenen Leistungserbringer selektivvertraglich erbrachten Fall handelt oder wenn die Abrechnungsart (noch) nicht vorliegt.

EBM ist ein Feld des Datensatzes DATENSATZ_295, es ist ein Muss-Feld, d.h. es ist ein Kann-Feld, wenn es sich um einen durch einen niedergelassenen Leistungserbringer selektivvertraglich erbrachten Fall handelt oder wenn die Abrechnungsart (noch) nicht vorliegt.



Achtung

Ergänzung des QS-Filter-Eingangsdatensatzes bei selektivvertraglichen Fällen eines niedergelassenen Leistungserbringers

Für selektivvertraglich erbrachte ambulante Fälle können (unabhängig von OPS-Kodes und Gebührennummern gemäß EBM-Katalog) spezifische Abrechnungskodes vorliegen, die nicht bundesweit einheitlich definiert sind und in der Beschreibung des QS-Filter-Datensatzes damit nicht aufgeführt werden können. Der Leistungserbringer hat gemeinsam mit seinem Softwareanbieter dafür Sorge zu tragen, dass die entsprechenden Codes bzw. Felder identifiziert und ergänzend im QS-Filter-Eingangsdatensatz berücksichtigt werden.

1.3.2 Der QS-Filter-Ausgangsdatensatz

Für die Erstellung der Sollstatistik zum Abschluss des Erfassungsjahres müssen sämtliche Felder des Ausgangsdatensatzes berechnet werden. Grundlage der Sollstatistik ist die gesamte beim Leistungserbringer vorliegende Datenbasis.

DATENSATZ_301 enthält den QS-Filter-Ausgangsdatensatz nach §301 Abs. 3 SGB V.

Der QS-Filter-Ausgangsdatensatz nach §301 Abs. 3 SGB V beschreibt die Grundlage für die Sollstatistik für Krankenhäuser. Dieser gilt sowohl für Verfahren gemäß QSKH-RL als auch für das Exportmodul PCI_LKG und NWIF gemäß Qesü-RL und das Exportmodul CHE_LKG gemäß DeQS-RL. Der QS-Filter-Ausgangsdatensatz besteht aus folgenden Teilen:

- Wiederholbare Angaben zu ausgelösten Modulen (technisch Teildatensatz QSMODUL)
- Nicht wiederholbare Falldaten (dem Teildatensatz FALL zugeordnet)
- Wiederholbare Angaben zu Fehlern (technisch Teildatensatz FEHLER)

Die Felder des QS-Filter-Ausgangsdatensatzes werden aus dem QS-Filter-Eingangsdatensatz berechnet.⁶³ Dabei werden die Felder des Ausgangsdatensatzes nach §301 aus dem DATENSATZ_301 oder aus dem allgemein gültigen Eingangsdatensatz ermittelt.

⁶³ Ausnahme ist das Datenfeld FALLNUMMER, welches für die relationalen Verknüpfungen zwischen den Teildatensätzen relevant ist.

Tabelle 30: Felder des QS-Filter-Ausgangsdatensatzes nach §301

| Feld | Beschreibung | M/K | Zeichenlänge | BasisTyp |
|------------------------------------|--|-----|--------------|----------------------------|
| Module (QSMODUL) | | | | |
| FALLNUMMER | Fallnummer | M | 15 | TEXT |
| MODUL | Dokumentationspflichtiges Modul | M | 20 | SCHLUESSEL (Modul) |
| DOKVERPFLICHT | Stufe der Dokumentationsverpflichtung | M | 1 | SCHLUESSEL (DokVerpflicht) |
| SOLLJAHR | Verfahrensjahr der Sollstatistik, in welchem der Datensatz gezählt wird | M | 4 | GANZEZAHL |
| berechnete Falldaten (FALL) | | | | |
| DRGFALL | <p>DRG-Fall</p> <p>1:= (TeilStrListe(ENTGELTARTSTAT;1;3) EINSIN ('701';'702';'703';'704';'705';'706';'707';'708')UND MODUL <> 'PCI') ODER (MODUL = 'PCI' UND ENTGELTARTSTAT EINSIN ('701';'702';'707';'708'))</p> <p>0:= (TeilStrListe(ENTGELTARTSTAT;1;3) KEINSIN ('701';'702';'703';'704';'705';'706';'707';'708') UND MODUL <> 'PCI') ODER (MODUL = 'PCI' UND ENTGELTARTSTAT KEINSIN ('701';'702';'707';'708'))</p> | M | 1 | NUMSCHLUESSEL (JN) |
| IVFALL | <p>IV-Fall</p> <p>1:= TeilStrListe(ENTGELTARTSTAT;1;3) EINSIN ('611';'612';'613';'614';'615';'616';'617';'618')</p> | M | 1 | NUMSCHLUESSEL (JN) |

| Feld | Beschreibung | M/K | Zeichenlänge | BasisTyp |
|-----------|--|-----|--------------|-----------------------|
| | 0:= TeilStr- Liste(ENTGELTART- STAT;1;3) KEINSIN ('611';'612';'613' ';'614';'615';'61 6';'617';'618') | | | |
| DMPFALL | DMP-Fall 1:= TeilStr- Liste(ENTGELTART- STAT;1;3) EINSIN ('650';'651';'652' ';'653';'654';'65 5';'657';'658';'6 59') 0:= TeilStr- Liste(ENTGELTART- STAT;1;3) KEINSIN ('650';'651';'652' ';'653';'654';'65 5';'657';'658';'6 59') | M | 1 | NUMSCHLUESSEL (JN) |
| SONSTFALL | Sonstiger Fall 1:= TeilStr- Liste(ENTGELTART- STAT;1;3) KEINSIN ('701';'702';'703' ';'704';'705';'70 6';'707';'708';'6 11';'612';'613';' 614';'615';'616'; '617';'618';'650' ';'651';'652';'653' ';'654';'655';'65 7';'658';'659') ODER ENTGEL- TARTAMB <> LEER 0:= TeilStr- Liste(ENTGELTART- STAT;1;3) EINSIN ('701';'702';'703' ';'704';'705';'70 6';'707';'708';'6 11';'612';'613';' 614';'615';'616'; '617';'618';'650' ';'651';'652';'653' ';'654';'655';'65 7';'65'8;'659') | M | 1 | NUMSCHLUESSEL (JN) |

| Feld | Beschreibung | M/K | Zeichenlänge | BasisTyp |
|---------------------------------------|----------------------------|-----|--------------|------------------------|
| | ODER ENTGEL-TARTAMB = LEER | | | |
| Fehler (wiederholbar) (FEHLER) | | | | |
| FALLNUMMER | Fallnummer | M | 15 | TEXT |
| FKODE | Fehler | M | 2 | NUMSCHLUESSEL (Fehler) |
| FMELDUNG | Fehlermeldung | M | 255 | TEXT |

Das `SOLLJAHR` definiert das Jahr, in welchem der Datensatz bei der Erstellung der Sollstatistik gezählt wird.

Wenn der QS-Filter-Algorithmus ein oder mehrere Module auslöst, werden ein oder mehrere Moduleinträge im Teildatensatz `QSMODUL` erzeugt.

Beim Auftreten eines Fehlers werden definierte Fehlercodes und eine Fehlermeldung in den Teildatensatz `FEHLER` eingetragen. Treten Fehler auf, werden für diesen Datensatz keine Teildatensätze vom Typ `Modul` ausgegeben (Abschnitt B 1.4.6). Für die verlaufs begleitende Auslösung sind lediglich die Datenfelder `MODUL`, `DOKVERPFLICHT` sowie der Teildatensatz `FEHLER` notwendig⁶⁴.

DATENSATZ_295 enthält den QS-Filter-Ausgangsdatensatz nach §295 SGB V.

Der QS-Filter-Ausgangsdatensatz nach §295 SGB V beschreibt die Grundlage für die Sollstatistik der kollektivvertraglich erbrachten ambulanten Fälle, die von den zuständigen Kassenärztlichen Vereinigungen berechnet wird. Die Felder des Ausgangsdatensatzes nach §295 basieren auf dem `DATENSATZ_295` oder dem allgemein gültigen Eingangsdatensatz.

Der QS-Filter-Ausgangsdatensatz besteht aus folgenden Teilen:

- Wiederholbare Angaben zu ausgelösten Modulen (technisch Teildatensatz `QSMODUL`)
- Nicht wiederholbare Falldaten (dem Teildatensatz `FALL` zugeordnet)
- Wiederholbare Angaben zu Fehlern (technisch Teildatensatz `FEHLER`)

Tabelle 31: Felder des QS-Filter-Ausgangsdatensatzes nach § 295

| Feld | Beschreibung | M/K | Zeichenlänge | BasisTyp |
|-------------------------|---------------------------------|-----|--------------|--------------------|
| Module (QSMODUL) | | | | |
| BEHANDLFALLID | Behandlungsfall-ID | M | 15 | TEXT |
| MODUL | Dokumentationspflichtiges Modul | M | 20 | SCHLUESSEL (Modul) |

⁶⁴ Für die unterjährige Auslösung brauchen die Falldaten aus den Entgeltangaben somit nicht berechnet werden.

| Feld | Beschreibung | M/K | Zeichenlänge | BasisTyp |
|---------------------------------------|---|-----|--------------|-------------------------------|
| DOKVERPFLICHT | Stufe der Dokumentationsverpflichtung | M | 1 | SCHLUESSEL (DokVerpflicht) |
| EBMJAH | Jahr der Operation | M | 4 | GANZEZAHL |
| SOLLJAHR | Verfahrensjahr der Sollstatistik, in welchem der Datensatz gezählt wird | M | 4 | GANZEZAHL |
| Berechnete Falldaten (FALL) | | | | |
| BELEGFALL | belegärztlicher Fall 1 := SATZART EINSIN (0103) 0 := SATZART KEINSIN (0103) | M | 1 | NUMSCHLUESSEL (JN) |
| Fehler (wiederholbar) (FEHLER) | | | | |
| BEHANDLFALLID | Behandlungsfall-ID | M | 15 | TEXT |
| FKODE | Fehler | M | 2 | NUMSCHLUESSEL (Fehler) |
| FMELDUNG | Fehlermeldung | M | 255 | TEXT |

Der Teildatensatz FALL enthält Felder zur Identifikation belegärztlicher Fälle.

Der Teildatensatz QSMODUL enthält beispielsweise das Feld SOLLJAHR (M). SOLLJAHR definiert das Jahr, in welchem der Datensatz bei der Erstellung der Sollstatistik gezählt wird. Dies geschieht für das Exportmodul PCI_KV mithilfe des Feldes EBMDATUM (jahreswertliste(EBMDATUM) EINSIN (<ERFASSUNGSJAHR>)) unter dem Ausschluss belegärztlicher Fälle bzw. mithilfe des Feldes ENTLDATUM ((ENTLDATUM >= '01.01.<ERFASSUNGSJAHR>' UND ENTLDATUM <= '31.12.<ERFASSUNGSJAHR>') für belegärztliche Fälle.

DATENSATZ_301 und DATENSATZ_295 definieren den QS-Filter-Ausgangsdatensatz bei selektivvertraglicher Leistungserbringung.

Der allgemein gültige QS-Filter-Ausgangsdatensatz beschreibt die Grundlage für die Sollstatistik der selektivvertraglich erbrachten ambulanten Fälle, die von den niedergelassenen Leistungserbringern zu erstellen ist.⁶⁵ Fälle, die kollektivvertraglich erbracht wurden, sind bei der Erstellung dieser Sollstatistik auszuschließen. Das SOLLJAHR definiert das Jahr, in welchem der Datensatz bei der Erstellung der Sollstatistik gezählt wird. Dies geschieht für das Exportmodul PCI_SV

⁶⁵ Da die Abrechnungsgrundlagen für selektivvertraglich erbrachte ambulante Leistungen nicht bundesweit eindeutig definiert sind, wird der QS-Filter-Datensatz für diese Fälle möglichst breit spezifiziert.

mithilfe des Behandlungsdatums, das zwischen dem 01.01. und dem 31.12. des gültigen Erfassungsjahres liegen muss.

**Achtung****Ergänzung des QS-Filter-Ausgangsdatensatzes bei selektivvertraglichen Fällen eines niedergelassenen Leistungserbringers**

Wurden im QS-Eingangsdatensatz für selektivvertraglich erbrachte ambulante Fälle spezifische Abrechnungskodes identifiziert, hat der Leistungserbringer gemeinsam mit seinem Softwareanbieter dafür Sorge zu tragen, dass die entsprechenden Codes bzw. Felder auch im QS-Filter-Ausgangsdatensatz berücksichtigt werden.

1.4 Der Algorithmus zur Ermittlung der Dokumentationspflicht

Dieser Abschnitt beschreibt den Algorithmus zur Ermittlung der Dokumentationsverpflichtung von Modulen und die erforderlichen Fehlerprüfungen.

1.4.1 Einleitung und Überblick

Für jedes Modul ist ein formaler, logischer Ausdruck definiert, welcher sich aus einer leistungs-bereichsbezogenen Teilbedingung (Abschnitt B 1.4.2) und einer administrativen Teilbedingung (Abschnitt B 1.4.3) zusammensetzt:

Auslösebedingung für ein Modul = ModulAusloeser.bedingung UND
AdminKriterium.bedingung

Wenn eine der beiden Teilbedingungen nicht erfüllt ist, so löst der QS-Filter für das jeweilige Modul keine Dokumentationspflicht aus.

Für jeden Fall evaluiert der QS-Filter-Algorithmus sämtliche der in der Spezifikationsdatenbank hinterlegten Auslösebedingungen. Wird die Auslösebedingung erfüllt, so ist der Fall für das verknüpfte Modul dokumentationspflichtig.

**Achtung**

Generell gilt die Einschränkung, dass ein gleiches Modul pro Fall nur einmal ausgelöst wird. Es können aber mehrere verschiedenartige Module ausgelöst werden.

Dokumentationspflicht eines Falles

Ein Fall ist dokumentationspflichtig, wenn für ihn mindestens ein QS-Modul dokumentationspflichtig ist.

Dokumentationspflicht eines Moduls

Ein Modul (Datensatz) ist dann dokumentationspflichtig, wenn folgende Bedingungen zutreffen:

- Bei der Fehlerprüfung tritt kein Fehler auf (Abschnitt B 1.4.6).
- Die jeweilige modulspezifische Auslösebedingung ist erfüllt (Abschnitt B 1.4.2).
- Die administrativen Bedingungen sind erfüllt (Abschnitt B 1.4.3).

Zu beachten sind die in Abschnitt B 1.4.5 beschriebenen Stufen der Dokumentationsverpflichtung.

1.4.2 Leistungsbereichsbezogene Einschlusskriterien

Überblick:

Für jede QS-Datensatz-Definition (= Modul) sind in der Tabelle `ModulAusloeser` (Tabelle 33) eine oder mehrere Auslösebedingungen (Attribut `bedingung`) hinterlegt.

Definition:

Jede Auslösebedingung der Tabelle `ModulAusloeser` definiert einen QS-Filter-Leistungsbereich.

Ein QS-Filter-Leistungsbereich ist somit ein Komplex von Leistungen, welche über den QS-Filter-Algorithmus zu einer Dokumentationsverpflichtung führen.

Beispiel:

Die Bedingung (Modul 17/1)

```
PROZ EINSIN SH_OPS UND DIAG EINSIN SH_ICD UND DIAG KEINSIN SH_ICD_EX UND PROZ KEINSIN SH_OPS_EX UND ALTER >= 18
```

setzt sich aus folgenden Teilbedingungen zusammen:

- PROZ EINSIN SH_OPS
- DIAG EINSIN SH_ICD
- DIAG KEINSIN SH_ICD_EX
- PROZ KEINSIN SH_OPS_EX
- ALTER >= 18

`SH_OPS`, `SH_ICD`, `SH_ICD_EX`, `SH_OPS_EX` stellen die Einschluss- bzw. Ausschlusslisten dar, die in den Tabellen `ICDListe` bzw. `OPSListe` spezifiziert wurden und deren Schlüsselwerte in den Tabellen `ICDWert` bzw. `OPSWert` zu finden sind.



Achtung

Einschlusskriterien mit `KASSEIKNR`, `VERSICHERTENIDNEU` und `PERSONENKREIS`

Leistungsbereichsbezogene Einschlusskriterien können die Felder `KASSEIKNR`, `VERSICHERTENIDNEU` und `PERSONENKREIS` enthalten, mithilfe dessen geprüft wird, ob es sich um einen GKV-Versicherten handelt. Liegt die elektronische Gesundheitskarte (eGK) bei der Prüfung der definierten Kriterien noch nicht vor, erlischt **nicht** die Dokumentationspflicht. Die Auslösebedingung ist bei Vorliegen der Angaben `KASSEIKNR/VERSICHERTENIDNEU/PERSONENKREIS/eGK` erneut zu prüfen.

Üblicherweise entspricht ein QS-Datensatz genau einem QS-Filter-Leistungsbereich. Zurzeit gibt es folgende Ausnahmen:

Ausnahmen für einzelne QS-Filter-Leistungsbereiche:

- Für den Minimaldatensatz MDS und die Follow-up-Datensätze⁶⁶ HTXFU, PNTXFU, NLSFU, LTXFU, LLSFU, LUTXFU sind keine QS-Filter-Leistungsbereiche definiert. Sie werden folglich nicht durch den QS-Filter ausgelöst.
- Für die QS-Filter-Leistungsbereiche PCI_LKG und CHE_LKG, PCI_KV sowie CHE_KV und PCI_SV und CHE_SV sind keine QS-Datensätze definiert. Es handelt sich um QS-Filter-Leistungsbereiche für die Sollstatistik, die Untermengen des Dokumentationsmoduls PCI nach der Art der Abrechnung darstellen. Bei der Sollstatistik zählt der QS-Filter-Leistungsbereich PCI_LKG nur die Datensätze, für die die LQS/LKG die zuständige Datenannahmestelle ist. Nur dieser QS-Filter-Leistungsbereich geht in die Sollstatistik der Krankenhäuser ein, die durch die Leistungserbringer selbst erstellt wird. Für die Sollstatistik zu ambulant selektivvertraglich erbrachten Fällen, die durch die niedergelassenen Leistungserbringer erstellt wird, gilt der QS-Filter-Leistungsbereich PCI_SV. Der QS-Filter-Leistungsbereich PCI_KV gilt nur für die Datenannahmestellen der entsprechenden Zuständigkeit (KVen), da diese die Sollstatistik zu den kollektivvertraglich erbrachten Fällen der niedergelassenen Tätigen erstellen.
- Für die Zählleistungsbereiche HEP_IMP, HEP_WE, KEP_IMP, KEP_WE, HTXM_TX, HTXM_MKU, HCH_AORT_KATH_ENDO und HCH_AORT_KATH_TRAPI sind keine QS-Datensätze definiert. Es handelt sich um Zählleistungsbereiche, die Untermengen der Leistungsbereiche HEP, KEP, HTXM und HCH darstellen und lediglich einen differenzierteren Soll-Ist-Abgleich für das jeweilige Modul ermöglichen sollen. Die folgende Tabelle zeigt beispielhaft einen Auszug aus der Sollstatistik eines Krankenhauses, das HCH-Daten erfasst.

Tabelle 32: Beispielhafter Auszug einer Sollstatistik 2015 für QS-Verfahren gemäß QSKH-RL

| Leistungsbe- reich | DS aus DRG- Fällen | DS aus iV-Fäl- len | DS aus DMP- Fällen | DS aus son- stige Fällen | Gesamt: Anzahl DS | Para- meter (B;L;K;Z) | Aufn.- Jahr |
|---------------------------|--------------------------|--------------------------|--------------------------|-----------------------------------|-------------------------|---------------------------------|----------------|
| HCH | 764 | 1 | 0 | 5 | 770 | B | 2015 |
| HCH AORTKA- THEENDO | 22 | 0 | 0 | 0 | 22 | Z | 2015 |
| HCH AORTKA- THTRAPI | 8 | 0 | 0 | 0 | 8 | Z | 2015 |

DS = Datensätze

In die Berechnung der Gesamtzahl der auf Bundesebene verpflichtend zu dokumentierenden Datensätze wird von den drei aufgeführten Leistungsbereichen lediglich der Leistungsbereich HCH mit dem Parameter B mit einbezogen.

⁶⁶ Gemeint sind die Follow-up-Datensätze zu Transplantationsmodulen (ohne PID-Daten).

Ausnahmen für alle Leistungsbereiche:

Bei stationären Fällen mit Fallzusammenführung gemäß § 2 FPV/DRG ist durch die Software sicherzustellen, dass nur ein Gesamtdatensatz erstellt wird, der alle Informationen von der ersten Aufnahme bis zur letzten Entlassung des Vorgangs enthält.

Es gibt folgende Ausnahmen für einzelne Bundesländer (zurzeit für Baden-Württemberg, Hessen, Rheinland-Pfalz und Bayern):

- Die QS-Dokumentation Gynäkologische Operationen (15 / 1) erfolgt im Rahmen der bundesweiten Dokumentationsverpflichtung (QS-Filter-Leistungsbereich GYN) und in Hessen auch im Rahmen der hessischen Zusatzerhebung zum Leiomyom des Uterus (QS-Filter-Leistungsbereich GYNHESSEN⁶⁷).
- Im Auftrag der GQH (Geschäftsstelle Qualitätssicherung Hessen bei der Hessischen Krankenhausesellschaft e.V.) wurden die Spezifikationsinhalte für QS-Dokumentation und QS-Filter in der QS-Basisspezifikation ergänzt.
- Zu folgenden QS-Filter-Leistungsbereichen existiert kein Datensatz in der QS-Spezifikation für QS-Dokumentation:
 - SA_HE (Schlaganfall-Akutbehandlung in Hessen)
 - SA_FRUEHREHA_HE (Schlaganfall-Frührehabilitation in Hessen)
 - SA_BW (Schlaganfall-Akutbehandlung in Baden-Württemberg)
 - APO_RP (Schlaganfall-Akutbehandlung in Rheinland-Pfalz)
 - NHS_RP (Neugeborenen-Hörscreening in Rheinland-Pfalz)
 - SA_BA (Schlaganfall-Akutbehandlung in Bayern)
 - MRE_HE (Multiresistente Erreger in Hessen)

Die Datensatzspezifikationen sind bei den verantwortlichen Landesgeschäftsstellen erhältlich.

Definition in der Datenbank

Tabelle 33 gibt einen Überblick über die Tabelle ModulAusloeser der Spezifikationsdatenbank.

Tabelle 33: Struktur der Tabelle ModulAusloeser

| Feldname | Datentyp | Bemerkung |
|------------------|----------|---|
| idModulAusloeser | INTEGER | Primärschlüssel |
| name | TEXT | Technischer Name des QS-Filter-Leistungsbereichs |
| bezeichnung | TEXT | Bezeichnungstext der Auslösebedingung |
| bedingung | MEMO | Auslösebedingung für den QS-Filter-Leistungsbereich |

⁶⁷ Die Auslösebedingungen für das Modul 15/1 (GYN und GYNHESSEN) sind der Datenbank für QS-Filtersoftware zu entnehmen.

| Feldname | Datentyp | Bemerkung |
|-----------------------|----------|--|
| textDefinition | MEMO | Medizinisch-inhaltliche Definition bzw. Erläuterung der Auslösebedingung |
| fkModul | INTEGER | Bezug zum Modul |
| verpflichtend | BOOLEAN | Wenn WAHR, ist das betreffende Modul bundesweit verpflichtend |
| fkAdminKriterium | INTEGER | Bezug zu den administrativen Abgrenzungskriterien |
| fkDs | INTEGER | Bezug zum entsprechenden Sollstatistikdatensatz |
| internBild | TEXT | Bild zum Modul |
| ambulant | BOOLEAN | Ist die Fallart ambulant? |
| stationaer | BOOLEAN | Ist die Fallart stationär? |
| krankenhausabrechnung | BOOLEAN | Wird der Fall stationär abgerechnet? |
| selektiv | BOOLEAN | Wird der Fall ambulant selektivvertraglich abgerechnet? |
| kollektiv | BOOLEAN | Wird der Fall ambulant kollektivvertraglich abgerechnet? |

Identifizierung der Fallart und der Art der Leistungserbringung

Zur Abbildung sektorenübergreifender Module wurden die booleschen Attribute zur Fallart (*ambulant*, *stationaer*) und zur Art der Leistungserbringung (*krankenhausabrechnung*, *kollektiv*, *selektiv*) in die Tabelle *ModulAusloeser* integriert. Diese können wie folgt identifiziert werden:

ambulant

Dieses Attribut ist WAHR, wenn Feld *FALLART* = 1 (Modulauslöser *PCIKORO* und Exportmodul *PCI_LKG*).

stationaer

Dieses Attribut ist WAHR, wenn Feld *FALLART* = 2 (Modulauslöser *PCIKORO* und Exportmodul *PCI_LKG*).



Hinweis

Im Exportmodul `PCI_KV` werden ambulante und stationäre Fälle berücksichtigt. Hierbei sind mit stationären Fällen belegärztliche Fälle gemeint, die über das Feld `SATZART` im `DATENSATZ_295` (ID 109 in Tabelle `TdsField`) identifiziert werden können.

Das Modul `CHE` bildet ausschließlich stationäre Fälle ab. Hierbei kann es sich um durch das Krankenhaus abgerechnete Fälle nach § 301 (`CHE_LKG`) oder um belegärztliche Leistungen (`CHE_KV`) handeln.

krankenhausabrechnung

Dieses Attribut ist WAHR, wenn Feld `ENTGELTARTAMB` <> LEER ODER (`ENTGELTARTSTAT` <> LEER UND `ENTGELTARTSTAT` KEINSIN `STAT_BELEG_ENTGELT`).

kollektiv/selektiv⁶⁸

Diese Attribute können über das Feld `EBM` identifiziert werden. Das Attribut `selektiv/kollektiv` ist WAHR, wenn `EBM` <> LEER. In diesem Fall ist das Attribut `krankenhausabrechnung` = FALSCH.

1.4.3 Administrative Einschlusskriterien

Überblick

Die administrativen Regelungen beschreiben diejenigen Auslösebedingungen, welche über die modulspezifischen ICD/OPS-Kriterien (Abschnitt B 1.4.2) hinausgehen. Moduldokumentationen werden nur dann durch die QS-Filter-Software ausgelöst, wenn

1. der Patient stationär/vor- und teilstationär/ambulant aufgenommen/behandelt wird (je nach Modul bzw. Richtlinie unterschiedlich),
2. der Behandlungsfall einen bestimmten zeitlichen Rahmen aufweist,
3. der Behandlungsfall keine Verbringungsleistung für Dritte ist.

Zu 1.

Stationär aufgenommen: Die Bedingung wird aus der Angabe im Datenfeld `Aufnahmegrund` (Abschnitt B 1.3.1) abgeleitet:

```
AUFNGRUND NICHTIN ('03';'04') UND AUFNGRUND <> LEER
```

Diese Teilbedingung ist Teil der administrativen Kriterien (siehe Attribut `bedingung` in Tabelle 34). Der zeitliche Rahmen wird bei stationären Fällen über das Aufnahme- und das Entlassungsdatum definiert.

⁶⁸ Die Attribute `kollektiv` und `selektiv` sollen die Unterscheidung beider Attribute z. B. durch Anbieter von Systemen in Arztpraxen ermöglichen. Die Attribute können ggf. nicht durch jeden Anbieter differenziert werden. Eine spätere Konkretisierung wird in der Spezifikation vorgenommen, sobald eindeutige Marker existieren bzw. identifiziert werden konnten, die eine bundesweite Differenzierung zwischen selektivvertraglichen und kollektivvertraglichen Fällen ermöglichen.

Vor- und teilstationär aufgenommen: Beim sektorenübergreifenden Verfahren PCI werden neben stationären Fällen auch vor- und teilstationäre Fälle berücksichtigt. Die Bedingung AUFNGRUND NICHTIN ('03'; '04') ist daher im administrativen Einschlusskriterium nicht enthalten:

```
AUFNGRUND <> LEER
```

Der zeitliche Rahmen wird bei vor- und teilstationären Fällen über das Aufnahme- und das Entlassungsdatum definiert.

Ambulant behandelt: Bei ambulant behandelten Patienten existiert kein Aufnahmegrund. Die Bedingung AUFNGRUND <> LEER ist daher im administrativen Einschlusskriterium nicht enthalten. Der zeitliche Rahmen wird bei ambulanten Fällen über das Behandlungsdatum definiert.

Zu 2⁶⁹

Beispiele für lange Überliegerverfahren:

Alle im Jahr 2019 stationär aufgenommenen Patienten, welche entweder noch nicht entlassen sind oder bis zum 31. Dezember des Folgejahres entlassen sind:

```
AUFNDATUM >= '01.01.2019' UND AUFNDATUM <= '31.12.2019' UND (ENT-  
LSDATUM = LEER ODER ENTLSDATUM <= '31.12.2020')
```



Hinweis

Der Ausdruck ENTLSDATUM = LEER wurde aufgenommen, damit auch diejenigen Patienten ausgelöst werden, welche bereits transplantiert oder behandelt, aber noch nicht entlassen sind.

Fallbeispiel für QS-Verfahren gemäß QSKH:

Spezifikation 2019

Stationärer Patient wird im Jahre 2019 aufgenommen und transplantiert, aber erst Mitte 2020 entlassen. Zum Zeitpunkt der Auslösung ist das Entlassungsdatum noch nicht bekannt. Dieser Patient wird als dokumentationspflichtig ausgelöst, da das Kriterium ENTLSDATUM = LEER ODER ENTLSDATUM <= '31.12.2020' zu diesem Zeitpunkt zutrifft. In der Sollstatistik wird das Entlassungsdatum berücksichtigt, sodass dieser Patient erst in der Sollstatistik 2020 als dokumentationspflichtig gezählt wird. Die Anzahl dieser Fälle wird im Formular zur Sollstatistik in der Spalte „Von Gesamtzahl: Aufnahme im Vorjahr“ zusätzlich ausgewiesen.

⁶⁹ Verbindlich sind die in der Spezifikationsdatenbank hinterlegten Kriterien.



Achtung

Alle gemäß Sollstatistik zum Erfassungsjahr 2019 gehörigen Fälle müssen pünktlich im Rahmen der Lieferfrist für die Auswertung 2019 an die zuständige Datenannahmestelle geliefert werden. Nach Ablauf der Lieferfrist ist ein nachträgliches Korrigieren oder Stornieren dieser Datensätze nicht vorgesehen.

Der jeweilige Datenannahmeschluss ist auch bei den Modulen der Transplantationsmedizin verbindlich! Nachlieferungen werden aber nicht mit `ERROR` abgewiesen, sondern mit `WARNING` gekennzeichnet. Verspätete Nachlieferungen und Updates werden nicht in der jährlichen Bundesauswertung berücksichtigt. Sie dienen lediglich der Korrektur von fehlerhaften Angaben (z. B. der ET-Nummern o.ä.). Die Exportfristen sind Tabelle `Exportmodul` der Datenbank zur QS-Dokumentation zu entnehmen.

Zu 3.

Dieses Kriterium ist nicht aus ein oder mehreren Datenfeldern des QS-Filter-Eingangsdatensatzes ableitbar. Hier ist seitens des Krankenhauses für eine organisatorische Regelung zu sorgen.

Regelungen für die Risikostatistik (QSKH)

Die Anzahl aller Fälle, die in der Risikostatistik zu berücksichtigen sind, werden ab dem Erfassungsjahr 2015 analog zur Spezifikation für Risikostatistik in der QS-Filter-Datenbank mit einem separaten administrativen Kriterium der Tabelle `AdminKriterium` definiert. Die entsprechende Anzahl ist in der Sollstatistik auszuweisen.

Definition in der Datenbank

Tabelle 34 gibt einen Überblick über die Tabelle `AdminKriterium`, welche die administrativen Abgrenzungskriterien definiert.

Tabelle 34: Struktur der Tabelle `AdminKriterium`

| Feldname | Datentyp | Bemerkung |
|-------------------------------|----------|---|
| <code>idAdminKriterium</code> | INTEGER | Primärschlüssel |
| <code>name</code> | TEXT | Technischer Name des administrativen Kriteriums |
| <code>bedingung</code> | MEMO | Auslösebedingung für den QS-Filter-Leistungsbereich |
| <code>info</code> | TEXT | Bezeichnungstext des administrativen Kriteriums |
| <code>sollBedingung</code> | MEMO | Auslösebedingung für die Sollstatistik des aktuellen Erfassungsjahres (enthält nur dann Informationen, wenn die Auslösebedingung der Sollstatistik von der Bedingung des QS-Filters zur Auslösung der Dokumentationspflicht abweicht) |

| Feldname | Datentyp | Bemerkung |
|-------------------------------|----------|--|
| sollBedingungUeberlie- ger | MEMO | Auslösebedingung für die Sollstatistik des Folgejahres (enthält nur dann Informationen, wenn die Auslösebedingung der Sollstatistik von der Bedingung des QS-Filters zur Auslösung der Dokumentationspflicht abweicht) |

1.4.4 Struktur und Syntax der Auslösebedingungen

Die Variablen der Auslösebedingungen

Die in den Auslösebedingungen erlaubten Variablen sind in der Tabelle `SyntaxVariable` definiert.

Die Variablennamen (Attribut `SyntaxVariable.name`) bestehen aus maximal 32 Zeichen. Sie dürfen nur die Buchstaben A bis Z (Großbuchstaben) und die Ziffern 0 bis 9 enthalten. Ein Feldname muss immer mit einem Buchstaben beginnen. Umlaute und Sonderzeichen sind in Feldnamen nicht erlaubt. Ein Feldname darf auch nicht ein reserviertes Wort sein (z. B. Namen von Operatoren wie `EINSIN`, siehe Tabelle 36).

Typen

Jede Variable hat einen Basistyp. Tabelle 35 gibt einen Überblick über die möglichen Basistypen.

Jeder der in der Tabelle `SyntaxVariable` definierten Variablen ist über den Wert des Attributs `SyntaxVariable.fkTdsFeld` ein Feld des QS-Filter-Eingangsdatensatzes zugeordnet. Jedes dieser Felder besitzt einen Basistyp.

Tabelle 35: Basistypen der Variablen

| Basistyp | Bezeichnung | Beispiele (Literele) |
|---------------|---|--|
| BOOL | Boolesche Variable | WAHR, FALSCH |
| TEXT | Zeichenkette (String) | „Spezifikation“ |
| GANZEZAHL | ... -2, -1, 0, 1, 2, 3, ... | 1 |
| ZAHL | Zahl (mit oder ohne Nachkommastellen) | Erfassung: 25,4 oder -100,8 Export: 25.4 oder 100.8 |
| DATUM | Zehnstelliges Datum | '01.01.2013' |
| NUMSCHLUESSEL | Numerisch kodierter Schlüssel (wie GANZEZAHL) | 0 |
| SCHLUESSEL | Alphanumerischer Schlüssel | '19.1', '07' |
| UHRZEIT | Uhrzeit | '10:15' |

| Basistyp | Bezeichnung | Beispiele (Literele) |
|-----------|-------------|----------------------|
| JAHRDATUM | Jahresdatum | 2018 |

Die meisten Schlüsselwerte werden als GANZEZAHL kodiert, d. h., dass die Codes nicht in Hochkommata gesetzt werden dürfen. Ausnahmen sind z. B. die OPS-Schlüssel (z. B. '5.282.0') oder die ICD-10-Schlüssel (z. B. 'C11.3').

Listen

Eine Variable wird als Liste interpretiert, wenn der Wert des Attributs `SyntaxVariable.istListe = WAHR` ist.

Literele

Alphanumerische Literale (z. B. SCHLUESSEL) werden von einfachen Hochkommata eingeschlossen, während Zeichenketten vom Datentyp TEXT in doppelte Anführungsstriche gesetzt werden müssen.

Dies gilt nicht für numerischen Literale (GANZEZAHL, ZAHL, NUMSCHLUESSEL) und Literale des Datentyps BOOL (Wahrheitswerte).

Diagnose-, Prozeduren, FAB- und Abrechnungslisten

Fest definierte Listen von Diagnosen, Prozeduren, Fachabteilungen oder Abrechnungsziffern werden in der Syntax ähnlich wie die Variable `TON_OPS` angesprochen.

Die erlaubten Namen von Diagnoselisten sind in der Tabelle `ICDListe` (Attribut `name`) zu finden. Analog sind die zulässigen Namen von Prozedurlisten in der Tabelle `OPSListe` (Attribut `name`), die zulässigen Namen von FAB-Listen in der Tabelle `FABListe` (Attribut `name`) und von Abrechnungslisten in der Tabelle `EBMListe` (Attribut `name`) definiert.

Die jeweiligen Codes (nach ICD-10-GM, OPS, FAB bzw. EBM) sind in der durch Fremdschlüsselbeziehungen verknüpften Tabelle `ICDWert`, `OPSWert`, `FABWert` und `EBMWert` enthalten. Alle Codes sind terminal. Die Diagnosecodes enthalten weder die Suffixe +, * oder ! noch die Zusatzkennzeichen zur Diagnosesicherheit A, G, V und Z.

Besonderheiten im QS-Filter-Algorithmus Neonatologie (NEO)

In der Tabelle `Syntaxfunktion` ist die Funktion `fabMinutenNichtGeb`⁷⁰ definiert. Diese wird im Modul NEO in folgender Teilbedingung des Modulauslösers verwendet:

```
(fabMinutenNichtGeb('ABTLG';FAB_GEB) = WAHR)
```

Folgend wird jeder Fachabteilungsschlüssel (Feld: `FACHABT`) der 'ABTLG'-Teildatensätze mit jedem Wert der Liste `FAB_GEB` verglichen. Falls der im Teildatensatz eingetragene Fachabteilungsschlüssel 'ABTLG.FACHABT' keinem Wert der Liste `FAB_GEB` entspricht, wird für diesen Fachabteilungsaufenthalt die Verweildauer in Minuten (Funktion: `abstandInMinuten`) berechnet und in der lokalen Variablen `vwdauerFAB` aufsummiert. Nach Durchlauf der Schleifen

⁷⁰ Der hier hinterlegte Pseudocode sowie die entsprechenden Parameter sind der QSFilter-Datenbank zu entnehmen.

beinhaltet die Variable `vwdauerFAB` die kumulierte Aufenthaltsdauer in Minuten außerhalb der geburtshilflichen Fachabteilungen. Wird eine außergeburtshilfliche Gesamtaufenthaltsdauer von mehr als 4320 Minuten (72 Stunden) überschritten, liefert die Funktion den booleschen Wert WAHR zurück und führt zur Auslösung einer Dokumentationspflicht.

Die Operatoren der Auslösebedingungen

Tabelle 36 gibt einen Überblick über die in der Syntax zulässigen Operatoren. Der aktuelle Überblick über alle zulässigen Operationen (inkl. Operanden) ist in Tabelle `SyntaxOperator` der QS-Filter-Datenbank zu finden.

Tabelle 36: Präzedenz und Assoziativität der Operatoren⁷¹

| Präzedenz | Assoziativität | Operator | Erläuterung |
|-----------|----------------|----------|-------------------------------------|
| 0 | links | EINSIN | Mengenoperator „EINSIN“ |
| | links | KEINSIN | Mengenoperator „KEINSIN“ |
| 3 | links | < | Vergleichsoperator „kleiner“ |
| | links | > | Vergleichsoperator „größer“ |
| | links | <= | Vergleichsoperator „kleiner gleich“ |
| | links | >= | Vergleichsoperator „größer gleich“ |
| 4 | links | = | Vergleichsoperator „gleich“ |
| | links | <> | Vergleichsoperator „ungleich“ |
| 5 | rechts | NICHT | Logischer Operator „NICHT“ |
| 6 | links | UND | Logischer Operator „UND“ |
| 7 | links | ODER | Logischer Operator „ODER“ |

Anmerkungen:

Operatoren mit beidseitigen Listenfeldern als Operanden:

EINSIN: Wenn mindestens ein Element aus der linken Liste in der rechten Liste enthalten ist, so ist der Ausdruck wahr (nichtleere Schnittmenge).

KEINSIN: Wenn kein Element der linken Liste in der rechten Liste enthalten ist, so ist der Ausdruck wahr (leere Schnittmenge)⁷².

⁷¹ In dieser Übersichtstafel hat jeder einzelne Operator eine Präzedenzstufe (höchste Präzedenzstufe ist 0). Operatoren, welche die gleiche Stufe haben, werden nach den Regeln der Assoziativität aufgelöst.

⁷² Dieser Operator ist redundant, da er auch durch Negation des `EINSIN`-Operators abgedeckt ist.

Beispiel:

Folgende Regel prüft, ob ein Element der Listenvariable PROZ einen bestimmten Kode besitzt:

```
PROZ EINSIN TON_OPS
```

String-Vergleich bei Diagnosen, Prozeduren und Abrechnungsziffern

Die Operation EINSIN basiert für Diagnosen, Prozeduren oder Abrechnungsziffern auf einem Vergleich von Zeichenketten (String-Vergleich): Für jedes Element der linken Liste wird über einen String-Vergleich geprüft, ob es in der rechten Liste enthalten ist.

Eine Modifikation des String-Vergleichs ist beim String-Vergleich von Diagnosen nötig: Hier werden die Suffixe +, * und ! sowie die Zusatzkennzeichen A, G, V und Z ignoriert. Nicht ignoriert wird das Sonderzeichen . (und – bei OPS).

**Achtung**

In der QS-Filter-Datenbank sind nur terminale Kodes enthalten. Entsprechend muss sichergestellt sein, dass auch die Kodes aus dem Eingangsdatensatz terminal sind. Darauf ist besonders zu achten, wenn Daten ungeprüft aus Subsystemen übernommen wurden.

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')
```

Eine analoge Regelung gilt für die Seitenlokalisationen der OPS-Kodes.

1.4.5 Stufen der Dokumentationsverpflichtung

Die Spezifikation für QS-Filter enthält:

- bundesweit **verpflichtende** (dokumentationspflichtige) Leistungsbereiche,
- bundesweit **freiwillige** Leistungsbereiche, z. B. freiwillige Leistungsbereiche oder landesweit verpflichtende Leistungsbereiche

Bundesweit dokumentationspflichtige Leistungsbereiche sind erkennbar am Attribut `verpflichtend` in der Tabelle `ModulAusloeser`.

Bundesweit freiwillig zu dokumentierende Leistungsbereiche können in einzelnen Krankenhäusern dokumentationspflichtig werden, wenn

- auf Landesebene eine zusätzliche Dokumentationspflicht besteht, oder
- ein einzelnes Krankenhaus intern die Dokumentation fordert.

Die Stufe der Dokumentationsverpflichtung der bundesweit freiwilligen Leistungsbereiche wird in jeder Installation von QS-Filter-Software konfiguriert.

In Tabelle 37 sind die Stufen der Dokumentationsverpflichtung (Schlüssel `DokVerpflicht` der Datenbank) aufgelistet.

Tabelle 37: Stufen der Dokumentationsverpflichtung

| Kennung | Beschreibung | nutzbar für Auswertungen der externen vergleichenden Qualitätssicherung |
|---------|---|---|
| B | Bundesweit verpflichtendes Modul | ja |
| L | Bundesweit freiwilliges Modul, aber im Land verpflichtend dokumentiert | ja |
| Z | Zählleistungsbereich (bundesweit erhoben, Dokumentation erfolgt im Elternmodul) | ja |

Die Stufe der Dokumentationsverpflichtung wird vor Beginn des Erfassungsjahres in der QS-Filter-Software eines Krankenhauses für alle Leistungsbereiche konfiguriert.

Die Konfiguration der Stufen darf im Laufe des Erfassungsjahres nicht mehr verändert werden, da sonst die Vollständigkeit der Dokumentationen nicht mehr gewährleistet ist.

Für jeden ausgelösten Leistungsbereich wird die Stufe der Dokumentationsverpflichtung in den QS-Filter-Ausgangsdatensatz (im Attribut DOKVERPFLICHT des Teildatensatzes QSMODUL) geschrieben.

1.4.6 Fehlerprüfung

Vor Evaluation der in den vorhergehenden Abschnitten beschriebenen Auslösebedingungen ist eine Fehlerprüfung durchzuführen. Die Fehlerprüfung bezieht sich auf die Daten des QS-Filter-Eingangsdatensatzes. Das Ergebnis der Fehlerprüfung wird im Teildatensatz FEHLER gespeichert.

Tabelle 38: Fehlercodes des QS-Filters

| Beschreibung des Fehlers | Fehler-kode | Formale Fehlerbedingung | Fehlermeldung |
|---------------------------|-------------|-------------------------|--|
| Überprüfung des Formats | 1 | Abschnitt B 2.4.8 | Der Wert '<WERT>' des Datenfeldes <Feld.name> '<TdsFeld.bezeichnung>' ist kein gültiger <BasisTyp.name>-Wert (<BasisTyp.bezeichnung> <BasisTyp.format>). |
| Überprüfung der Feldlänge | 2 | Abschnitt B 2.4.8 | Der Wert '<WERT>' des Datenfeldes <Feld.name> '<TdsFeld.bezeichnung>' überschreitet die zulässige Feldlänge <Feld.laenge>. |

| Beschreibung des Fehlers | Fehler-kode | Formale Fehlerbedingung | Fehlermeldung |
|---|-------------|---|---|
| Sind in den Datenfeldern mit internen und externen Schlüsseln (Basistyp SCHLUESSEL oder NUM-SCHLUESSEL) gültige Schlüsselkodes eingetragen? ⁷³ | 3 | Abschnitt B 2.4.8 | Ungültiger Schlüsselcode <Wert> des Schlüssels <Schlüssel.name> im Datenfeld <Feld.name> '<TdsFeld.bezeichnung>'! |
| Überprüfung numerischer Wertebereiche | 4 | Abschnitt B 2.4.8 | Der Wert '<WERT>' des Datenfeldes <Feld.name> '<TdsFeld.bezeichnung>' ist kleiner als '<Feld.min>' oder Der Wert '<WERT>' des Datenfeldes <Feld.name> '<TdsFeld.bezeichnung>' ist größer als '<Feld.max>' |
| Ist ein Muss-Feld ausgefüllt? | 5 | Abschnitt B 2.4.8: Attribut <TdsFeld.fkMussKann> | Das Datenfeld <Feld.name> '<TdsFeld.bezeichnung>' muss einen gültigen Wert enthalten. |
| Ist der Fall dem Erfassungsjahr der Spezifikation zugeordnet? (Spezifikation <ERFASSUNGSJAHR>) | 6 | (MODUL <> 'PCI' ODER (MODUL = 'PCI' UND FALLART = 2) UND AUFNDATUM <'01.01.<ERFASSUNGSJAHR>' ODER AUFNDATUM <'31.12.<ERFASSUNGSJAHR>') ODER (MODUL = 'PCI' UND FALLART <> 2 UND (OPDATUM <'01.01.<ERFASSUNGSJAHR>' ODER OPDATUM | Der Fall ist im Jahr <ERFASSUNGSJAHR> nicht dokumentationspflichtig: Aufnahmedatum/OP-Datum = <Wert> |

⁷³ Die Korrektheit (z. B. ausschließlich gültige terminale Codes oder Verwendung gültiger Katalogversionen) der extern definierten Codes (ICD und OPS) muss vor der Prüfung des Datensatzes durch die QS-Filter-Software sichergestellt sein.

| Beschreibung des Fehlers | Fehler-kode | Formale Fehlerbedingung | Fehlermeldung |
|--------------------------|-------------|--|---------------|
| | | > ' 31 . 12 . <ER- FASSUNGS- JAHR> ')) | |

In <Wert> ist der Wert des Datenfeldes der Falldaten einzusetzen, auf den sich die Fehlermeldung bezieht. Ansonsten sind in die <...> -Ausdrücke die entsprechenden Einträge aus der Datenbank einzusetzen.

Im Fehlerfall sind entsprechende Einträge im Teildatensatz FEHLER zu generieren:

- Fehlercode in Attribut FKODE,
- Fehlermeldung in Attribut FMELDUNG.

Es wird beim Auftreten eines Fehlers nicht weiter geprüft, ob für den Behandlungsfall Moduldokumentationen ausgelöst werden.

1.5 Sollstatistik

Die Sollstatistik ist in der Datenbank für QS-Filtersoftware spezifiziert. Weitere Spezifikationskomponenten sind die Vorlagen zur Sollstatistik:

- „Vorlage_Sollstatistik_<ERFASSUNGSJAHR>_methodisch_QSKH.doc“
- „Vorlage_Sollstatistik_<ERFASSUNGSJAHR>_methodisch_Qesue_DeQS.doc“
- „Vorlage_Sollstatistik_<ERFASSUNGSJAHR>_Budget_QSKH.doc“
- „Vorlage_Sollstatistik_<ERFASSUNGSJAHR>_Budget_Qesue_DeQS.doc“

Ergänzend zu den beschriebenen Prozessen zur Erstellung der Sollstatistik (Kapitel A 3) und den Erläuterungen zum zugrunde liegenden Ausgangsdatensatz des QS-Filters (siehe Abschnitt B 1.3) werden im Folgenden die Besonderheiten der Sollstatistik des Moduls PCI beschrieben.

1.5.1 Sollstatistik des Moduls PCI gemäß Qesü-RL

In der Spezifikationsdatenbank zur Sollstatistik (QSF) erfolgt für das Modul PCI eine Aufteilung analog zu den drei verschiedenen Exportmodulen (Abschnitt B 2.6.1). Anders als im Modulauslöser wird hier ein konkreter Bezug zur Abrechnungsart hergestellt.

Die Sollstatistik muss im Krankenhausbereich sowie für ambulant-selektivvertraglich erbrachte Eingriffe durch die Leistungserbringer geführt werden.

PCI_LKG zählt hierbei nur die Datensätze, die durch das Krankenhaus dokumentiert und an die LQS/LKG zu übermitteln sind. Von den Arztpraxen ist die Sollstatistik nur für selektivvertraglich erbrachte Leistungen (PCI_SV) zu erstellen. Für die kollektivvertraglich erbrachten Leistungen (PCI_KV) wird die Sollstatistik dagegen über die Abrechnungseinrichtungen geführt und gemeldet. PCI_KV gilt somit nur für die Datenannahmestellen der entsprechenden Zuständigkeit (KVen).

Die Datengrundlage ist den jeweiligen administrativen Einschlusskriterien der QS-Verfahren gemäß Qesü-RL zu entnehmen. Für `PCI_LKG` werden hierbei alle Fälle gemäß § 301 berücksichtigt, die das administrative Einschlusskriterium erfüllen.

Analog zur Auslösung wird der zeitliche Rahmen zur Zuordnung eines Falles zum Erfassungsjahr für den stationären Sektor über Aufnahme- und Entlassungsdatum und im ambulanten Sektor über das Datum der Prozedur festgelegt. Ein Ausschluss von vor- oder teilstationären Behandlungen wird auch in der Sollstatistik nicht vorgenommen.

PCI_LKG: Durch ein Krankenhaus erbrachte Leistungen

berücksichtigt bei am Krankenhaus erbrachten Leistungen⁷⁴:

- definierte Einschlussprozeduren und Gebührenordnungspositionen
- den Ausschluss definierter Ausschlussdiagnosen
- den Einschluss von Patienten im Alter von 18 Jahren oder älter
- definierte Kriterien zur Identifizierung von GKV-Versicherten

PCI_KV: Durch Vertragsärzte kollektivvertraglich erbrachte Leistungen

berücksichtigt bei Leistungen, die durch einen kollektivvertraglich tätigen Vertragsarzt erbracht werden⁷⁵:

- definierte Gebührenordnungspositionen
- den Ausschluss definierter Ausschlussdiagnosen
- den Einschluss von Patienten im Alter von 18 Jahren oder älter
- definierte Kriterien zur Identifizierung von GKV-Versicherten

PCI_SV: Durch Vertragsärzte selektivvertraglich erbrachte Leistungen

berücksichtigt bei Leistungen, die selektivvertraglich durch einen Vertragsarzt erbracht werden:

- definierte Einschlussprozeduren und Gebührenordnungspositionen sowie ggf. vertragspezifische Abrechnungskodes (Abschnitt B 1.3.1)
- den Ausschluss definierter Ausschlussdiagnosen
- den Einschluss von Patienten im Alter von 18 Jahren oder älter
- definierte Kriterien zur Identifizierung von GKV-Versicherten

1.5.2 Sollstatistik des Moduls CHE gemäß DeQS-RL

Analog zum Modul `PCI` erfolgt in der Spezifikationsdatenbank zur Sollstatistik (QSF) für das Modul `CHE` eine Aufteilung analog zu den drei verschiedenen Exportmodulen (Abschnitt B 2.6.1). Anders als im Modulauslöser wird hier ein konkreter Bezug zur Abrechnungsart hergestellt. Gemäß DeQS-RL gilt im Verfahren Cholezystektomie die Besonderheit, dass die Erhebung und Weiterleitung der belegärztlichen Daten an die Datenannahmestellen nach Teil 1 § 9 Absatz 1 Satz 3 (LQS/LKG) erfolgen soll.

⁷⁴ Darunter fallen auch belegärztliche Leistungen nach §121 Abs. 5 SGB V (Honorarbelegärzte). Belegärztliche Leistungen nach §121 Abs. 3 SGB V fallen nicht unter diese Definition und werden hier ausgeschlossen.

⁷⁵ Darunter fallen auch belegärztliche Leistungen nach § 121 Abs. 3 SGB V und Leistungen von Krankenhausärzten im Rahmen einer Ermächtigung zur Teilnahme an der vertragsärztlichen Versorgung der Versicherten nach § 116 SGB V. Belegärztliche Leistungen nach § 121 Abs. 5 SGB V (Honorarbelegärzte) fallen nicht unter diese Definition und werden hier ausgeschlossen.

Die Sollstatistik muss im Krankenhausbereich durch die Leistungserbringer geführt werden. Die Zuordnung erfolgt in Anlehnung an PCI (Abschnitt B 1.5.1).

CHE_LKG: Durch ein Krankenhaus erbrachte Leistungen

berücksichtigt bei am Krankenhaus erbrachten Leistungen⁷⁶:

- definierte Einschlussprozeduren und -diagnosen
- den Ausschluss definierter Ausschlussdiagnosen und -prozeduren
- definierte Kriterien zur Identifizierung von GKV-Versicherten

CHE_KV: Durch Vertragsärzte kollektivvertraglich erbrachte Leistungen

berücksichtigt bei Leistungen, die durch einen kollektivvertraglich tätigen Vertragsarzt erbracht werden⁷⁷:

- definierte Einschlussprozeduren und -diagnosen
- den Ausschluss definierter Ausschlussdiagnosen und -prozeduren
- definierte Kriterien zur Identifizierung von GKV-Versicherten

CHE_SV: Durch Vertragsärzte selektivvertraglich erbrachte Leistungen

berücksichtigt bei Leistungen, die selektivvertraglich durch einen Vertragsarzt erbracht werden:

- definierte Einschlussprozeduren und -diagnosen
- den Ausschluss definierter Ausschlussdiagnosen und -prozeduren
- definierte Kriterien zur Identifizierung von GKV-Versicherten

⁷⁶ Darunter fallen auch belegärztliche Leistungen nach §121 Abs. 5 SGB V (Honorarbelegärzte). Belegärztliche Leistungen nach §121 Abs. 3 SGB V fallen nicht unter diese Definition und werden hier ausgeschlossen.

⁷⁷ Darunter fallen nur belegärztliche Leistungen nach § 121 Abs. 3 SGB V.

2 QS-Dokumentation

Die vorliegenden Spezifikationskomponenten für die QS-Dokumentation dienen der Erstellung von Software zur Datenerfassung, Plausibilitätsprüfung und Datenübermittlung für die externe vergleichende Qualitätssicherung im Rahmen der QSKH-, der Qesü-RL und der DeQS-RL. Diese sollen die Bereitstellung valider und vergleichbarer Daten gewährleisten. Neben der Datenbank zur QS-Dokumentation zählen zu den Spezifikationskomponenten der QS-Dokumentation die Ausfüllhinweise, die Dokumentationsbögen und die Erläuterungen zum Minimaldatensatz. Weiterhin werden ergänzende Informationen zur Verfügung gestellt, sowie der Java-Code für ein Testprogramm und eine Testklasse mit Testfällen. Die Delta-Tabellen in der Datenbank zur QS-Dokumentation stellen geänderte, gelöschte und neue Datenbankinhalte im Vergleich zur letzten gültigen Version des Vorjahres sowie, sofern vorhanden, zur letzten Version des aktuellen Erfassungsjahres dar.

2.1 Anmerkungen zur Struktur der Spezifikation zur QS-Dokumentation

Die Spezifikation zur QS-Dokumentation ist in einer relationalen Datenbank abgelegt. Zurzeit wird sie ausschließlich als Access-Datenbank (MS Access 2007-2013) zur Verfügung gestellt. Der Name der Spezifikation richtet sich nach folgendem Schema:

`<Erfassungsjahr>_Basis_QSDOK_V<Versionsnummer>.mdb`

`<Erfassungsjahr>` bezeichnet das Jahr, in dem die QS-Dokumentation stattfindet. `<Versionsnummer>` bezeichnet die 2-stellige Versionsnummer (z. B. 01).

Beispiel:

Im Erfassungsjahr 2019 ist die Spezifikation `2019_Basis_QSDOK_V01.mdb`⁷⁸ gültig.

Weiterführende Erläuterungen zum Benennungsschema für Spezifikationskomponenten sind der Einleitung, Abschnitt 1.1.2, zu entnehmen. Die Tabellenstruktur der Spezifikationsdatenbank wird in Abschnitt B beschrieben.

Folgende Attribute treten in vielen Tabellen auf:

- `name` ist in der Regel als technischer Name zu verstehen. Zum Beispiel wird `Feld.name` als Variablenname in den Plausibilitätsregeln verwendet.
- `bezeichnung` ist eine kurze Beschreibung. Zum Beispiel ist `BogenFeld.bezeichnung` der Text, welcher ein Feld auf einem Eingabeformular beschreibt.
- `bedingung` enthält einen logischen Ausdruck. Prominentester Vertreter dieses Attributtyps ist das Attribut `bedingung` in der Tabelle `Regeln`.

⁷⁸ Die Versionsnummer der gültigen Spezifikation (z. B. V01, V02 usw.) ist dem zuletzt veröffentlichten Update zu entnehmen.

Abfragen der Datenbank

Die Abfragen der Access-Datenbank geben einen vereinfachenden Überblick über die Inhalte der Spezifikation.

- `Datensätze`
- Diese Abfrage liefert einen Überblick über die in der Spezifikation enthaltenen Module (verpflichtende und freiwillige Module), die der aktuell gültigen Version entsprechen.
- `Datensätze_sektorenübergreifend`
- Diese Abfrage liefert einen Überblick über die in der Spezifikation enthaltenen einrichtungs- und sektorenübergreifenden Export- und Dokumentationsmodule der QSDOK-Leistungsbereiche. Ab der Spezifikation 2016 wird hier das Modul `PCI` angezeigt. Ab der Spezifikation 2017 ist weiterhin das Modul `NWIF` enthalten.
- `Datenfelddbeschreibung`
- Hier sind alle Bogenfelder der spezifizierten Module, sortiert nach Modulname, Bogenname und Zeilennummer der Bogenfelder dargestellt (Abschnitt B 2.3).
- `DatenfelddbeschreibungFürEinModul`
- Wenn man diese Abfrage aufruft, so muss der Modulname (z. B. „15/1“) angegeben werden und man erhält eine entsprechende modulbezogene Auswahl der Datenfelddbeschreibung.
- `Plausibilitätsregeln`
- Diese Abfrage enthält alle Plausibilitätsregeln der spezifizierten Module, sortiert nach Modulname und Nummer der Regel (Abschnitt B 2.4).
- `PlausibilitätsregelnFürEinModul`
- Wenn man diese Abfrage aufruft, so muss der Modulname (z. B. „17/1“) angegeben werden und man erhält eine entsprechende modulbezogene Auswahl der Plausibilitätsregeln.
- `Teildatensätze`
- Diese Abfrage liefert einen Überblick über die Teildatensätze und die Regeln für das Anlegen von Teildatensätzen (Abschnitt B 2.3).
- `Ersatzfelder`
- Dies ist eine Auflistung der Ersatzfelder für die Bogenfelder, die modulspezifisch anonymisiert werden müssen (Export von Teildatensätzen).
- `OPSListen`
- Diese Abfrage liefert einen Überblick über die Codes der OPS-Listen (Abschnitt B 2.5.1).
- `ICDListen`
- Diese Abfrage liefert einen Überblick über die Codes der ICD-Listen (Abschnitt B 2.5.2).
- `Exportfelder`
- Wenn man diese Abfrage aufruft, erhält man eine Übersicht über alle Exportfelder. Exportfelder für Listenelemente werden nicht pro Listenelement, sondern pro Listenelement dargestellt. Die Anzahl der Elemente ist der Abfrage zu entnehmen (`Exportfelder.elements`).
- `ExportfelderFürEinModul`
- Diese Abfrage zeigt eine Auswahl der Exportfelder eines Moduls (Modulname ist explizit anzugeben). Man erhält eine Übersicht über die zu exportierenden Felder inkl. Zuordnung zum Teildatensatz. Exportfelder für Listenelemente werden nicht pro Listenelement, sondern pro Listenelement dargestellt. Die Anzahl der Elemente ist der Abfrage zu entnehmen (`ExportfelderFürEinModul.elements`) (Abschnitt B 2.6.2).

- Feldgruppen
- Diese Abfrage liefert eine Übersicht über alle Feldgruppen (Abschnitt B 2.4.7).
- `FeldgruppenFürEinModul`
- Wenn man diese Abfrage aufruft, so muss der Modulname (z. B. „HCH“) angegeben werden und man erhält eine entsprechende modulbezogene Auswahl der Feldgruppen eines Moduls.
- `WertebereicheNumerischerFelder`
- Diese Abfrage liefert eine modulübergreifende Anzeige der numerischen Datenfelder (Typ `ZAHL` und `GANZEZAHL`) und ihrer Wertebereiche.
- `WertebereicheNumerischerFelderFürEinModul`
- Hier werden die numerischen Datenfelder (Typ `ZAHL` und `GANZEZAHL`) und ihrer Wertebereiche für ein Modul angezeigt. Das Modul muss bei Aufrufen der Abfrage angegeben werden.
- `ÜberschriftenFürEinModul`
- Diese Abfrage liefert eine Anzeige der Überschriften für das angegebene Modul. Angegeben werden Start- und Ende-Felder der Überschriften, sowie die Ebene der Überschriften.
- `Schlüsselcodes`
- Diese Abfrage zeigt alle Schlüssel und die zugehörigen Schlüsselwerte an.
- `Ausfüllhinweise`
- Hier wird die Zuordnung von Ausfüllhinweisen (htm.Dateien) zu den Feldern in den einzelnen Modulen angezeigt.
- `AusfüllhinweiseFürEinModul`
- Hier wird die Zuordnung von Ausfüllhinweisen (htm.Dateien) zu den Feldern eines Moduls angezeigt. Das Modul muss bei Aufrufen der Abfrage angegeben werden.

2.2 Patientenidentifizierende Daten zur Follow-up-Erhebung

Mit der Spezifikation 2015 wurden erstmals QS-Daten mithilfe sogenannter patientenidentifizierender Daten (PID) im Regelbetrieb patientenbezogen zusammengeführt.

Mit den QS-Daten werden für jeden Vorgang die folgenden berechneten Ersatzfelder

- `versichertenidgkv`⁷⁹
- `versichertenstatusgkv`⁸⁰ und
- `kasseiknr2Stellen`⁸¹ exportiert.

In der Spezifikationsdatenbank werden die Module mit PID-Verfahren über das Attribut `pid` der Tabelle `Modul` gekennzeichnet. Für welche Module dasselbe Patientenpseudonym zu generieren ist, wird über das Attribut `fkPseudonymVerfahren` beschrieben. Ob auch von der DAS eine verfahrensbezogene Pseudonymisierung der LE-Daten durchzuführen ist, wird durch das Feld `das` (WAHR/FALSCH) festgelegt.

⁷⁹ Im Modul NEO werden zusätzlich bzw. im Modul 16/1 stattdessen die Ersatzfelder `vornameMutter`, `nachnameMutter`, `gebdatumNeo` und `gebdatumNeo` exportiert

⁸⁰ Im Minimaldatensatz wird das Ersatzfeld `vstatusgkvmds` exportiert.

⁸¹ Im Minimaldatensatz wird das Ersatzfeld `kasseiknr2Stellenmds` exportiert.

Tabelle 39: Struktur der Tabelle PseudonymVerfahren

| Feldname | Datentyp | Bemerkung |
|----------------------|--------------|--|
| idPseudonymVerfahren | INTEGER | Primärschlüssel |
| bezeichnung | TEXT (255) | Bezeichnung des Pseudonymverfahrens, z. B. Knieendoprothesenversorgung (KEP) |
| name | TEXT (255) | Name des Pseudonymverfahrens, z. B. KEP |
| vst | BOOLEAN | PID-Pseudonymisierung durch die Vertrauensstelle |
| das | BOOLEAN | LE-Pseudonymisierung (Qesü- und DeQS-Verfahren) durch die Datenannahmestelle |

2.3 Datenfelddescription

Für jedes Modul existiert eine eigene Datenfelddescription. Sie spezifiziert alle auszufüllenden Datenfelder (Bogenfelder, auch Items genannt) und besteht aus mehreren Tabellen (Abbildung 15), die in den nachfolgenden Abschnitten erläutert werden.

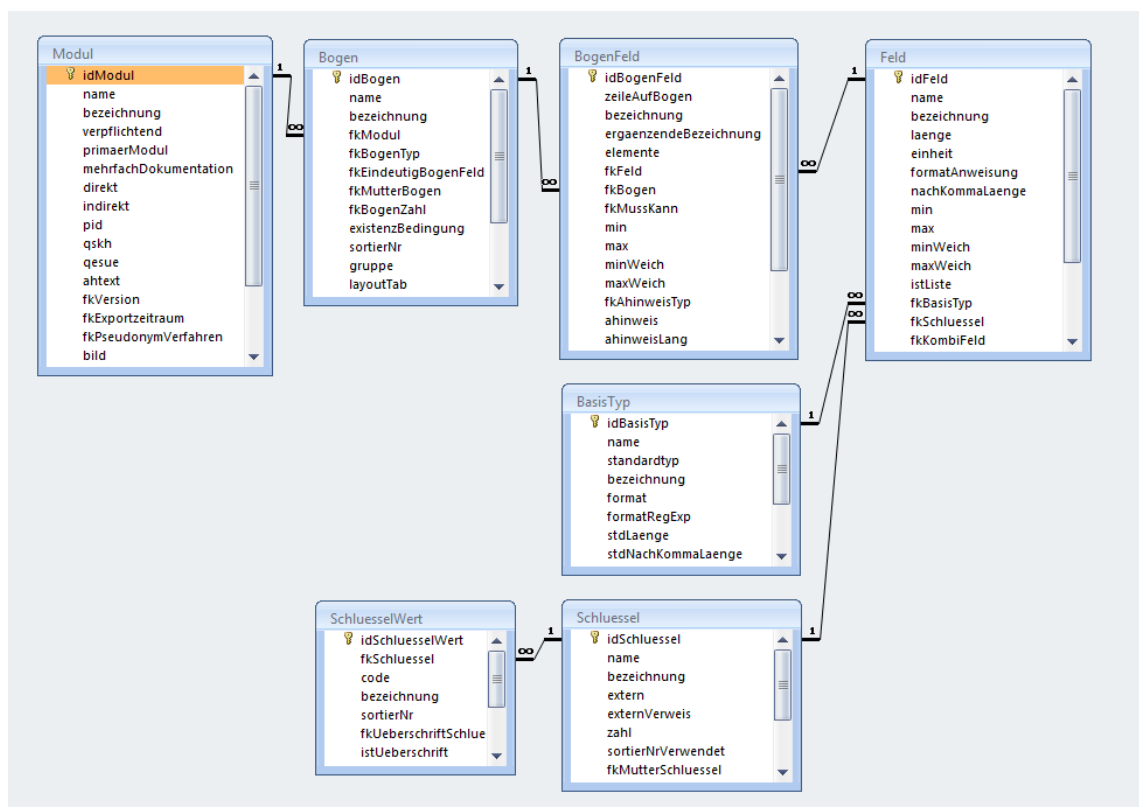


Abbildung 15: Tabellen und Relationen der Datenfelddescription

Die Abfragen Datenfelddescription und DatenfelddescriptionFürEinModul der Access-Datenbank ermöglichen den Überblick über diese Struktur.

Die Beschreibung der Datenfelder hat folgende Ziele:

- Bereitstellung der Informationen, welche für die Programmierung des Eingabeformulars und für die Sicherung der eingegebenen Daten nötig sind
- Vermeidung von Redundanzen
- Typisierung der Felder nach fachlichen und datentechnischen Kriterien

Das für den Anwender wichtigste Merkmal ist die Bezeichnung des Datenfelds (Attribut `BogenFeld.bezeichnung`).

Die Datenfeldbeschreibung ist teilweise auf dem jeweiligen Dokumentationsbogen eines Moduls („Bogensicht“) abgebildet. Zu beachten ist dabei, dass die „Bogensicht“ lediglich die Sicht der medizinischen Fachgruppen, die die Module entwickeln, darstellt. Bei verteilten Softwarelösungen für einen Leistungserbringer hingegen ist die Bogensicht dann nicht mehr adäquat, wenn die Bestandteile eines Bogens auf verschiedene Teilsysteme verteilt sind. Die Daten eines Bogens werden in diesen Fällen für den Export aus den einzelnen Teilsystemen zusammengestellt.



Hinweis

Die Papierbögen werden lediglich zu Illustrationszwecken zur Verfügung gestellt. Sie sind zur Dokumentation nicht zugelassen.

Im Kontext einer integrierten, prozessorientierten Dokumentationssoftware müssen die Teildatensätze nicht direkt in Eingabeformulare umgesetzt werden. Es ist sinnvoller, die Teile eines Dokumentationsbogens zu dem Zeitpunkt und in dem Dokumentationskontext zu erfragen, der sich in den Prozessablauf⁸² eines Leistungserbringers einordnet.

2.3.1 Dokumentationsmodule (Datensätze)

Ein Modul der Spezifikation enthält die Datensatzdefinition von einem (Beispiel: Geburtshilfe) oder mehreren (Beispiel: Herzchirurgie) Leistungsbereichen (QSKH-RL) bzw. Verfahren (Qesü-RL und DeQS-RL). Abhängig von (inhaltlich oder organisatorisch) abzugrenzenden Bereichen kann ein Leistungsbereich/Verfahren mehrere Module umfassen (Beispiel: Transplantationen). Die QS-Dokumentationssoftware kann für einen Behandlungsfall eine oder mehrere Moduldokumentationen anlegen, die nach Dokumentationsabschluss an die Datenannahmestelle übermittelt werden. Fehlerfreie Moduldokumentationen (verkürzt „Module“), die die Basis der Datenauswertungen bilden, werden dem Leistungserbringer von der Datenannahmestelle bestätigt. Aus technischer Sicht ist ein Modul durch einen eindeutigen Namen gekennzeichnet. Es umfasst mindestens einen Teildatensatz. In der Tabelle `Modul` der QS-Spezifikation finden sich die zentralen Definitionen eines Moduls.

⁸² Zum Beispiel in den Prozessablauf eines Krankenhauses.

Tabelle 40: Struktur der Tabelle Modul

| Feldname | Datentyp | Bemerkung |
|-----------------------|--------------|---|
| idModul | INTEGER | Primärschlüssel |
| Name | TEXT (32) | Eindeutiger technischer Name |
| bezeichnung | TEXT (255) | Erläuternde Bezeichnung |
| verpflichtend | BOOLEAN | Besteht für das Modul eine QS-Dokumentationsverpflichtung? |
| primaerModul | BOOLEAN | Ist das Modul ein Primärmodul? |
| mehrfachDokumentation | BOOLEAN | Ist ein mehrfaches Anlegen eines gleichartigen Datensatzes pro Fall zulässig (WAHR / FALSCH)? |
| direkt | BOOLEAN | Handelt es sich um ein direktes Datenexportverfahren? (Abschnitt A 5.3) |
| indirekt | BOOLEAN | Handelt es sich um ein indirektes Datenexportverfahren? (Abschnitt A 5.3) |
| pid | BOOLEAN | Handelt es sich um ein Modul zur Follow-up-Erhebung? ⁸³ (Abschnitt B 2.2) |
| qskh | BOOLEAN | Handelt es sich um ein Modul der QSKH-Richtlinie? |
| quesue | BOOLEAN | Handelt es sich um ein Modul der Qesü-Richtlinie? |
| deqs | BOOLEAN | Handelt es sich um ein Modul der DeQS-Richtlinie? |
| ahtext | MEMO | Einleitender Text für den Ausfüllhinweis eines Moduls |
| fkVersion | INTEGER | Gültige Version des jeweiligen Moduls |
| fkPseudonymVerfahren | INTEGER | Pseudonymisierung von FU-Verfahren ⁸⁴ |
| bild | TEXT (20) | Modulspezifisches Bild |

Auslösung der Moduldokumentation

Der auslösende Sachverhalt für die Dokumentationspflicht ist in der Spezifikation für QS-Filter-Software definiert. Die QS-Filter-Software greift zu diesem Zweck unter anderem auf ICD- und OPS-Kodes (z. B. Haupt- und Nebendiagnosen und Prozeduren) und Gebührenordnungspositionen (EBM-Kodes) zurück, die im Krankenhausinformationssystem (KIS) oder Arzteinformationssystem (AIS) verfügbar sind. Die Dokumentationspflicht des Moduls *NWIF* wird unter anderem über Einschlussdiagnosen ermittelt, die nach Primär- und Sekundärdiagnosen im Sinne der

⁸³ Gemeint ist eine Follow-up-Erhebung mittels PID-Verfahren.

⁸⁴ Gemeint ist ein Follow-up-Verfahren anhand von PID-Daten.

Kreuz-Stern-Notation zu differenzieren sind. Die Zusammengehörigkeit bestimmter Sekundärdiagnosen zu bestimmten Primärdiagnosen ist bei der Auslösung zu berücksichtigen.

Primärmodule – Minimaldatensatz

Für Primärmodule sind in der Spezifikation für QS-Filter-Software Auslösebedingungen definiert. Das Modul MDS (Minimaldatensatz) verfügt über einen Sonderstatus: Es besitzt keinen definierten Auslöser für die Dokumentationspflicht. Das Modul MDS muss manuell als Ersatz für ein Primärmodul angelegt werden, sofern für dieses keine reguläre Dokumentation mit Fallabschluss möglich ist.

Sekundärmodule

Die im Modulnamen mit der Ergänzung FU gekennzeichneten Follow-up-Module⁸⁵, wie z. B. das Modul HTXFU (Follow-up-Herztransplantation), sind Sekundärmodule. Der Datensatz HTXFU ist jeweils nach 1, 2 oder 3 Jahren von demjenigen Krankenhaus zu dokumentieren, in dem die Transplantation erbracht worden ist. Da für das Follow-up kein neuer OPS-Kode erbracht worden ist, wird dieser Datensatz nicht direkt vom QS-Filter ausgelöst.

Mehrfachdokumentation

Pro Fall darf höchstens ein Datensatz eines Moduls angelegt und exportiert werden, wenn `mehrfachDokumentation = FALSCH` ist (Abschnitt B 1.4).

Beispiel für in `mehrfachDokumentation` festgelegte Regelungen:

Werden während eines stationären Aufenthalts bei einer Patientin zwei Brustoperationen durchgeführt, so dürfen hierfür nicht zwei Datensätze für das Modul 18/1 für eine Patientin angelegt werden. Stattdessen sind die Operationen in mehreren Teildatensätzen **eines** Datensatzes zu dokumentieren.

Die Dokumentation aller notwendigen Teildatensätze soll durch die Software sichergestellt werden.



Achtung

Die QS-Dokumentationssoftware muss sicherstellen, dass die Mehrfachdokumentation gleichartiger Datensätze für einen Fall unterbunden wird, sofern diese nicht zulässig ist. Stattdessen sind separate Teildatensätze zu dokumentieren

2.3.2 Teildatensätze

Die Begriffe „Teildatensatz“ und „Bogen“ werden synonym gebraucht. In den der Illustration dienenden Dokumentationsbögen werden alle Teildatensätze aufgeführt. Dabei erfolgt eine chronologische Anordnung, was dazu führen kann, dass ein Teildatensatz durch einen anderen, hierarchisch untergeordneten Teildatensatz unterbrochen wird. Manche Teildatensätze (z. B. Kindteildatensatz in Geburtshilfe, 16/1) müssen unter bestimmten Umständen mehrfach pro

⁸⁵ Gemeint sind die Follow-up-Datensätze zu Transplantationsmodulen (ohne PID-Daten).

Datensatz ausgefüllt werden. Diese mehrfach dokumentierbaren Teildatensätze sind im Muster-Dokumentationsbogen mit entsprechendem Hinweis nur einmal abgebildet.

Ein Teildatensatz

- ist jeweils einem Modul zugeordnet,
- besitzt einen Namen, der innerhalb eines Moduls eindeutig ist,
- kann unter definierten Bedingungen mehrfach pro Fall erzeugt werden.

Die Teildatensätze der QS-Spezifikation sind in der Tabelle *Bogen* definiert (Tabelle 41).

Tabelle 41: Struktur der Tabelle *Bogen*

| Feldname | Datentyp | Bemerkung |
|----------------------|------------|---|
| idBogen | INTEGER | Primärschlüssel |
| name | TEXT | Technischer Name des Teildatensatzes |
| bezeichnung | TEXT | Beschreibender Text |
| existenzBedingung | MEMO | Logische Bedingung (Regeln für das Anlegen von Teildatensätzen) |
| sortierNr | INTEGER | Reihenfolge der Unterbögen bei der Erfassung und beim Export |
| fkModul | INTEGER | Obligatorischer Fremdschlüssel zu einem Modul |
| fkBogenZahl | TEXT (1) | Anzahl der auszufüllenden Teildatensätze pro Patient (bezogen auf den Basisbogen oder ggf. auf den Mutterteildatensatz) |
| fkMutterBogen | INTEGER | Optionalen Fremdschlüssel, welcher den Mutterteildatensatz eines Teildatensatzes definiert |
| fkBogenTyp | TEXT (1) | Spezifiziert den für den Export relevanten Bogen-typ: Mögliche Werte B, K oder O. Die Angabe ist obligatorisch. |
| fkEindeutigBogenFeld | INTEGER | Fremdschlüssel auf ein Bogenfeld, das mehrfach vorhandene Teildatensätze eines Datensatzes identifiziert |

Benennung von Teildatensätzen

Ein Teildatensatz wird durch die folgende Kombination von Modulnamen und Bogenamen identifiziert und angesprochen:

<Modul.name> : <Bogen.name>

Beispiele:

PNEU : B ist der Basisbogen des Moduls Ambulant erworbene Pneumonie

PCI : PCI ist der PCI-Teildatensatz des Moduls Perkutane Koronarintervention und Koronarangiographie

HCH : O ist der Teildatensatz Operation des Moduls Herzchirurgie

18/1 : O ist der Teildatensatz Operation des Moduls Mammachirurgie

Bogentyp

Der Kerndatensatz besteht aus mindestens einem Basisteildatensatz und kann durch einen oder mehrere Teildatensätze ergänzt werden. Das Attribut `Bogen.fkBogenTyp` definiert für jeden Teildatensatz seine Rolle im und seine Zugehörigkeit zum Kerndatensatz. In Tabelle 42 sind die Bezeichnungen der einzelnen Bogentypen definiert.

Tabelle 42: Inhalte der Tabelle `BogenTyp`

| idBogenTyp | Bezeichnung |
|------------|--|
| B | Basisteildatensatz (Teil des Kerndatensatzes) |
| K | Teildatensatz ist Teil des Kerndatensatzes und kein Basisteildatensatzes |
| O | Teildatensatz ist Teil des optionalen Datensatzes |

Hierarchie von Teildatensätzen

Der Ausgangspunkt („root“) für die Teildatensatzhierarchie eines Moduls ist immer der Basisteildatensatz (Wert B des Attributs `Bogen.fkBogenTyp`). Ein abhängiger Teildatensatz besitzt einen Mutterteildatensatz, der über das Attribut `fkMutterBogen` der Tabelle `Bogen` definiert ist.⁸⁶

Auf diese Weise lässt sich für jedes Modul ein „Hierarchiebaum“ der Teildatensätze aufbauen.

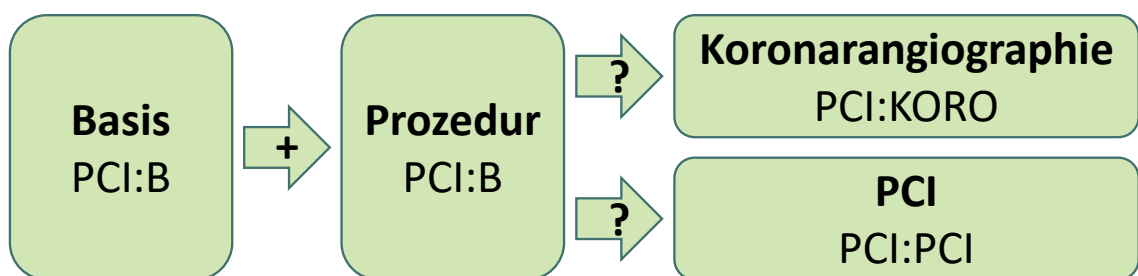


Abbildung 16: Teildatensatzstruktur des Datensatzes PCI

⁸⁶ Falls der Mutterteildatensatz nicht über das Attribut `fkMutterBogen` explizit definiert ist, so gilt implizit der Basisteildatensatz des Moduls als Mutterteildatensatz.

Das Modul `PCI` (Abbildung 16) enthält die Teildatensätze:

- `PCI:B` = Basis
- `PCI:PROZ` = Prozedur
- `PCI:KORO` = Koronarangiographie
- `PCI:PCI` = PCI

In der Tabelle `Bogen` sind folgende Bezüge zum Mutterteildatensatz definiert:

- `PCI:B` hat keinen Mutterteildatensatz
- `PCI:PROZ` hat den Mutterteildatensatz `PCI:B`
- `PCI:KORO` hat den Mutterteildatensatz `PCI:PROZ`
- `PCI:PCI` hat den Mutterteildatensatz `PCI:PROZ`

Regeln für das Anlegen von Teildatensätzen

Jedes Modul muss die Definition genau eines Basisteildatensatzes enthalten (Wert `B` des Attributs `fkBogenTyp` der Tabelle `Bogen`). Wenn die Dokumentation eines Moduls durchgeführt wird, muss der Basisteildatensatz genau einmal angelegt werden (z. B. in der Exportdatei). Das Attribut `fkBogenZahl` gibt Auskunft darüber, wie oft ein Teildatensatz pro Vorgang angelegt werden darf. Folgende Werte des Attributs sind möglich:

- `1` = Genau ein Teildatensatz muss ausgefüllt werden
- `+` = Mindestens ein Teildatensatz muss ausgefüllt werden
- `?` = Höchstens ein Teildatensatz darf ausgefüllt werden
- `*` = Eine beliebige Anzahl von Teildatensätzen kann ausgefüllt werden

Die Kardinalität eines abhängigen Teildatensatzes bezieht sich auf den Mutterteildatensatz. Der Basisteildatensatz hat immer die Kardinalität `1`.

Die Ausprägung `fkBogenZahl = *` definiert eine 1-n-Beziehung. Man beachte, dass das Attribut `fkBogenZahl` wichtig für den XML-Aufbau des QS-Datensatzes ist und im Schema Berücksichtigung findet.

Beispiele:

Der Teildatensatz `18/1:B` muss als Basisteildatensatz genau einmal ausgefüllt werden (`fkBogenZahl = 1`).

Der Teildatensatz `HCH:O` muss mindestens einmal pro Datensatz angelegt werden (`fkBogenZahl = +`).

Der Teildatensatz `PCI:PCI` im Modul `Perkutane Koronarintervention und Koronarangiographie` muss nur dann angelegt werden, wenn auch wirklich eine PCI durchgeführt wurde. Es kann also eine beliebige Anzahl von Teildatensätzen angelegt werden. Trotzdem gilt `fkBogenZahl = ?`, da – bezogen auf jeden Mutterteildatensatz `PCI:PROZ` – maximal ein Teildatensatz existieren darf.

Man beachte weiterhin, dass die im Attribut `fkBogenZahl` der Tabelle `Bogen` definierten Kardinalitäten durch Definitionen in den nachfolgend beschriebenen Attributen `existenzBedingung` bzw. `fkEindeutigBogenFeld` eingeschränkt werden können.

Inhaltliche Voraussetzung für das Anlegen von Teildatensätzen

Das Attribut `existenzBedingung` ist eine logische Bedingung (Syntax gemäß Abschnitt B 2.4.2) für das Anlegen eines Teildatensatzes. Die referenzierten Bogenfelder der Existenzbedingung beziehen sich auf den Mutterteildatensatz.

Die Dokumentationssoftware muss die Existenzbedingung als Trigger für das Anlegen eines abhängigen Teildatensatzes nutzen. Wenn die Existenzbedingung eines potenziellen Kindteildatensatzes erfüllt ist, so muss der Kindteildatensatz auch angelegt und übermittelt werden.



Hinweis

Beim Anlegen von Prozedurbögen ohne Existenzbedingung können Softwareprodukte die auslösenden OPS-Kodes als Existenzbedingung heranziehen. Bei einer solchen Umsetzung ist darauf zu achten, dass auslösende Codes innerhalb einer Operation (pro OPS-Datum) nicht zur fehlerhaften Mehrfachdokumentation von Teildatensätzen führen dürfen.

Andererseits gilt: Wenn die entgegennehmende Stelle einen Kindteildatensatz erhält, für den die zugehörige Existenzbedingung im Mutterteildatensatz nicht erfüllt ist, so ist das eine relationale Plausibilitätsverletzung.

Beispiel:

Modul `PCI` :

Der Teildatensatz `PCI : KORO` darf nur innerhalb eines Vorgangs angelegt werden, wenn im zugehörigen Mutterteildatensatz `PCI : PROZ` folgende Bedingung erfüllt ist:

`ARTPROZEDUR IN (1;3)`

Wenn ein Benutzer im Feld `ARTPROZEDUR` den Code 1 (`isolierte Koronarangiographie`) auswählt, so

- muss der abhängige Teildatensatz `PCI : KORO` angelegt werden,
- darf der Teildatensatz `PCI : PCI` nicht angelegt werden,
- muss ein bereits angelegter Teildatensatz `PCI : PCI` wieder gelöscht werden.

Zählleistungsbereiche

Zählleistungsbereiche erfassen einzelne Teilmengen indem sie prüfen, ob ein Fall eine bestimmte Operation beinhaltet. Wird für einen Wechseleingriff in der Hüftendoprothesenversorgung ein OPS-Kode kodiert, wird im Zählleistungsbereich `HEP_WE` ein Fall mit entsprechendem Prozedurbogen erwartet.

Das folgende Beispiel zeigt einen Fall, bei dem während eines Aufenthaltes sowohl eine Implantation als auch ein Wechseleingriff durchgeführt wurde und deswegen zwei Prozeduren in den QS-Daten erwartet werden.

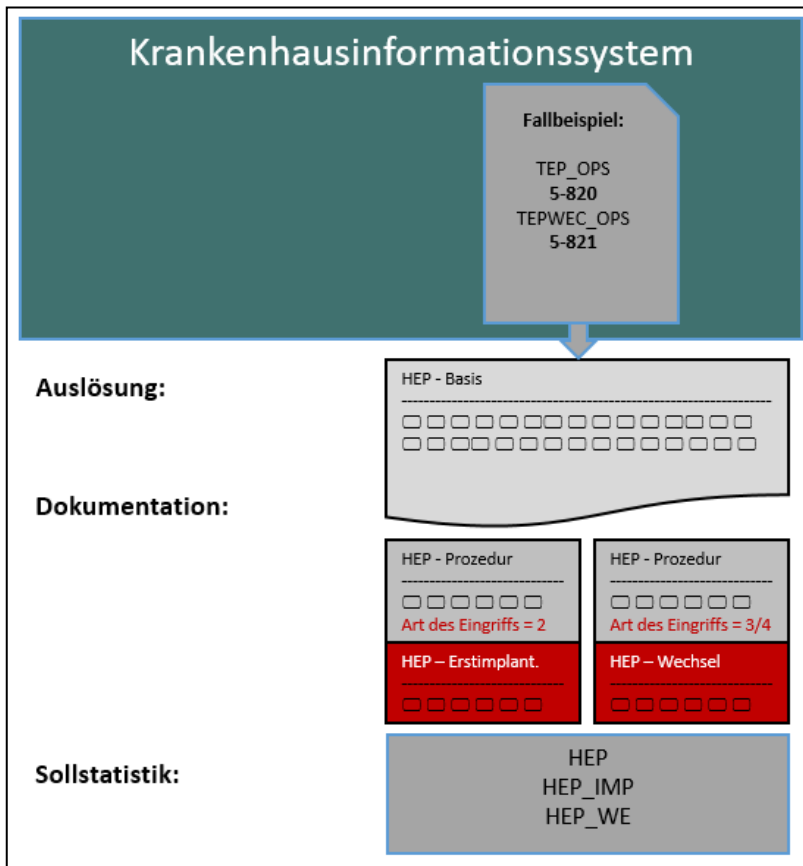


Abbildung 17: Fallbeispiel HEP in Bezug auf Dokumentation und Sollstatistik

Bei dieser Konstellation wird der Fall in der Sollstatistik sowohl bei HEP als auch bei den Zählleistungsbereichen HEP_IMP und HEP_WE gezählt.

Identifizierende Attribute mehrfach vorhandener Teildatensätze

Teildatensätze, die mehr als einmal ausgefüllt werden dürfen (Werte + und * des Attributs fkBogenZahl), sind nicht mehr durch die Vorgangsnummer voneinander unterscheidbar. Diese Teildatensätze benötigen ein zusätzliches identifizierendes Bogenfeld, das im Attribut fkEindeutigBogenFeld festgelegt wird. Beim Teildatensatz PCI:PCI ist es das Bogenfeld LFDNRPCI.

Beim Anlegen einer Tabelle für die Speicherung eines mehrfach vorhandenen Teildatensatzes muss der Primärschlüssel mindestens die Attribute Vorgangsnr⁸⁷, VersionsNr⁸⁸ und das in fkEindeutigBogenFeld definierte Feld umfassen.

⁸⁷ Bei den Zusatzfeldern ist zu beachten, dass die Feldnamen beim Export durch die entsprechenden XML-Elemente zu ersetzen sind.

⁸⁸ Bei der entgegennehmenden Stelle kommt noch das Feld RegistrierNr hinzu, da dort Datensätze verschiedener Krankenhäuser gesammelt werden.

Beispiel:

Modul HCH :

Die Follow-up-Dokumentation⁸⁹ des Moduls HCH ist freiwillig. Daher hat der Teildatensatz HCH:FU die Kardinalität (?). Höchstens ein Bogen darf ausgefüllt werden. Die Software muss dem Anwender alle freiwillig auszufüllenden Teildatensätze zur Verfügung stellen.

Wenn es bei den Teildatensätzen mehr als eine Ebene gibt, muss der Wert des Attributs `fkEindeutigBogenFeld` eines Kindbogens eindeutig in Bezug auf den übergeordneten Bogen sein. Hierbei kann sich die Eindeutigkeit des Wertes auf den Elternbogen (z. B. den Prozedurbogen) beschränken, sodass die Kombination beider Werte in Bezug auf den gesamten Vorgang eindeutig ist. Diese Bedingung wird auch erfüllt, wenn das Attribut `fkEindeutigBogenFeld` in Bezug auf den übergeordneten Basisbogen und damit auf den gesamten Vorgang eindeutig ist.

Beispiel (PCI):

Die zweite PCI während des Aufenthaltes kann auch als erste PCI der zweiten Prozedur interpretiert werden. Die eindeutige Zuordnung des PCI-Teildatensatzes zu einer Prozedur (Mutterbogen) erfolgt über das Attribut `IdBogenFeldMutter`. In Bezug auf die `Vorgangsnr` des Basisbogens lässt sich der einzelne PCI-Teildatensatz daher anhand der Attribute `LFDNRPCI` (`fkEindeutigBogenFeld` des Teildatensatzes) und `IdBogenFeldMutter` eindeutig identifizieren.

2.3.3 Datenfelder (Bogenfelder)

Jedes auf einem Teildatensatz vorhandene und auszufüllende Feld wird als Datenfeld (Item, Bogenfeld) bezeichnet. Datenfelder sind charakterisiert durch ihren Namen (Bezeichnung) und die Spezifikation des einzutragenden Inhalts.

Die Bezeichnung⁹⁰ wird so gewählt, dass sie einem medizinischen Experten unmittelbar verständlich ist. Die Spezifikation des Inhalts umfasst hingegen sowohl eine fachliche (medizinische) als auch datentechnische Typisierung. Dagegen repräsentieren die in der Tabelle `Feld` aufgelisteten Felder inhaltlich gleiche Dokumentationsfelder mehrerer Module (Abschnitt B 2.3.1), der datentechnische Typ (`BasisTyp`) charakterisiert das Format des Feldes (Abschnitt B 2.3.1).

Jedes Datenfeld hat zwingend einen Bezug zu einem Teildatensatz und zu einem technischen Feld. Weitere Eigenschaften sind die Bogenfeldbezeichnung und die fortlaufende Nummer im Teildatensatz. Die Datenfelder sind in der Tabelle `BogenFeld` gespeichert.

Identifizierendes Merkmal eines Datenfelds ist eine Kombination aus `fkBogen` und `fkFeld`. Das bedeutet, dass das Datenbankschema gewährleistet, dass der technische Feldname (`Feld.name`) in einem Teildatensatz maximal einmal vorkommt. Per definitionem muss ein

⁸⁹ Im Modul HCH erfolgt eine freiwillige Follow-up-Dokumentation ohne PID-Daten mit Hilfe eines eigenen Teildatensatzes.

⁹⁰ Gegebenenfalls im Kontext der Überschriften (Abschnitt B 1.3.4).

Datenfeldname sogar innerhalb eines Moduls eindeutig sein, d.h. dass eine Abfrage mit dem Primärschlüsselpaar (`modulNr`, `feldNr`) genau einen Primärschlüssel `idBogenFeld` liefert.

Tabelle 43: Struktur der Tabelle `BogenFeld`

| Feldname | Datentyp | Bemerkung |
|-------------------------------------|-------------|--|
| <code>idBogenFeld</code> | INTEGER | Primärschlüssel |
| <code>zeileAufBogen</code> | DOUBLE | bestimmt die Reihenfolge von Datenfeldern im Dokumentationsbogen |
| <code>gliederungAufBogen</code> | TEXT | Gliederungsnummer, die im Dokumentationsbogen angezeigt wird |
| <code>bezeichnung</code> | TEXT | Beschreibender Text zum Feld auf dem Dokumentationsbogen |
| <code>ergaenzendeBezeichnung</code> | TEXT | Optionale ergänzende Bezeichnung zu einem Bogenfeld. |
| <code>elemente</code> | INTEGER | Anzahl der Elemente bei Listenfeldern |
| <code>fkFeld</code> | INTEGER | Fremdschlüssel zu dem Teildatensatz und zu dem Feld, bilden zusammen die identifizierenden Merkmale |
| <code>fkBogen</code> | INTEGER | |
| <code>fkMussKann</code> | TEXT (1) | M oder K, Unterscheidung zwischen Muss- und Kann-Feldern |
| <code>min</code> | DOUBLE | Harte Untergrenze des Wertebereichs eines numerischen Datenfeldes (modulspezifisch). Die Definition ist optional. |
| <code>max</code> | DOUBLE | Harte Obergrenze des Wertebereichs eines numerischen Datenfeldes (modulspezifisch). Die Definition ist optional. |
| <code>minWeich</code> | DOUBLE | Weiche Untergrenze des Wertebereichs eines numerischen Datenfeldes (modulspezifisch). Die Definition ist optional. |
| <code>maxWeich</code> | DOUBLE | Weiche Obergrenze des Wertebereichs eines numerischen Datenfeldes (modulspezifisch). Die Definition ist optional. |
| <code>ahinweis</code> | TEXT (32) | Name des HTML-Ausfüllhinweises ohne Endung <code>.htm</code> (Abschnitt B 2.3.5) |

Muss- und Kann-Felder

Jedes Bogenfeld ist als Muss- oder Kann-Feld zu deklarieren:

- Ein Muss-Feld (M) muss innerhalb eines angelegten Teildatensatzes immer ausgefüllt sein (Abschnitt B 2.3.2).⁹¹

⁹¹ In jedem Muss-Feld muss für jeden angelegten Teildatensatz einmal eine Angabe erfolgen.

- Kann-Felder (\mathbb{K}) sind optionale Felder.
- Abhängige Muss-Felder (\mathbb{K}) müssen nur unter bestimmten Bedingungen ausgefüllt werden. Wenn also logische Sachverhalte dem Ausfüllen von Kann-Feldern entgegenstehen, so dürfen sie nicht ausgefüllt werden. Diese Felder unterliegen Feldgruppenregeln und verfügen wie optionale Felder über den Attributwert \mathbb{K} .

Anzahl der Elemente von Listenfeldern

Das Attribut `elemente` ist nur relevant bei von Listenfeldern (vgl. Attribut `istListe` der Tabelle `Feld`) abgeleiteten Bogenfeldern (Bogenfeldlisten). Es gibt die Größe der Bogenfeldliste an. Wenn für eine Bogenfeldliste das Attribut `elemente` leer ist, so ist die Größe per Definition 1.

Wenn ein Listenfeld als Muss-Feld deklariert ist, so ist nur das erste Exportfeld der Liste ein Muss-Feld, die restlichen Elemente sind Kann-Felder. Wenn ein Listenfeld als Kann-Feld deklariert ist, so sind alle weiteren exportierten Elemente ebenfalls Kann-Felder.

Felder – ein erster Schritt zur Prozess- und Datenintegration

Die Tabelle `Feld` (Tabelle 44) erleichtert dem Softwarehersteller den Abgleich seines Datenmodells mit dem Datenmodell des IQTIG. Gleiche Informationen in der Menge aller Dokumentationsbögen müssen dadurch nicht redundant abgebildet werden.

Beispielsweise taucht das Feld `ENTLGRUND` (Entlassungsgrund) in den meisten Modulen auf. Um die mehrfache Pflege dieser Felder zu vermeiden, wird ein Feld mit dem Namen `ENTLGRUND` definiert und jeweils nur noch in der Tabelle `BogenFeld` referenziert.

Jedem Feld ist zwingend ein Basistyp zugeordnet (Abschnitt B 2.3.1). Bei Schlüsselfeldern muss auch ein Schlüssel assoziiert sein. Im Gegensatz zu den (technischen) Basistypen enthalten die Felder die medizinisch-fachliche Information der Datenfelder. Der fachliche Inhalt wird durch den Text im Attribut `bezeichnung`⁹² beschrieben.

Identifizierendes Attribut eines Felds ist allein sein technischer Name (Attribut `name`). Dies ist wichtig für die Eindeutigkeit von Feldnamen innerhalb eines Moduls: Felder mit unterschiedlichen Typen oder unterschiedlichen Schlüsseln müssen unterschiedliche Namen haben.

Ein Feld kann als Skalar oder als Liste definiert sein. Diese Eigenschaft wird über das Attribut `istListe` gesteuert. Jedes von einem Listenfeld abgeleitete Bogenfeld ist automatisch eine Liste.⁹³ Die Anzahl der Elemente des von einem Feld abgeleiteten Bogenfelds wird über das Attribut `elemente` der Tabelle `BogenFeld` gesteuert.

⁹² Das Attribut `bezeichnung` ist ein Standardtext für das gleichnamige Attribut der Tabelle `BogenFeld`. Im Eingabeformular wird die Bezeichnung aus der Tabelle `BogenFeld` angezeigt.

⁹³ Man beachte die Besonderheiten der Listenfelder beim Datenexport und in der Syntax der Plausibilitätsregeln.

Beispiel:

Das Feld AUFNDIAG (Aufnahmediagnosen) ist als Liste definiert.

Im Modul 15/1 enthält das entsprechende Bogenfeld fünf Elemente (BogenFeld.elements), im Modul 16/1 hat das Bogenfeld lediglich ein Element.

Insbesondere für die Verwendung der richtigen Operatoren in den Plausibilitätsregeln und Feldgruppen ist die Listendefinition eines Felds wichtig.

Grundsätzlich gilt: Die Festlegung, ob ein Bogenfeld ein Skalar oder Listenfeld ist, wird durch die Tabelle `Feld` vorgegeben. Alle von einem Listenfeld abgeleiteten Bogenfelder sind automatisch auch Listenfelder. Die Größe der Liste wird individuell in der Tabelle `BogenFeld` konfiguriert.

Die Tabelle `Feld` bietet über die „Bogensicht“ hinausgehende Informationen.

Tabelle 44: Struktur der Tabelle `Feld`

| Feldname | Datentyp | Bemerkung |
|-----------------|----------|---|
| idFeld | INTEGER | Primärschlüssel |
| name | TEXT | Technischer Name |
| bezeichnung | TEXT | (Erlaubte Zeichen: A-Z, 0-9, Ziffer nicht am Anfang) Beschreibender Text auf dem Dokumentationsbogen (Standardwert für gleichnamiges Feld in Tabelle <code>BogenFeld</code>) |
| laenge | INTEGER | Anzahl der Zeichen in der Feldeingabemaske, enthält beim Typ ZAHL auch das Komma, bei SCHLUESSEL die Trennzeichen |
| einheit | TEXT(50) | Einheit des Felds (z. B. mm, Stunden) |
| formatAnweisung | TEXT | Regulärer Ausdruck für die Formatprüfung (z. B. <code>[0-9]{9}</code>) |
| funktion | TEXT | Formel zur Generierung des Feldinhaltes, z. B. durch Aufruf einer Syntaxfunktion |
| nachKommaLaenge | INTEGER | Anzahl der Nachkommastellen in der Feldeingabemaske (muss kleiner als <code>laenge</code> sein) |
| min | DOUBLE | Harte Untergrenze des Wertebereichs eines numerischen Datenfelds (modulübergreifend). Die Definition ist optional. |
| max | DOUBLE | Harte Obergrenze des Wertebereichs eines numerischen Datenfelds (modulübergreifend). Die Definition ist optional. |

| Feldname | Datentyp | Bemerkung |
|--------------|----------|---|
| minWeich | DOUBLE | Weiche Untergrenze des Wertebereichs eines numerischen Datenfelds (modulübergreifend). Die Definition ist optional. |
| maxWeich | DOUBLE | Weiche Obergrenze des Wertebereichs eines numerischen Datenfelds (modulübergreifend). Die Definition ist optional. |
| istListe | BOOLEAN | Wenn <code>istListe = WAHR</code> , so sind die vom betreffenden Feld abgeleiteten Bogenfelder Listenfelder. |
| fkBasisTyp | INTEGER | Fremdschlüssel zur Tabelle Basistypen |
| fkSchluessel | INTEGER | Fremdschlüssel zur Tabelle Schlüsseltypen |
| fkKombiFeld | INTEGER | Optionaler Fremdschlüssel auf ein anderes Feld, welches Kombinationsfelder kennzeichnet |

Kombinationsfelder

Für manche Bogenfelder ist zwingend vorgeschrieben, dass sie innerhalb eines Moduls in Kombination mit einem anderen Bogenfeld existieren. Die Definition von Kombinationsfeldern geschieht mithilfe des optionalen Fremdschlüssels `fkKombiFeld` in der Tabelle `Feld`.

Basistypen

Das Hauptmerkmal eines Basistyps ist der technische Typ eines Eingabefelds (z. B. Zeichenkette, numerischer Typ, Datum usw.). Wichtiges Charakteristikum ist die Beschreibung des Eingabeformats. Die Basistypen sind Voraussetzung für die Beschreibung einer formalen Regelsyntax (Abschnitt B 2.4.2).

Das identifizierende Merkmal eines Basistyps ist sein technischer Name (Attribut `name`).

Tabelle 45: Struktur der Tabelle `Basistyp`

| Feldname | Datentyp | Bemerkung |
|--------------|----------|--|
| idBasistyp | INTEGER | Primärschlüssel |
| Name | TEXT | Technischer Name (muss eindeutig sein) |
| standardtyp | TEXT | Entsprechender Standarddatentyp |
| bezeichnung | TEXT | Beschreibender Text |
| format | TEXT | Formatdefinition, z. B. <code>TT.MM.JJJJ</code> beim Basistyp <code>Datum</code> |
| formatRegExp | TEXT | Regulärer Ausdruck für die Formatprüfung |

| Feldname | Datentyp | Bemerkung |
|--------------------|----------|---|
| stdLaenge | INTEGER | Vorschlagsfeld für das gleichnamige Feld in der Tabelle <code>Feld</code> (einschließlich Vorzeichen und Komma) |
| stdNachKommaLaenge | INTEGER | Vorschlagsfeld für das gleichnamige Feld in der Tabelle <code>Feld</code> |



Hinweis

- In Zeichenketten (Basistyp `TEXT`) sind alle Zeichen des ASCII-Formats mit einem Kode ≥ 32 erlaubt. Ausgenommen sind das Semikolon, die doppelten Anführungsstriche und Hochkommata.
- Es gibt zwei Arten von Schlüsseln: numerische und nichtnumerische.
- Das Komma trennt die Nachkommastellen, Vorzeichen `+` und `-` sind erlaubt.
- Das Datumstrennzeichen ist der Punkt.

Schlüssel

Identifizierendes Merkmal eines Schlüssels (Kodesystem) ist sein technischer Name. Die meisten Schlüsselcodes sind in der Tabelle `Schluesse1Wert` (Tabelle 47) definiert.

Tabelle 46: Struktur der Tabelle `Schluesse1`

| Feldname | Datentyp | Bemerkung |
|--------------------|----------|---|
| idSchluesse1 | INTEGER | Primärschlüssel |
| Name | TEXT | Technischer Name (muss eindeutig sein) |
| bezeichnung | TEXT | Beschreibender Text |
| extern | BOOLEAN | Zeigt an, ob der Schlüssel in der Tabelle <code>Schluesse1</code> (= <code>FALSCH</code>) oder in einer externen Tabelle gespeichert (= <code>WAHR</code>) ist. |
| externVerweis | TEXT | Verweis auf die Quelle des externen Schlüssels |
| Zahl | BOOLEAN | Wenn <code>WAHR</code> , sind die Werte im Attribut <code>code</code> der zugehörigen Schlüsselwerte als ganze Zahl kodiert, ansonsten als Zeichenkette. |
| sortierNrVerwendet | BOOLEAN | Flag, das anzeigt, ob für die Reihenfolge das Attribut <code>sortierNr</code> der Tabelle <code>Schluesse1Wert</code> herangezogen wird. |
| fkMutterSchluesse1 | INTEGER | Referenz auf einen übergeordneten Schlüssel. Beispielsweise enthält der Schlüssel <code>MaSarkome</code> ausschließlich Codes des Schlüssels <code>ICDO3Mamma</code> . Abgeleitete Schlüssel ent- |

| Feldname | Datentyp | Bemerkung |
|----------|----------|---|
| | | halten in der Regel keine Bezeichnungen (Datenbanktabelle Schlüsselwert), da diese bereits im „Mutterschlüssel“ definiert sind. |

Schlüsselcodes können auf zwei Arten interpretiert werden: Wenn das Attribut `zahl` gesetzt ist, so werden die Codes als ganze Zahl gedeutet, ansonsten werden sie als Zeichenketten interpretiert. In der Syntax der Plausibilitätsregeln werden die letztgenannten Codes in einfache Hochkommata gesetzt (Abschnitt B 2.4.2).

Beispiel:

Attribut `zahl` bei Schlüsselfeldern

- Felder des Basistyps NUMSCHLUESSEL haben das Attribut `zahl` = WAHR.
- Felder des Basistyps SCHLUESSEL haben das Attribut `zahl` = FALSCH. Es handelt sich um alphanumerische Schlüssel, die Buchstaben, Ziffern oder Sonderzeichen verwenden (z. B. `ypN0`). Hierbei kann es sich auch um Werte handeln, die lediglich Ziffern verwenden, aber mit führender Null beginnen (z. B. `01`).

Externe Schlüsselkataloge

Externe Schlüsselkataloge sind über das Attribut `extern` deklariert. Hinweise zu den Bezugsquellen sind in der Spalte `externVerweis` zu finden (z. B. www.dimdi.de). Diese externen Schlüsselkataloge werden nicht vom IQTIG bereitgestellt und somit auch nicht verantwortet.



Achtung

Der Softwareanbieter hat dafür Sorge zu tragen, dass die jeweils aktuellen externen Schlüsselkataloge in der Software verwendet werden.

Die datenentgegennehmenden Stellen müssen ebenfalls die aktuellen Schlüsselkataloge verwenden und fehlerhafte Datensätze abweisen.

Hinweise zu den Bezugsquellen sind in der Spalte `externVerweis` zu finden (z. B. www.dimdi.de). Ein Verweis auf eine Bezugsquelle kann unabhängig vom Attribut `extern` angegeben werden (siehe Schlüssel `EntlGrund`).

Der Fachabteilungsschlüssel (`Fachabt`) ist solch ein externer Schlüsselkatalog: Schlüsseländerungen bzw. Fortschreibungen müssen vom Softwareanbieter und von der datenentgegennehmenden Stelle selbstständig und zeitnah über die §301-Vereinbarung (www.dkgev.de) bezogen werden. Der Fachabteilungsschlüssel wird alphanumerisch interpretiert.

Der Schlüssel Entlassungsgrund (`EntlGrund`) basiert auf einem externen Schlüssel, der als Schlüssel 5 in Anlage 2 der §301-Vereinbarung definiert ist: Die 1. und 2. Stelle dieses §301-Schlüssels werden im Rahmen der QS-Spezifikation alphanumerisch kodiert (Attribut `zahl` = FALSCH). Dabei sind nicht alle Schlüsselwerte des Schlüssels 5 in Anlage 2 der §301-Vereinbarung für die QS-Dokumentation zulässig.

**Achtung**

Der Schlüssel `EntlGrund` ist kein externer Schlüssel (`extern = FALSCH`). Das bedeutet, dass die in der Spezifikation enthaltenen Werte in der Software zu verwenden sind, auch wenn diese von dem Entlassungsgrund nach §301 abweichen.

Die Schlüsselcodes sind in der Tabelle `SchluesseWert` enthalten. Spätere Schlüsseländerungen bzw. -fortschreibungen werden vom IQTIG zeitnah übernommen.

**Achtung**

Das Datenfeld `PERSONENKREIS` verwendet den Schlüssel `Personenkreis`⁹⁴. Da dieses Feld sowohl die Schlüsselwerte nach § 301 als auch nach § 295 (KVD-Datensatzbeschreibung) abbildet, ist kein externer Verweis im Attribut `Schlues-sel.externVerweis` hinterlegt.

Um die verschiedenen Quellen voneinander abgrenzen zu können, sind zwei separate Töchterschlüssel (`Personenkreis301`, `PersonenkreisKVDT`) hinterlegt, die jeweils über einen eindeutigen externen Verweis verfügen. Diese Töchterschlüssel (ohne hinterlegte Schlüsselwerte) sind keinem technischen Feld zugeordnet, sondern dienen lediglich der Zuordnung der im übergreifenden Schlüssel definierten Werte zu den verschiedenen technischen Anlagen.

Schlüsselwerte

Tabelle 47 gibt einen Überblick über die Datenbanktabelle `SchluesseWert`, in der die Codes und Bezeichnungen der Schlüssel hinterlegt sind. Identifizierendes Merkmal ist hier eine Kombination der Spalten `fkSchluesse` und `code`. Das bedeutet, dass jeder Schlüsselcode innerhalb eines Schlüssels nur einmal vorkommen darf.

Tabelle 47: Struktur der Tabelle `SchluesseWert`

| Feldname | Datentyp | Bemerkung |
|------------------------------|-------------|---|
| <code>idSchluesseWert</code> | INTEGER | Primärschlüssel |
| <code>fkSchluesse</code> | INTEGER | Fremdschlüssel zur Tabelle Schlüssel |
| <code>code</code> | TEXT (50) | Schlüsselcode (entweder numerisch oder alphanumerisch kodiert) |
| <code>bezeichnung</code> | TEXT | Textliche Definition des Schlüsselwertes |
| <code>sortierNr</code> | INTEGER | Optionale Angabe zur Reihenfolge der Schlüsselwerte: Wenn belegt, so ist diese Reihenfolge bei der Anzeige in der Erfassungssoftware einzuhalten. |

⁹⁴ Gemäß neuer Version der technischen Anlage zu Anlage 4a BMV-Ä der Kassenärztlichen Bundesvereinigung, sind die Schlüsselwerte der Quellen § 301 und § 295 mit Gültigkeit zum 1. Juli 2018 nicht mehr identisch. Der Schlüssel `Personenkreis` ist nicht als externer Schlüssel definiert.

Das Attribut `code` der Tabelle `SchluesseWert` ist ein Textfeld, das in Abhängigkeit vom Wert des Attributes `zahl` im zugeordneten Schlüssel entweder numerisch oder nichtnumerisch interpretiert wird. Wenn in einer Plausibilitätsregel (Abschnitt B 2.4.2 und B 2.4.7) Felder mit numerischen Schlüsseln (Basistyp `NUMSCHLUESSEL`) vorkommen, so werden bei der Evaluierung der Regel die Schlüsselcodes wie ganze Zahlen behandelt.

Sortierung der Codes

- Für die Codes (Attribut `SchluesseWert.code`) eines Schlüssels ist eine Sortierung definiert. Die Art der Sortierung wird über die Attribute `zahl` und `sortierNrVerwendet` der Tabelle `Schluesse` festgelegt.
- Numerische Sortierung: Wenn `sortierNrVerwendet = FALSCH` und `zahl = WAHR`, so sind die Codes nach der Spalte `code` der Tabelle `Schluesse` numerisch zu sortieren.
- Alphanumerische Sortierung: Wenn `sortierNrVerwendet = FALSCH` und `zahl = FALSCH`, so sind die Codes nach der Spalte `code` der Tabelle `Schluesse` alphanumerisch zu sortieren.
- Spezielle Sortierung: Wenn `sortierNrVerwendet = WAHR`, so sind die Codes nach den Werten in der Spalte `sortierNr` der Tabelle `Schluesse` numerisch zu sortieren.

Beispiel:

Das Datenfeld `pT` des Datensatzes 18/1 besitzt den Schlüssel `pTMamma`, für den die spezielle Sortierung (`sortierNrVerwendet = WAHR`) definiert ist (Tabelle 48).

Tabelle 48: Schlüssel mit spezieller Sortierung (`pTMamma`)

| <code>code</code> | <code>sortierNr</code> |
|---------------------|------------------------|
| <code>pT0</code> | 1 |
| <code>pTis</code> | 2 |
| <code>pTlmic</code> | 3 |
| <code>pTla</code> | 4 |
| <code>pTlb</code> | 5 |
| <code>pTlc</code> | 6 |
| ... | ... |

Suchfunktion bei Schlüsseln mit einer großen Anzahl von Codes

Bei Schlüsseln mit einer großen Anzahl von Codes (z. B. Schlüssel `ICDO3Mamma` mit 138 Einträgen) soll eine anwenderfreundliche Möglichkeit zur Auswahl der passenden Codes bereitgestellt werden. Die Umsetzung als Auswahlliste (z. B. Combobox) führt zu erhöhtem Dokumentationsaufwand, falls der Anwender über Pfeiltasten oder Schieberegler zum passenden Code navigieren muss. Ergänzend soll daher eine Suchfunktion realisiert werden, die eine Suche über die

Attribute `SchlüsselWert.code` oder `SchlüsselWert.bezeichnung` ermöglicht. Die zu realisierenden Anwendungsfälle werden in den folgenden Beispielen erläutert.

Beispiel: Suche über Kode

Der Anwender möchte beim Datenfeld `maligne Neoplasie` (Schlüssel `ICDO3Mamma`, Modul 18/1) einen ICDO3-Kode eingeben, der mit der Ziffernfolge „8523“ beginnt. Über ein geeignetes Suchfenster gelangt der Anwender zu einer Teilliste, die die nachfolgend aufgelisteten Codes und die hinterlegten Bezeichnungen anzeigt:

8523/3 = invasives duktales Karzinom gemischt mit anderen Karzinom-Typen

8523/6 = invasives duktales Karzinom gemischt mit anderen Karzinom-Typen, Metastase

8523/9 = invasives duktales Karzinom gemischt mit anderen Karzinom-Typen, unbestimmt ob Primärtumor oder Metastase

Beispiel: Suche über Bezeichnung

Der Anwender möchte beim Datenfeld `maligne Neoplasie` (Schlüssel `ICDO3Mamma`, Modul 18/1) einen ICDO3-Kode eingeben, dessen Bezeichnung die Zeichenfolge „Adenokarzinom“ enthält. Über ein geeignetes Suchfenster gelangt der Anwender zu einer Teilliste, die die nachfolgend aufgelisteten Codes und die hinterlegten Bezeichnungen anzeigt:

8140/3 = Adenokarzinom o.n.A.

8140/6 = Adenokarzinom-Metastase o.n.A.

8140/9 = Adenokarzinom o.n.A., unbestimmt ob Primärtumor oder Metastase

8211/3 = Tubuläres Adenokarzinom

8211/6 = Tubuläres Adenokarzinom, Metastase

8211/9 = Tubuläres Adenokarzinom, unbestimmt ob Primärtumor oder Metastase

8401/3 = Apokrines Adenokarzinom

8401/6 = Apokrines Adenokarzinom, Metastase

8401/9 = Apokrines Adenokarzinom, unbestimmt ob Primärtumor oder Metastase

8410/3 = Talgdrüsenadenokarzinom

8410/6 = Talgdrüsenadenokarzinom, Metastase

8410/9 = Talgdrüsenadenokarzinom, unbestimmt ob Primärtumor oder Metastase

8480/3 = Muzinöses Adenokarzinom

8480/6 = Muzinöses Adenokarzinom, Metastase

8480/9 = Muzinöses Adenokarzinom, unbestimmt ob Primärtumor oder Metastase

8572/3 = Adenokarzinom mit Spindelzellmetaplasie

8572/6 = Adenokarzinom mit Spindelzellmetaplasie, Metastase

8572/9 = Adenokarzinom mit Spindelzellmetaplasie, unbestimmt ob Primärtumor oder Metastase

8574/3 = Adenokarzinom mit neuroendokriner Differenzierung

2.3.4 Überschriften

Die Überschriften der Dokumentationsbögen in der Spezifikation sind in der Tabelle Abschnitt zu finden.

Tabelle 49: Struktur der Tabelle *Abschnitt*

| Feldname | Datentyp | Bemerkung |
|------------------|----------|---|
| idAbschnitt | INTEGER | Primärschlüssel |
| bezeichnung | TEXT | Text der Überschrift |
| ebene | INTEGER | Zeigt die Hierarchie der Überschriften an |
| fkStartBogenFeld | INTEGER | Fremdschlüssel auf das erste zur Überschrift gehörende Bogenfeld |
| fkEndeBogenFeld | INTEGER | Fremdschlüssel auf das letzte zur Überschrift gehörende Bogenfeld |

Zu jeder Überschrift ist angegeben, bei welchem Bogenfeld sie beginnt und bei welchem Bogenfeld sie endet. Über das Attribut *ebene* lassen sich auch Teilüberschriften realisieren. Ein Bogenfeld kann somit mehreren Überschriften zugeordnet sein.



Achtung

Die in der Spezifikationsdatenbank hinterlegten Überschriften sind in die Eingabemasken der QS-Dokumentationssoftware zu integrieren. Viele Datenfelder sind für den Anwender erst im Kontext der Überschriften verständlich.

2.3.5 Ausfüllhinweise

Die Ausfüllhinweise zu den Datenfeldern sind in einem separaten ZIP-Archiv enthalten, das nach dem Benennungsschema für Spezifikationskomponenten bezeichnet wird (Einleitung, Abschnitt 1.1). Jeder Ausfüllhinweis ist ein HTML-Dokument.

Beispiel:

Ausfüllhinweis IDNRPAT.htm

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
<title>IDNRPAT</title>
```

```

</head>
<body>
<!--BLOCKANFANG-->
<div class=„AH“><p>

```

Die (einrichtungsinterne) Identifikationsnummer wird dem Patienten von der Einrichtung zugewiesen. Sie verbleibt in der Einrichtung und wird nicht an die Datenannahmestelle übermittelt.

```

</p></div>
<!--BLOCKENDE-->
</body>
</html>

```

In der Spalte `ahinweis` der Tabelle `BogenFeld` ist festgelegt, welcher HTML-Ausfüllhinweis mit einem Datenfeld verknüpft ist:

`<aHinweis>.htm` = Name der HTML-Datei

Beispiel:

Das Bogenfeld 6028 (Spezifikation 2018) hat in der Spalte `ahinweis` den Eintrag „OPSCHLUESSEL“. Der zugeordnete Ausfüllhinweis des ZIP-Archivs heißt `OPSCHLUESSEL.htm`.

Wenn der Eintrag in `ahinweis` leer ist, so existiert für das betreffende Bogenfeld kein Ausfüllhinweis. Das Attribut `fkAhinweisTyp` lässt die Differenzierung drei verschiedener Arten von Ausfüllhinweisen zu:

Tabelle 50: Arten von Hinweistypen

| <code>fkAhinweisTyp</code> | Beschreibung | Beispiel |
|----------------------------|---|--|
| Feldbezogen | Der Ausfüllhinweis bezieht sich auf den entsprechenden Datensatz in der Tabelle <code>Feld</code> . Der Ausfüllhinweis ist modulunabhängig. | <code>IDNRPAT.htm</code> Der Ausfüllhinweis bezieht sich auf verschiedene Module, beispielsweise auf <code>HEP</code> , <code>KEP</code> und <code>PCI</code> . |
| Modulspezifisch | Soll sich ein Ausfüllhinweis nur auf ein bestimmtes Modul beziehen, kann der Attributwert <code>modulspezifisch</code> ausgewählt werden. | <code>IDNRPAT\$DEK.htm</code> Der Ausfüllhinweis bezieht sich nur auf das Modul <code>DEK</code> . |
| Speziell | Soll es für verschiedene Datenfelder der Tabelle <code>Feld</code> einen gemeinsamen Ausfüllhinweis geben, kann dieser als speziell deklariert werden. Der Attributwert <code>ahinweis</code> definiert den Namen des Ausfüllhinweises. | <code>NICHTGEMESSENVE.htm</code> Die Felder <code>ASONVEREIZN</code> und <code>ASONVERAMPN</code> haben denselben Ausfüllhinweis. |

Die Zuordnung von Bogenfeldern und Ausfüllhinweisen ist auch in der Abfrage `Ausfüllhinweise` dargestellt. Sie zeigt Modul/Teildatensatz, Zeile, Feldname, Bezeichnung und den HTML-Dateinamen des Ausfüllhinweises zu dem Bogenfeld. Im Gegensatz zur Tabelle `Bogenfeld` ist hier die Endung `.htm` mit angegeben.

2.4 Plausibilitätsprüfungen

Es wird zwischen drei Arten von Plausibilitätsprüfungen unterschieden, die in Tabelle `RegelTyp` definiert sind:

- harte Prüfungen
- weiche Prüfungen in der QS-Dokumentationssoftware
- warnende Prüfungen bei der Datenentgegennahme

Tabelle 51: Tabelle `RegelTyp`

| <code>idRegelTyp</code> | <code>bezeichnung</code> |
|-------------------------|----------------------------|
| D | Warnung Datenentgegennahme |
| H | hart |
| W | weich |

Weiterhin wird zwischen sogenannten **Einzelregeln** (Abschnitt B 2.4.4) und **Feldgruppenregeln** (Abschnitt B 2.4.7) unterschieden.

2.4.1 Die Regeltabelle

Die Bedingungen für unplausible Angaben⁹⁵ sind in der Tabelle `Regeln` abgelegt. Die hier beschriebenen Prüfungen sind in der Spezifikationsdatenbank für QS-Dokumentation hinterlegt. Die Syntax ist in Abschnitt B 2.4.4 beschrieben. Die Bedingungen sind möglichst kurz gefasst (Vermeidung von durch `ODER` verknüpften Teilbedingungen). Jede Bedingung kommt nur einmal innerhalb eines Moduls vor.

Tabelle 52: Struktur der Tabelle `Regeln`

| <code>Feldname</code> | <code>Datentyp</code> | <code>Bemerkung</code> |
|------------------------|-----------------------|--|
| <code>idRegeln</code> | INTEGER | Primärschlüssel |
| <code>fkModul</code> | INTEGER | Fremdschlüssel zur Tabelle <code>Modul</code> |
| <code>bedingung</code> | MEMO | Entsprechend der Syntax definierte Regeln |
| <code>meldung</code> | MEMO | Fehlermeldung: Diese Texte sind bei Regeln mit Bezug zu Feldgruppen automatisch generiert. |

⁹⁵ Eine Plausibilitätsregel müsste eigentlich „Unplausibilitätsregel“ heißen, weil sie unplausible Zustände beschreibt, die zu Fehlermeldungen führen.

| Feldname | Datentyp | Bemerkung |
|-------------------|----------|--|
| alternativMeldung | MEMO | Alternative Fehlermeldung: Wenn hier ein Text vorhanden ist, so ist dieser anstelle des Textes in der Spalte <code>meldung</code> zu verwenden. |
| fkMehrfachRegel | INTEGER | Fremdschlüssel zur Tabelle mit mehrfach vorkommenden Regeln, die mithilfe von Ersatzbedingungen nach dem Export gültig sind. |
| fkFeldGruppe | INTEGER | Optionaler Fremdschlüssel zur Tabelle <code>FeldGruppe</code> : Indikator dafür, ob eine Regel aus einer Feldgruppe generiert wurde. |
| fkRegelTyp | TEXT(1) | Fremdschlüssel zur Tabelle <code>RegelTyp</code> : Die Regeltypen sind die in Abschnitt A 2.4 beschriebenen Arten der Plausibilitätsprüfungen: H, W oder D |
| gueltigNachExport | BOOLEAN | Regeln, die den Wert <code>FALSCH</code> haben, können von Datenannahmestellen nicht evaluiert werden. Stattdessen werden die referenzierten Ersatzbedingungen der Tabelle <code>MehrfachRegel</code> evaluiert (falls definiert). |

Regelfelder (Bogenfelder einer Regel)

Die Tabelle `RegelFelder` (Tabelle 53) ist eine Verknüpfungstabelle zwischen den Tabellen `Regeln` und `BogenFeld`. Durch gezielte Abfragen erhält man unter Verwendung dieser Tabelle einen Überblick über Folgendes:

- Bogenfelder, die in einer Regel verwendet werden.
- Regeln, die sich auf ein Bogenfeld beziehen.

Tabelle 53: Struktur der Tabelle `RegelFelder`

| Feldname | Datentyp | Bemerkung |
|-------------|----------|--|
| fkBogenFeld | INTEGER | Fremdschlüssel zu den Tabellen <code>Feld</code> und <code>Regeln</code> , bilden zusammen den Primärschlüssel |
| fkRegeln | INTEGER | |

Mehrfachregeln (Ersatzregeln zur Prüfung nach dem Export)

Wenn in einer Regel von der Pseudonymisierung betroffene Datenfelder benutzt werden, so kann diese von Datenannahmestellen nicht evaluiert werden. Stattdessen wird für solche Regeln in der Tabelle `MehrfachRegel` eine Ersatzbedingung definiert, deren Referenzierung in der Tabelle `Regeln` definiert ist. Die Ersatzbedingung ist von den Datenannahmestellen zu evaluieren.

Tabelle 54: Struktur der Tabelle *MehrfachRegel*

| Feldname | Datentyp | Bemerkung |
|-----------------|----------|---|
| idMehrfachRegel | INTEGER | Primärschlüssel |
| bedingung | MEMO | Entsprechend der Syntax definierte Regeln |
| meldung | MEMO | Kontextunabhängige Fehlermeldung |
| ersatzBedingung | MEMO | Ersatzregel für den pseudonymisierten Datensatz |
| fkRegelTyp | TEXT(1) | Fremdschlüssel zur Tabelle <i>RegelTyp</i> : Die Regeltypen sind die in Abschnitt A 2.4 beschriebenen Arten der Plausibilitätsprüfungen: H, W oder D |

Beispiel:

Die Datenfelder der Regel 578 ($OPDATUM > ENTLDATUM$) werden nicht an die Datenannahmestellen übermittelt. Bei der Entgegennahme ist die in der Tabelle *MehrfachRegel* definierte Ersatzregel ($idMehrfachRegel = 133$) anzuwenden: $poopvwdauer < 0$.

Weitere Regeln

Weitere feldübergreifende Regeln sind die in Abschnitt B 2.3.2 beschriebenen Existenzbedingungen für das Anlegen von abhängigen Teildatensätzen (Attribut *existenzBedingung* in Tabelle *Bogen*).

2.4.2 Regelsyntax

Bedingungen sind in den Tabellen *Regeln*, *MehrfachRegel* und *Bogen* definiert. Die den Bedingungen zugrunde liegende Regelsyntax wird in diesem Abschnitt beschrieben. Jede Regel ist ein logischer Ausdruck, dessen Ergebnis WAHR oder FALSCH lautet. Jede Regel bezieht sich auf einen eingegebenen Datensatz eines Moduls, dessen Daten in Variablen gespeichert sind.

Die Regelsyntax lehnt sich an die logischen Ausdrücke in bekannten Programmiersprachen an. Jedoch haben die Operatoren deutsche Namen, z. B. UND statt AND oder ODER statt OR. Die Regelsyntax ist als Pseudocode zu verstehen.

Typen

Die möglichen Typen der Datenfelder sind in Tabelle 55 aufgelistet.

Tabelle 55: Basistypen der Datenfelder in den Plausibilitätsregeln

| Basistyp | Bezeichnung | Beispiele (Literele) |
|----------|-----------------------|----------------------|
| BOOL | Boolesche Variable | WAHR, FALSCH |
| TEXT | Zeichenkette (String) | "Spezifikation" |

| Basistyp | Bezeichnung | Beispiele (Literele) |
|-------------------------|--|----------------------|
| GANZEZAHL ⁹⁶ | ... -2, -1, 0, 1, 2, 3, ... | 1 |
| ZAHL | Zahl (mit oder ohne Nachkommastellen) | 25,4 oder -100,8 |
| DATUM | Zehnstelliges Datum | '01.01.2012' |
| MONDATUM | Monatsdatum | '04.2012' |
| QUARTDATUM | Quartalsdatum | '3/2012' |
| JAHRDATUM | Jahresdatum | 2012 |
| NUMSCHLUESSEL | Numerisch kodierter Schlüssel (wie GANZEZAHL) | 1 |
| SCHLUESSEL | Alphanumerischer Schlüssel | '19.1', '07' |
| UHRZEIT | Uhrzeit | '10:15' |

In der Spezifikation für die QS-Dokumentation wird zwischen NUMSCHLUESSEL und SCHLUESSEL unterschieden:

- Schlüsselwerte verfügen über den Datentyp NUMSCHLUESSEL, wenn es sich bei den Codes um ganze Zahlen handelt. Da dies ein numerischer Schlüssel ist, darf er nicht in Hochkommata gesetzt werden.
- Schlüsselwerte, die alphanumerische Codes beinhalten, haben den Basistyp SCHLUESSEL. Die OPS-Schlüssel (z. B. '5-282.0') und die ICD-10-GM-Schlüssel verfügen über diesen Datentyp, der in Hochkommata geschrieben wird.



Achtung

Datumsangaben (Datum, Monats-, Quartalsdatum) müssen in Hochkommata gesetzt werden. Eine Ausnahme ist das Jahresdatum (JAHRDATUM), da es sich hierbei um eine ganze Zahl handelt.

Felder

Feldnamen bestehen aus maximal 32 Zeichen und dürfen nur die Buchstaben A bis Z (Großbuchstaben) und die Ziffern 0 bis 9 enthalten. Ein Feldname muss immer mit einem Buchstaben beginnen. Umlaute und Sonderzeichen sind in Feldnamen nicht erlaubt. Ein Feldname darf kein reserviertes Wort sein (z. B. LEER).



Achtung

In einer Regel dürfen nur die Feldnamen der im betreffenden Modul definierten Bogenfelder⁹⁷ enthalten sein. Bei der Evaluierung von Regeln werden die aktuellen Werte der referenzierten Bogenfelder eingesetzt. Kann-Bogenfelder können auch un- ausgefüllt sein, also den Wert LEER haben.

⁹⁶ Beim Typ GANZEZAHL sind auch negative ganze Zahlen erlaubt.

⁹⁷ Bei den Ersatzregeln in Tabelle `MehrfachRegel` sind stattdessen die Exportfelder des Moduls erlaubt.

Listenfelder

Ein Bogenfeld wird dann als Liste interpretiert, wenn im referenzierten Feld (Tabelle `Feld`) der Wert des Attributs `Feld.istListe = WAHR` ist. Andernfalls ist das Bogenfeld ein Skalar. Bei der Formulierung von Regeln ist darauf zu achten, dass Listenfelder nicht bei jedem Operator als Operand fungieren können. Listenfelder dürfen z. B. nicht voneinander subtrahiert werden.

Beispiel:

Hinter dem Feld `SSBEFUND`, welches im Modul `16/1` als Bogenfeld vorkommt, verbirgt sich eine Liste mit 56 Elementen, die nachfolgend als Variable in einer Regel angesprochen wird:

```
SSBEFUNDE <> 1 UND SSBEFUND EINSNICHTIN (LEER)
```

Literale

Alphanumerische Literale (z. B. `SCHLUESSEL`) werden von einfachen Hochkommata eingeschlossen, während Zeichenketten vom Datentyp `TEXT` in Anführungszeichen gesetzt werden müssen.⁹⁸

Dies gilt nicht für numerische Literale (`GANZEZAHL`, `ZAHL`, `NUMSCHLUESSEL` und `JAHR-DATUM`) und Literale des Datentyps `BOOL` (Wahrheitswerte).

Beispiel für Regeln mit Literalen:

```
POKOMPLIKAT <> 1 UND PNEUMONIE <> LEER
AUFNVONSTATPFLEGE = 1 UND ENTLGRUND NICHTIN ('07'; '10')
aktuellesJahr() - jahreswert(GEBDATUM) > 100
```

Listen von Literalen

Literale können sowohl als Skalare als auch als Listen angesprochen werden. Der Separator einer Liste von Literalen ist das Semikolon. Um zu prüfen, ob alle Listenfelder ausgefüllt sind, wird die Liste über den Wert `LEER` angesprochen.

Beispiele für Listen von Literalen:

- Liste von Literalen vom Typ `GANZEZAHL` oder `NUMSCHLUESSEL`:
(1;2;3)
 - Liste von Literalen vom Typ `SCHLUESSEL` (alphanumerisch):
('5-740.0' ; '5-740.1' ; '5-740.y')
-

Längere Listen von Prozedurcodes (`OPS`) oder Diagnosecodes (`ICD-GM-10`) werden als Variable angesprochen, deren Namen einem festen Namensschema gehorchen. Diese Listen werden in

⁹⁸ Beim Export entfallen die begrenzenden Zeichen.

separaten Tabellen definiert, die den Variablennamen⁹⁹ (z. B. GYN_OPS) und die darin enthaltenen Prozedur- und Diagnosecodes beinhalten (Abschnitt B 2.5).

Beispiel:

In der Regel 8686 aus Modul 15 / 1 (Gynäkologische Operationen) wird die OPS-Liste GYN_OPS verwendet:

```
GYNZUSATZ = 1 UND OPSCHLUESSEL EINSIN GYN_OPS
```

Außerdem gibt es Teildatensatz-Listenfelder, die im Abschnitt B 2.4.6 beschrieben werden.

Operatoren

Tabelle 56 gibt einen Überblick über die in der Syntax zulässigen Operatoren. Der aktuelle Überblick über alle zulässigen Operationen (inkl. Operanden) ist in Tabelle `SyntaxOperator` in der QSDOK-Datenbank zu finden.

In Tabelle 56 hat jeder einzelne Operator eine Präzedenzstufe (höchste Präzedenzstufe ist 0). Operatoren, die die gleiche Stufe haben, werden nach den Regeln der Assoziativität aufgelöst.

Tabelle 56: Präzedenz und Assoziativität der Operatoren

| Präzedenz | Assoziativität | Operator | Erläuterung |
|-----------|----------------|-------------|---|
| 0 | links | IN | Operator zum Vergleich einer Variablen mit einer Liste (z. B. ein Datenfeld mit Schlüsselwerten). Die Variable und die Feldelemente müssen gleichen Typs sein. |
| | links | NICHTIN | |
| | links | EINSIN | Operator zum Vergleich einer Liste mit einer anderen Liste oder einem Listenelement (z. B. ein Listenfeld mit einem Schlüsselwert oder ein Listenfeld mit einer OPS-Liste). Die Listenelemente müssen gleichen Typs sein. |
| | links | JEDESIN | |
| | links | EINSNICHTIN | |
| | links | KEINSIN | |
| 1 | links | * | Operator für die Multiplikation „mal“ |
| | links | / | Operator für die Division „geteilt“ |
| 2 | links | + | Operator für die Addition „plus“ |
| | links | - | Operator für die Subtraktion „minus“ |
| 3 | links | < | Vergleichsoperator „kleiner“ |
| | links | > | Vergleichsoperator „größer“ |
| | links | <= | Vergleichsoperator „kleiner gleich“ |

⁹⁹ Der Variablenname ist synonym mit dem Listennamen (z. B. `OPSListe.name`) der Prozedur- bzw. Diagnoseliste.

| Präzedenz | Assoziativität | Operator | Erläuterung |
|-----------|----------------|----------|------------------------------------|
| | links | >= | Vergleichsoperator „größer gleich“ |
| 4 | links | = | Vergleichsoperator „gleich“ |
| | links | <> | Vergleichsoperator „ungleich“ |
| 5 | rechts | NICHT | Logischer Operator: „NICHT“ |
| 6 | links | UND | Logischer Operator: „UND“ |
| 7 | links | ODER | Logischer Operator: „ODER“ |

Prüfung auf LEER mit Vergleichsoperatoren

Die Prüfung auf LEER von in Regeln verwendeten Kann-Feldern, welche an anderer Stelle in der Regel mit einem anderen Operator als <> oder = geprüft werden sollen, findet auf der linken Seite einer ODER-Verknüpfung statt. Hintergrund dieser Syntaxregel ist, dass die Vermeidung von Laufzeitfehlern bei der Evaluation ermöglicht werden soll.

Beispiel:

FELD = LEER ODER FELD OPERATOR OPERAND

Beispielsweise kann bei leeren Feldwerten und der vorgegebenen Linksassoziativität des ODER-Operators die Evaluation bei leerem Feldwert vor der Evaluation des rechtsstehenden Ausdrucks mit der Rückgabe von WAHR abgebrochen werden. Ein Laufzeitfehler, der sich z. B. bei einem Vergleich von LEER < WERT ergeben würde, kann so nicht entstehen.

Operatoren zum Vergleich einer Variablen mit einer Liste

Folgende Operatoren erfordern entweder nur rechts oder links und rechts Listenfelder:

- nur rechts: IN, NICHTIN
- links und rechts: EINSIN, KEINSIN, JEDESIN, EINSNICHTIN

Operatoren mit beidseitigen Listenfeldern als Operanden:

- EINSIN: Wenn mindestens ein Element aus der linken Liste in der rechten Liste enthalten ist, so ist der Ausdruck wahr (nichtleere Schnittmenge).
- KEINSIN: Wenn kein Element der linken Liste in der rechten Liste enthalten ist, so ist der Ausdruck wahr (leere Schnittmenge). Dieser Operator ist redundant, da er auch durch Negation des EINSIN-Operators abgedeckt ist.
- JEDESIN: Der Ausdruck ist dann wahr, wenn jedes Element der linken Liste in der rechten Liste enthalten ist (Teilmenge).
- EINSNICHTIN: Der Ausdruck ist dann wahr, wenn mindestens ein Element der linken Liste nicht in der rechten Liste enthalten ist (nichtleere Differenz).

Beispiel:

- Die Operation `GANZEZAHL := DATUM1 - DATUM2` liefert als Ergebnis die Differenz zwischen zwei Kalenderdaten in Tagen.
- Die Operation `ZAHL := UHRZEIT1 - UHRZEIT2` liefert als Ergebnis die Differenz zwischen zwei Uhrzeiten in Minuten.

Beispiel:

Folgende Regel prüft, ob kein Element des Listenfeldes `OPSCHLUESSEL` (4 Elemente) einen bestimmten Code besitzt:

```
OPSCHLUESSEL KEINSIN ('5-983')
```

Wenn z. B. `OPSCHLUESSEL := ('5-661.3y';LEER;LEER;LEER)`, so ist die Regel erfüllt.

Gleichwertig ist die Regel:

```
NICHT OPSCHLUESSEL EINSIN ('5-983')
```

Eine Besonderheit bei Listenoperationen ist die Prüfung, ob alle Elemente einer Liste ausgefüllt sind:

Beispiel:

```
NICHT OPSCHLUESSEL JEDESIN (LEER)
```

Diese Bedingung erfordert, dass zumindest ein Listenelement ausgefüllt ist. Beispielsweise erfüllt `OPSCHLUESSEL := ('5-661.3y';LEER;LEER;LEER)` die Bedingung.

Gleichwertig ist die Regel:

```
OPSCHLUESSEL EINSNICHTIN (LEER)
```

Folgende Operatoren sind komplementär:

`IN` und `NICHTIN`

`EINSIN` und `KEINSIN`

`JEDESIN` und `EINSNICHTIN`

Folgende Ausdrücke sind gleich:

`A EINSNICHTIN B`

`NICHT A JEDESIN B`

Plausibilitätsprüfungen mit OPS- und ICD-Listen

Die OPS- und ICD-Listen enthalten ausschließlich Normcodes. Die vom Leistungserbringer dokumentierten Codes enthalten ggf. auch Zusatzkennzeichen (Bsp.: Seitenlokalisierung). Bei der Evaluation der Regeln werden die dokumentierten Zusatzkennzeichen ignoriert (Abschnitt B 2.5).

Beispiel:

Die OPS-Liste KAT OPS enthält unter anderem den Code 5-144.x5.

Die Evaluation der Regel OPSCHLUESSEL EINSIN KAT OPS führt auch dann zu einem positiven Ergebnis, wenn OPSCHLUESSEL = ('5-144.x5:R';LEER;LEER;LEER)

**Hinweis**

Wird das Zusatzkennzeichen direkt in der Plausibilitätsregel abgefragt, wird dieses bei der Evaluation der entsprechenden Regel nicht ignoriert. Hierbei kann es sich beispielsweise um die Prüfung der Diagnosesicherheit mithilfe der Funktion `format` handeln.

Beispiel

```
STATUSLE = 1 UND ARTLE IN (2;3) UND format(ENTLDIAG;'[a-zA-Z][0-9]{2}(\.[0-9]{1,2})?([#\*\+!])?([RLBr1b])?$',) <> WAHR
```

2.4.3 Funktionen

Eine Funktion ist gekennzeichnet durch ihren Namen, an den sich unmittelbar (ohne Leerzeichen) ein Listenausdruck anschließt. Funktionen ohne Übergabeparameter werden ähnlich wie in C oder Java durch ein Klammerpaar abgeschlossen. Funktionen können nicht nur in Regeln, sondern auch zur Berechnung von Exportfeldern genutzt werden (Abschnitt B 2.6.2).

Beispiel:

Das Ersatzfeld `entlquartal` wird mithilfe der Syntaxfunktion `quartal` berechnet:
`entlquartal = quartal(ENTL DATUM)`

Der aktuelle Stand der in der Syntax verwendeten Funktionen ist in der Tabelle `SyntaxFunktion` der Spezifikation zu finden.

In den nachfolgenden Beispielen gilt folgende Notation für Funktionen:

```
<BASISTYP> <FUNKTIONSNAME> ([ <BASISTYP> <VARNAME> { ;  
<BASISTYP> <VARNAME> } ] )
```

mit

- { } Wiederholung
- [] Option
- <BASISTYP> Basistyp der Variablen
- <VARNAME> Name der Variablen

Beispiele:

```
DATUM aktuellesDatum()
```

Funktion ohne Übergabeparameter und mit Ergebnistyp DATUM

```
DATUM Minimum(DATUM DATUMLISTE)
```

Funktion mit Ergebnis vom Typ DATUM, die das Minimum einer Liste von Datumsangaben (DATUMLISTE) liefert.

```
JAHRDATUM jahreswert(DATUM EINDATUM)
```

Funktion mit Ergebnis vom Typ JAHRDATUM

Es kommen auch verschachtelte Funktionsaufrufe (z. B. funktionA(funktionB())) oder arithmetische Ausdrücke als Funktionsargumente (z. B. funktion(x+y)) vor (Beispiel: gewichtssperzentile). Häufig wird nur die Signatur von Funktionen bereitgestellt.

Hinweise für die Implementierung von Funktionen

Als Hilfestellung für die Ausprogrammierung wird bei manchen Funktionen ein Pseudocode bereitgestellt. Der Pseudocode ergänzt die Syntax der Plausibilitätsregeln um folgende Sprachelemente:¹⁰⁰

- Befehlszeilen werden mit Semikolon abgeschlossen ;
- Wertzuweisungen mit dem Operator :=

```
A := B + C;
```

- Auswahlanweisungen

```
if (<Bedingung>){
...
}
else {
...
}
```

Hinter <Bedingung> verbirgt sich ein logischer Ausdruck, der der Syntax der Plausibilitätsregeln gehorcht.

- Blöcke werden durch geschweifte Klammern definiert.

```
{
...
}
```

- Innerhalb einer Funktion sind die Argumentvariablen verfügbar.

¹⁰⁰ Der Pseudocode erhebt nicht den Anspruch auf formale Korrektheit.

- Eine Variable, die den gleichen Namen wie die Funktion hat, muss am Ende mit `return` zurückgegeben werden.

Hinweise zur Funktion `gestAlter`

Die Funktion `gestAlter` berechnet das Gestationsalter in Tagen. Signatur und Pseudocode sind der Datenbank zur QS-Dokumentation zu entnehmen. Alternativ kann der Java-Kode herangezogen werden, der in der Spezifikationskomponente `erginformationen` enthalten ist (Abschnitt B).

Hinweise zur Funktion `gewichtsPerzentile`

Innerhalb der Funktion `gewichtsPerzentile` wird die Funktion `gestAlter` aufgerufen. Der Rückgabewert dieser Funktion (das Gestationsalter in Tagen) wird bei Aufruf der Funktion `gewichtsPerzentile` durch 7 dividiert. Da die Division durch 7 einen Dezimalwert zurückliefern kann, wird bei Aufruf der Funktion `gewichtsPerzentile` gleichzeitig eine `trunc`-Funktion verwendet, sodass die Schwangerschaftswoche als ganze Zahl zurückgegeben wird (vgl. Regel-ID 13988, 13989). Anhand des Gestationsalters und der Angaben in den Feldern `GESCHLECHTK`, `ANZMEHRLINGE` und `KG` wird ermittelt, ob das Geburtsgewicht eines Kindes ober- oder unterhalb der 97. bzw. 3. Perzentilgrenzen liegt.

Beispiel:

Ein Mädchen (Mehrling), für das ein Gestationsalter von 161 Tagen ermittelt wird (also 23. SSW), wiegt 721 g. Die Funktion `gewichtsPerzentile` liefert in diesem Fall den Wert „2“, da das Gewicht des Mädchens die 97. Perzentilgrenze (> 720 g) überschreitet.

Hinweise zur Funktion `format (Feld, pattern)`

Die Funktion prüft, ob der erste Parameter (`Feld`) dem regulären Ausdruck (`pattern`) entspricht. Gibt es eine Übereinstimmung, gibt die Funktion ein `WAHR` zurück.

Beispiel:

```
format (STANDORT, [0-9]{2})
```

[0-9]: der Wert darf nur die Zahlen 0–9 enthalten

{2}: der Wert muss genau 2-stellig sein

Die konkrete Implementierung dieser Funktion ist von der eingesetzten Programmiersprache abhängig.

| JAVA | C# | C++ |
|-------------------------------------|---|--|
| <code>feld.matches (pattern)</code> | <code>Regex.IsMatch (feld,pattern)</code> | <code>Regex::IsMatch (feld,pattern)</code> |

Hinweise zur Funktion `verkettenmt`

Die Funktion `verkettenmt` (zwei oder mehrere) Zeichenfolgen zu einer Zeichenfolge. Dabei wird die als erster Parameter übergebene `TRENNZEICHENFOLGE` zwischen alle nicht leeren Elemente der `TEXTLISTE` eingesetzt, d.h. sollte ein Element der `TEXTLISTE` leer sein, findet für dieses keine Verkettung mit der `TRENNZEICHENFOLGE` statt.



Achtung

Ist lediglich eine übergebene Zeichenfolge nicht leer, wird diese ohne Verkettung zurückgeliefert.

Hinweise zur Funktion `transPeri`

Die Funktion transformiert Zeichen (z. B. im Vornamen, Nachnamen, Vorsatzworte) basierend auf einer definierten Zeichenkodierung. Diese ist Kapitel "Transformationsregeln zwischen Datenerhebung und Datenexport" der technischen Dokumentation zur „Verknüpfung der Module Geburtshilfe und Neonatologie des QS-Verfahrens Perinatalmedizin“ zu entnehmen¹⁰¹.

Hinweise zur Funktion `personenkreismapping`

Mithilfe der Funktion `personenkreismapping` werden die Angaben zum besonderen Personenkreis aus zwei verschiedenen technischen Anlagen vor Übertragung in den Dokumentationsbogen in das passende Format übersetzt. Beispielsweise wird der Wert ' 00 ' auf den Wert ' 0 ' im Dokumentationsbogen gemappt.

2.4.4 Syntaxvariablen

Der Eingangsdatensatz bildet die medizinische Routinedokumentation ab, die in jedem Krankenhausinformationssystem (KIS) enthalten ist. Syntaxvariablen in der QS-Dokumentation dienen der technischen Darstellung der automatischen Generierung von Angaben aus dem Eingangsdatensatz. Diese sind in Tabelle `SyntaxVariable` hinterlegt.

Derzeit werden Syntaxvariablen in der QSDOK-Datenbank insbesondere für den Leistungsbereich Dekubitusprophylaxe verwendet, da die Angaben der Bogenfelder im Modul `DEK` aus den Routinedaten im KIS generiert werden sollen. Auch die automatische Befüllung des Datenfeldes `PERSONENKREIS` in FU-Modulen wird mittels Syntaxvariablen gesteuert.

Der Eingangsdatensatz wird in der QSDOK nicht explizit aufgeführt, da er der zugehörigen Risikostatistik bzw. der QS-Filterdatenbank entnommen werden kann.

Jeder der in der Tabelle `SyntaxVariable` definierten Variablen ist über den Wert des Attributes `SyntaxVariable.fkTdsFeld` ein Feld des Risikostatistik-Eingangsdatensatzes zugeordnet.

Jedes dieser Felder besitzt somit einen Basistyp.

Die in den Bedingungen erlaubten Variablen sind in der Tabelle `SyntaxVariable` definiert.

¹⁰¹ <https://iqtig.org/datenerfassung/spezifikationen/qs-basisspezifikation-fuer-leistungserbringer>

Die Variablennamen (Attribut `SyntaxVariable.name`) bestehen aus maximal 32 Zeichen. Sie dürfen nur die Buchstaben A bis Z (Großbuchstaben) und die Ziffern 0 bis 9 enthalten. Ein Feldname muss immer mit einem Buchstaben beginnen. Umlaute und Sonderzeichen sind in Feldnamen nicht erlaubt. Ein Feldname darf auch kein reserviertes Wort sein (z. B. Namen von Operatoren wie `EINSIN`).

2.4.5 Einzelregeln

Sogenannte Einzelregeln können sich als feldbezogene Prüfungen auf ein einziges Datenfeld oder als feldübergreifende Prüfungen auf mehrere Datenfelder beziehen. Einzelregeln sind von den in Abschnitt B 2.4.7 beschriebenen Feldgruppen zu unterscheiden.

Feldbezogene Prüfungen – beispielsweise Wertebereichsüberprüfungen – sind in der formalen Regelsyntax in Tabelle **Regeln** formuliert (Beispiel: `OPDAUER > 600`).

Unter feldbezogenen Prüfungen sind aber auch die in Abschnitt B 2.4.8 beschriebenen Prüfungen des Formates, der Feldlänge, der Wertebereiche, Prüfungen von Schlüsselkodes und von Muss-Feldern zu verstehen. Für diese Prüfungen gibt es keine formale Regelsyntax in Tabelle **Regeln**.

Feldübergreifende Regeln

- haben eine eigene Syntax,
- haben geringe Komplexität,
- haben einfache, dem Anwender verständliche Fehlertexte,
- enthalten alle Teilregeln der Feldgruppen,
- haben gewöhnlich den Bezug zu zwei oder mehreren Feldern,
- können zum Teil direkt nach der Benutzereingabe in ein Feld geprüft werden,
- enthalten Bedingungen für unplausible Angaben¹⁰².

Feldübergreifende Regeln können auch teildatensatzübergreifende Regeln sein, wenn die Datenfelder der Regel aus mehreren Teildatensätzen eines Moduls stammen (Abschnitt B 2.4.6).

2.4.6 Teildatensatzübergreifende Regeln

Eine Regel ist teildatensatzübergreifend, wenn die Datenfelder der Regel aus mehreren Teildatensätzen eines Moduls stammen.

Es gibt zwei Arten von teildatensatzübergreifenden Regeln:

- Die Felder sind in verschiedenen Teildatensätzen eines Moduls definiert.
- Ein Feld der Regel ist in einem wiederholbaren Teildatensatz definiert und die Regel bezieht sich auf alle Werte des Datenfeldes innerhalb eines Datensatzes (= Summe aller Teildatensätze eines Vorgangs).

¹⁰² Eine Plausibilitätsregel müsste eigentlich „Unplausibilitätsregel“ heißen, weil sie unplausible Zustände beschreibt, die zu Fehlermeldungen führen.

Beispiel:

Definition auf verschiedenen Teildatensätzen eines Moduls

Regel Modul HEP : OPDATUM > ENTLDATUM

Das Bogenfeld OPDATUM ist in Teildatensatz HEP : PROZ, das Bogenfeld ENTLDATUM in HEP : B definiert.

Beispiel:

Definition in einem wiederholbaren Teildatensatz

Die Werte des Bogenfeldes LFDNREINGRIFF müssen – bezogen auf alle PCI : PROZ-Teildatensätze einer QS-Dokumentation – eindeutig sein. Diese Regel findet sich nicht in der Tabelle Regeln, sondern ist über das Attribut `fkEindeutigBogenFeld` der Tabelle Bogen definiert (Abschnitt B 2.3.2).

Regeln mit Teildatensatz-Listenfeldern

Zu jedem skalaren Datenfeld eines wiederholbaren Teildatensatzes existiert ein Teildatensatz-Listenfeld (kurz TDS-Listenfeld), das über das @-Zeichen vor dem Feldnamen angesprochen wird. Das TDS-Listenfeld enthält sämtliche Werte des betreffenden Datenfeldes, die innerhalb der QS-Dokumentation eines Falles existieren.

Beispiel 1:

Der Teildatensatz 18 / 1 : BRUST enthält das Datenfeld 48 OPTHERAPIEENDE (primär-operative Therapie abgeschlossen).

Wurden während eines Aufenthalts zwei Operationen (Prozedurnummer 1 und 2) durchgeführt, so werden zwei Teildatensätze 18 / : BRUST angelegt, und das Datenfeld OPTHERAPIEENDE muss zweimal dokumentiert werden:

18 / 1 : PROZ [ZUOPSEITE=1] : OPTHERAPIEENDE = 0 (nein)

18 / 1 : PROZ [ZUOPSEITE=2] : OPTHERAPIEENDE = 1 (ja)

Das entsprechende TDS-Listenfeld lautet:

@ OPTHERAPIEENDE = (0 ; 1)

TDS-Listenfelder können in Feldgruppen und Plausibilitätsregeln verwendet werden. Bei Verwendung in Feldgruppen hat das Attribut `tdsListe` in der Tabelle `FeldGruppeFelder` den Wert WAHR.

Beispiel 2:

Die Regel 10604

@OPTHERAPIEENDE KEINSIN (1) UND ADJUTHERAPIEPLANUNG <> LEER

erzwingt die korrekte Dokumentation des Datenfeldes ADJUTHERAPIEPLANUNG (postoperative Therapieplanung in interdisziplinärer Tumorkonferenz) auf dem Basisteildatensatz 18/1 : B, wenn der Patient eine primär-operative Therapie abgeschlossen hat.

2.4.7 Feldgruppenregeln

Logische Abhängigkeiten von Bogenfeldern werden über Feldgruppen dargestellt. Die Plausibilitätsregeln, die einen Bezug zu einer Feldgruppe aufweisen (Tabelle `Regeln`), werden anhand der Feldgruppendefinition (Tabelle `FeldgruppeFelder`) automatisch generiert. Die Menge der abgeleiteten Einzelregeln wird in diesem Abschnitt erläutert.

Die möglichen Antworten¹⁰³ eines jeden Datenfeldes werden in zwei Gruppen aufgeteilt. Die erste Gruppe ist die Menge der positiven, die zweite Gruppe die Menge der negativen Antworten.¹⁰⁴

Typische positive Antworten sind beispielsweise:

`Feld <> LEER` oder `Feld IN (2;3)`

Die komplementären negativen Antworten würden entsprechend wie folgt lauten:

`Feld = LEER` oder `Feld NICHTIN (2;3)`

Eine Feldgruppe kann ein Filterfeld haben. Wenn die Antwort dieses Filterfeldes negativ ausfällt (Bspw. Bedingung: `Feld = 3`; Antwort: `Feld <> 3`), so darf keines der abhängigen Felder positiv beantwortet werden.

Tabelle 57: Typen von Feldgruppen

Die folgende Tabelle gibt einen Überblick über die Typen von Feldgruppen. Der aktuelle Stand findet sich in der Tabelle `FeldGruppenTyp` der Spezifikation.

Tabelle 57: Typen von Feldgruppen

| Name | Bemerkung |
|--------------------|---|
| mit Filterfeld | |
| EF_FILTER | Einfachauswahl, genau ein abhängiges Feld muss positiv beantwortet sein |
| EF_OPTIONAL_FILTER | Einfachauswahl, genau ein abhängiges Feld kann positiv beantwortet sein |
| MF_OPTIONAL_FILTER | Mehrfachauswahl, alle abhängigen Felder können positiv beantwortet sein |

¹⁰³ Die Antworten eines Datenfeldes umfassen hier neben möglichen Werten (z. B. Schlüsselwerten) oder Wertemengen auch die Kategorie „nicht ausgefüllt“ (LEER).

¹⁰⁴ Die negativen Antworten sind abhängig von der definierten Bedingung eines Feldes in der entsprechenden Feldgruppe.

| Name | Bemerkung |
|-----------------------|---|
| MF_MINDESTENS1_FILTER | Mehrfachauswahl, mindestens ein abhängiges Feld muss positiv beantwortet sein |
| MF_ALLES_FILTER | Mehrfachauswahl, alle abhängigen Felder müssen positiv beantwortet sein |
| ohne Filterfeld | |
| EF | Einfachauswahl, genau ein Feld muss positiv beantwortet sein |
| MF_OPTIONAL | Mehrfachauswahl, alle Felder können positiv beantwortet sein |
| MF_MINDESTENS1 | Mehrfachauswahl, mindestens ein Feld muss positiv beantwortet sein |
| UND | Einfache Regel mit Und-Verknüpfungen |

In der Tabelle `BogenFeld` sind abhängige Datenfelder einer Feldgruppe immer als Kann-Felder definiert. Nach Abhängigkeit der Feldgruppenlogik können/müssen diese Felder leer bleiben oder zwingend ausgefüllt werden. Im letztgenannten Fall können die Datenfelder auch als bedingte Muss-Felder bezeichnet werden.

Die Muss- oder Kann-Definition der Datenfelder (Bogen- und Ersatzfelder) im Exportformat unterliegt ebenfalls der Feldgruppenlogik. Ist die Berechnung eines Ersatzfeldes von bedingten Datenfeldern abhängig, so gilt die Feldgruppenlogik auch für diese Ersatzfelder. Wenn die bedingten Datenfelder zwingend ausgefüllt werden müssen, so muss auch das Ersatzfeld zwingend berechnet bzw. exportiert werden.¹⁰⁵

Struktur der Tabellen `FeldGruppe` und `FeldgruppeFelder`

Die Feldgruppen sind in den Tabellen `FeldGruppe` und `FeldgruppeFelder` definiert. In der Tabelle `FeldGruppe` (Tabelle 58) sind Name, Typ und die Zuordnung zu einem Modul definiert. Die Verknüpfungstabelle `FeldgruppeFelder` (Tabelle 59) definiert die abhängigen Bogenfelder. Zusätzlich wird hier festgelegt, welche Bogenfelder der Feldgruppe als Filterfeld dienen.

Tabelle 58: Struktur der Tabelle `FeldGruppe`

| Feldname | Datentyp | Bemerkung |
|---------------------------|-------------|---------------------------------|
| <code>idFeldGruppe</code> | INTEGER | Primärschlüssel |
| <code>name</code> | TEXT (64) | Technischer Name der Feldgruppe |

¹⁰⁵ Die Funktion `verkettenmt` verkettet (zwei oder mehrere) Zeichenfolgen zu einer Zeichenfolge. Hierbei müssen nicht alle im Attribut `Ersatzfeld.formel` aufgeführten Datenfelder ausgefüllt sein. Ist lediglich eine übergebene Zeichenfolge nicht leer, wird diese ohne Verkettung zurückgeliefert.

| Feldname | Datentyp | Bemerkung |
|------------------|----------|---|
| fkModul | INTEGER | Obligatorischer Fremdschlüssel zu einem Modul |
| fkFeldgruppenTyp | INTEGER | Obligatorischer Fremdschlüssel zu einem Feldgruppentyp |
| hinweis | TEXT | Bei Filter-Feldgruppen relevant für die Gestaltung der Eingabemaske. Der Hinweistext informiert den Anwender über die Bedingungen, welche das Ausfüllen von ein oder mehreren abhängigen Feldern erforderlich machen. Der Hinweistext kann bei der Erstellung der Eingabemasken verwendet werden. Beispiel: Der Hinweistext „Bei postoperativen Komplikationen“ wird oberhalb eines Blocks von zusammengehörigen Feldgruppenfeldern angezeigt. |
| fkFilterFeldTyp | CHAR(1) | Attribut wird bei Feldgruppen mit mehreren Filterfeldern gesetzt: O = Oder-Verknüpfung der positiven Filterbedingungen U = Und-Verknüpfung der positiven Filterbedingungen |
| fkRegelTyp | CHAR(1) | Fremdschlüssel zur Tabelle RegelTyp: Die Regeltypen sind die in Abschnitt A 2.4 beschriebenen Arten der Plausibilitätsprüfungen: H, W oder D Die generierten Einzelregeln der Feldgruppe haben den gleichen Regeltyp. |
| nurPositiv | BOOLEAN | Wenn WAHR, dann umfasst die Feldgruppe nur diejenigen Regeln, welche sich auf die positive (Filter)bedingung beziehen. |
| grauWennNegativ | BOOLEAN | Definiert eine Layout-Feldgruppe, wenn WAHR (siehe unten, Abschnitt Layout-Feldgruppen) |

Tabelle 59: Struktur der Tabelle *FeldgruppeFelder*

| Feldname | Datentyp | Bemerkung |
|----------------------------|----------|--|
| idFeldgruppeFelder | INTEGER | Primärschlüssel |
| fkFeldGruppe | INTEGER | Obligatorischer Fremdschlüssel zur Feldgruppe |
| fkBogenFeld | INTEGER | Obligatorischer Fremdschlüssel zum Bogenfeld |
| bedingung | TEXT | Positive Bedingung für das jeweilige Bogenfeld |
| istFilter | BOOL | Legt fest, ob das jeweilige Bogenfeld ein Filterfeld ist |
| bezeichnungSchluesselListe | TEXT | Abkürzende Bezeichnung für eine Schlüsseliste in der Bedingung, wird beim Generieren von Fehlermeldungen verwendet. |
| tdsFilter | BOOLEAN | Das Bogenfeld wird in Regeln als TDS-Listenfeld (Abschnitt B 2.4.6) verwendet (Vorstellen des @-Zeichens vor Feldnamen). |

Syntax der Feldgruppenregeln

In den Tabellen *FeldGruppe* bzw. *FeldgruppeFelder* sind die positiven Bedingungen für das Filterbogenfeld bzw. die abhängigen Bogenfelder einer Feldgruppe definiert. Jede Bedingung hat folgenden Aufbau:

```
<Operator> <Operand>
```

Der linke Operand wird hier weggelassen, weil er immer der Name des jeweiligen Bogenfeldes ist. Die komplette Bedingung für das Bogenfeld einer Feldgruppe lautet also:

```
<Bogenfeld> <Operator> <Operand>
```

Als Operator kann jeder dyadische Operator der Tabelle 56 verwendet werden. Die auf der rechten Seite erlaubten Operanden sind nachfolgend aufgelistet:

- Literale (Tabelle 55)
- LEER
- Kodelisten, in denen auch die Codes eines Schlüssels referenziert werden können; Beispiel: (1 ; 2 ; 3) oder (MaDCIS)
- ICD-Listen oder OPS-Listen (z. B. OPS_AxLKEntfOhneMark)



Hinweis

Der rechte Operand darf kein Bogenfeld sein, da sich eine Feldbedingung immer genau auf ein Bogenfeld bezieht.

Im Folgenden sind einige mögliche Bedingungen von Feldgruppen dargestellt:

Beispiele:

ERSCHWNEBDG = 1

HERZFEHLER <> LEER

OPSCHLUESSEL EINSIN OPS_ AxLKEntfOhneMark

POSTICDO3 IN (MaDCIS) ist gleichbedeutend mit:

POSTICDO3 IN ('8500/2'; '8503/2'; '8504/2'; '8507/2'; '8543/3'),
da der Schlüssel MaDCIS die aufgelisteten Codes umfasst.

Formale Definition von Feldgruppen

A sei ein Bogenfeld in einer Feldgruppe. Dann seien $p(A)$ die positiven und $n(A)$ die negativen Bedingungen, welche jeweils das Ergebnis wahr oder falsch haben können.

Eine Feldgruppe kann ggf. ein Filterfeld haben, das mit F bezeichnet wird. Eine Feldgruppe lässt sich dann in folgender Tabelle darstellen:

Tabelle 60: Formale Definition einer Feldgruppe

| Feld | Positive Bedingung | Negative Bedingung | Bemerkung |
|------|--------------------|--------------------|---------------------------------|
| F | $p(F)$ | $n(F)$ | falls Feldgruppentyp mit Filter |
| A1 | $p(A1)$ | $n(A1)$ | |
| A2 | $p(A2)$ | $n(A2)$ | |
| A3 | $p(A3)$ | $n(A3)$ | |
| ... | | | |
| An | $p(An)$ | $n(An)$ | |

Eine Feldgruppe besteht insgesamt aus n abhängigen Bogenfeldern:

A1, A2, ... An

In Abhängigkeit von den Feldgruppentypen werden unterschiedliche Einzelregeln generiert.

Feldgruppen mit Filter

- Regeln der Feldgruppe „Optionale Mehrfachauswahl mit Filterfeld“ (MF_OPTIONAL_FILTER)
- $n(F) \text{ UND } p(A_i) \quad i = 1, \dots, n$

Insgesamt sind n Einzelregeln mit der Feldgruppe verknüpft.

- Regeln der Feldgruppe „Obligatorische Mehrfachauswahl mit Filterfeld“ (MF_MINDESTENS1_FILTER)
- $n(F) \text{ UND } p(A_i) \quad i = 1, \dots, n$

- $p(F) \text{ UND } n(A1) \text{ UND } n(A2) \text{ UND } \dots \text{ UND } n(An)$

Insgesamt sind $n+1$ Einzelregeln mit der Feldgruppe verknüpft.

- Regeln der Feldgruppe „Mehrfachauswahl mit Filterfeld, alle abhängigen Felder müssen positiv beantwortet sein“ (MF_ALLES_FILTER)
- $n(F) \text{ UND } p(Ai) \quad i = 1, \dots, n$
- $p(F) \text{ UND } n(Ai) \quad i = 1, \dots, n$

Insgesamt sind $2n$ Einzelregeln mit der Feldgruppe verknüpft.

- Regeln der Feldgruppe „Einfachauswahl mit Filter“ (EF_FILTER)
- $n(F) \text{ UND } p(Ai) \quad i = 1, \dots, n$
- $p(F) \text{ UND } n(A1) \text{ UND } n(A2) \text{ UND } \dots \text{ UND } n(An)$
- $p(F) \text{ UND } p(Aj) \text{ UND } p(Ai) \quad \text{für alle unterschiedlichen } i, j = 1, \dots, n$

Insgesamt sind $n(n+1)/2+1$ Einzelregeln mit der Feldgruppe verknüpft.

- Regeln der Feldgruppe „Optionale Einfachauswahl mit Filter“ (EF_OPTIONAL_FILTER)
- $n(F) \text{ UND } p(Ai) \quad i = 1, \dots, n$
- $p(F) \text{ UND } p(Aj) \text{ UND } p(Ai) \quad \text{für alle unterschiedlichen } i, j = 1, \dots, n$

Insgesamt sind $n(n+1)/2$ Einzelregeln mit der Feldgruppe verknüpft.

Feldgruppen mit Filter: Attribut nurPositiv

Wenn in einer Feldgruppe mit Filter das Attribut `nurPositiv` gesetzt ist, so sind nur die Einzelregeln mit positiver Filterbedingung Bestandteil der Feldgruppe.

Beispiel:

Die Feldgruppe `EF_FILTER` mit `nurPositiv=ja` hat folgende Einzelregeln:

$p(F) \text{ UND } n(A1) \text{ UND } n(A2) \text{ UND } \dots \text{ UND } n(An)$

$p(F) \text{ UND } p(Aj) \text{ UND } p(Ai) \quad \text{für alle unterschiedlichen } i, j = 1, \dots, n$

Feldgruppen ohne Filter

- Regeln der Feldgruppe „Einfachauswahl“ (EF)
- $n(A1) \text{ UND } n(A2) \text{ UND } \dots \text{ UND } n(An)$
- $p(Aj) \text{ UND } p(Ai) \quad \text{für alle unterschiedlichen } i, j = 1, \dots, n$

Insgesamt sind $n(n-1)/2+1$ Einzelregeln mit der Feldgruppe verknüpft.

- Regeln der Feldgruppe „Obligatorische Mehrfachauswahl“ (MF_MINDESTENS1)
- $n(A1) \text{ UND } n(A2) \text{ UND } \dots \text{ UND } n(An)$

Insgesamt ist eine Einzelregel mit der Feldgruppe verknüpft.

- Regeln der Feldgruppe „Und-Regel“ (UND)
- $p(A1) \text{ UND } p(A2) \text{ UND } \dots \text{ UND } p(An)$

Insgesamt ist eine Einzelregel mit der Feldgruppe verknüpft.

Feldgruppen mit mehreren Filterfeldern

Es besteht die Möglichkeit, Feldgruppen mit mehr als einem Filterfeld zu definieren:

Formal gibt es dann die Filterfelder F_1, F_2, \dots, F_n mit den positiven bzw. negativen Bedingungen $p(F_j)$ bzw. $n(F_j)$. Für alle Filterfelder wird eine positive Bedingung $p(F_1, \dots, F_n)$ und eine negative Bedingung $n(F_1, \dots, F_n)$ gebildet. Diese modifizierten Filterbedingungen ersetzen die im Abschnitt B 2.4.7 definierten Filterbedingungen $p(F)$ und $n(F)$ bei den Einzelregeln.

Die Filterfelder können entweder über eine ODER-Verknüpfung oder eine UND-Verknüpfung miteinander verbunden sein:

$$p(F_1, \dots, F_n) = p(F_1) \text{ ODER } p(F_2) \text{ ODER } \dots \text{ ODER } p(F_n)$$

(ODER-Verknüpfung)

$$p(F_1, \dots, F_n) = p(F_1) \text{ UND } p(F_2) \text{ UND } \dots \text{ UND } p(F_n)$$

(UND-Verknüpfung)

Der Verknüpfungstyp ist im Attribut `fkFilterFeldTyp` der Tabelle `FeldGruppe` hinterlegt.

Layout-Feldgruppen

Feldgruppen, bei denen das Attribut `grauWennNegativ` in der Datenbanktabelle `FeldGruppe` WAHR ist, werden nachfolgend als Layout-Feldgruppen bezeichnet. Der Attributname `grauWennNegativ` wurde gewählt, weil die abhängigen Felder der Layout-Feldgruppen auf den generierten Dokumentationsbögen eingegraut sind. Abbildung 18 zeigt die Layout-Feldgruppe `NEO:OPArt` mit einem Filterfeld und drei abhängigen Feldern am Beispiel der Spezifikation 2017.

| 73-76 Operation(en) und Prozeduren | |
|---|--|
| 73 | Operation(en) und Prozeduren während des aktuellen stationären Aufenthaltes <input type="checkbox"/> 0 = nein 1 = ja |
| wenn Operation(en) und Prozeduren während des aktuellen stationären Aufenthaltes = ja | |
| 74> | OP oder Therapie einer ROP (Frühgeborenenretinopathie) <input type="checkbox"/> 0 = nein 1 = ja, Lasertherapie oder Kryotherapie 2 = ja, intravitreale Anti-VEGF-Therapie 3 = ja, sonstige |
| 75> | OP einer NEK (nekrotisierende Enterokolitis) <input type="checkbox"/> 0 = nein 1 = ja |
| 76> | Indikation(en) zu(r) weiteren Operation(en) <small>http://www.dtmtd.de</small> 1. <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> . <input type="checkbox"/> <input type="checkbox"/> 2. <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> . <input type="checkbox"/> <input type="checkbox"/> 3. <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> . <input type="checkbox"/> <input type="checkbox"/> 4. <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> . <input type="checkbox"/> <input type="checkbox"/> 5. <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> . <input type="checkbox"/> <input type="checkbox"/> |

Abbildung 18: Feldgruppe NEO:OPart auf dem Dokumentationsbogen (Spezifikation 2017)

Layout-Feldgruppen haben folgende Eigenschaften:

- Sie haben mindestens ein Filterfeld.
- Jedes abhängige Feld hat die Bedingung <> LEER oder EINSNICHTIN (LEER) (Attribut bedingung in Tabelle FeldGruppeFelder).
- Das Attribut `nurPositiv` hat den Wert `FALSCH`.

Bei Vorliegen dieser drei Eigenschaften müssen die abhängigen Felder leer bleiben, wenn die negative Filterbedingung bei der Dokumentation eines Falles erfüllt ist.

Beispiel:

Wenn in Datenfeld 73 = 0 (nein) angegeben ist, so müssen die Datenfelder 74, 75 und 76 leer bleiben. Die folgenden Plausibilitätsprüfungen stellen dies sicher.

Tabelle 61: Plausibilitätsregeln der Feldgruppe NEO:OPart in Spezifikation 2017

| Bedingung | Fehlermeldung | fkFeld-Gruppe |
|--|---|---------------|
| OP <> 1 UND OPROP <> LEER | "Operation(en) und Prozeduren während des aktuellen stationären Aufenthaltes" ist nicht "ja" [1], obwohl "OP oder Therapie einer ROP (Frühgeborenenretinopathie)" ausgefüllt ist | NEO:O-Part |
| OP <> 1 UND OPNEC <> LEER | "Operation(en) und Prozeduren während des aktuellen stationären Aufenthaltes" ist nicht "ja" [1], obwohl "OP einer NEK (nekrotisierende Enterokolitis)" ausgefüllt ist | NEO:O-Part |
| OP <> 1 UND OPDIAG EINSNICHTIN (LEER) | "Operation(en) und Prozeduren während des aktuellen stationären Aufenthaltes" ist nicht "ja" [1], obwohl "Indikation(en) zu(r) weiteren Operation(en)" angegeben ist | NEO:O-Part |
| OP = 1 UND OPROP = LEER UND OPNEC = LEER UND OPDIAG JEDESIN (LEER) | "Operation(en) und Prozeduren während des aktuellen stationären Aufenthaltes" ist "ja" [1] und keines der Felder "OP oder Therapie einer ROP (Frühgeborenenretinopathie)" und "OP einer NEK (nekrotisierende Enterokolitis)" ist ausgefüllt, obwohl "Indikation(en) zu(r) weiteren Operation(en)" nicht angegeben ist | NEO:O-Part |

Die zugehörige Feldgruppe (Abfrage `FeldgruppeFürEinModul` = Zusammenschau der Tabellen `Feldgruppe` und `FeldgruppeFelder`) ist wie folgt definiert:

Tabelle 62: Definition der Feldgruppe NEO:OPart in Spezifikation 2017

| Bogenfeld | Bedingung | istFilter | Feldgruppentyp | grauWenn-Negativ |
|-----------------|--------------------|-----------|-----------------------|------------------|
| NEO.B.74.OPROP | <> LEER | FALSE | MF_MINDESTENS1_FILTER | TRUE |
| NEO.B.75.OPNEC | <> LEER | FALSE | MF_MINDESTENS1_FILTER | TRUE |
| NEO.B.76.OPDIAG | EINSNICHTIN (LEER) | FALSE | MF_MINDESTENS1_FILTER | TRUE |

| Bogenfeld | Bedingung | istFilter | Feldgruppentyp | grauWenn-Negativ |
|-------------|-----------|-----------|-----------------------|------------------|
| NEO.B.73.OP | = 1 | TRUE | MF_MINDESTENS1_FILTER | TRUE |

2.4.8 Prüfung von Feldeigenschaften

Die in diesem Abschnitt behandelten feldbezogenen Prüfungen ergeben sich direkt aus den Tabellen `Feld` (bzw. `ErsatzFeld` oder `ZusatzFeld`) und `BogenFeld` (bzw. `ExportFormat`) und werden vor Evaluation der in Abschnitt B 2.4.4 beschriebenen feldübergreifenden Regeln durchgeführt.

Die hier beschriebenen Prüfungen sind nur in Form von Feldeigenschaften – nicht aber in Regelsyntax – in der Datenbank für QS-Dokumentation hinterlegt.

Überprüfung des Formats

Die Formatprüfung bezieht sich auf den Exportdatensatz (Abschnitt B 2.6.2): Die QS-Dokumentations- bzw. Exportsoftware muss Daten im korrekten Format generieren, was durch die datenentgegennehmenden Stellen geprüft wird. Für Exportfelder (Tabelle `ExportFormat`), die einen Bezug zu einem Ersatzfeld (Tabelle `ErsatzFeld`) bzw. zu einem Zusatzfeld (Tabelle `ZusatzFeld`) haben, sind die standardisierten Fehlertexte anzupassen.

Die Prüfung bezieht sich insbesondere darauf, ob der Feldinhalt dem in der Spezifikation definierten Basistyp entspricht. Beispielsweise sind Buchstaben beim Basistyp `GANZEZAHL` nicht erlaubt.

Standardisierter Fehlertext für Formatfehler eines Ersatzfeldes

Der Wert '`<WERT>`' des Datenfeldes `<Modul.name>:<Bogen.name>:<Ersatzfeld.name>` '`<ErsatzFeld.bezeichnung>`' ist kein gültiger `<Basistyp.name>` Wert (`<Basistyp.bezeichnung>` `<Basistyp.format>`).

Standardisierter Fehlertext für Formatfehler eines Bogenfeldes

Der Wert '`<WERT>`' des Datenfeldes `<Modul.name>:<Bogen.name>:<Feld.name>` '`<BogenFeld.bezeichnung>`' (Zeile `<BogenFeld.gliederungAufBogen>`) ist kein gültiger `<Basistyp.name>` Wert (`<Basistyp.bezeichnung>` `<Basistyp.format>`).

Standardisierter Fehlertext für Muss-Fehler eines Zusatzfeldes

Das Zusatzfeld `<Modul.name>:<Bogen.name>:<ZusatzFeld.name>` '`<ZusatzFeld.bezeichnung>`' ist kein gültiger `<Basistyp.name>` Wert (`<Basistyp.bezeichnung>` `<Basistyp.format>`).

Beispiel (Bogenfeld):

Der Wert '3A.06.2012' des Datenfeldes 15/1:B:AUFNDATUM 'Aufnahmedatum Krankenhaus' (Zeile 8) ist kein gültiger DATUM-Wert (Zehnstelliges Datum TT.MM.JJJJ).

Überprüfung der Feldlänge

Die Feldlängenprüfung bezieht sich darauf, ob die Anzahl der Zeichen eines Wertes die spezifizierte Länge¹⁰⁶ (Attribut laenge in Tabelle Feld) des Feldes überschreitet.

Standardisierter Fehlertext für Längenfehler eines Ersatzfeldes

Der Wert '<WERT>' des Datenfeldes <Modul.name>:<Bogen.name>:<ErsatzFeld.name> '<ErsatzFeld.bezeichnung>' überschreitet die zulässige Feldlänge <Feld.laenge>.

Standardisierter Fehlertext für Längenfehler eines Bogenfeldes

Der Wert '<WERT>' des Datenfeldes <Modul.name>:<Bogen.name>:<Feld.name> '<BogenFeld.bezeichnung>' (Zeile <BogenFeld.gliederungAufBogen>) überschreitet die zulässige Feldlänge <Feld.laenge>.

Standardisierter Fehlertext für Muss-Fehler eines Zusatzfeldes

Das Zusatzfeld <Modul.name>:<Bogen.name>:<ZusatzFeld.name> '<ZusatzFeld.bezeichnung>' überschreitet die zulässige Feldlänge <ZusatzFeld.laenge>.

Beispiel (Bogenfeld):

Der Wert '31.06.20040' des Datenfeldes 15/1:B:AUFNDATUM 'Aufnahmedatum Krankenhaus' (Zeile 8) überschreitet die zulässige Feldlänge 10.

Überprüfung der Schlüsselcodes

Die Überprüfung von Schlüsselcodes bezieht sich darauf, ob bei Schlüsselfeldern nur zulässige Schlüsselcodes verwendet werden.

Standardisierter Fehlertext bei unzulässigen Schlüsselcodes eines Ersatzfeldes

Der Wert '<WERT>' des Datenfeldes <Modul.name>:<Bogen.name>:<ErsatzFeld.name> '<ErsatzFeld.bezeichnung>' ist kein zulässiger Code des Schlüssels <Schlüssel.name> (<Schlüssel.bezeichnung>).

Standardisierter Fehlertext bei unzulässigen Schlüsselcodes eines Bogenfeldes

Der Wert '<WERT>' des Datenfeldes <Modul.name>:<Bogen.name>:<Feld.name> '<BogenFeld.bezeichnung>' (Zeile <BogenFeld.gliederungAufBogen>) ist

¹⁰⁶ Wenn bei einem Ersatz die Länge nicht spezifiziert ist, so entfällt die Prüfung.

kein zulässiger Kode des Schlüssels <Schluessel.name> (<Schluessel.bezeichnung>).

Standardisierter Fehlertext für Muss-Fehler eines Zusatzfeldes

Das Zusatzfeld <Modul.name>:<Bogen.name>:<ZusatzFeld.name> '<ZusatzFeld.bezeichnung>' ist kein zulässiger Kode des Schlüssels <Schluessel.name> (<Schluessel.bezeichnung>).

Beispiel (Bogenfeld):

Der Wert 'xyz' des Datenfeldes HEP:PROZ:OPSCHUESSEL 'Prozedur(en)' (Zeile 29) ist kein zulässiger Kode des Schlüssels OPS301.

Besonderheiten bei externen Schlüsseln

- Bei externen DIMDI-Schlüsseln (ICD-10-GM oder OPS) sind die jeweils gültigen amtlichen Kataloge zu verwenden. Alle Codes eines Behandlungsfalles müssen in derjenigen Katalogversion dokumentiert sein, die am Aufnahmetag des Patienten in das Krankenhaus bzw. am Follow-up-Erhebungsdatum bei den Datensätzen HTXFU, NTFUFU, LTFUFU, NLSFU und LLSFU gültig ist.¹⁰⁷
- Nicht-terminale ICD- oder OPS-Kodes sind unzulässig!
- Fehlen bei OPS-Kodes Seitenlokalisationen, obwohl diese erforderlich sind, so ist der OPS-Kode fehlerhaft (siehe auch Abschnitt B 2.5.1).

Überprüfung numerischer Wertebereiche

Bei numerischen Datenfeldern (Typ ZAHL oder GANZEAHL) ist hart zu überprüfen, ob der Zahlenwert im durch die Attribute min und max (Tabelle FELD) definierten Wertebereich liegt:

- wert < min (nur prüfen, wenn min <> LEER)
- wert > max (nur prüfen, wenn max <> LEER)

In der Tabelle Feld sind weitere Unter- und Obergrenzen (Attribute min/max und minWeich/maxWeich) für Prüfungen definiert. In wenigen Fällen sind auch in der Tabelle BogenFeld Grenzen definiert. Falls vorhanden (= not null), werden die in der Tabelle BogenFeld gesetzten speziellen Wertgrenzen statt der Grenzen in der Tabelle Feld angewandt.

Beispiel:

Tabelle 63 zeigt für das Feld OPDAUER die Wertebereichsdefinitionen auf Feld- und Bogenfeld-ebene für die Module 09/1, 17/1 und HCH. Tabelle 64 stellt die in den Modulen anzuwendenden Prüfungen dar.

¹⁰⁷ Analog zur Regelung zur Abgrenzung von Erfassungsjahren und Datensatzformaten in Abschnitt B 1.7.3.

Tabelle 63: Beispiel für Wertebereichsgrenzen (Datenfeld OPDAUER)¹⁰⁸

| | min | | max | | minWeich | | maxWeich | |
|------|------|-----------|------|-----------|----------|-----------|----------|-----------|
| | Feld | Bogenfeld | Feld | Bogenfeld | Feld | Bogenfeld | Feld | Bogenfeld |
| 09/1 | 1 | NULL | NULL | NULL | NULL | 5 | 240 | NULL |
| 17/1 | 1 | 10 | NULL | NULL | NULL | 15 | 240 | NULL |
| HCH | 1 | 0 | NULL | NULL | NULL | 20 | 240 | 480 |

Tabelle 64: Beispiel für Prüfungen von Wertebereichsgrenzen

| 09/1 | 17/1 | HCH |
|-----------------------|-----------------------|-----------------------|
| OPDAUER < 1 (hart) | OPDAUER < 10 (hart) | OPDAUER = 0 (hart) |
| OPDAUER < 5 (weich) | OPDAUER < 15 (weich) | OPDAUER < 20 (weich) |
| OPDAUER > 240 (weich) | OPDAUER > 240 (weich) | OPDAUER > 480 (weich) |

Eine Übersicht über die in numerischen Datenfeldern definierten harten und weichen Wertebereiche bietet die Abfrage WertebereicheNumerischerFelder.

Außerdem sind die Wertebereiche in den Ausfüllhinweisen zu den jeweiligen Datenfeldern angegeben.

Beispiel:

Wertebereiche des Datenfeldes OPDAUER in 09/1 in den Ausfüllhinweisen:

- Gültige Angabe: ≥ 1 Minuten
- Angabe ohne Warnung: 5–240 Minuten

Standardisierter Fehlertext bei Unterschreitung einer Wertebereichsgrenze eines Ersatzfeldes

Der Wert '<WERT>' des Datenfeldes <Modul.name>:<Bogen.name>:<Ersatzfeld.name> '<Ersatzfeld.bezeichnung>' ist kleiner als '<Feld.min>'.

Standardisierter Fehlertext bei Unterschreitung einer Wertebereichsgrenze eines Bogenfeldes

Der Wert '<WERT>' des Datenfeldes <Modul.name>:<Bogen.name>:<Feld.name> '<Bogenfeld.bezeichnung>' (Zeile <Bogenfeld.gliederungAufBogen>) ist kleiner als '<Feld.min>'.

¹⁰⁸ Zeile 1 = Attribute der Tabelle Feld/Zeile 2 = Tabelle

Beispiel (Bogenfeld):

Der Wert '-90' des Datenfeldes HEP:B:POSTOPEXFLEX1 'Extension/Flexion 1 bei Entlassung' (Zeile 77) ist kleiner als '0'.

Standardisierter Fehlertext bei Überschreitung einer Wertebereichsgrenze eines Ersatzfeldes

Der Wert '<WERT>' des Datenfeldes <Modul.name>:<Bogen.name>:<Ersatzfeld.name> '<Ersatzfeld.bezeichnung>' ist größer als '<Feld.max>'.

Standardisierter Fehlertext bei Überschreitung einer Wertebereichsgrenze eines Bogenfeldes

Der Wert '<WERT>' des Datenfeldes <Modul.name>:<Bogen.name>:<Feld.name> '<Bogenfeld.bezeichnung>' (Zeile <Bogenfeld.gliederungAufBogen>) ist größer als '<Feld.max>'.

Beispiel (Bogenfeld):

Der Wert '370' des Datenfeldes HEP:B:POSTOPEXFLEX1 'Extension/Flexion 1 bei Entlassung' (Zeile 77) ist größer als '10'.

Bei weichen Plausibilitätsverletzungen ist dem Fehlertext das Wort „Hinweis“ voranzustellen.

Überprüfung der Muss-Felder

Ein nicht ausgefülltes Muss-Feld (Abschnitt B 2.3.3) führt zu einer Regelverletzung.

Standardisierter Fehlertext für Muss-Fehler eines Ersatzfeldes

Das Datenfeld '<Modul.name>:<Bogen.name>:<Ersatzfeld.name>' '<Ersatzfeld.bezeichnung>' muss einen gültigen Wert enthalten.

Standardisierter Fehlertext für Muss-Fehler eines Bogenfeldes

Das Datenfeld '<Modul.name>:<Bogen.name>:<Feld.name>' '<Bogenfeld.bezeichnung>' (Zeile <Bogenfeld.gliederungAufBogen>) muss einen gültigen Wert enthalten.

Standardisierter Fehlertext für Muss-Fehler eines Zusatzfeldes

Das Zusatzfeld <Modul.name>:<Bogen.name>:<Zusatzfeld.name> '<Zusatzfeld.bezeichnung>' muss einen gültigen Wert enthalten.

Beispiel (Bogenfeld):

Das Datenfeld '15/1:B:AUFNDATUM' 'Aufnahmedatum Krankenhaus' (Zeile 8) muss einen gültigen Wert enthalten.

2.4.9 Verfahren für die Evaluation von Regeln

Grundsätzlich muss jede gem. Abschnitt B 2.4.2 formulierte Regel evaluiert werden, wenn keine der folgenden drei Bedingungen zutrifft:

1. Für mindestens ein referenziertes Bogenfeld¹⁰⁹ schlägt eine harte Feldprüfung (Abschnitt B 2.4) fehl.¹¹⁰
2. Ein Feld der Regel ist nicht ausgefüllt (LEER) und **keine** der folgenden Teilbedingung trifft in Bezug auf das leere Feld zu:
 - Es ist in einer Liste enthalten, die mit einem Listenoperator (EINSIN, KEINSIN, JEDESIN, EINSNICHTIN) geprüft wird bzw. wird direkt gegen eine Liste geprüft (IN, NICHTIN).
 - Es wird in der Regel explizit mit <> oder = auf LEER geprüft.
 - Das Feld der Regel befindet sich auf einem vorhandenen Teildatensatz.¹¹¹
 - Jeder Operation auf einen Wert <> LEER ist eine ODER-Verknüpfte Prüfung auf LEER direkt vorgeschaltet (Feld = LEER ODER Feld Operator Operand).
3. Eine Funktion der Regel hat das Ergebnis LEER und wird in der Regel nicht explizit mit <> oder = auf LEER geprüft.

Der Algorithmus zur Evaluation einer Plausibilitätsregel ist in Abbildung 19 beschrieben.

Umgang mit Laufzeitfehlern

Bei der Evaluation von Regeln können, z. B. bei einem Vergleich von LEER < WERT, Laufzeitfehler entstehen. Solche Laufzeitfehler sind bei der Evaluation zu verhindern.

Laufzeitfehler bei der Evaluation von Regeln nach den Vorgaben sind immer ein Hinweis auf Syntaxfehler in der Regel. Das Ausbleiben von Laufzeitfehlern ist noch kein Beweis für die syntaktische Fehlerfreiheit einer Regel, da es auch fehlertolerante Parser geben könnte, die beim Verlassen des definierten Wertebereichs der Regel ein FALSCH zurückgeben können. Tatsächlich liegt aber hier ein undefinierter Zustand vor, der weder FALSCH noch WAHR ist.

Ein undefinierter Zustand an einem beliebigen Punkt im Evaluationsprozess muss für alle dort noch möglichen Werte durch die Struktur der Regel immer vermieden werden.

¹⁰⁹ Die Verbindung zwischen Regeln und Bogenfeldern geschieht über die Tabelle RegelFelder, siehe Regelfelder (Bogenfelder einer Regel), Abschnitt B 1.4.1.

¹¹⁰ Erst bei Fehlerfreiheit der feldbezogenen Prüfungen werden die feldübergreifenden Prüfungen durchgeführt.

¹¹¹ Eine Regel wird nicht geprüft, wenn der entsprechende Teildatensatz eines leeren Feldes optional und im konkreten Fall nicht vorhanden ist.

Algorithmus zur Evaluation einer Plausibilitätsregel

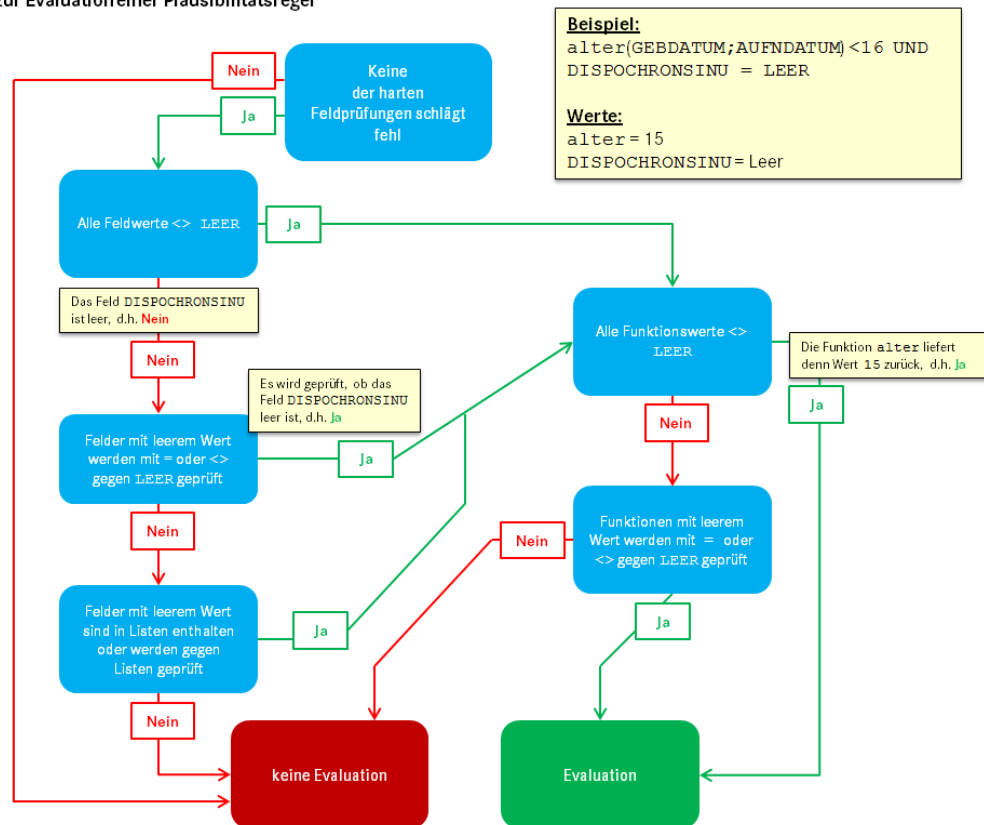


Abbildung 19: Algorithmus zur Evaluation von Plausibilitätsregeln

Die in Tabelle 65 dargestellten Plausibilitätsregel zeigen Beispiele für die Evaluation gemäß dem beschriebenen Algorithmus.

Tabelle 65: Beispiele für die Evaluation von Regeln

| Beispiel | Erläuterung | Evaluation |
|---|--|--|
| HTX MECHKREISLUNTER STUETZ <> 0 UND abstKreislau UntDatum = LEER | Die Prüfung <> Wert bezieht sich auf ein Muss-Feld. | Die Regel wird evaluiert. |
| PNTXFU (TRANSPLANTATVERS = LEER ODER TRANSPLANTATVERS <> 1) UND TRANSPLANTATVERSDATUM <> LEER | Die Prüfung <> Wert bezieht sich auf ein Kann-Feld, es wird aber explizit gegen LEER geprüft. | Die Regel wird evaluiert. |
| HCH ENTLGRUND = '07' UND abstTodEntldatum <> 0 | Die Prüfung <> Wert bezieht sich auf ein Kann-Feld, es wird nicht explizit gegen LEER geprüft. | Die Regel wird nicht evaluiert, wenn das Feld abstTod-Entldatum leer ist |

| Beispiel | | Erläuterung | Evaluation |
|----------|--|---|---|
| HCH | ENTLGRUND = '07' UND MaxGanzeZahl (@poopvwdauer) <= 30 UND STATUS30TAGEPOSTOP <> 0 | Die Prüfung <> Wert bezieht sich auf ein Feld eines optionalen Bogens (Bogen.fkBogen- Zahl = ?). | Die Regel wird nicht evaluiert, wenn der Bogen HCH:FU nicht ausgefüllt wurde. |

Teildatensatzübergreifende Regeln

Teildatensatzübergreifende Regeln (Abschnitt B 2.4.6) müssen u.U. mehrfach evaluiert werden (für jede Kombination von Teildatensätzen, die von der Regel betroffen ist).

Beispiel (Spezifikation 2018):

In Modul PCI (Perkutane Koronarintervention und Koronarangiographie) gibt es die Regel 8867:

```
STATUSLE IN (1;2) UND ARTLE IN (2;3) UND OPDATUM > ENTLDATUM
```

Diese Regel hat einen Bezug zum letzten Teildatensatz PCI:B (Feld ENTLDATUM) über die Felder STATUSLE und ARTLE zum ersten Teildatensatz PCI:B und zum Teildatensatz PCI:PROZ (Feld OPDATUM).

Es wird angenommen, dass bei einem Patienten (in einem Krankenhaus oder durch den Vertragsarzt in Form einer belegärztlichen Leistung) folgende Eingriffe (stationär oder vorstationär) durchgeführt wurden:

- 03.05.2018: Eingriff (Eingriffsnummer 1) Koronarangiographie
- 06.05.2018: Eingriff (Eingriffsnummer 2) PCI
- 10.05.2018: Entlassung aus dem Krankenhaus

In der QS-Dokumentation wird das Modul PCI mit den Teildatensätzen PCI:B, PCI:PROZ[1] und PCI:PROZ[2] angelegt. In den eckigen Klammern findet sich der Wert des Datenfeldes Eingriffsnummer LFDNREINGRIFF aus dem Teildatensatz PCI:PROZ.

Für die Felder ENTLDATUM und OPDATUM gibt es dann folgende Werte:

- PCI:B.ENTLDATUM = '10.05.2018'
- PCI:PROZ[LFDNREINGRIFF=1].OPDATUM = '03.05.2018'
- PCI:PROZ[LFDNREINGRIFF=2].OPDATUM = '06.05.2018'

Die Regel muss somit zweimal evaluiert werden. Nachfolgend sind die Datumswerte bereits eingesetzt:

- STATUSLE IN (1;2) UND ARTLE IN (2;3) UND '03.05.2018' > '10.05.2018'
- STATUSLE IN (1;2) UND ARTLE IN (2;3) UND '06.05.2018' > '10.05.2018'

**Hinweis**

In wenigen Einzelfällen beziehen sich Plausibilitätsregeln auf mehr als zwei Teildatensätze.

2.5 Listen von Schlüsselcodes (OPS, ICD-10-GM)

In der Spezifikation sind Listen von OPS- bzw. ICD-10-GM-Kodes¹¹² in separaten Tabellen definiert. Jede Liste hat einen technischen Namen (z. B. GEB_ICD) und eine erläuternde Bezeichnung (z. B. Einschlussdiagnosen Perinatalmedizin). Die Listen sind in insgesamt vier Tabellen der Spezifikationsdatenbank definiert. Die technischen Namen und erläuternden Bezeichnungen sind in den Tabellen OPSListe bzw. ICDListe definiert. Die Codes finden sich in den Tabellen OPSWert bzw. ICDWert.

Beispiel:

Die Einschlussprozeduren Perinatalmedizin (GEB_OPS) sind in der Spezifikation für QS-Filter zu finden.

**Achtung**

Einige Listen der Tabellen OPSListe oder ICDListe sind als QS-Filter-Listen deklariert (Attribut qsFilter). Die Definitionen der QS-Filter-Listen sind in der für das Erfassungsjahr gültigen Spezifikation für QS-Filter zu finden. Dort existieren die gleichen Tabellen OPSListe, ICDListe, OPSWert und ICDWert mit den entsprechenden Listendefinitionen.

2.5.1 OPS-Listen

Jede OPS-Liste ist charakterisiert durch ihren Namen (Attribut name in Tabelle OPSListe), welcher nach folgendem Schema gebildet wird:

```
{<TEXT>_}OPS{_<TEXT>}
```

Hinter <TEXT> verbirgt sich ein frei wählbarer Name (Erlaubte Zeichen: A–Z, a–z, 0–9, _ . Umlaute sind nicht erlaubt.). Die {}-Ausdrücke sind optional.

Beispiele:

| | |
|-----------------|---------------------------------------|
| OPS_Mastektomie | Prozeduren Mastektomie |
| GEB_OPS | Einschlussprozeduren Perinatalmedizin |
| HCH_OPS_EX | Ausschlussprozeduren Herzchirurgie |

¹¹² Die aktuell gültigen Kataloge sind über das DIMDI (<http://www.dimdi.de>) zu beziehen.

Umgang mit Seitenlokalisationen

Die Codes der OPS-Listen enthalten keine Seitenlokalisationen, obwohl die Zusatzkennzeichen für Seitenbezeichnung R, L oder B für Prozeduren an Lokalisationen, die paarig vorhanden sind (z. B. Leiste, Niere, Oberschenkel) verpflichtend zu dokumentieren sind.¹¹³ Für die Prüfung, ob zwei Codes identisch sind, genügt kein einfacher Stringvergleich. Stattdessen wird ein Stringvergleich der Normcodes¹¹⁴ durchgeführt, um die Übereinstimmung zwischen dem dokumentierten Code und dem einer OPS-Liste zu ermitteln (siehe Tabelle 66).

Tabelle 66: Identitätsprüfung zwischen dokumentierten OPS-Codes und Codes von OPS-Listen

| Dokumentierter OPS-Code | OPS-Code der OPS-Liste | Bedingung für Gleichheit (= ist Stringvergleich) |
|---|----------------------------|--|
| <code>normCodeDok + seiteDok</code> | <code>normCodeListe</code> | <code>normCodeDok = normCodeListe</code> |
| <code>normCodeDok</code> | <code>normCodeListe</code> | <code>normCodeDok = normCodeListe</code> |

Bei allen Prüfungen mit OPS-Listen (z. B. OPSCHLUESSEL EINSIN HCH_OPS_EX) sind diese Regeln zu beachten (Abschnitt B 2.4.4).

2.5.2 ICD-Listen

Jede ICD-Liste ist charakterisiert durch ihren Namen (Attribut `name` in Tabelle `ICDListe`), welcher per definitionem folgendem Schema folgt:

```
{ <TEXT>_ } ICD { _<TEXT> }
```

Hinter `<TEXT>` verbirgt sich ein frei wählbarer Name (Erlaubte Zeichen: A–Z, a–z, 0–9, _, Umlaute sind nicht erlaubt).

Die in der Tabelle `ICDWert` (Attribut `code`) definierten Codes entsprechen der Systematik der Spalte `NormCode` aus Tabelle `Codes` in den Katalogen des DIMDI:

Der ICD-10-GM wird 4- oder 5-stellig kodiert, kann aber durch ein Suffix bestehend aus `[A|V|Z][L|R|B]` (ohne Leerzeichen, z. B. „K41.9ZL“) ergänzt werden.

¹¹³ In der QS-Dokumentation wird das Zusatzkennzeichen für die Seitenbezeichnung getrennt durch einen Doppelpunkt dem OPS-Code angehängt. Fehlt ein erforderliches Zusatzkennzeichen, so ist die Dokumentation unplausibel (Abschnitt B 1.4.4).

¹¹⁴ Jeder OPS-Code `code` lässt sich entweder als Code mit Seitenlokalisation: `code=normCode+seite` oder als Code ohne Seitenlokalisation `code=normCode` darstellen.



Hinweis

Die Suffixe *, +, ! entfallen in der Spezifikation!

Es ist zu beachten, dass im Krankenhaus dokumentierte ICD-Kodes die Suffixe *, +, ! enthalten können.

Beispiel:

In der ICD-Liste GEB_ICD ist der Code **Z37.9** definiert. Bei der Prüfung, ob der im Krankenhaus dokumentierte Code **Z37.9!** in der Liste GEB_ICD enthalten ist, muss die Software zu einem positiven Ergebnis kommen.

Umgang mit Kennzeichen zur Diagnosesicherheit

Im ambulanten Sektor muss ein Kennzeichen zur Diagnosesicherheit angegeben werden. Das Zusatzkennzeichen zur Diagnosesicherheit ist in den ICD-Listen der Spezifikationsdaten-banken nicht enthalten. Das Zusatzkennzeichen ist für ambulante Fälle dennoch zu dokumentieren. Die Diagnosesicherheit ist im Modul PCI wie folgt zu dokumentieren:

- Die Diagnosesicherheit ist **nicht** zu dokumentieren, wenn
STATUSLE = 1 UND ARTLE IN (2;3)
- Die Diagnosesicherheit ist **verpflichtend** zu dokumentieren, wenn
STATUSLE = 2 ODER (STATUSLE = 3 UND ARTLE = 1)
- Die Diagnosesicherheit ist **optional** anzugeben, wenn
STATUSLE = 3 UND ARTLE IN (2;3) ODER (STATUSLE = 1 UND ARTLE = 1)

Die Diagnosesicherheit wird in Plausibilitätsregeln mithilfe von regulären Ausdrücken geprüft.

Beispiel PCI:

Bei der **Dokumentation** des ICD-Kodes Z52.7 soll es nur zu einer Fehlermeldung kommen, wenn dieser nicht mit A (Ausschluss) gekennzeichnet ist.

```
format(ENTLDIAG; '[a-zA-Z][0-9]{2}(\.[0-9]{1,2})?([#\+*\+!])?([GVZgvz])?([RLBr1b])?$', ' ) = WAHR UND ENL-
LDIAG EINSIN PCI_ICD_EX
```

Die **Auslösung** eines Falls soll nicht stattfinden, wenn der ICD-Kode Z52.7 nicht mit A (Ausschluss), V (Verdacht auf) oder Z (Zustand nach) gekennzeichnet ist.

```
format(DIAG; '[a-zA-Z][0-9]{2}(\.[0-9]{1,2})?([#\+*\+!])?([Gg])?([RLBr1b])?$', ' ) = WAHR UND DIAG
KEINSIN PCI_ICD_EX
```

2.6 Exportfeldbeschreibung

Neben der Datenfeldbeschreibung (Abschnitt B 2.3) enthält die Spezifikationsdatenbank die Beschreibung der Exportfelder für ein Modul (Exportdatensatz). Diese werden zum Teil über Ersatzfelder berechnet. Ab der Spezifikation 2016 wird zudem zwischen Dokumentationsmodulen (Abschnitt B 2.3.1) und Exportmodulen unterschieden.

2.6.1 Exportmodule

Um die unterschiedlichen Abrechnungswege und davon abhängige Datenflüsse mit unterschiedlichen Datenannahmestellen für die Datenübermittlung zu berücksichtigen, wird ab der Spezifikation 2016 zwischen den bisherigen Dokumentationsmodulen (Tabelle Modul) und den neu entwickelten Exportmodulen (Tabelle Exportmodul) unterschieden.

Tabelle 67: Struktur der Tabelle *ExportModul*

| Feldname | Datentyp | Bemerkung |
|-------------------------|----------|--|
| idExportModul | INTEGER | Primärschlüssel |
| fkModul | INTEGER | Bezug zum Dokumentationsmodul |
| name | TEXT | Technischer Name (muss eindeutig sein) |
| bezeichnung | TEXT | Bezeichnung des Exportmoduls |
| existenzBedingung | TEXT | Definiert, unter welcher Bedingung das Modul in ein definiertes Exportmodul transformiert wird. |
| type_QS_data | TEXT | Datentyp im XML-Schema |
| ersatzBedingungMDS | TEXT | Definiert, unter welcher Bedingung das Modul MDS in ein definiertes Exportmodul transformiert wird. |
| type_QS_data_mds | TEXT | Datentyp im XML-Schema |
| fkExportzeitraumEntlEJ | INTEGER | Definiert den Exportzeitraum für ein Exportmodul bei Entlassung im Erfassungsjahr. Die hinterlegten Zeitpunkte sind in der Tabelle <i>Exportzeitraum</i> definiert. <i>Exportzeitraum.exportBis</i> = Datenlieferfrist inkl. Korrekturfrist <i>Exportzeitraum.exportBisQ4</i> = Datenlieferfrist ohne Korrekturfrist |
| fkExportzeitraumEntlEJ1 | INTEGER | Definiert den Exportzeitraum für ein Exportmodul bei Entlassung im Erfassungsjahr + 1 (Langlieger). Die hinterlegten Zeitpunkte sind in der Tabelle <i>Exportzeitraum</i> definiert. <i>Exportzeitraum.exportBis</i> = Datenlieferfrist inkl. Korrekturfrist <i>Exportzeitraum.exportBisQ4</i> = Datenlieferfrist ohne Korrekturfrist |

Beispiel:

Das Modul `PCI` gliedert sich nach Abschluss der Dokumentation bei Erfüllen der Existenzbedingung in drei Exportmodule, die jeweils einem separaten Datenfluss entsprechen (`PCI_LKG`, `PCI_KV`, `PCI_SV`). Wird im Dokumentationsmodul `PCI` im Feld `STATUSLE` beispielsweise `1 = Krankenhaus` kodiert, wird dieses nach Abschluss der Dokumentation in das Exportmodul `PCI_LKG` transformiert. Damit wird die korrekte Datenannahmestelle definiert, an die der Datensatz zu übermitteln ist – in diesem Fall die `LQS/LKG`.

Das Attribut `type_QS_data` gibt Auskunft darüber, welchem Datentyp dies im XML-Schema entspricht.

Beispiel:

Beim Exportmodul `PCI_LKG` ist der Datentyp im XML-Schema `qs_data_pci_type`.

Softwareanbietern soll hiermit die Integration eines Mechanismus der automatischen Datentypzuweisung ermöglicht werden, um den Aufwand zu reduzieren und Fehler zu vermeiden. Für Dokumentationsmodule, die das Anlegen eines Minimaldatensatzes zulassen, wird im Attribut `ersatzBedingungMDS` der Tabelle `ExportModul` definiert, unter welcher Bedingung das Modul `MDS` in ein definiertes Exportmodul transformiert wird.

Beispiel:

Da für das Dokumentationsmodul `PCI` die Möglichkeit besteht, einen Minimaldatensatz anzulegen, wird im Attribut `ersatzBedingungMDS` definiert, unter welcher Bedingung das Modul `MDS` in ein definiertes Exportmodul transformiert wird. Wird im Modul `MDS` im Feld `ZUQSMODUL` beispielsweise `'PCI'` und im Feld `STATUSLE` `1 = Krankenhaus` kodiert, wird dieses nach Abschluss der Dokumentation in das Exportmodul `PCI_LKG` transformiert.

Das Attribut `type_QS_data_mds` gibt Auskunft darüber, welchem Datentyp dies im XML-Schema entspricht.

Beispiel:

Beim Exportmodul `PCI_LKG` ist der Datentyp im XML-Schema `qs_data_mds_type`.

Attributwerte liegen hierbei nur vor, wenn für das Dokumentationsmodul ein Minimaldatensatz angelegt werden kann.

2.6.2 Exportdatensatz

Der Exportdatensatz enthält die Exportfelder für ein Modul. Welche Zusatzfelder, Bogenfelder und/oder Ersatzfelder den Exportdatensatz pro Modul bilden, ist in Tabelle `ExportFormat` definiert.

Zusatzfelder¹¹⁵

Ein Exportfelddatensatz beginnt immer mit den folgenden Zusatzfeldern:

| | | |
|--------------|---|--|
| RegistrierNr | = | Registriernummer des Dokumentationssystems (Länderkode + Registrierkode) ¹¹⁶ |
| Vorgangsnr | = | Vorgangsnummer |
| VersionNr | = | Versionsnummer |
| Storno | = | Stornierung eines Datensatzes (inkl. aller Teildatensätze) |
| Modul | = | Bezeichnung des Exportmoduls |
| Bogen | = | Teildatensatz (Bogen) |
| DokAbschlDat | = | Dokumentationsabschlussdatum |

Ein neuer Teildatensatz beginnt mit den Zusatzfeldern `RegistrierNr`, `Vorgangsnr` und `VersionNr`. Teildatensätze mit einem definierten Mutterteildatensatz beinhalten zusätzlich das Zusatzfeld `IdBogenFeldMutter` (= Wert des eindeutigen Bogenfeldes des Mutterteildatensatzes).

Zusatzfelder, welche nicht in der Datenfeldbeschreibung (Tabelle `BogenFeld`) eines Moduls enthalten sind, werden von der QS-Dokumentationssoftware ausgefüllt.¹¹⁷

Einige der in der Tabelle `ZusatzFeld` definierten Zusatzfelder werden nachfolgend erläutert:

- Das übertragene Speicherdatum `DokAbschlDat` (Datum des Dokumentationsabschlusses bzw. der Freigabe des Datensatzes für den Export) ist nicht Teil der Datenbank für Auswertungen und wird nur für organisatorische Zwecke verwendet. Das `DokAbschlDat` ist das Datum der letzten Änderung des gesamten Datensatzes.

Beispiel:

Wird z. B. ein neuer Teildatensatz `FU` für einen bereits abgeschlossenen oder ggf. exportierten Basisdatensatz `HCH` erzeugt, muss auch das `DokAbschlDat` entsprechend geändert werden.

- Die Versionsnummer (`VersionNr`) gibt an, welche Version des Datensatzes übertragen wird.
- In der Regel wird die Versionsnummer 1 lauten, d.h., dass der nach dem ersten Dokumentationsabschluss freigegebene Datensatz übertragen wird. Muss ein korrigierter Datensatz erneut eingesandt werden, so muss die Versionsnummer vom dokumentierenden System um eins erhöht werden. Die neue Version des Datensatzes wird bei der Entgegennahme geprüft und überschreibt bei Korrektheit die alte Version des Datensatzes.

¹¹⁵ Dabei ist zu beachten, dass die Feldnamen beim Export durch die XML-Elemente zu ersetzen sind

¹¹⁶ Für die Datenübermittlung an KVen und DAS-SV gelten andere Regelungen als bei der Datenübermittlung an die LQS/LKG.

¹¹⁷ Hier gilt also nicht der Grundsatz, dass Felder nicht vorbelegt sein dürfen.

**Achtung**

Wenn die entgegennehmende Stelle einen Datensatz mit derselben Versionsnummer ein zweites Mal erhält, so wird dieser zurückgewiesen.

- Der Eintrag 1 im Zusatzfeld `Storno` veranlasst die datenentgegennehmende Stelle, den übermittelten Datensatz einschließlich seiner Vorversion(en) als „storniert“ zu kennzeichnen.
- Das Zusatzfeld `IdBogenFeldMutter` wird bei Teildatensätzen eingefügt, welche einen mehrfach anlegbaren Elternteildatensatz (Attribut `fkBogenZahl = '*'` oder `'+'`) haben. In diesem Fall wird die identifizierende Nummer des Elternteildatensatzes (konfiguriert über `Bogen.fkEindeutigBogenFeld`) im Kindteildatensatz übermittelt.

Beispiel:

Beim Übermitteln des Teildatensatzes `PCI:KORO` muss als `IdBogenFeldMutter` der Wert des Bogenfeldes `LFDNREINGRIFF` des Mutterteildatensatzes `PCI:PROZ` eingetragen werden.

Eine vollständige Liste der möglichen Zusatzfelder findet sich in der Tabelle `ZusatzFeld` der Spezifikationsdatenbank zur QS-Dokumentation. Zusatzfelder sind in Tabelle 68 definiert.

Tabelle 68. Struktur der Tabelle `ZusatzFeld`

| Feldname | Datentyp | Bemerkung |
|------------------------------|----------|---|
| <code>idZusatzFeld</code> | INTEGER | Primärschlüssel |
| <code>name</code> | TEXT | Technischer Name (muss eindeutig sein) |
| <code>bezeichnung</code> | TEXT | Bezeichnung des Zusatzfeldes |
| <code>fkBasisTyp</code> | INTEGER | Fremdschlüssel zur Tabelle <code>BasisTyp</code> |
| <code>fkSchluessel</code> | INTEGER | Fremdschlüssel zur Tabelle <code>Schlüssel</code> |
| <code>laenge</code> | INTEGER | Feldlänge des Zusatzfeldes |
| <code>nachKommaLaenge</code> | INTEGER | Anzahl der Nachkommastellen |
| <code>istListe</code> | BOOLEAN | Wenn <code>istListe = WAHR</code> , so sind die vom betreffenden Feld abgeleiteten Bogenfelder Listenfelder. |
| <code>min</code> | INTEGER | Harte Untergrenze des Wertebereichs eines numerischen Datenfeldes (modulübergreifend). Die Definition ist optional. |
| <code>max</code> | INTEGER | Harte Obergrenze des Wertebereichs eines numerischen Datenfeldes (modulübergreifend). Die Definition ist optional. |

| Feldname | Datentyp | Bemerkung |
|-----------------------|----------|---|
| sortierNr | INTEGER | Sortiernummer |
| nurBasisTDS | BOOLEAN | Wenn nurBasisTDS = WAHR, so ist das Zusatzfeld im Exortdatensatz nur im Basisbogen enthalten |
| nurKreuzStern-KindTDS | BOOLEAN | Wenn nurKreuzSternKindTDS = WAHR, so ist das Zusatzfeld nur bei Teildatensätzen mit einem definierten Mutterteildatensatz enthalten |

Die Exportfelder werden pro Exportmodul exportiert. Hierfür erhält das Zusatzfeld Modul ab der Spezifikation 2016 den Datentyp SCHLUESSEL und lässt nur die im Schlüssel Exportmodul definierten Codes zu.



Hinweis

Bei den Zusatzfeldern ist zu beachten, dass die Feldnamen beim Export durch die entsprechenden XML-Elemente zu ersetzen sind (Abschnitt B 2.8.1).

Ersatzfelder für den Datenexport

Ersatzfelder werden aus einem oder mehreren Feldern der Datenfeldbeschreibung berechnet. Mit einem Ersatzfeld verknüpfte Bogenfelder werden nicht exportiert, wenn sie nicht als <bleibt> gekennzeichnet sind. Stattdessen werden ein oder mehrere Ersatzfelder exportiert. Vorrangig dienen Ersatzfelder der Anonymisierung beim Datenexport. Die verwendeten Ersatzfelder sind in der Tabelle ErsatzFeld gespeichert.

Tabelle 69. Struktur der Tabelle Ersatzfeld

| Feldname | Datentyp | Bemerkung |
|--------------|----------|---|
| idErsatzFeld | INTEGER | Primärschlüssel |
| name | TEXT | Technischer Name (muss eindeutig sein) |
| bezeichnung | TEXT | Bezeichnung des Ersatzfeldes |
| formel | TEXT | Berechnungsformel der Ersatzfelder |
| fkBasisTyp | INTEGER | Obligatorischer Fremdschlüssel zum Basistyp |
| fkSchluessel | INTEGER | Optionaler Fremdschlüssel zum Schlüssel |

Beispiel:

Das Ersatzfeld „postoperative Verweildauer“ wird folgendermaßen berechnet:

$$\text{poopvwdauer} = \text{ENTLDATUM} - \text{OPDATUM}$$

Ersatzfelder, die nicht berechnet werden können

Es kann auch vorkommen, dass Ersatzfelder für einen Datensatz nicht berechnet werden können, weil die der Berechnung zugrunde liegenden Bogenfelder nicht ausgefüllt (LEER) sind. Folgende allgemeine Regeln gelten für die Berechnung von Ersatzfeldern:

- Wenn die Bogenfelder, aus denen ein Ersatzfeld berechnet wird, dokumentiert sind (<> LEER), so ist das entsprechende Ersatzfeld zu berechnen und zu exportieren.
- Wenn eines der beteiligten Bogenfelder nicht ausgefüllt ist und somit auch kein Ersatzfeld berechnet werden kann, wird kein Wert für das Ersatzfeld exportiert (bleibt LEER). Eine Ausnahme bildet die Funktion `verkettenmt`. Die Funktion `verketten` (zwei oder mehrere) Zeichenfolgen zu einer Zeichenfolge. Ist lediglich eine übergebene Zeichenfolge nicht leer, wird diese ohne Verkettung zurückgeliefert.

Anonymisierungsvorschriften

Die Anonymisierung von Datenfeldern wird wie aus der folgenden Tabelle 70 Struktur der Tabelle `ErsatzFuerFeld` ersichtlich konfiguriert. Für die Programmierung der Exportfelder ist dieser Abschnitt nicht relevant, da die Exportfelder direkt über die Abfrage `ExportFelderFürEinModul` bzw. die Tabelle `ExportFormat` ermittelt werden können. Die Tabelle `ErsatzFuerFeld` ordnet einem Feld (Tabelle `Feld`) oder Bogenfeld ein oder mehrere Ersatzfelder zu. Die über das Attribut `fkFeld` definierte Anonymisierung ist die Standardanonymisierung für alle Module. Sie kann jedoch durch eine modulspezifische Anonymisierung überschrieben werden: Wenn ein Ersatzfeld mit einem Bogenfeld (über Attribut `fkBogenFeld`) verknüpft ist, wird statt des Bogenfelds das berechnete Ersatzfeld in die Exportdatei des Teildatensatzes geschrieben.

Tabelle 70. Struktur der Tabelle `ErsatzFuerFeld`

| Feldname | Datentyp | Bemerkung |
|-------------------------------|----------|--|
| <code>idErsatzFuerFeld</code> | INTEGER | Primärschlüssel |
| <code>fkFeld</code> | INTEGER | Optionaler Fremdschlüssel zur Tabelle <code>Feld</code> |
| <code>fkBogenFeld</code> | TEXT | Optionaler Fremdschlüssel zur Tabelle <code>BogenFeld</code> |
| <code>fkErsatzFeld</code> | TEXT | Obligatorischer Fremdschlüssel zur Tabelle <code>ErsatzFeld</code> |
| <code>parametrierbar</code> | BOOL | Kennzeichen, ob das Feld parametrierbar ist |

Beispiel:

Das Feld `ENTL DATUM` in Modul 15/1 (QSKH-RL) hat die Ersatzfelder `vwDauer`, `entlquartal`, `entlwochtag` und das parametrierbare Ersatzfeld `<bleibt>`.

Wenn das Ersatzfeld `<entfällt>` mit einem Bogenfeld verknüpft ist, entfällt das Bogenfeld in der Exportdatei. Wenn das Ersatzfeld `<bleibt>` mit einem Bogenfeld verknüpft ist, wird das Bogenfeld unverändert in die Exportdatei übernommen.

Wenn ein einziges Ersatzfeld sowohl über `fkFeld` als auch über `fkBogenFeld` definiert ist („doppelte Definition“), hat die spezielle Anonymisierung (über `fkBogenFeld`) Vorrang. Die allgemeinen Anonymisierungen (`fkFeld`) werden ignoriert. Die allgemeine Definition kommt nur in den Modulen zur Anwendung, in denen keine spezielle Anonymisierung vorliegt.

Beispiel:

Dem Feld `ENTLDATE` sind über das Attribut `fkFeld` die Ersatzfelder `vwdauer`, `entlwochtag` und `entlquartal` zugewiesen. In einigen Bundesländern kann das Entlassungsdatum auch direkt übermittelt werden. Bei einigen Modulen sind daher in `fkBogenFeld` das parametrierbare Ersatzfeld `<bleibt>` und die anonymisierten Ersatzfelder `vwdauer`, `entlwochtag` und `entlquartal` definiert.

Parametrierung

Die Verknüpfung zwischen `Feld` bzw. `Bogenfeld` und `Ersatzfeld` kann parametriert werden (Attribut `parametrierbar`). Parametrierbare Ersatzfelder erscheinen immer als eigenes Element in der Exportdatei. Es ist aber über die Dokumentationssoftware konfigurierbar, ob die Werte auch tatsächlich exportiert werden. Auf diese Weise können spezifische Erfordernisse zum Datenschutz auf Landesebene berücksichtigt werden.

Beispiel:

Im Modul 16/1 gibt es für das Feld `PLZ` (Postleitzahl) drei parametrierbare Ersatzfelder: `<bleibt>`, `PLZ3stellig` und `PLZ4stellig`.

In manchen Bundesländern dürfen Postleitzahlen unverändert exportiert werden, in anderen gibt es strengere Anforderungen: Dort werden z. B. nicht alle Stellen einer Postleitzahl exportiert.

**Achtung**

Die Übermittlung von parametrierbaren Exportfeldern wird von der jeweils zuständigen Landesebene bzw. bei Direktverfahren von der Bundesebene festgelegt. Wenn keine Festlegung getroffen worden ist, bleiben die parametrierbaren Exportfelder leer¹¹⁸.

Muss-Felder des Exportdatensatzes

Verbindlich für die Muss/Kann-Prüfung ist die Definition in der Tabelle `BogenFeld`. Die Muss/Kann-Zuordnungen im Exportdatensatz werden hieraus abgeleitet:

- Für Exportfelder, die nicht pseudonymisiert werden und die keine Listenfelder sind, entspricht die Muss/Kann-Zuordnung der Definition in der Tabelle `BogenFeld`.
- Die Muss/Kann-Zuordnung der pseudonymisierten Datenfelder (Ersatzfelder) ergibt sich logisch aus der Berechnungsformel (Attribut `formel` in Tabelle `ErsatzFeld`). Beispielsweise

¹¹⁸ Die entsprechenden Exportfelder existieren zwar inkl. Feldnamen in der Exportdatei, die Werte werden jedoch nicht eingetragen.

ist ein Ersatzfeld ein Muss-Feld, wenn alle an der Berechnung beteiligten Bogenfelder Muss-Felder sind.

- Bei Muss-Listenfeldern der Tabelle `BogenFeld` ist grundsätzlich nur das erste Element ein Muss-Feld, die weiteren Elementfelder sind Kann-Felder. Hierbei ist zu beachten, dass Exportfelder für Listenfelder seit der Spezifikation 2015 nicht mehr pro Listenelement, sondern pro Listefeld dargestellt werden. Die Anzahl der Elemente von Listenfeldern ist den Abfragen `Exportfelder` und `ExportfelderFürEinModul` zu entnehmen (`ExportfelderFürEinModul.elemente`).
- Zusatzfelder der Tabelle `ExportFormat` sind Muss-Felder, außer wenn sie Schlüsselfelder mit einem `Jleer`-Schlüssel (z. B. das Zusatzfeld `Storno`) sind.

Beispiel:

Das Feld `entlquartal` berechnet sich über die Formel `quartal(ENTLDATEUM)`.

In Modulen, in denen das Datenfeld `ENTLDATEUM` ein Muss-Feld ist, ist auch das Ersatzfeld ein Muss-Feld und die Datenannahmestelle ist verpflichtet, die Muss-Prüfung hier auch durchzuführen. Ansonsten ist das Ersatzfeld ein Kann-Feld.

**Achtung**

Wenn ein Listefeld als Muss-Feld deklariert ist, so ist nur das erste Exportfeld der Liste ein Muss-Feld, die restlichen Elemente sind Kann-Felder. Wenn ein Listefeld als Kann-Feld deklariert ist, so sind alle exportierten Elemente ebenfalls Kann-Felder.

Das Nichtausführen der erforderlichen Muss-Prüfungen kann gravierende Folgen für die Auswertung haben!

Als Hilfestellung für Datenannahmestellen bei der Umsetzung gilt das Attribut `fkMussKann` in der Tabelle `ExportFormat`, deren Inhalte automatisch generiert werden.

2.7 Versionierung

Im Folgenden werden die Tabelle `Version`, der Abgleich zu vorherigen Versionen, die Abgrenzung zwischen Erfassungsjahren und Datensatzformaten sowie die Version von Exportverfahren und -dateien beschrieben.

2.7.1 Grundlegende Definitionen

In der Tabelle `Version` finden sich Informationen zur Version der Spezifikationsdatenbank. Die wichtigsten Eigenschaften einer Version sind der Versionsname (Attribut `name`) und die Gültigkeitszeiträume (Attribute `ab` und `bis`). Der Gültigkeitszeitraum einer Version ist in der Regel ein Erfassungsjahr (z. B. Aufnahme zwischen dem 1. Januar 2019 und dem 31. Dezember 2019).

Die QS-Dokumentationssoftware eines Erfassungsjahres wird für diejenigen Behandlungsfälle verwendet, deren Aufnahmedatum in das Krankenhaus (stationäre Fälle) bzw. deren Behandlungsdatum (ambulante Fälle) in den oben definierten Gültigkeitszeitraum fällt. Zudem wird im

stationären Sektor bei sogenannten „Überliegern“ (Aufnahmedatum im alten Jahr, Entlassungsdatum im nachfolgenden Jahr) die QS-Dokumentationssoftware auch für Behandlungsfälle genutzt, die nach dem in der Datenbank definierten Gültigkeitszeitraum (in der Regel nach dem 31. Dezember) entlassen worden sind.

Jedes Modul der Datenbank hat eine Version (vgl. Attribut `fkVersion` in Tabelle `Modul`). In einer Spezifikationsdatenbank haben alle Module dieselbe Version. Diese entspricht immer der in Tabelle `Version` als gültig gekennzeichneten Version. Über die in der Datenbank definierten Relationen sind auch für alle Bogenfelder (Tabelle `BogenFeld`), Exportfelder (Tabelle `ExportFormat`) und Plausibilitätsregeln (Tabelle `Regeln`) Versionen definiert.

Status der Spezifikation

Versionen können den Status `in Entwicklung`, `final` oder `Update` der finalen Spezifikation haben. Diese Zustände werden zum Nachschlagen in der Tabelle `VersStatus`¹¹⁹ verwaltet. Das Attribut `gueltig` zeigt die gültige Version der Datenbank an. Es darf nur eine einzige Version als gültig markiert sein.

Beispiel:

Die Version 2019 V01, einschließlich ihrer Updates, sind finale Versionen.

Hat eine Spezifikationsdatenbank den Status `in Entwicklung`, kann `Modul.fkVersion` als `ungueltig` markierte Versionen enthalten, um Zwischenstände abzubilden.

Historie der Versionen

Die Tabelle `Version` enthält auch einen Selbstbezug (Attribut `fkVersion`), der die Identifizierung der Vorgängerversion ermöglicht.

Die Vorgängerversion der Spezifikation 2019 V02 ist die Version 2019 V01.

2.7.2 Delta-Informationen zur vorhergehenden Version

Um den Benutzern der Spezifikation umfassende Informationen zu den jeweiligen Änderungen zur Verfügung zu stellen, enthält die Spezifikationsdatenbank Tabellen, die den Änderungsstand im Vergleich mit der letzten gültigen Version des Vorjahres und zur Vorversion der Datenbank anzeigen. Diese sogenannten Delta-Tabellen werden automatisch generiert.

Beispiel:

Die Datenbank 2019_BASIS_QSF_V01 stellt die Änderungen im Vergleich zur letzten gültigen Spezifikationsdatenbank des Vorjahres 2018_BASIS_QSF_V04 dar.

Die Datenbank 2019_BASIS_QSF_V02 stellt zusätzlich die Änderungen im Vergleich zur Vorversion 2019_BASIS_QSF_V01 dar.

¹¹⁹ Der Status „Service Release zur finalen Spezifikation“ wird seit der Spezifikation 2014 nicht mehr verwendet.

Es gibt drei Delta-Tabellen, die die neuen Entitäten, die gelöschten Entitäten und die geänderten Attributwerte weiter bestehender Entitäten aufzeigen.

Neue Entitäten

Über die Attribute `id` und `fkTabellenStruktur` in der Tabelle `DeltaNeu` (siehe Tabelle 71) ist die Zuordnung zu den Entitäten der Datenbank möglich.

Tabelle 71. Struktur der Tabelle `DeltaNeu`

| Feldname | Datentyp | Bemerkung |
|---------------------------------|----------|---|
| <code>idDeltaNeu</code> | INTEGER | Primärschlüssel |
| <code>id</code> | TEXT | ID der Entität, die eingefügt wurde |
| <code>fkTabellenStruktur</code> | ZAHL | Bezug zur Tabelle, in die die Entität eingefügt wurde |
| <code>bemerkung</code> | MEMO | Begründung für die Ergänzung |

Beispiel:

Die Zeile `idDeltaNeu = 10914` enthält in der Spezifikation 2015 V01 folgende Werte:

`id = 8063`

`fkTabellenStruktur = <BogenFeld>`

`bemerkung = HCH.B.KOERPERGEWICHTNB[12.0]`

Dies bedeutet, dass in der Tabelle `BogenFeld` ein neuer Eintrag mit `idBogenFeld = 8063` angelegt worden ist. Unter `bemerkung` finden sich weitere Erläuterungen zum neuen Tabelleneintrag.

Geänderte Entitäten

Über die Attribute `id` und `fkTabellenFeldStruktur` der Tabelle `DeltaAttribut` (siehe Tabelle 72) ist die Zuordnung zu den Attributen der Entitäten der Datenbank möglich.

Tabelle 72: Struktur der Tabelle `DeltaAttribut`

| Feldname | Datentyp | Bemerkung |
|-------------------------------------|----------|--|
| <code>idDeltaAttribut</code> | AUTOWERT | Primärschlüssel |
| <code>id</code> | TEXT | ID der Entität, die geändert wurde |
| <code>fkTabellenFeldStruktur</code> | ZAHL | Bezug zum Attribut einer Tabelle, in der die Entität geändert wurde |
| <code>alterInhalt</code> | MEMO | Alter Inhalt der geänderten Entität in der letzten finalen Spezifikation |

| Feldname | Datentyp | Bemerkung |
|-------------|----------|--|
| neuerInhalt | MEMO | Neuer Inhalt dieser Entität in der aktuellen Spezifikation |
| bemerkung | MEMO | Begründung für die Ergänzung |

Beispiel:

Die Zeile `idDeltaAttribut = 16312` enthält in der Spezifikation 2015 V01 folgende Werte:

`Id = 5972`

`Tabellenattribut = <BogenFeld.bezeichnung>`

`alterInhalt = Hypertonie präoperativ`

`neuerInhalt = arterielle Hypertonie präoperativ`

`bemerkung = NLS.B.AUFNHYPERTONIE[11.0]`

Das heißt, dass in der Zeile `idBogenfeld = 5972` der Tabelle `BogenFeld` das Attribut `bezeichnung` von `Hypertonie präoperativ` auf `arterielle Hypertonie präoperativ` geändert wurde. Die Spalte `bemerkung` enthält ggf. weitere Hinweise zum Kontext der Änderung.

Gelöschte Entitäten

Über die Attribute `id` und `fkTabellenStruktur` in der Tabelle `DeltaGeloescht` (siehe Tabelle 73) ist die Zuordnung zu den Entitäten der Datenbank möglich.

Tabelle 73: Struktur der Tabelle `DeltaGeloescht`

| Feldname | Datentyp | Bemerkung |
|---------------------------------|----------|---|
| <code>idDeltaGeloescht</code> | INTEGER | Primärschlüssel |
| <code>id</code> | TEXT | ID der Entität, die gelöscht wurde |
| <code>fkTabellenStruktur</code> | ZAHL | Bezug zum Attribut einer Tabelle, in der die Entität gelöscht wurde |
| <code>bemerkung</code> | MEMO | Ggf. Bemerkung zur Ergänzung |

Beispiel:

Die Zeile `idDeltaGeloescht = 3012` enthält in der Spezifikation 2015 V01 folgende Werte:

`Id = 6052`

`fkTabellenStruktur = <BogenFeld>`

`bemerkung = LLS.B.ERYTROZYTENKONZENTRAT[37.0]`

Das heißt, dass aus der Tabelle `BogenFeld` der Vorgängerdatenbank das `BogenFeld` mit der `id = 6052` gelöscht wurde. Unter `bemerkung` finden sich ggf. weitere Hinweise zum Kontext der gelöschten Entität.

Konfiguration der Delta-Berechnung

Es werden nur dann Delta-Informationen zu einer Entität angezeigt, wenn für mindestens ein Attribut der Entität in der Tabelle `TabellenFeldStruktur` das Attribut `deltaAktiv = WAHR` gesetzt ist.

Tabelle 74: Inhalt der Tabelle `TabellenFeldStruktur` (`fkTabellenFeldStruktur = Regeln`)

| idTabelle | feldName | ... | deltaAktiv | fk TabellenFeldStruktur |
|-----------|-------------------|-----|------------|-------------------------|
| 445 | idRegeln | | Nein | Regeln |
| 450 | fkMehrfachRegel | | Ja | Regeln |
| 451 | fkFeldGruppe | | Ja | Regel |
| 449 | fkRegelTyp | | Ja | Regeln |
| 446 | bedingung | | Ja | Regeln |
| 447 | meldung | | Ja | Regeln |
| 448 | fkModul | | Ja | Regeln |
| 454 | gueltigNachExport | | Ja | Regeln |

Beispiel:

Das Delta für die Tabelle `Regeln` wird für die Attribute `fkFeldGruppe`, `fkMehrfachRegel`, `fkRegelTyp`, `bedingung`, `meldung`, `fkModul` und `gueltigNachExport` erzeugt.

2.7.3 Abgrenzung zwischen Erfassungsjahren und Datensatzformaten

Die Datenannahmestellen müssen Datensätze von Leistungserbringern entgegennehmen, falls die Aufnahmedaten bei stationären Fällen bzw. die Eingriffsdaten ambulanter Fälle in den Gültigkeitszeitraum der Version einer finalen Datenbank fallen.

Das Abgrenzungskriterium zwischen den Erfassungsjahren¹²⁰ ist:

- das Datum der Aufnahme in das Krankenhaus (`AUFNDATUM`) bei Datensätzen stationär aufgenommenener Patienten
- das Erhebungsdatum des Follow-up (`FUERHEBDATUM`) bei Datensätzen ohne Datum der Aufnahme in das Krankenhaus: `HTXFU`, `LUTXFU`, `PNTXFU`, `LTXFU`, `NLSFU` und `LLSFU`

¹²⁰ Das Abgrenzungskriterium definiert somit die Zuordnung des Datensatzes zu einer Version der Spezifikation bzw. das Format des Datensatzes.

- das Datum des Eingriffs (OPDATUM) bei Datensätzen ambulanter Behandlungsfälle.



Achtung

Dem Erfassungsjahr 2019 zugeordnete Fälle müssen im Format der Spezifikation 2019 an die Datenannahmestellen gesandt werden, sonst ist die Datenlieferung zurückzuweisen. Im stationären Bereich müssen auch Datenlieferungen angenommen und verarbeitet werden, die gemäß der Spezifikation aus dem Vorjahr erhoben und dokumentiert wurden und gemäß Entlassungsdatum dem aktuellen Erfassungsjahr zuzuordnen sind.

Die entsprechenden Exportzeiträume sind den Attributen `Exportmodul.fkExportzeitraumEntLEJ` und `Exportmodul.fkExportzeitraumEntLEJ1` der Datenbank zur QS-Dokumentation zu entnehmen.

2.7.4 Version des Exportverfahrens

In der Spezifikationsdatenbank für die QS-Dokumentation wird in der Tabelle `Version` neben der gültigen Version (`gueltig`) auch das gültige Exportverfahren (`gueltigExportVerfahren`) angegeben. Dieses Attribut gibt an, welches Versionskürzel (Attribut `Version.name`) im XML-Dokument im Headerbereich unter `header/document/software/specification@V` verwendet werden muss.

Nur eine Version ist für das Exportverfahren als gültig markiert. Die Version des Exportverfahrens kann eine Vorversion der gültigen Version der Spezifikation sein.

2.8 Administrative Objekte

Die Datenbank für QS-Dokumentation beinhaltet neben den Dokumentationsobjekten eine Gruppe von administrativen Objekten, die Hilfestellungen für die Einhaltung eines korrekten Datenflusses und standardisierter Prüfprozesse geben.

Zu den administrativen Objekten gehören **Mapping-Informationen**, mit denen QS-Daten, die außerhalb des eigentlichen QS-Containers (dem Element `<qs-daten>`) im XML verortet werden können. Außerdem gibt es eine **Auflistung und Kategorisierung von Prüfschritten** zur Implementierung eines Datenservices (Abschnitt B 2.8.3). Um eine Ansicht der administrativen Objekte zu erhalten, ist in Access (Spezifikationsdatenbank zur QS-Dokumentation) oberhalb der Objektübersicht das Drop-Down-Menü zu öffnen und der Menüpunkt „Benutzerdefiniert“ auszuwählen.

In einer separaten Datenbank ist eine Übersicht über die für ein Modul innerhalb einer Region relevanten technischen **Datenservices mit Empfängeradressen** und zu verwendenden XML-Schlüsseln (Abschnitt B 2.8.2) enthalten.

Abbildung 20 und Abbildung 21 stellen die Beziehungen der administrativen Objekte zueinander dar.¹²¹

¹²¹ Die Tabellen zu Datenserviceinformationen sind in einer separaten Datenbank abgelegt

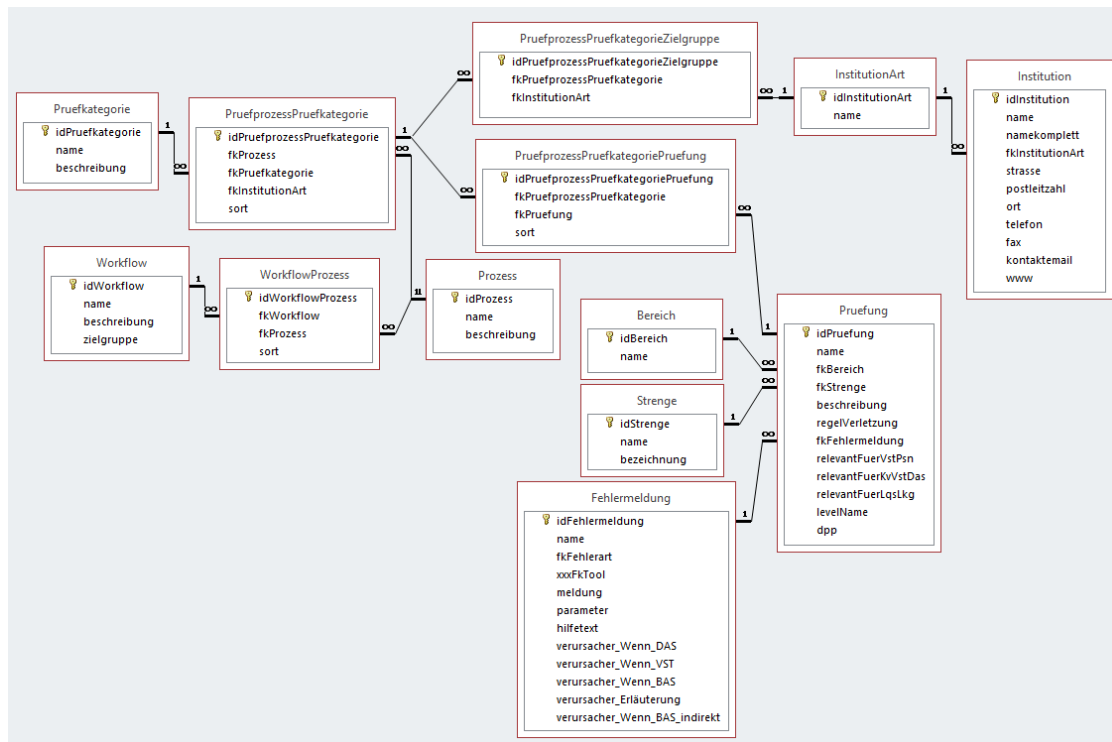


Abbildung 20: Beziehungen der administrativen Objekte (Prüfungen)

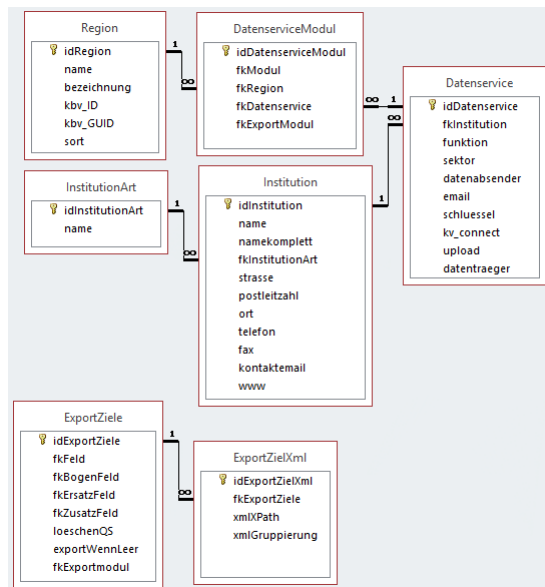


Abbildung 21: Beziehungen der administrativen Objekte (Datenservices, Mapping-Informationen)

2.8.1 CSV/XML-Mapping in der Spezifikationsdatenbank (QSDOK)

Es gibt eine Reihe von XML-Elementen, die zurzeit nicht in der Access-DB hinterlegt sind. Insgesamt gibt es vier Kategorien von Abweichungen:

- Neu eingeführte XML-Elemente
- Header-Informationen, wie die GUID des Dokuments
- Umbenannte und aus dem QS-Datensatz ausgelagerte CSV-Felder wie beispielsweise kasseiknr2Stellen

- Gelöschte Felder wie beispielsweise das Feld `IKNRKH`

In Abstimmung mit den Softwareanbietern werden die CSV-Felder nicht an die XML-Struktur angepasst. Stattdessen wird für betroffene Felder eine Mapping-Tabelle in der Access-DB hinterlegt.

Die durch Bögen (Teildatensätze) für das Element `<qs_data>` festgelegte Struktur bleibt durch die aufgeführten Mappings unangetastet. Die Mappings erlauben alleine eine Festlegung darüber, ob und falls ja wo ein betroffenes Datum außerhalb des `<qs_data>` Elements in das XML geschrieben wird und ob das Datum für das Element `<qs_data>` gelöscht werden muss. Ggf. nötige Umbenennungen außerhalb von `<qs_data>` werden implizit durch die Ortsangabe vorgenommen.

Die für das Mapping geführten Tabellen heißen `ExportZiele` und `ExportZielXml`. In der Abfrage `vExportZieleXml` sind diese Tabellen zu einer Übersicht zusammengefasst. Hierbei ist zu beachten, dass nicht jedem Eintrag in der Tabelle `ExportZiele` auch ein Eintrag in der Tabelle `ExportZielXML` zugewiesen sein muss. Soll beispielsweise ein Datum aus dem Element `<qs_data>` lediglich gelöscht werden, ohne jedoch an andere Stelle im XML verschoben zu werden, ist keine dementsprechende `xmlXPath`-Angabe erforderlich.

Beispiel:

Für das in der Tabelle `ExportZiele` aufgeführte, modulspezifische Bogenfeld `PCI:B:IKNRKH` (`idExportZiele = 1` referenziert mittels `fkBogenFeld`) soll für das Exportmodul `PCI_KV` verhindert werden, dass dieses, sofern vorhanden, analog zur allgemein gültigen Regelung für das Feld `IKNRKH` (`idExportZiele = 19` referenziert mittels `fkFeld`) in das Element `<care_provider>` verschoben wird. Daher gibt es in der Tabelle `ExportZielXML` keinen Eintrag, welcher als Fremdschlüssel das Feld `PCI:B:IKNRKH` (also `fkExportZiele = 19`) aufweist.

Das Feld `PCI:B:IKNRKH` wird für das Exportmodul `PCI_KV` aus den QS-Daten gelöscht und nicht in das Element `<care_provider>` verschoben, sodass die Angabe nicht exportiert oder übermittelt wird.

Es können alle Feldarten referenziert sein. Referenziert sind aber nur solche Felder, die nicht oder nicht nur im Element `<qs_data>` aufgeführt werden. Die Felder werden referenziert und über diese Tabelle mit zusätzlichen Informationen in Bezug auf das XML verknüpft.

Die über das Attribut `fkFeld` referenzierten Felder gelten für alle Module, in denen dieses Feld verwendet wird. Wird ein Feld (gleichzeitig) über unterschiedliche Feldarten referenziert, überlagern detailliertere Angaben die allgemeinen. Wird so beispielsweise in der Tabelle `ExportZiele` ein Feld allgemein über den Fremdschlüssel `fkFeld` und in einer weiteren Zeile modulspezifisch über `fkBogenFeld` referenziert, werden die allgemeinen Regelungen (`loeschenQS`, `exportWennLeer`, `xmlXPath` und `xmlGruppierung`) für das Feld mittels der Referenzierung über `fkFeld` mit den spezifischeren Regelungen für das

modulbezogene Feld mittels der Referenzierung über `fkBogenFeld` für das konkret aus der Referenzierung hervorgehende Modul überschrieben.

Dabei ist ebenfalls eine etwaige Konkretisierung über `fkExportModul` zu beachten, welche Ihrerseits konkretisierenden Charakter hat. Die über das Attribut `fkErsatzFeld` referenzierten Ersatzfelder gelten für alle Exportmodule, in denen dieses Ersatzfeld verwendet wird. Bei Referenzierung des gleichen Feldes (insbesondere bezogen auf `fkFeld`, `fkErsatzFeld` bzw. `fkZusatzfeld`) in zwei Zeilen – einmal ohne und einmal mit Angabe von `fkExportModul` – so überschreibt der Eintrag mit Angabe den Eintrag ohne Angabe für das konkret benannte Exportmodul.

Beispiel:

Für das Feld `IKNRKH` gibt es zwei Referenzierungen. Zunächst gelten modulunabhängig die Regelungen für die Referenzierung über die Feldart `fkFeld` (`idExportZiele = 1` referenziert mittels `fkFeld = IKNRKH`). Die zweite Referenzierung auf dieses Feld, modulspezifisch festgelegt über die Feldart `fkBogenFeld` (`idExportZiele = 19` referenziert mittels `fkBogenFeld = PCI:B:IKNRKH`) überschreibt die allgemeinen Regelungen konkret für das Modul `PCI`. Darüber hinaus führt die Konkretisierung über `fkExportModul = PCI_KV` dazu, dass die weiterführenden Regelungen lediglich im Rahmen des Datenexports für den Datenfluss über die KV als Datenannahmestelle greifen. Für die Exportmodule `PCI_LGK` sowie `PCI_SV` gelten weiterhin die Regelungen mit der Referenz über die Feldart `fkFeld`.

| idExportZieleXml | fkFeld | fkBogenFeld | fkErsatzFeld | fkZusatzfeld | loeschenQS | exportWennLeer | xmlXPath | xmlGruppierung | fkExportModul |
|------------------|----------|------------------|-------------------|--------------|-------------------------------------|-------------------------------------|---|-------------------------------------|---------------|
| 2 | IKNRKH | | | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | ancestor::data_container/care_provider/IKNR/@V | <input checked="" type="checkbox"/> | |
| 3 | | | | Modul | <input type="checkbox"/> | <input type="checkbox"/> | ancestor::cases/@module | <input checked="" type="checkbox"/> | |
| 4 | | | | Modul | <input type="checkbox"/> | <input type="checkbox"/> | case_admin/module/@V | <input type="checkbox"/> | |
| 5 | | | | Modul | <input type="checkbox"/> | <input type="checkbox"/> | qs_data/@module | <input type="checkbox"/> | |
| 7 | | | versichertenidgv | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | patient/pid/VERSICHERTENIDNEU/@V | <input type="checkbox"/> | |
| 8 | | | kasseiknr2stellen | | <input type="checkbox"/> | <input type="checkbox"/> | patient/@twodigitik | <input type="checkbox"/> | |
| 10 | BSNRAMBU | | | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | ancestor::data_container/care_provider/BSNRAMBU | <input checked="" type="checkbox"/> | |
| 11 | NBSNRAMB | | | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | ancestor::data_container/care_provider/NBSNRAMB | <input checked="" type="checkbox"/> | |
| 12 | LANR | | | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | ancestor::data_container/care_provider/LANR/@V | <input checked="" type="checkbox"/> | |
| 13 | | | | Registriernr | <input checked="" type="checkbox"/> | <input type="checkbox"/> | ancestor::root/header/provider/@registration | <input type="checkbox"/> | |
| 14 | | | | VersionNr | <input checked="" type="checkbox"/> | <input type="checkbox"/> | case_admin/version/@V | <input type="checkbox"/> | |
| 15 | | | | Vorgangsnr | <input checked="" type="checkbox"/> | <input type="checkbox"/> | case_admin/id/@V | <input type="checkbox"/> | |
| 16 | | | | Storno | <input checked="" type="checkbox"/> | <input type="checkbox"/> | case_admin/action/@V="delete" | <input type="checkbox"/> | |
| 17 | | opquartal | | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | case_admin/quarter/@V | <input type="checkbox"/> | PCI_KV |
| 18 | | vornameMutter | | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | patient/perineo_pid/vorname_mutter/@V | <input type="checkbox"/> | |
| 19 | | nachnameMutter | | | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | patient/perineo_pid/nachname_mutter/@V | <input type="checkbox"/> | |
| 22 | | gebdatumNeo | | | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | patient/perineo_pid/GEBDATUMK/@V | <input type="checkbox"/> | |
| 23 | | gebdatumGeb | | | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | patient/perineo_pid/GEBDATUMK/@V | <input type="checkbox"/> | |
| 24 | | versichertenidgv | | | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | patient/perineo_pid/VERSICHERTENIDNEUK/@V | <input type="checkbox"/> | NEO |

Abbildung 22: Ergebnis der Abfrage `vExportZieleXml`

Die booleschen Spalten `loeschenQS` und `exportWennLeer` geben folgende Informationen an:

- `loeschenQS`: Das Datum wird nur an den/die alternativen Ort(e) geschrieben und taucht in den QS-Daten nicht mehr auf. Hier geht es zum Beispiel um vom Datenschutz betroffene Felder, die nur in bestimmte Bereiche des XML geschrieben werden dürfen, wo sie dann verschlüsselt werden können.
- `exportWennLeer`: Hier wird ein leeres Feld nur dann berücksichtigt und leer eingetragen, wenn dieser Wert auf `true` steht.

In der Spalte `xmlXPath` vom Typ String steht ein XPath-Ausdruck, der ein Mapping auf einen im XML liegenden Ort des betroffenen Feldes enthält. Ausgangspunkt des XPath-Ausdrucks ist immer der aktuelle Datensatz im XML, also das `<case>` Element.

Die boolesche Spalte `xmlGruppierung` gibt an, dass ein neuer Knoten nur eingesetzt wird, wenn das Datum einen neuen Wert enthalten sollte.

| loeschenQS | exportWennLeer | fkExportZiel | xmlXPath | xmlGruppierung |
|-------------------------------------|--------------------------|--------------|---|-------------------------------------|
| <input checked="" type="checkbox"/> | <input type="checkbox"/> | | 1 /root/body/data_container/care_provider/IKNR/@V | <input checked="" type="checkbox"/> |
| <input type="checkbox"/> | <input type="checkbox"/> | | 2 ancestor::cases/@module | <input checked="" type="checkbox"/> |
| <input type="checkbox"/> | <input type="checkbox"/> | | 2 case_admin/module/@V | <input type="checkbox"/> |
| <input type="checkbox"/> | <input type="checkbox"/> | | 2 qs_data/@module | <input type="checkbox"/> |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> | | 4 patient/pid/VERSICHERTENIDNEU/@V | <input type="checkbox"/> |

Abbildung 23: Beispiel für XPath-Ausdrücke in der Tabelle `ExportZieleXml` in Verbindung mit weiteren Informationen

2.8.2 Datenservices

In einer separaten Datenbank sind Angaben über die beim Export relevanten technischen Datenservices und ihre verfahrensbezogene und regionale Zuordnung zu finden. Diese Datenbank zu Datenserviceinformationen ist auf der Homepage <https://www.iqtig.org/> zu finden.

Die einzelnen Datenservices werden zusammen mit den E-Mail-Adressen¹²², an die die Exportdateien zu versenden sind, und den für die Verschlüsselung der QS-Daten zu verwendenden XML-Schlüsseln in der Tabelle `Datenservice` abgebildet:

| fkInstitution | Funktion | Sektor | Datenabsender | Email | Schlüssel |
|---------------|----------|-------------|---------------|--------------------------|--|
| Geqik | DAS | Krankenhaus | LE | daten@geqik.de | https://www.sqg.de/downloads/2013/xml/schluesel/Pub_key_Datenannahmesstelle_BW_LQS.pub |
| BAQ | DAS | Krankenhaus | LE | daten@baq-bayern.de | https://www.sqg.de/downloads/2013/xml/schluesel/Pub_key_Datenannahmesstelle_BA_LQS.pub |
| GQH | DAS | Krankenhaus | LE | xmldaten@gqhnet.de | https://www.sqg.de/downloads/2013/xml/schluesel/Pub_key_Datenannahmesstelle_HE_LQS.pub |
| PGS-QS | DAS | Krankenhaus | LE | qs-daten@nkgev.de | https://www.sqg.de/downloads/2013/xml/schluesel/Pub_key_Datenannahmesstelle_NI_LQS.pub |
| SLAEK | DAS | Krankenhaus | LE | qs-daten@slaek.de | https://www.sqg.de/downloads/2013/xml/schluesel/Pub_key_Datenannahmesstelle_SN_LQS.pub |
| KGSH | DAS | Krankenhaus | LE | proqs@kgsh.de | https://www.sqg.de/downloads/2013/xml/schluesel/Pub_key_Datenannahmesstelle_SH_LQS.pub |
| BQS-Institut | DAS | Krankenhaus | LE | xmldaten@bqs-institut.de | https://www.sqg.de/downloads/2013/xml/schluesel/Pub_key_Datenannahmesstelle_DL_BQS.pub |
| AQUA-Institut | BAS | | VST | | https://www.sqg.de/downloads/2013/xml/schluesel/Pub_key_Bundesauswertungsstelle.pub |
| SCI | VST | | DAS | | https://www.sqg.de/downloads/2013/xml/schluesel/Pub_key_Vertrauensstelle.pub |

Abbildung 24: Tabelle `Datenservice`

Die einzelnen Datenannahmestellen, die ihre Datenannahme mithilfe eines oder mehrerer dieser Datenservices realisieren, sind in der Tabelle `Region` hinterlegt, während die Zuordnung eines Datenservices zu einer Region abhängig vom Modul in der Tabelle `DatenserviceModul` festgelegt ist:

¹²² Die KVen und die DAS-SV erhalten die Daten in der Regel auf anderen Übertragungswegen.

| Region | | | | |
|--------|----------|------|------------------------|--------|
| | idRegion | name | bezeichnung | kbv_ID |
| + | 73 | BW | Baden-Württemberg | |
| + | 74 | BA | Bayern | |
| + | 75 | BE | Berlin | |
| + | 76 | BB | Brandenburg | |
| + | 77 | HB | Bremen | |
| + | 78 | HH | Hamburg | |
| + | 79 | HE | Hessen | |
| + | 80 | MV | Mecklenburg-Vorpommern | |
| - | 81 | NI | Niedersachsen | |

| | idDat | fkExportModul | fkModul | fkDatenservice |
|--|-------|---------------|---------|----------------------|
| | 2863 | PNEU | PNEU | PGS-QS/DAS |
| | 2735 | PCI_LKG | PCI | PGS-QS/DAS |
| | 3196 | PCI_KV | PCI | KV-Niedersachsen/DAS |
| | 3119 | NEO | NEO | PGS-QS/DAS |
| | 3167 | KEP | KEP | PGS-QS/DAS |
| | 3151 | HEP | HEP | PGS-QS/DAS |

Abbildung 25: Verfahrensbezogene Datenservices

2.8.3 Prüfschritte

Innerhalb der administrativen Objekte der QSDOK-Datenbank befinden sich Tabellen zum Beschreiben von Prüfschritten, die Institutionen als Vorlage für die Implementierung eines Datenservices nutzen können. Diese reichen von der Prüfung einer korrekten Übermittlung der Exportdatei bis hin zur Schemaprüfung der durch die Institution modifizierten XML-Datei.

Zurzeit beschränken sich diese Prüfungen auf die QS-Dokumentation für Datenannahmestellen. Zukünftig können damit weitere Workflows abgebildet werden, wie der folgende Inhalt der Tabelle Workflow zeigt:

Tabelle 75: Überblick über neben der QS-Dokumentation weitere potenzielle Workflows mit definierten Prüfschritten

| Name | Beschreibung | Zielgruppe | In derzeitiger Spezifikation enthalten |
|-----------------------|------------------------------|------------|--|
| QS-Dokumentation | Prüfung der QS-Dokumentation | DAS | Ja |
| QS-Sollstatistik | Prüfung der QS-Statistik | DAS | Nein |
| PID-Pseudonymisierung | Pseudonymisierung der PID | VST | Nein |

Für jede Prüfung (Tabelle `Pruefung`) existiert eine Fehlermeldung (Tabelle `Fehlermeldung`), die als Validierungsergebnis ausgegeben werden muss, wenn diese Prüfung fehlschlägt.

Um Prüfungen chronologisch, d.h. von der Ankunft einer Transportdatei bis zum Abschicken an die Vertrauensstelle zu gruppieren, sind die Prüfungen Kategorien (Tabelle `PruefprozessPruefkategorie`) zugeordnet und die Kategorien wiederum sogenannten Workflowprozessen (Tabelle `Prozess`). Über die Tabelle `PruefprozessPruefkategorie` werden die Kategorien den Workflowprozessen zugeordnet, während über die Tabelle `PruefprozessPruefkategoriePruefung` einzelne Prüfungen den Kategorien innerhalb der Workflowprozesse zugeordnet werden. Innerhalb dieser Zuordnungstabellen gibt die Spalte `sort` die oben genannte chronologische Reihenfolge wieder. Da nicht alle Prüfungen auf alle Arten von Datenannahmestellen zutreffen, ordnet die Tabelle `PruefprozessPruefkategorieZielgruppe` einzelne Kategorien unterschiedlichen Arten von Datenannahmestellen zu.

Eine aus diesen Daten generierte HTML-Ansicht, abrufbar unter <https://iqtig.org/datenerfassung/datenannahmestellen/pruefschritte-qs-dokumentation/>, ist dem folgenden Screenshot zu entnehmen.

nahmestelle: LQS / LKG KV VST-DAS

| | | | | |
|-------------------|----------------------------|---------------|-------------------|------------------------------|
| Paket-Prüfungen | Entschlüsselung des Pakets | Prüfung | Datei-Prüfungen | Leistungserbringer-Prüfungen |
| Formale Prüfungen | Formale Prüfungen | Schemaprüfung | Formale Prüfungen | Formale Prüfungen |

Prüfdetails

Prüfung: Ist eine bereits übermittelte Exportdatei erneut mit derselben Dokumenten-ID (GUID) übermittelt word (Die GUID muss für mindestens für ein Erfassungsj gespeichert werden.)

Beschreibung: -

Bereich: Transaktion

Fehlerart: ERROR

Regelverletzung: -

Id der Fehlermeldung: 1002004

Fehlermeldung: Die übermittelte GUID (<guid>) wurde bereits verwendet.

Parameter: <params guid="Eindeutige Identifikation" />

Verursacher wenn DAS: Leistungserbringer

Abbildung 26: HTML-Ansicht der Prüfschritte innerhalb der QS-Dokumentation

Ein schneller Überblick über alle Prüfungen ist der Abfrage `vPruefung` zu entnehmen, deren Felder folgend kurz erläutert werden:

Tabelle 76: Felder der Abfrage vPruefung

| Feldname | Bedeutung | Bezugselement im XML |
|-------------------------------|---|---|
| Prüfung | Kurzbeschreibung der Prüfung | - |
| Beschreibung | ggf. erweiterte Erläuterungen zur Prüfung | - |
| Id-Fehlermeldung | Fehlerreferenz | validation_item/error/rule_id[@V] |
| Bereich | Einordnung der Prüfung | validation_item[@V] |
| Strenge | Konsequenz im Fehlerfall (Hinweis, Fehler) | validation_item/error/rule_type[@V] |
| Fehlerart | Legacy -> Bezug zu CSV | validation_item/error/error_type[@V] |
| Regelverletzung | Bedingungen, die zu einer Regelverletzung führen | - |
| Fehlermeldung | Standardisierte Fehlermeldung | validation_item/error/error_message[@V] |
| Parameter | Parameter für Fehlermeldung | - |
| Hilfetext | ggf. weitere Hinweise zum Fehler | - |
| Verursacher_Wenn_DAS | Verursacher wenn Regelverstoß in DAS festgestellt | validation_item/error/originator[@V] |
| Verursacher_Wenn_VST | Verursacher wenn Regelverstoß in VST festgestellt | validation_item/error/originator[@V] |
| Verursacher_Wenn_BAS | Verursacher wenn Regelverstoß in BAS festgestellt | validation_item/error/originator[@V] |
| Verursacher_Wenn_BAS_indirekt | Verursacher wenn Regelverstoß in BAS festgestellt | validation_item/error/originator[@V] |
| Verursacher_Erläuterung | ggf. Erläuterung zum Verursacher eines Fehlers | - |

| Feldname | Bedeutung | Bezugselement im XML |
|--|---|----------------------|
| Relevant_Für_VST-PSN ¹²³ | M=obligatorische Prüfung, K= freiwillige Prüfung | - |
| Relevant_Für_KV _VST-DAS ¹²³ | M=obligatorische Prüfung, K= freiwillige Prüfung | - |
| Relevant_Für_LQS_LKG ¹²³ | M=obligatorische Prüfung, K= freiwillige Prüfung | - |
| Protokoll-Level | Zuordnung der Prüfung: DK-> betrifft Dokument DS-> betrifft Datensatz | - |

¹²³ Gibt an, ob Prüfungen bei VST, DAS ambulant oder DAS stationär vorgeschrieben (M) sind oder freiwillig (K) durchgeführt werden können.

3 Risikostatistik

Die technische Umsetzung der Risikostatistik ist weitgehend an die bekannten Strukturen und Vorgehensweisen im Rahmen der QS-Filter-Software angelehnt. Es handelt sich jedoch um einen separaten Prozess.

3.1 Allgemeine Anmerkungen über die Struktur der Spezifikationsdatenbank zur Risikostatistik

Die Spezifikation zur Risikostatistik ist in einer relationalen Datenbank abgelegt. Sie wird als eigenständige Access-Datenbank (MS Access 2007-2013) zur Verfügung gestellt. Der Name der Risikostatistik-Spezifikation richtet sich nach folgendem Schema:

`<Erfassungsjahr>_Risikostatistik_V<Versionsnummer>.mdb`

`<Versionsnummer>` bezeichnet die 2-stellige Versionsnummer (z. B. 01).

3.2 Tabellenstruktur der Datenbank

Die Tabellenstruktur der Datenbank zur Risikostatistik ist an die Tabellenstruktur der Spezifikationsdatenbank für QS-Filter angelehnt. Diese wird in Abschnitt B 1.2.2 beschrieben.

Folgende Attribute treten in den Tabellen der Datenbank zur Risikostatistik auf und sind hier kurz erläutert:

- `name` ist in der Regel als „technischer Name“ zu verstehen.
- `bezeichnung` ist eine kurze Beschreibung.

3.3 Struktur der Datensatzdefinitionen

In den nachfolgenden Abschnitten dieses Unterkapitels wird die grundlegende Struktur der Tabellen der Datensatzdefinitionen beschrieben.

Jeder Datensatz besteht aus Teildatensätzen (Abschnitt B 3.3.2), die, ausgehend von einem Basis-teildatensatz, hierarchisch angeordnet sind. Die Definitionen der Datenfelder werden in den Abschnitten B 3.3.3 bis B 3.3.7 erläutert.

3.3.1 Datensätze

In der Spezifikation zur Risikostatistik sind zwei Datensätze definiert:

- Risikostatistik-Eingangsdatsatz (Abschnitt B 3.7)
Dieser beschreibt die Datengrundlage zur Erstellung der Risikostatistik.
- Datensatz zur Risikostatistik (Abschnitt B 3.8)
Für jeden dieser Datensätze wird in der Tabelle `Ds` ein Eintrag angelegt.

Tabelle 77: Struktur der Tabelle *Ds*

| Feldname | Datentyp | Bemerkung |
|-------------|----------|---|
| idDs | INTEGER | Primärschlüssel |
| name | TEXT | Technischer Name des Datensatzes (identifizierendes Attribut) |
| bezeichnung | TEXT | Beschreibender Text |

3.3.2 Teildatensätze

Die Definition von Teildatensätzen findet sich in der Tabelle *Tds* der Datenbank. Jeder Teildatensatz ist eindeutig durch seinen Namen (z. B. *FALL* oder *DIAG*) charakterisiert.

Tabelle 78: Struktur der Tabelle *Tds*

| Feldname | Datentyp | Bemerkung |
|---------------|----------|---|
| idTds | INTEGER | Primärschlüssel |
| name | TEXT | Technischer Name des Teildatensatzes (identifizierendes Attribut) |
| bezeichnung | TEXT | Beschreibender Text |
| fkTds | INTEGER | Optionaler Fremdschlüssel zu einem Mutter-Teildatensatz |
| fkDs | INTEGER | Bezug des Teildatensatzes zum übergeordneten Datensatz in der Tabelle <i>Ds</i> (Risikostatistik-Eingangsdatsatz oder Datensatz für Risikostatistik) |
| fkRelationTyp | TEXT | Relationstyp, bezieht sich auf die Relation zum Mutter-Teildatensatz: * Eine beliebige Anzahl von Teildatensätzen darf angelegt werden! ? Höchstens ein Teildatensatz darf angelegt werden! + Mindestens ein Teildatensatz muss angelegt werden! 1 Genau ein Teildatensatz muss angelegt werden! |

In der Tabelle *Tds* ist eine Hierarchie der Teildatensätze definiert. Der Ausgangspunkt („root“) für die Teildatensatzhierarchie eines Datensatzes (z. B. Eingangsdatsatz) ist immer der Basis-Teildatensatz (= Teildatensatz¹²⁴ *fkTds* = NULL in der Tabelle *Tds*). Über die in den restlichen Teildatensätzen definierten Bezüge zu den Mutter-Teildatensätzen und Relationstypen lässt sich ein „Hierarchiebaum“ der Teildatensätze (= Datensatz) aufbauen.

¹²⁴ Es darf nur eine Definition eines Basis-Teildatensatzes existieren.

Jeder Datensatz besteht aus

- genau einem Basis-Teildatensatz und
- ggf. einem weiteren Teildatensatz oder mehreren weiteren Teildatensätzen (= Kind-Teildatensätze).

Beispiele:

- Der Risikostatistik-Eingangsdatensatz besitzt den Basis-Teildatensatz `FALL` und die Kind-Teildatensätze `DIAG`, `PROZ` und `ABTLG`.
 - Der Datensatz Risikostatistik besitzt den Basis-Teildatensatz `RISIKOBASIS` und den Kind-Teildatensatz `RISIKOSTATISTIK`.
-

3.3.3 Felder der Teildatensätze

Die Felder eines Teildatensatzes sind in der Tabelle `TdsField` definiert. Jedes Feld eines Teildatensatzes (kurz TDS-Feld) ist eindeutig charakterisiert durch die Zugehörigkeit zu einem Teildatensatz und zum referenzierten Feld. Jedes Feld darf also nur einmal in einem Teildatensatz verwendet werden.

3.3.4 Felder

Ein Feld wird eindeutig über seinen technischen Namen definiert. Jedes Feld hat einen Basistyp (z. B. `SCHLUESSEL`, `NUMSCHLUESSEL`, `ZAHL`). Schlüsselfelder erfordern zusätzlich einen Schlüssel (Abschnitt B 3.3.6).

3.3.5 Basistypen

Das Hauptmerkmal eines Basistyps ist der technische Typ eines Eingabefeldes (z. B. Zeichenkette, numerischer Typ, Datum usw.). Weiteres Charakteristikum ist die Beschreibung des Eingabeformats. Die Basistypen sind Voraussetzung für die Beschreibung einer formalen Regelsyntax. Das identifizierende Merkmal eines Basistyps ist sein technischer Name (Attribut `name`).

Tabelle 79: Struktur der Tabelle `BasiSTyp`

| Feldname | Datentyp | Bemerkung |
|------------------------------|----------|--|
| <code>idBasiSTyp</code> | INTEGER | Primärschlüssel |
| <code>name</code> | TEXT | Technischer Name (muss eindeutig sein) |
| <code>bezeichnung</code> | TEXT | Beschreibender Text |
| <code>formatAnweisung</code> | TEXT | Regulärer Ausdruck für die Formatprüfung |

**Hinweis**

- In Zeichenketten (Basistyp `TEXT`) sind alle Zeichen des ASCII-Formats mit einem Kode ≥ 32 erlaubt. Ausgenommen sind das Semikolon, die doppelten Anführungsstriche und Hochkommata.
- Es gibt zwei Arten von Schlüsseln: numerische und nichtnumerische.
- Das Komma trennt die Nachkommastellen, Vorzeichen `+` und `-` sind erlaubt.
- Das Datumstrennzeichen ist der Punkt.

3.3.6 Schlüssel

Identifizierendes Merkmal eines Schlüssels ist sein technischer Name. Die meisten Schlüsselkodes sind in der Tabelle `Schluesse1Wert` (Abschnitt B 3.3.7) definiert. Externe Schlüsselkataloge (z. B. ICD10) sind von den entsprechenden Anbietern zu beziehen.

Tabelle 80: Struktur der Tabelle `Schluesse1`

| Feldname | Datentyp | Bemerkung |
|----------------------------|----------|---|
| <code>idSchluesse1</code> | INTEGER | Primärschlüssel |
| <code>name</code> | TEXT | Technischer Name (muss eindeutig sein) |
| <code>bezeichnung</code> | TEXT | Beschreibender Text |
| <code>extern</code> | BOOL | Zeigt an, ob der Schlüssel in der Tabelle <code>Schluesse1</code> oder in einer externen Tabelle gespeichert ist. |
| <code>externVerweis</code> | TEXT | Verweis auf eine externe Quelle des Schlüsselkataloges |
| <code>zahl</code> | BOOL | Wenn WAHR, wird das Attribut <code>code</code> der zugehörigen Schlüsselwerte als ganze Zahl interpretiert, ansonsten als Zeichenkette. |

Schlüsselkodes können auf zwei Arten kodiert werden. Wenn das Attribut `zahl` gesetzt ist, werden die Codes als ganze Zahl interpretiert. Ansonsten werden sie als Zeichenketten angesehen.

Beispiel:

Attribut `zahl` bei Schlüsselfeldern

Felder des Basistyps `NUMSCHLUESSEL` haben das Attribut `zahl = WAHR`.

Felder des Basistyps `SCHLUESSEL` haben das Attribut `zahl = FALSCH`. Es handelt sich um alphanumerische Schlüssel, die Buchstaben, Ziffern oder Sonderzeichen verwenden (z. B. `ypN0`). Hierbei kann es sich auch um Werte handeln, die lediglich Ziffern verwenden, aber mit führender Null beginnen (z. B. `01`).

Externe Schlüsselkataloge

Externe Schlüsselkataloge sind über das Attribut `extern` deklariert. Externe Schlüsselkataloge werden nicht vom IQTIG bereitgestellt und somit auch nicht verantwortet.



Achtung

Der Softwareanbieter oder die Datenannahmestelle hat dafür Sorge zu tragen, dass die aktuellen externen Schlüsselkataloge in der Software verwendet werden.

Hinweise zu den Bezugsquellen sind in der Spalte `externVerweis` zu finden (z. B. <http://www.dimdi.de>). Ein Verweis auf eine Bezugsquelle kann unabhängig vom Attribut `extern` angegeben werden (siehe Schlüssel `EntlGrund`).

3.3.7 Schlüsselwerte

Identifizierendes Merkmal ist hier eine Kombination der Spalten `fkSchluessel` und `code`. Das bedeutet, dass jeder Schlüsselcode innerhalb eines Schlüssels nur einmal vorkommen darf.

Tabelle 81: Struktur der Tabelle `SchluesseWert`

| Feldname | Datentyp | Bemerkung |
|-------------------------------|----------|--|
| <code>idSchluesselWert</code> | INTEGER | Primärschlüssel |
| <code>fkSchluessel</code> | INTEGER | Fremdschlüssel zur Tabelle <code>Schlüssel</code> |
| <code>code</code> | TEXT | Schlüsselcode (entweder numerisch oder alphanumerisch kodiert) |
| <code>bezeichnung</code> | TEXT | Textliche Definition des Schlüsselwertes |

3.4 ICD-Listen und OPS-Listen

Die Struktur der ICD-Listen und OPS-Listen (Tabellen `ICDListe` bzw. `OPSListe` und `ICDWert` bzw. `OPSWert`) entspricht den gleichnamigen Tabellen der QS-Spezifikationen für QS-Dokumentationssoftware. Diese sind im Abschnitt B 2.5.1 erläutert.

3.5 Delta-Informationen zur vorhergehenden Version

Um den Nutzern der Risikostatistik umfassende Informationen zu den jeweiligen Änderungen zur Verfügung zu stellen, enthält die Risikostatistik-Spezifikation Tabellen, die den Änderungsstand der Datenbank im Vergleich zur letzten gültigen Version des Vorjahres anzeigen.

Es gibt drei Delta-Tabellen:

- `DeltaNeu`: neue Entitäten
- `DeltaGeloescht`: gelöschte Entitäten
- `DeltaAttribut`: geänderte Entitäten

Es werden nur dann Delta-Informationen zu einer Entität angezeigt, wenn für mindestens ein Attribut der Entität in der Tabelle `TabellenFeldStruktur` das Attribut `deltaAktiv` = WAHR gesetzt ist.

Die Delta-Tabellen entsprechen den gleichnamigen Tabellen der QS-Spezifikationen für QS-Dokumentationssoftware und werden in Abschnitt B 2.7.2 erläutert.

3.6 Versionierung

In der Tabelle `Version` finden sich Informationen zur Version der Spezifikationsdatenbank. Die wichtigsten Eigenschaften einer Version sind der Versionsname (Attribut `name`) und die Gültigkeitszeiträume (Attribute `ab` und `bis`). Der Gültigkeitszeitraum einer Version ist in der Regel ein Erfassungsjahr (z. B. Entlassung im Jahr 2019), wobei die Aufnahme auch bereits im vorherigen Jahr liegen kann (z. B. Aufnahme zwischen dem 1. Januar 2018 und dem 31. Dezember 2019).

Versionen können seit der Risikostatistik 2015 V01 den Status „in Entwicklung“, „final“ oder „Update der finalen Spezifikation“ haben¹²⁵. Diese Zustände werden in der Nachschlagetabelle `VersStatus` verwaltet. Das Attribut `gueltig` zeigt die gültige Version der Datenbank an. Es darf nur eine einzige Version als gültig markiert sein.

Die Tabelle `Version` enthält seit der Risikostatistik 2015 V01 auch einen Selbstbezug (Attribut `fkVersion`), der die Identifizierung der Vorgängerversion ermöglicht.

Beispiel:

Die Vorgängerversion der Spezifikation 2019 V01 ist die Version 2018 V01.

3.7 Der Risikostatistik-Eingangsdatensatz

Die Algorithmen zur Berechnung der Risikostatistik basieren auf administrativen und medizinischen Datenfeldern. Für die Erstellung der Risikostatistik sollen die im KIS befindlichen Routinedaten des Krankenhauses herangezogen werden. Die für die Risikostatistik benötigten Datenfelder finden sich im Aufnahmesatz und in der Entlassungsanzeige des §301-Datenformats. Fast alle benötigten Datenfelder können aus der Struktur des §301-Datensatzes abgeleitet werden. Einzige Ausnahme ist das Feld `PATALTER`.

Der Risikostatistik-Eingangsdatensatz besteht aus drei Teildatensätzen:

- Behandlungsfall (Teildatensatz `FALL`)
- Diagnoseangaben des Behandlungsfalls (Teildatensatz `DIAG`, wiederholbar)
- Prozedurangaben des Behandlungsfalls (Teildatensatz `PROZ`, wiederholbar)

Tabelle 82: Felder des Risikostatistik-Eingangsdatensatzes

| Feld | Beschreibung | M/K | Feldlänge | Basistyp |
|--------------------------------|--|-----|-----------|----------|
| Behandlungsfall (FALL): | | | | |
| FALLNUMMER | KH-interne Fallnummer Die Fallnummer dient der eindeutigen Identifizierung eines Kranken- | M | 15 | TEXT |

¹²⁵ Eine Ausnahme ist der Status „Service Release zur finalen Spezifikation“. Dieser ist lediglich für die Versionen RS 2013 SR3 und RS 2013 SR4 gültig.

| Feld | Beschreibung | M/K | Feldlänge | Basistyp |
|--|---|-----|-----------|-------------------------------|
| | hausfalls. Sie wird vom Krankenhaus vergeben. Im Teildatensatz FALL ist sie ein Primärschlüssel, in den anderen Teildatensätzen ist sie als Fremdschlüssel einzufügen. | | | |
| AUFNDATUM | Datum Aufnahme in Krankenhaus | M | 10 | DATUM (TT.MM.JJJJ) |
| AUFNZEIT | Aufnahmeuhrzeit | K | 5 | UHRZEIT (hh:mm) |
| ENTLDATEM | Datum Entlassung aus Krankenhaus Es gelten die Daten, mit denen die Verweildauer der DRG berechnet wird. | M | 10 | DATUM (TT.MM.JJJJ) |
| PATALTER | Alter in Jahren am Aufnahmetag | M | 3 | GANZEZAHL |
| DAUBEAT | Beatmungsstunden (vgl. Entlassungsanzeige §301) | K | 4 | GANZEZAHL |
| GESCHLECHT | Geschlecht | M | 1 | NUMSCHLUESSEL (Geschlecht) |
| AUFNGRUND | Aufnahmegrund §301–Aufnahmesatz (1. und 2. Stelle): 01 – Krankenhausbehandlung, vollstationär 02 – Krankenhausbehandlung, vollstationär mit vorausgegangener vorstationärer Behandlung 03 – Krankenhausbehandlung, teilstationär 04 – vorstationäre Behandlung ohne anschließende vollstationäre Behandlung 05 – Stationäre Entbindung 06 – Geburt 07 – Wiederaufnahme wegen Komplikationen (Fallpauschale) nach KFPV 2003 08 – Stationäre Aufnahme zur Organentnahme | M | 2 | SCHLUESSEL (AufnGrund) |
| Diagnoseangaben (wiederholbar) (DIAG) | | | | |
| FALLNUMMER | KH-interne Fallnummer | M | 15 | TEXT |

| Feld | Beschreibung | M/K | Feldlänge | Basistyp |
|---|---|-----|-----------|-----------------------------|
| ICD | Diagnoseschlüssel (ICD-Kode) | M | 9 | SCHLUESSEL (ICD-Katalog) |
| Prozedurenangaben (wiederholbar) (PROZ) | | | | |
| FALLNUMMER | KH-interne Fallnummer | M | 15 | TEXT |
| OPS | Prozedurenschlüssel (OPS-Kode) | M | 13 | SCHLUESSEL (OPS-Katalog) |
| Fachabteilungsangaben (wiederholbar) (ABTLG) | | | | |
| FALLNUMMER | KH-interne Fallnummer | M | 15 | TEXT |
| FACHABT | Fachabteilung | M | 4 | SCHLUESSEL |
| FABAUFNDATUM | Datum der Aufnahme in die Fachabteilung | M | 10 | DATUM (TT.MM.JJJJ) |
| FABAUFNZEIT | Uhrzeit der Aufnahme in die Fachabteilung | M | 5 | UHRZEIT (hh:mm) |

M = Muss; K = Kann

3.8 Felder der Risikostatistik

Die Felder der Risikostatistik werden aus dem Risikostatistik-Eingangsdatensatz berechnet (siehe Tabelle 83).

Für die Berechnung der Risikostatistik wird als Exportdatei eine Tabelle erzeugt. Dabei stellen die Felder DEK2 bis SONSTERKRANKUNG die zu prüfenden Risikofaktoren dar. Für einen Fall kann der jeweilige Risikofaktor erfüllt sein oder nicht. Daher sind die Spalten DEK2 bis SONSTERKRANKUNG jeweils mit 1 und 0 (WAHR oder FALSCH) kodiert. Ausnahmen bilden die Felder PATALTER und DAUBEAT, die das Alter bzw. die Dauer der Beatmung als GANZE-ZAHL beinhalten.

Tabelle 83: Felder der Risikostatistik

| Feld | Bezeichnung |
|-------------|---------------------------|
| IKNRKH | Institutionskennzeichen |
| STANDORT | entlassender Standort |
| LFDNR | laufende Nummer |
| AUFNFACHABT | aufnehmende Fachabteilung |
| DEK2 | Dekubitus Grad 2 |
| DEK3 | Dekubitus Grad 3 |
| DEK4 | Dekubitus Grad 4 |

| Feld | Bezeichnung |
|-----------------|----------------------------------|
| DEKnnb | Dekubitus nicht näher bezeichnet |
| PATALTER | Alter |
| DIABETES | Diabetes mellitus |
| DAUBEAT | Dauer der Beatmung |
| MOBILITAET | eingeschränkte Mobilität |
| INFEKTION | Infektion |
| DEMENZ | Demenz und Vigilanzstörung |
| INKONTINENZ | Inkontinenz |
| UNTERGEWICHT | Untergewicht und Mangelernährung |
| ADIPOSITAS | Adipositas |
| SONSTERKRANKUNG | weitere schwere Erkrankungen |

Ab dem Erfassungsjahr 2018 erfolgt die Erstellung der Risikostatistik fallbezogen, d.h. für jeden Fall wird eine Zeile in der Risikostatistik erstellt.

Tabelle 84: fallbezogene Risikostatistik

| TKNRKH | STAND- ORT | AUFN- FACH- ABT | LFDR | DEK2 | DEK3 | DEK4 | DEKnnb | ALTER | .. |
|-----------|---------------|-----------------------|------|------|------|------|--------|-------|----|
| 123456789 | 01 | 1500 | 1 | 1 | 0 | 0 | 1 | 67 | .. |
| 123456789 | 01 | 1500 | 2 | 0 | 0 | 1 | 0 | 75 | .. |
| 123456789 | 01 | 1500 | .. | .. | .. | .. | .. | .. | .. |
| .. | .. | 1500 | .. | .. | .. | .. | .. | .. | .. |

3.9 Algorithmus zur Berechnung der Risikostatistik

Der Risikostatistik-Datensatz ist in der Tabelle TdsFeld (fkTds = RISIKOSTATISTIK) hinterlegt. Hierbei sind alle Fälle zu berücksichtigen, die das administrative Kriterium erfüllen.

Dies trifft zu wenn:

1. der Patient stationär aufgenommen wird.
2. der Behandlungsfall einen bestimmten zeitlichen Rahmen aufweist.
3. der Patient 20 Jahre oder älter ist.

Zu 1.

Die Bedingung „stationär aufgenommen“ wird aus der Angabe im Datenfeld Aufnahmegrund abgeleitet:

```
AUFNGRUND NICHTIN ('03';'04') UND AUFNGRUND <> LEER
```

Diese Teilbedingung ist Teil des administrativen Kriteriums (siehe Attribut `bedingung` in Tabelle `AdminKriterium`).

Zu 2.¹²⁶

In die Berechnung der Risikostatistik sollen nur Patienten einbezogen werden, die in den Jahren 2018 oder 2019 aufgenommen und im Jahr 2019 entlassen wurden:

```
AUFNGRUND <> LEER UND AUFNGRUND NICHTIN ('03';'04') UND
AUFNDATUM >= '01.01.2018' UND AUFNDATUM <= '31.12.2019' UND
ENTLDATUM >= '01.01.2019' UND ENTLDATUM <= '31.12.2019' UND
ALTER >= 20
```

Zu 3.

In die Berechnung der Risikostatistik sollen nur Patienten einbezogen werden, die 20 Jahre oder älter sind:

```
ALTER >= 20
```

Nachfolgende Tabelle gibt einen Überblick über die Tabelle `AdminKriterium`.

Tabelle 85. Struktur der Tabelle `AdminKriterium`

| Feldname | Datentyp | Bemerkung |
|-------------------------------|----------|---|
| <code>idAdminKriterium</code> | INTEGER | Primärschlüssel |
| <code>name</code> | TEXT | Technischer Name des administrativen Kriteriums |
| <code>bedingung</code> | TEXT | Definition der Einschlussbedingung |
| <code>info</code> | TEXT | Bezeichnungstext des administrativen Kriteriums |

Die Summe aller Fälle, die das administrative Kriterium erfüllen, wird in dem Feld `Faelle_geprueft` der `RISIKOBASIS_<Erfassungsjahr>.txt` erfasst. Somit werden alle vollstationären Fälle ab 20 Jahren im Erfassungszeitraum gezählt.

3.10 Struktur und Syntax der Bedingungen

Im Folgenden werden die Variablen und Operatoren der Bedingungen sowie Diagnose- und Prozedurenlisten erläutert.

3.10.1 Die Variablen der Bedingungen

Die in den Bedingungen erlaubten Variablen sind in der Tabelle `SyntaxVariable` definiert.

¹²⁶ Verbindlich sind die in der Spezifikationsdatenbank hinterlegten Kriterien.

Die Variablennamen (Attribut `SyntaxVariable.name`) bestehen aus maximal 32 Zeichen. Sie dürfen nur die Buchstaben A bis Z (Großbuchstaben) und die Ziffern 0 bis 9 enthalten. Ein Feldname muss immer mit einem Buchstaben beginnen. Umlaute und Sonderzeichen sind in Feldnamen nicht erlaubt. Ein Feldname darf auch kein reserviertes Wort sein (z. B. Namen von Operatoren wie `EINSIN`).

Typen

Jede Variable hat einen Basistyp. Einen Überblick über mögliche Basistypen liefert die in Abschnitt B 1.4.4 dargestellte Tabelle „Basistypen der Variablen“.

Jeder der in der Tabelle `SyntaxVariable` definierten Variablen ist über den Wert des Attributes `SyntaxVariable.fkTdsFeld` ein Feld des Risikostatistik-Eingangsdatensatzes zugeordnet. Jedes dieser Felder besitzt somit einen Basistyp.

Listen

Eine Variable wird als Liste interpretiert, wenn der Wert des Attributs `SyntaxVariable.istListe = WAHR` ist.

Literale

Alphanumerische Literale (z. B. `SCHLUESSEL`) werden von einfachen Hochkommata eingeschlossen, während Zeichenketten vom Datentyp `TEXT` in doppelte Anführungsstriche gesetzt werden müssen.

Dies gilt nicht für numerische Literale (`GANZEZAHL`, `ZAHL`, `NUMSCHLUESSEL`) und Literale des Basistyps `BOOL` (Wahrheitswerte).

3.10.2 Diagnose und Prozedurenlisten

Die erlaubten Namen von Diagnose- bzw. Prozedurenlisten sind in der Tabelle `ICDListe` und `OPSListe` (Attribut `name`) zu finden.

Die jeweiligen Codes (nach ICD-10-GM bzw. OPS) sind in der durch Fremdschlüsselbeziehungen verknüpften Tabelle `ICDWert` bzw. `OPSWert` enthalten. Alle Codes sind terminal. Die Diagnosekodes enthalten nicht die Suffixe „+“, „*“ oder „!“.

3.10.3 Die Operatoren der Bedingungen

Die Operatoren der Bedingungen der Risikostatistik entsprechen den Operatoren der Auslösebedingungen der Spezifikation für QS-Filter. Die in Abschnitt B 1.4.4 dargestellte Tabelle „Präzedenz und Assoziativität der Operatoren“ gibt einen Überblick über die in der Syntax zulässigen Operatoren. Der aktuelle Überblick über alle zulässigen Operationen (inkl. Operanden) ist in Tabelle `SyntaxOperator` der Spezifikation zu finden.

Operatoren mit beidseitigen Listenfeldern als Operanden

`EINSIN`: Wenn mindestens ein Element aus der linken Liste in der rechten Liste enthalten ist, ist der Ausdruck `WAHR` (nichtleere Schnittmenge).

KEINSIN: Wenn kein Element der linken Liste in der rechten Liste enthalten ist, ist der Ausdruck WAHR (leere Schnittmenge)¹²⁷.

Beispiel:

Folgende Regel prüft, ob ein Element der Listenvariable DIAG einen bestimmten Code besitzt:

DIAG EINSIN DEK4_ICD

String-Vergleich bei Diagnosen

Die Operation EINSIN basiert für Diagnosen auf einem Vergleich von Zeichenketten (String-Vergleich): Für jedes Element der linken Liste wird über einen String-Vergleich geprüft, ob es in der rechten Liste enthalten ist.

Eine Modifikation des String-Vergleichs ist beim String-Vergleich von Diagnosen nötig: Hier werden die Suffixe „+“, „*“ und „!“ ignoriert. Nicht ignoriert wird das Sonderzeichen „.“.

3.11 Fehlerprüfung

Vor Evaluation der in den vorhergehenden Abschnitten beschriebenen Zählbedingungen ist eine Fehlerprüfung durchzuführen. Die Fehlerprüfung bezieht sich auf die Daten des Risikostatistik-Eingangsdatensatzes.

Tabelle 86. Fehlerkodes

| Beschreibung des Fehlers | Fehlercode | Formale Fehlerbedingung | Fehlermeldung |
|--|------------|--|---|
| Überprüfung des Formats | 1 | siehe Abschnitt B 2.4.8 („Überprüfung des Formats“) | Der Wert '<WERT>' des Datenfelds <Feld.name> '<TdsFeld.bezeichnung>' ist kein gültiger <BasisTyp.name>-Wert (<BasisTyp.bezeichnung> <BasisTyp.format>). |
| Überprüfung der Feldlänge | 2 | siehe Abschnitt B 2.4.8 („Überprüfung der Feldlänge“) | Der Wert '<WERT>' des Datenfelds <Feld.name> '<TdsFeld.bezeichnung>' überschreitet die zulässige Feldlänge <Feld.laenge>. |
| Sind in den Datenfeldern mit internen und externen Schlüsseln (BasisTyp SCHLUESSEL oder NUMSCHLUESSEL) | 3 | siehe Abschnitt B 2.4.8 („Überprüfung der Schlüsselkodes“) | Ungültiger Schlüsselcode <Wert> des Schlüssels <Schlüssel.name> im Datenfeld <Feld.name> '<TdsFeld.bezeichnung>' |

¹²⁷ Dieser Operator ist redundant, da er auch durch Negation des EINSIN-Operators abgedeckt ist.

| Beschreibung des Fehlers | Fehlercode | Formale Fehlerbedingung | Fehlermeldung |
|---|------------|---|--|
| gültige Schlüsselkodes eingetragen? ¹²⁸ | | | |
| Überprüfung numerischer Wertebereiche | 4 | siehe Abschnitt B 2.4.8 („Überprüfung numerischer Wertebereiche“) | Der Wert '<WERT>' des Datenfelds <Feld.name> '<TdsFeld.bezeichnung>' ist kleiner als '<Feld.min>' oder Der Wert '<WERT>' des Datenfelds <Feld.name> '<TdsFeld.bezeichnung>' ist größer als '<Feld.max>' |
| Ist ein Muss-Feld ausgefüllt? | 5 | siehe Abschnitt B 2.4.8 („Überprüfung der Muss-Felder“) Kriterium: Attribut <TdsFeld.fkMuss Kann> | Das Datenfeld <Feld.name> '<TdsFeld.bezeichnung>' muss einen gültigen Wert enthalten. |
| Ist der Fall dem Erfassungsjahr der Spezifikation zugeordnet? | 6 | AUFNDATUM < '01.01.2018' ODER AUFNDATUM > '31.12.2019' | Der Fall ist nicht im Jahr 2018 oder 2019 aufgenommen: Aufnahmedatum = <Wert> |



Hinweis

- In <Wert> ist der Wert des Datenfelds der Falldaten einzusetzen, auf den sich die Fehlermeldung bezieht.
- Ansonsten sind in die <...>-Ausdrücke die entsprechenden Einträge aus der Datenbank einzusetzen.
- Im Fehlerfall sind entsprechende Einträge im Teildatensatz FEHLER zu generieren:
- Fehlercode in Attribut FKODE
- Fehlermeldung in Attribut FMELDUNG
- Es wird beim Auftreten eines Fehlers nicht weiter geprüft, ob der Behandlungsfall in die Risikostatistik einbezogen wird.

¹²⁸ Die Korrektheit (z. B. ausschließlich gültige terminale Codes oder Verwendung gültiger Katalogversionen) der extern definierten Codes (ICD) muss vor der Prüfung des Datensatzes sichergestellt sein.

4 XML-Schema

Die XML-Schema-Datei (XSD) ist eine Empfehlung des W3C¹²⁹ zum Definieren von Strukturen für XML-Dokumente.

In der Basisspezifikation werden die XML-Schemata aus der Spezifikationsdatenbank abgeleitet und haben die Aufgabe, die aktuellen Datenflussmodelle der G-BA-Richtlinien abzubilden:

- Abbildung der zu exportierenden QS-Daten
- Abbildung der Schnittstellen an den Institutionsübergängen (z. B. die Schnittstelle zwischen DAS und 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.

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.

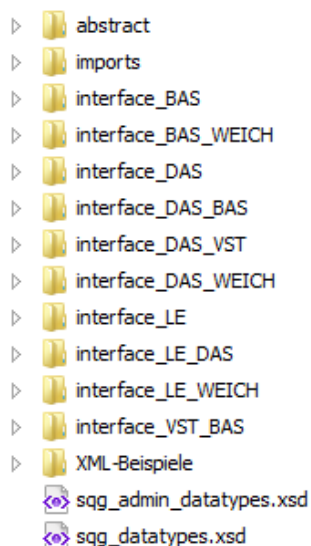


Abbildung 27: Dateiordner der Schnittstellen-Schemata

¹²⁹ <http://www.w3.org/XML/Schema>

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 wird hier näher erläutert.

Beispiel:

Die leistungserbringeridentifizierenden Daten existieren gemäß Datenflussmodell der Qesü-RL bzw. der DeQS-RL in drei Ausprägungen:

- im Klartext (Schnittstelle LE)
- pseudonymisiert (Schnittstelle DAS)
- 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.

4.2 Schnittstellen

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

Tabelle 87: Verwendbare Schemata und Ablageort

| Schnittstelle | Schema | Verfahren | Absender | Empfänger |
|----------------------------|-------------------------|------------------------------|-----------------|-----------|
| LE interface_LE | 2017_bas_dv_1.0_Export | Direkte Verfahren (dv) | KH | BAS |
| | 2017_kv_pid_1.0_Export | PID-Verfahren (pid) | KH/PR-Kollektiv | KV |
| | 2017_lqs_iv_1.0_Export | Indirekte Verfahren (iv) | KH | LQS |
| | 2017_lqs_pid_1.0_Export | PID-Verfahren (pid) | KH | LQS/LKG |
| | 2017_sv_pid_1.0_Export | PID-Verfahren (pid) | PR-Selektiv | DAS-SV |
| LE/DAS interface_LE_DAS | interface_LE_BAS | Direkte Verfahren (dv) | KH | BAS |
| | interface_LE_KV | PID-Verfahren (pid) | PR-Kollektiv | KV |
| | | PID-Verfahren (pid) (CHE_KV) | KH | LQS/LKG |
| | interface_LE_LQS_iv | Indirekte Verfahren (iv) | KH | LQS |

| Schnittstelle | Schema | Verfahren | Absender | Empfänger |
|------------------------------|-----------------------|---|--|-----------|
| | interface_LE_LQS_pid | PID-Verfahren | KH | LQS/LKG |
| | interface_LE_SV | PID-Verfahren | PR-Selektiv | DAS-SV |
| DAS interface_DAS | interface_KV_psn | PID-Verfahren | Nach der QS-Entschlüsselung in der KV oder LQS/LKG | |
| | interface_KV_psn | PID-Verfahren | Nach der LE-Pseudonymisierung in der KV oder LQS/LKG | |
| | interface_LQS_iv | Indirekte Verfahren (iv) | Nach der QS-Entschlüsselung in der LQS/LKG | |
| | interface_LQS_pid | PID-Verfahren | Nach der QS-Entschlüsselung in der LQS/LKG | |
| | interface_LQS_psn | Verfahren nach Qesü-RL/DeQS-RL | Nach der LE-Pseudonymisierung in der LQS/LKG | |
| | interface_SV_psn | PID-Verfahren | Nach der LE-Pseudonymisierung | |
| | Interface_KV_soll.xsd | Sollstatistik nach Qesü-RL | Erstellung der Sollstatistik | |
| | Interface_SV_soll.xsd | Sollstatistik | Erstellung der Sollstatistik | |
| DAS/VST interface_DAS_VST | interface_DAS_VST | PID-Verfahren | DAS | VST |
| DAS/BAS interface_DAS_BAS | interface_DAS_BAS.xsd | Indirekte Verfahren (iv) und Sollstatistik(KV/SV) | LQS | BAS |
| VST/BAS interface_VST_BAS | interface_VST_BAS.xsd | PID-Verfahren | VST | BAS |

Um nach einer Schemavalidierung der XML-Dateien die Weiterverarbeitung und dementsprechend die spezifikationskonforme Protokollierung auf Datensatzebene weiterhin zu ermöglichen, wurden neben der oben in Tabelle 87 beschriebenen Schemavariante ein „weiches“ Schema für die Schnittstellen LE und DAS eingeführt. Diese weiche Variante wird ausschließlich mit dem Datenprüfprogramm verwendet (Abschnitt B 5.2). Es ist dabei zu beachten, dass nur die DAS, die die QS-Daten entschlüsseln, diese weiche Variante benötigen. Diese sind LQS/LKG, KV und BAS.

In der folgenden Tabelle werden die Schema-Dateien aufgeführt, die im Rahmen der Prüfung mit dem Datenprüfprogramm verwendet werden.

Tabelle 88: Weiche Schemavarianten für das Datenprüfprogramm

| Schnittstelle | Schema | Verwendungszweck in Zusammenhang mit dem DPP |
|--------------------------------|-------------------------|---|
| LE Interface _LE_WEICH | 2017_bas_dv_1.0_Export | Validierung der direkten Module mit dem Datenprüfprogramm (DPP) |
| | 2017_kv_pid_1.0_Export | Validierung der Module (für die KV) mit dem DPP |
| | 2017_lqs_iv_1.0_Export | Validierung der indirekten Module für die LQS mit dem DPP |
| | 2017_lqs_pid_1.0_Export | Validierung der PID-Verfahren ¹³⁰ für die LQS mit dem DPP |
| | 2017_sv_pid_1.0_Export | Validierung der Module für die DAS-SV mit dem DPP |
| DAS Interface _DAS_WEICH | interface_KV_pid | Validierung der PID-Verfahren mit dem DPP nach der Entschlüsselung der QS-Daten in der KV |
| | interface_KV_psn | Validierung der PID-Verfahren mit dem DPP nach der Entschlüsselung der QS-Daten und der Pseudonymisierung der LE-Daten (in der KV) |
| | interface_LQS_iv | Validierung der indirekten Module mit dem DPP nach der Entschlüsselung der QS-Daten in der LQS/LKG |
| | interface_LQS_pid | Validierung der PID-Verfahren mit dem DPP nach der Entschlüsselung der QS-Daten in der LQS/LKG |
| | interface_LQS_psn | Validierung der PID-Verfahren mit dem DPP nach der Entschlüsselung der QS-Daten und der Pseudonymisierung der LE-Daten in der LQS/LKG |
| BAS interface _BAS_WEICH | interface_BAS_dv.xsd | Validierung der direkten Module mit dem Datenprüfprogramm (DPP) |
| | interface_BAS_iv.xsd | Validierung der indirekten Module mit dem DPP |
| | interface_BAS_pid.xsd | Validierung der PID-Verfahren mit dem DPP |

In der folgenden Tabelle werden Schema-Dateien aufgeführt, die im Rahmen der Rückprotokollierung Verwendung finden.

¹³⁰ Follow-up-Verfahren mit PID


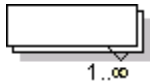



Tabelle 89: XML-Schemata für die Rückprotokollierung

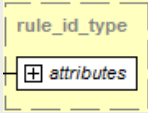
| Schnittstelle | Schema | Verwendungszweck in Zusammenhang mit dem DPP |
|------------------------------|----------------------|--|
| LE/DAS interface_LE_DAS | response_DAS_LE.xsd | Datenflussprotokoll der DAS |
| | response_receipt.xsd | Empfangsbestätigung der DAS |
| DAS/VST interface_DAS_VST | response_VST_DAS.xsd | Datenflussprotokoll der VST |
| | response_receipt.xsd | Empfangsbestätigung der VST |
| DAS/BAS interface_DAS_BAS | response_BAS_DAS.xsd | Datenflussprotokoll der BAS |
| VST/BAS interface_VST_BAS | response_BAS_VST.xsd | Datenflussprotokoll von der BAS an die VST |
| | response_receipt.xsd | Empfangsbestätigung |

4.3 Darstellung der XML-Struktur

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

Tabelle 90: Symbole in den XML-Schema-Diagrammen

| Symbol | Beschreibung |
|---|--|
|  | Optionales Element Kardinalität 0..1 („0 oder 1“) |
|  | Obligatorisches Element Kardinalität 1: das Element muss genau einmal vorkommen |
|  | Mehrfach wiederholbares Element Kardinalität: die erlaubte Anzahl der Elemente wird unter dem Symbol dargestellt (Beispiel: 1..n, n..m). |
|  | Referenzelement Das referenzierte globale Element ist an anderer Stelle im Schema definiert. |
|  | Eine Folge von Elementen Die Elemente müssen genau in der Reihenfolge vorkommen, in der sie im Schemadiagramm angezeigt sind. |
|  | Eine Auswahl von Elementen Nur ein einziges Element aus der Liste kann ausgewählt werden. |
|  | Ein Element mit Kind-Elementen |

| Symbol | Beschreibung |
|---|--|
|  | Komplexer Datentyp Der komplexe Datentyp wird mit einem Rahmen mit einem gelben Hintergrund angezeigt. |

Die wesentlichen Bestandteile der XML-Schemata werden in den nachfolgenden Abschnitten dargestellt. Die Darstellung umfasst folgende Eigenschaften des betrachteten Elements:

- Grafische Abbildung der Kind-Elemente und -Attribute
- Auflistung der Kind-Elemente
- Auflistung der Kind-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“)

4.4 Aufbau der XML-Exportdatei

Grundsätzlich beginnt jede XML-Exportdatei mit einer Headerzeile gefolgt vom Wurzelement `<root>`, das den gesamten Inhalt einschließt. Als Zeichensatz wird „UTF-8“ (Unicode-Codierung) verwendet.

Beispiel:

Headerzeile

```
<?xml version="1.0" encoding="UTF-8"?>
<root> </root>
```

4.4.1 Namensräume

Die Schemata für den XML-Datenfluss sind für den Namensraum mit der id "urn:gba:sqg" definiert. Dieser soll ohne Präfix-Mapping im `<root>`-Element in das XML eingebunden werden.

Falls ein Ablageort des zugrunde liegenden Schemas angegeben werden soll, wird dieses im `<root>`-Element mit dem Attribut "schemaLocation" vorgenommen. Da dieses Attribut ebenfalls aus einem externen Namensraum stammt, wird dieser Namensraum dem reservierten Präfix "xsi" zugeordnet, was sich dann als "xsi:schemaLocation='...'" liest:

- `xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance`
- `xsi:schemaLocation="urn:gba:sqg interface_LE_KV.xsd"`

Für die Verschlüsselung der XML-Elemente werden zwei externe Namensräume mit der id:

- " `xmlns:ds=http://www.w3.org/2000/09/xmldsig#`"
- und

- " xmlns:xenc=http://www.w3.org/2001/04/xmlenc#"

verwendet.

Diese sollen auf die reservierten Präfixe "ds" und "xenc" zugeordnet eingebunden werden. Es ist zu empfehlen, die externen Namensräume in das <root>-Element einzubinden, um lokale Wiederholungen auf Elementebene zu vermeiden.

Beispiel:

```
<root container_version="2.0" content_version="1.0"
xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance
xmlns="urn:gba:sqg"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
</root>
```

4.4.2 Wurzelement <root>

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

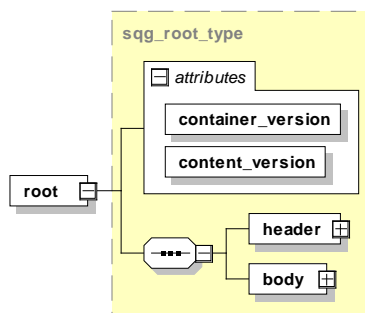


Abbildung 28: Root-Element und Kind-Elemente header und body

Das Root-Element hat zusätzlich zwei Attribute (Tabelle 91).

Tabelle 91: Root-Element – Attribute

| Name | 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 kleinen optionalen Änderungen wird die Versionsnummer |

| Name | Type | Use | Fixed | Beschreibung |
|-----------------|--------------------------|----------|-------|---|
| | | | | <p>beibehalten, um die Aufwärtskompatibilität zu gewährleisten.</p> <p>Ein XML-Dokument, das einen alten Wert dieses Attributs enthält, muss von der Datenannahmestelle zurückgewiesen werden.</p> |
| content_version | content_version_Datentyp | required | 1.0 | <p>Ist ein fixer Wert und definiert die aktuell gültige Versionsnummer des Inhalts der QS-Daten.</p> <p>Die Versionsnummer wird erhöht, wenn unterjährig das Schema unabhängig von der zugrundeliegenden Spezifikationsdatenbank geändert wird.</p> |

4.4.3 Header-Bereich

Element header

Das Element `Header` besteht aus Metadaten (administrative und meldebezogene Daten) zu den QS-Daten, die im `<body>` enthalten sind.

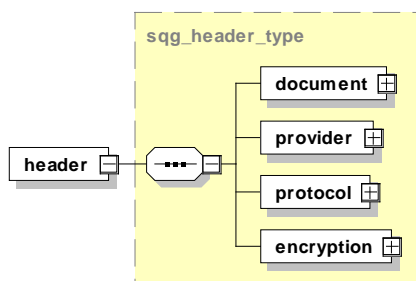


Abbildung 29: Aufbau des Elements header

Element header/document

Das Element enthält allgemeine Informationen zum erstellten Dokument. Dieses Element ist weitestgehend über den gesamten Datenfluss hinweg beständig. Nur das Element `<modification_dttm>` (Modifikationsdatum) wird bei jeder Bearbeitung neu gesetzt.

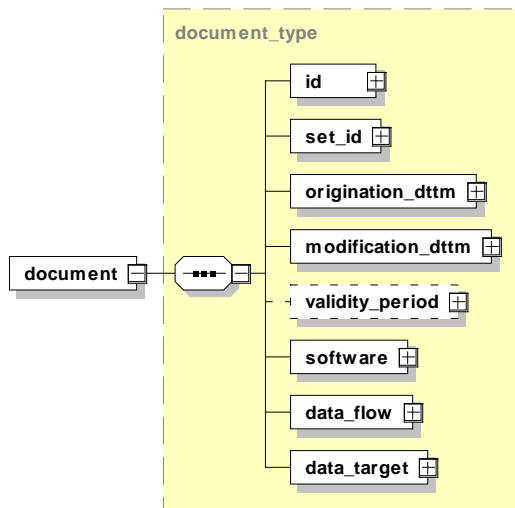


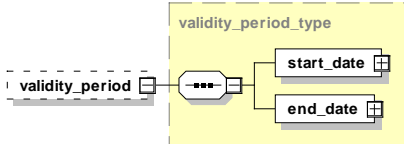
Abbildung 30: Aufbau des Elements document

Dieses Element hat weitere Kind-Elemente, die in der folgenden Tabelle beschrieben werden.

Tabelle 92: Kind-Elemente des Elements document

| Kind-Elemente | Beschreibung |
|---------------------|---|
| <id> | Nach Erstellung nicht mehr modifizierbar. Eindeutige ID des Dokuments, wird vom Dokumentenersteller als GUID erzeugt. ¹³¹ |
| <set_id> | Nach Erstellung nicht mehr modifizierbar. Eindeutige ID für mehrere Dokumente, die im selben Zusammenhang stehen; wird vom Leistungserbringer erzeugt. Dafür könnte z. B. die GUID vom ersten Dokument des Zusammenhangs verwendet werden. |
| <origination_dttm> | Das Element ist der Zeitstempel der ursprünglichen Dokumenterzeugung. Darf nach seiner Erstellung nicht mehr modifiziert werden. Format: CCYY-MM-DDThh:mm:ss |
| <das_receive_dttm> | Ist der Zeitstempel des Dateneingangs nach Übermittlung von LE an DAS. Nach Einfügen durch die DAS nicht mehr modifizierbar. Format: CCYY-MM-DDThh:mm:ss |
| <modification_dttm> | Dieses Element ist ein Zeitstempel und muss bei jeder Modifikation des Dokuments aktualisiert werden. Das Modifikationsdatum darf nicht vor dem Erstelldatum liegen. Format: CCYY-MM-DDThh:mm:ss |

¹³¹ Ein Globally Unique Identifier oder kurz GUID ist eine global eindeutige Zahl mit 128 Bit (16 Bytes), die ein Dokument weltweit eindeutig identifiziert.

| Kind-Elemente | Beschreibung |
|-------------------|--|
| <validity_period> |  <p>Mithilfe des optionalen <validity_period>-Elements wird der Bezugszeitraum der Datei angegeben. Dies kann ein Jahr oder ein Quartal oder ein frei wählbarer Bereich sein. Zur Qualifizierung sind die Felder <start_date> und <end_date> entsprechend zu füllen. Im QSKH-Bereich hat das Element noch keine Anwendung.</p> |

<origination_dttm> und <modification_dttm> sind vom Datentyp dateTime, der einen Zeitpunkt darstellt (ISO 8601). Es handelt sich um das Format CCYY-MM-DDThh:mm:ss:

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

Dieser Darstellung kann direkt ein „Z“ nachgestellt werden, um anzuzeigen, dass es sich um die Universal Time Coordinated (UTC) handelt. Folgt der Zeitangabe statt eines „Z“ ein Plus- oder Minuszeichen bedeutet das, dass die darauf folgende Angabe im Format „hh:mm“ die Differenz zur UTC angibt (der Minutenanteil ist erforderlich).

Beispiele:

- 2011-11-01T21:32:52
- 2011-11-01T21:32:52+02:00 (Zeitzonendifferenz von plus 2 Stunden)
- 2011-11-01T19:32:52Z

Sammelelement für Angaben zur eingesetzten QS-Software.

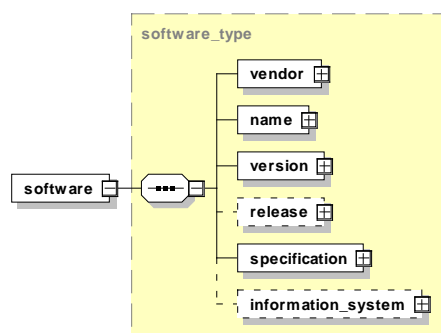


Abbildung 31: Aufbau des Elements software

Dieses Element enthält Kind-Elemente, die in der folgenden Tabelle beschrieben werden.

Tabelle 93: Kind-Elemente des Elements *software*

| Kind-Elemente | Beschreibung |
|----------------------|--|
| <vendor> | Enthält Informationen über den Softwarehersteller |
| <name> | Enthält den Softwarenamen der eingesetzten Software |
| <version> | Enthält die Version der eingesetzten Software |
| <release> | Enthält das Release der eingesetzten Software |
| <specification> | Enthält einen Wert aus der enumeration enum_spez_type: Version der Spezifikation, auf deren Basis die QS-Dokumentationssoftware entwickelt wurde |
| <information_system> | Enthält Angaben zum eingesetzten Informationssystem (KIS/AIS). |

Element header/document/software/information_system

Sammelement für Angaben zum eingesetzten Informationssystem (KIS/AIS).

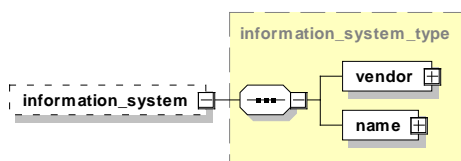


Abbildung 32: Aufbau des Elements *information_system*

Tabelle 94: Kind-Elemente des Elements *information_system*

| Kind-Elemente | Beschreibung |
|---------------|--|
| <vendor> | Enthält Informationen über den Softwarehersteller. |
| <name> | Enthält den Softwarenamen der eingesetzten Software. |

Beim Element <software>/<vendor>/<registration> müssen die ambulanten Leistungserbringer die KBV-Prüfnummer eintragen. Im stationären Bereich müssen die Softwareanbieter die vom IQTIG vergebene Registriernummer verwenden.

Element header/document/data_flow

Dieses Element gibt an, für welchen Datenfluss (Datenannahmestelle) dieses Dokument erzeugt wurde.

Tabelle 95: Angabe des betreffenden Datenflusses

| Ausprägung | Beschreibung |
|-------------------------|---|
| QS-Bundesbezogen | für direkte Verfahren, die direkt vom LE an die BAS übermittelt werden müssen |
| QS-Laenderbezogen | für indirekte Verfahren (mit und ohne PID), die an die Landesgeschäftsstellen (LQS/LKGen) übermittelt werden müssen |
| QS-Kollektivvertraglich | für die Verfahren, die an die kassenärztlichen Vereinigungen (KVen) übermittelt werden müssen |
| QS-Selektivvertraglich | für die Verfahren, die an die Vertrauensstelle als Datenannahmestelle (DAS-SV) übermittelt werden müssen |
| Strukturierter Dialog | <noch nicht benötigt> |
| Datenvalidierung | <noch nicht benötigt> |
| Protokoll | <noch nicht benötigt> |
| Sollstatistik | für die Übermittlung der Sollstatistik |

Element header/document/data_target

Dieses Element gibt an, welches Ziel der Datenfluss hat.

Tabelle 96: Angabe des betreffenden Datenfluss-Ziels

| Datenfluss | Ziel |
|----------------|--|
| Echtdatenpool | Echtdaten für den Echtbetrieb |
| Probedatenpool | Echtdaten für vorläufige Auswertungen wie der Sonderexport |
| Testdatenpool | Testdaten für Testzwecke |

**Achtung
Datenverlust**

Nur Daten mit der Kennzeichnung „Echtdatenpool“ werden für die Erstellung der Bundes-, Landes- und Rückmeldeberichte berücksichtigt.

Element header/provider

Das Element <provider> gibt an, welche Institution dieses Dokument zuletzt bearbeitet hat. Es wird in jeder am Datenfluss beteiligten Instanz durch diese ersetzt und so zur nächsten Instanz geschickt. Einzige Ausnahme bildet die Vertrauensstelle. Diese lässt das Provider-Element unberührt.

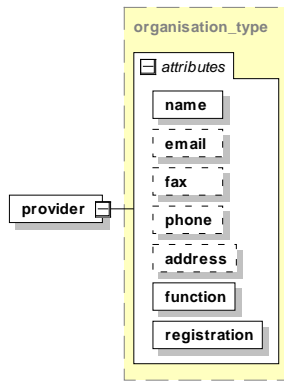


Abbildung 33: Aufbau des Elements provider

Tabelle 97: Attribute des Elements header/provider

| Name | Type | Use | Beschreibung |
|--------------|-----------------------------|----------|---|
| name | xs:string | required | Name der Institution |
| email | emailAddress_type | optional | E-Mail-Adresse |
| fax | xs:string | optional | Faxnummer |
| phone | xs:string | optional | Telefonnummer |
| address | xs:string | optional | Adresse |
| function | enum_organisa- tion_type | required | Bundesauswertestelle/Datenannah- mestelle Vertrauensstelle/Softwarehersteller/ Undefined |
| registration | registration_type | required | Registrierungsnummer. Sollte für die Datenübertragung an die DAS (DAS-SV) keine Registrierungs- nummer erforderlich, ist das Element mit einem Dummy-Wert „000“ zu füllen. Sollte für die Datenübertragung an die DAS (KV) keine Registrierungsnummer erforderlich sein, kann dieses Element für den Datenfluss vom ambulanten Leistungserbringer an die KV wegge- lassen werden. Für alle folgenden Da- tenflüsse ist die Registriernummer eine Pflichtangabe. |

Element header/protocol

Das Element `<protocol>` nimmt Informationen zu Prüfungen auf, die im Datenfluss durchgeführt wurden. Es ist Teil der Rückprotokollierung. Dieses Element ist nicht optional und soll gemeinsam mit dem Unterelement `<status_document>` von Anfang an im Datenfluss vorhanden sein, um nachfolgende im Datenfluss vorgenommene Prüfergebnisse aufzunehmen.

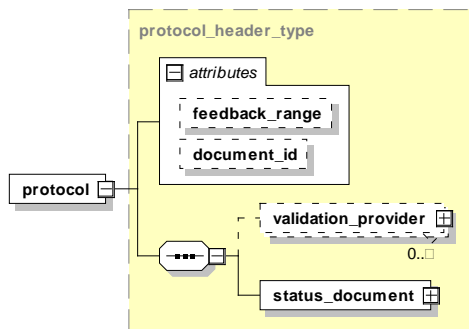


Abbildung 34: Aufbau des Elements header/protocol

Dieses Element hat zusätzlich zu den optionalen Attributen `feedback_range` und `document_id` zwei Kind-Elemente:

`<validation_provider>` und `<status_document>`

Tabelle 98: Attribute des Elements header/protocol

| Name | Type | Use | Beschreibung |
|----------------|------|----------|---|
| feedback_range | | optional | Da die Transaktionsprotokolle durch die Empfangsbestätigungen ersetzt wurden, ist nur der Wert „dataflow“ zu verwenden. |
| document_id | | optional | Soweit die GUID des Exportdokuments lesbar ist, muss sie in das Attribut <code>document_id</code> eingetragen werden. |

Element header/protocol/validation_provider

Hier gibt sich die Stelle zu erkennen, die einen oder mehrere Prüfungsschritte durchgeführt hat. Die Ergebnisse der Prüfung werden in diesem Container abgelegt und werden Teil der Rückprotokollierung.

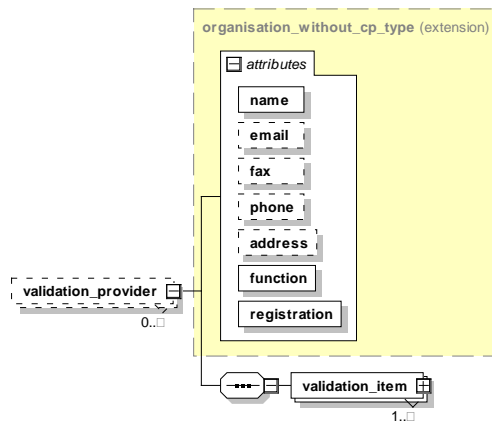


Abbildung 35: Aufbau und Kind-Elemente des Elements validation_provider

Tabelle 99: Attribute des Elements validation_provider

| Name | 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 ihre durchgeführten Prüfungen in der Auflistung `<validation_item>`.

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

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

Das Ergebnis der Prüfung wird in das Attribut @V des Elements `<status>` im Element `<validation_item>` eingetragen.

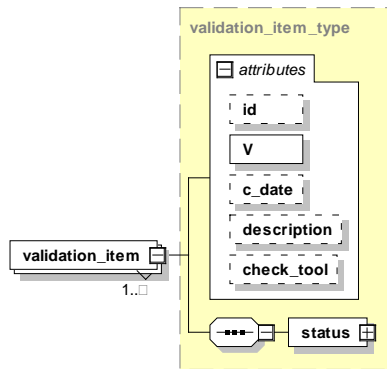


Abbildung 36: Aufbau und Kind-Elemente des Elements validation_item

Tabelle 100: Attribute des Elements validation_item

| Name | Type | Beschreibung |
|-------------|----------------------|---|
| id | xs:int | 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. Die ID muss nur dann vergeben werden, wenn eine Prüfung auf Fallebene stattfindet. |
| V | enum_validation_type | Dieser Wert bezeichnet die durchgeführte Prüfung anhand einer „enumeration“, die in sqg_protocol.xsd definiert wird. Gültige Werte sind: Dechiffrierung, LE_Pseudonym, PID_Pseudonym, Protokoll, Schema, Spezifikation, Transaktion, sonstige Prüfung. |
| c_date | xs:dateTime | Hier kann ein Zeitstempel für die Verarbeitung angegeben werden. |
| description | xs:string | Prüfungsbeschreibung laut Spalte „Prüfung“ in Sicht vPruefung in QSDOK-Datenbank |
| check_tool | xs:string | Versionsnummer des Prüftools (z. B. das Datenprüfprogramm) |

Attribut /protocol/validation_provider/validation_item/@check_tool

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 XSLT-Skripts eingetragen).

Das Datenprüfprogramm trägt automatisch die Versionsnummer in das <validation_item>-Element ein. Damit das Protokoll nicht unnötig groß wird, wird die Information über das Tool nur auf Dokumentenebene aufgenommen (header/protocol/validation_provider/validation_item/).

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 verschlechtert oder gleich bleibt. `ERROR` bedeutet, dass das Dokument komplett zurückgewiesen werden muss.

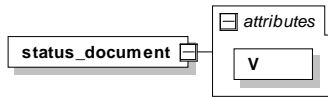


Abbildung 37: Aufbau des Elements `status_document`

Tabelle 101: Attribute des Elements `status_document`

| Name | Type | Use | Beschreibung |
|------|-------------|----------|----------------------------------|
| v | status_type | required | Mögliche Werte: OK/WARNING/ERROR |

Element header/protocol/validation_provider/validation_item/status

Das Element gibt an, ob die betroffene Testeinheit 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.

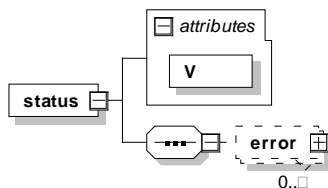


Abbildung 38: Aufbau und Kind-Elemente des Elements `status`

Tabelle 102: Attribut des Elements `status`

| Name | Type | Use | Beschreibung |
|------|-------------|----------|--|
| v | status_type | required | Status einer Prüfung mit folgenden, möglichen Werten: <code>OK</code> , <code>WARNING</code> oder <code>ERROR</code> . |

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 einzig verpflichtendes Unterelement gilt das `<error_message>`-Element. Die Elemente `<rule_id>` und `<rule_type>` sind spezifisch für die Anwendung von Plausibilitätsregeln für die Spezifikation:

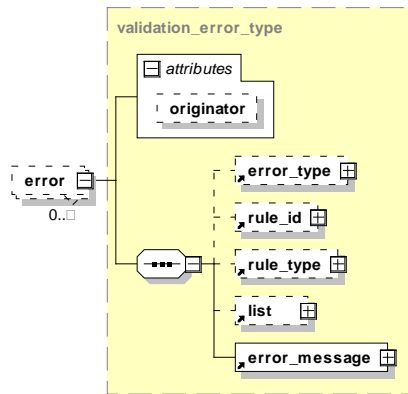


Abbildung 39: Aufbau des Elements error

Tabelle 103: Attribut des Elements error

| Name | Type | Use | Beschreibung |
|------------|------------------------|----------|---|
| originator | enum_organisation_type | optional | Mögliche Werte: Bundesauswertestelle, Datenannahmestelle, Vertrauensstelle, Leistungserbringer, Softwarehersteller, undefined |

Tabelle 104: Kind-Elemente des Elements error

| Kind-Elemente | Beschreibung |
|-----------------|--|
| <rule_id> | Nummer der Regel (idRegeln in Tabelle Regeln) oder Nummer der Fehlermeldung aus der Tabelle Fehlermeldung (idFehlermeldung) |
| <rule_type> | Werte H (=hart) oder W (=Warnung bzw. weich) |
| <liste> | In Abhängigkeit von der Fehlerart entweder Liste von Teildatensätzen oder von Bogenfeldern |
| <error_message> | Fehlermeldung als Freitext |
| <error_type> | Hat folgende Ausprägungen ¹³² : EXPORT = Formatfehler der Exportdatei DOPPELT = bereits vorhandener Datensatz wird erneut übermittelt TDS = Vollständigkeit und Version der Teildatensätze |

¹³² Die Ausprägung STEUER (= Formatfehler der Steuerdatei) wurde im Rahmen eines csv-Exports eingeführt und wird seit Überführung in das XML-Format nicht mehr verwendet

| Kind-Elemente | Beschreibung |
|---------------|---|
| | <p>WERT = Wertebereichsverletzung</p> <p>REGEL = Plausibilitätsverletzung</p> <p>KOLLISION = Patientenpseudonym mit unterschiedlichen Alters-/Geschlechtsangaben</p> <p>LE = LE nicht entschlüsselbar</p> <p>FEHLT = Angabe fehlt</p> <p>PID = PID nicht entschlüsselbar (in Kombination mit Dechiffrierung von „validation_item“)</p> <p>QS = QS-Daten nicht entschlüsselbar (in Kombination mit Dechiffrierung von „validation_item“)</p> <p>IST_Statistik = Fehlende/falsche Angaben zur IST-Statistik</p> <p>SOLL_Statistik = Fehlende/falsche Angaben zur SOLL-Statistik</p> |

Element header/encryption

Das Element nimmt Informationen über den Schlüssel auf, mit dem die Daten verschlüsselt worden sind. Das Attribut `id` enthält den Namen des symmetrisch verschlüsselten XML-Knotens. Eine Beispielimplementierung dieser Spezifikation ist ein öffentliches Verschlüsselungsprogramm des IQTIG (XPacker).

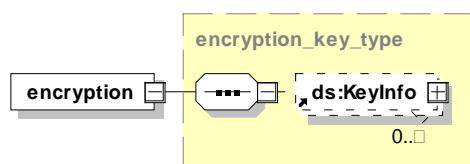


Abbildung 40: Aufbau und Attribute des Elements `encryption`(Krankenhaus)

Das Programm dient zur Ver- und Entschlüsselung einzelner XML-Elemente (Tags) innerhalb einer XML-Datei, basierend auf einem hybriden Verfahren, das aus folgenden Einzelschritten besteht:

- Ein zufälliger symmetrischer Schlüssel wird erzeugt.
- Mit diesem Schlüssel wird ein XML-Element (z. B. `qs_data`) chiffriert.
- Der Schlüssel wird nun mit dem „public Key“ des Empfängers (z. B. LQS/LKG) verschlüsselt.
- Der mit dem „public Key“ chiffrierte symmetrische Schlüssel wird dem Empfänger zusammen mit den verschlüsselten Daten übergeben.
- Der Empfänger dechiffriert den Schlüssel mit seinem „private Key“ und erhält so den symmetrischen Schlüssel.
- Mit diesem symmetrischen Schlüssel dechiffriert der Empfänger die verschlüsselten Daten.

Weitere Informationen sind der Dokumentation des Verschlüsselungsprogramms (Abschnitt B 5.3.1) zu entnehmen.

Element header/encryption/feedback_key

Gemäß der Qesü-RL und der DeQS-RL erstellt die BAS Rückmeldeberichte (RMB) für die LE. Diese RMB werden von der BAS über die DAS an die jeweiligen LE geleitet. Dabei ist sicherzustellen, dass die DAS keine Möglichkeit zur Einsicht in die Berichte hat.



Hinweis

Die Erhebung des Feedbackkeys im Rahmen der Dokumentation durch den Leistungserbringer ist fallbezogen lediglich für die Verfahren QS PCI und QS CHE erforderlich. Für das Verfahren QS WI muss der Feedbackkey im Rahmen der einrichtungsbezogenen QS-Dokumentation übermittelt werden.

Krankenhäuser sowie kollektivvertraglich tätige LE müssen daher, damit sie den RMB von der BAS erhalten können, eine Zusatzinformation an die BAS liefern. In den Header der QS-Datenlieferung ist ein mit dem öffentlichen Schlüssel der BAS verschlüsseltes Passwort (Feedbackkey) zu übermitteln. Dieser Feedbackkey wird von der BAS entschlüsselt und später wiederum zur Verschlüsselung des RMB verwendet. Auf diese Weise ist sichergestellt, dass nur die BAS und der jeweilige LE Einblick in den RMB erhalten.

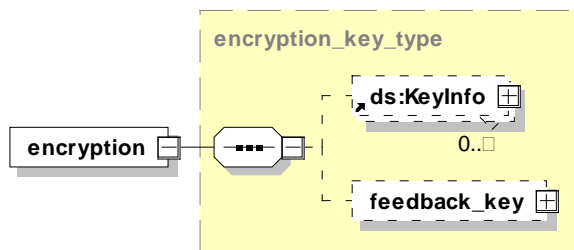


Abbildung 41: Aufbau und Attribute des Elements encryption(AP-selektiv/-kollektiv)

Die Kind-Elemente werden in folgender Tabelle beschrieben.

Tabelle 105: feedback_key: Kind-Elemente

| Kind-Elemente | Beschreibung |
|---------------|---|
| <key_name> | Enthält den Namen des Berichts (hier "Rueckmeldebericht") |
| <key_value> | Hier wird das Passwort (Feedbackkey) aufgenommen |

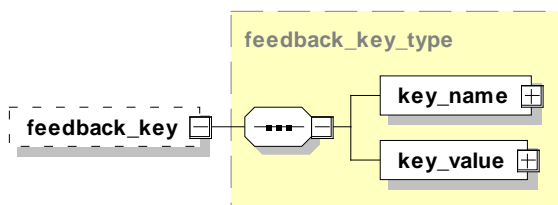


Abbildung 42: Aufbau des Elements feedback_key

Mehrfache Datenlieferung

Übermittelt der LE eine Datenlieferung mehr als einmal, hat er dafür zu sorgen, dass in jeder Datenlieferung das verschlüsselte Passwort (Feedbackkey) enthalten ist.

Die BAS verschlüsselt den Bericht mit dem zuletzt an sie gesendeten Passwort (Feedbackkey).

4.4.4 Body-Bereich

Im `<body>`-Element liegen die eigentlichen PID, QS- und LE-Daten. Der Body-Bereich kann einen oder mehrere `<data_container>` enthalten, die einem bestimmten Leistungserbringer zugeordnet sind.

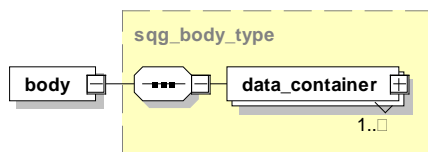


Abbildung 43: Aufbau des Elements body

Kind-Element body/data_container

Ein `<data_container>` ist einem bestimmten Leistungserbringer zuzuordnen. In der Regel sollte in einem Dokument nur ein `<data_container>` vorhanden sein. Da aber mehr als ein `<data_container>` erlaubt ist, können ggf. auch mehrere `<data_container>` für mehrere Leistungserbringer verwendet werden, wenn das Dokument z. B. von einer Stelle (z. B. einer Datenannahmestelle) erstellt wird, die Daten mehrerer Leistungserbringer sammelt. Dies ist der Fall bei der Erstellung der Sollstatistik durch die KV und die DAS-SV.

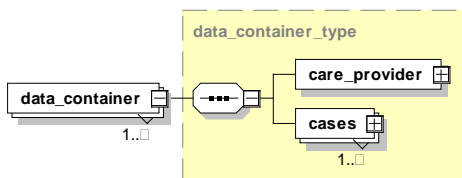


Abbildung 44: Aufbau des Elements body/data_container

Element body/data_container/care_provider

Die Zuordnung zu einem Leistungserbringer erfolgt durch das Element `care_provider`. Dies erfolgt in Abhängigkeit zum Status des Leistungserbringers (Krankenhaus oder vertragsärztliche Praxis) durch einen unterschiedlichen Aufbau. Die Inhalte des `<care_provider>`-Elements ergeben sich aus Anwendung der Abfrage `vExportZieleXml` (Abschnitt B 2.8.1) der QSDOK-Datenbank.

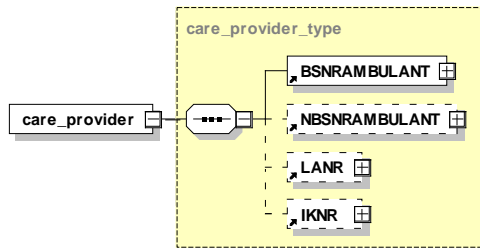


Abbildung 45: Aufbau des Elements care_provider – kollektiv-, selektivvertraglich

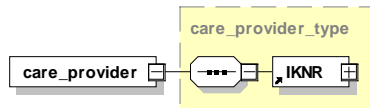


Abbildung 46: Aufbau des Elements care_provider – Krankenhaus

Die Kind-Elemente für die vertragsärztliche Praxis oder das Krankenhaus werden in der folgenden Tabelle beschrieben.

Tabelle 106: Leistungserbringeridentifizierende Daten im kollektiven, selektiven und stationären Bereich

| Kind-Elemente | Beschreibung |
|--|--|
| Leistungserbringeridentifizierende Daten im kollektiven oder selektiven Bereich | |
| <LANR> | Lebenslange Arztnummer – LANR. Für die persönliche Kennzeichnung seiner Leistungen hat jeder Vertragsarzt und -psychotherapeut zum 1. Juli 2008 eine „Lebenslange Arztnummer“ (LANR) erhalten. Diese muss er bei jeder von ihm abgerechneten Leistung und Verordnung angeben. |
| <BSNRAMBULANT> | Betriebsstättennummer ambulant – BSNR Die BSNR identifiziert die Arztpraxis als abrechnende Einheit und ermöglicht die Zuordnung ärztlicher Leistungen zum Ort der Leistungserbringung. Dabei umfasst der Begriff Arztpraxis auch Medizinische Versorgungszentren (MVZ), Institute, Notfallambulanzen sowie Ermächtigungen von am Krankenhaus beschäftigten Ärzten. |
| <NBSNRAMBULANT> | Nebenbetriebsstättennummer – NBSNR |
| <IKNR> ¹³³ | Institutionskennzeichen – IKNRKH. Gemäß §293 SGB V wird bei der Datenübermittlung zwischen den gesetzlichen Krankenkassen und den Leistungserbringern ein Institutionskennzeichen (IK) als eindeutige Identifizierung verwendet. Die IK werden durch die „Sammel- und Vergabestelle Institutionskennzeichen (SVI)“ der Arbeitsgemeinschaft Institutionskennzeichen in Sankt Augustin (SVI, Alte Heerstraße 111, 53757 Sankt Augustin) vergeben und gepflegt. Hier ist das bei der Registrierung für die Qualitätssicherung angegebene IK zu verwenden. |

¹³³ Die IKNRKH ist ein optionales Element im kollektivvertraglichen Bereich. Derzeit ist dieses Element im KV-Bereich nicht zu verwenden.

| Kind-Elemente | Beschreibung |
|--|--|
| Leistungserbringeridentifizierende Daten im Bereich Krankenhaus | |
| <IKNR> | Institutionskennzeichen – IKNRKH. Gemäß §293 SGB V wird bei der Datenübermittlung zwischen den gesetzlichen Krankenkassen und den Leistungserbringern ein Institutionskennzeichen (IK) als eindeutige Identifizierung verwendet. Die IK werden durch die „Sammel- und Vergabestelle Institutionskennzeichen (SVI)“ der Arbeitsgemeinschaft Institutionskennzeichen in Sankt Augustin (SVI, Alte Heerstraße 111, 53757 Sankt Augustin) vergeben und gepflegt. Hier ist das bei der Registrierung für die Qualitätssicherung angegebene IK zu verwenden. |

Element `body/data_container/cases`

Container-Element für eine Liste von gleichartigen Fällen (Vorgängen). „Gleichartig“ meint hier Fälle des gleichen Primärmoduls. Das Element enthält einen oder mehrere Vorgänge¹³⁴.

Für unterschiedliche Module müssen jeweils mehrere `<cases>` angelegt werden. Die Ausweisung eines `<cases>`-Elements für Daten eines bestimmten Primärmoduls erfolgt über dessen Attribut `module`.

Die Minimaldatensätze sind unbedingt im korrespondierenden `<cases>`-Element des ausgelösten Primärmoduls unterzubringen. MDS gelten nicht als unabhängige Module, sondern sind Ersatzmodule eines jeweiligen Primärmoduls.

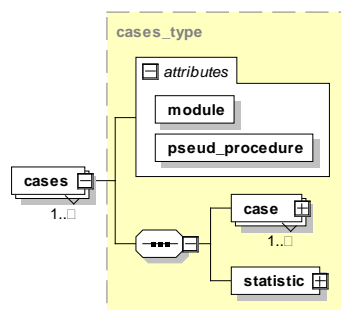


Abbildung 47: Aufbau des Elements `cases`

¹³⁴ Fälle und Vorgänge werden hier als Synonyme verwendet.

Tabelle 107: Attribute des Elements cases

| Name | Type | Use | Beschreibung |
|-----------------|---------------------|----------|---|
| module | enum_module_type | required | Das Erfassungsmodul |
| pseud_procedure | enum_procedure_type | required | Zuordnung des Moduls zu einem Pseudonymisierungsverfahren. Gehört dem Modul kein Pseudonymisierungsverfahren, ist das Attribut auf „undefined“ zu setzen. |

Das Element <cases> enthält das Attribut „pseud_procedure“: „pseud_procedure“ ist eine Verfahrenskennung, die eindeutig zusammengehörende Exportmodule (z. B. 09/1, 09/2 und 09/3) vermerkt und die es der Vertrauensstelle ermöglicht, die PID verfahrensbezogen zu pseudonymisieren.

Tabelle 108: Verfahrenskennung: „pseud_procedure“

| Betrieb | Exportmodul | Verfahrenskennung | XML (Umsetzung) |
|-----------|-----------------------------|-------------------|--|
| QSKH 2015 | HEP | HEP | <cases module="HEP" pseud_procedure="HEP"> |
| | KEP | KEP | <cases module="KEP" pseud_procedure="KEP"> |
| | 09/1 09/2 09/3 | 09/1_09/2_09/3 | <cases module="09/1" pseud_procedure="09/1_09/2_09/3"> <cases module="09/2" pseud_procedure="09/1_09/2_09/3"> <cases module="09/3" pseud_procedure="09/1_09/2_09/3"> |
| Qesü 2016 | PCI_KV PCI_LKG PCI_SV | PCI | <cases module="PCI_KV" pseud_procedure="PCI"> <cases module="PCI_LKG" pseud_procedure="PCI"> <cases module="PCI_SV" pseud_procedure="PCI"> |
| Qesü 2017 | NWIF | NWI | <cases module="NWIF" pseud_procedure="NWI"> |
| QSKH 2018 | 09/4 09/5 09/6 | 09/4_09/5_09/6 | <cases module="09/4" pseud_procedure="09/4_09/5_09/6"> <cases module="09/5" pseud_procedure="09/4_09/5_09/6"> |
| | | | |

| Betrieb | Exportmodul | Verfahrenskennung | XML (Umsetzung) |
|-----------------------------|-------------|-------------------|---|
| | | | <cases module="09/6" pseud_procedure="09/4_09/5_0 9/6"> |
| QSKH 2018 | 16/1 NEO | 16/1_NEO | <cases module="16/1" pseud_procedure="16/1_NEO"> <cases module="NEO" pseud_procedure="16/1_NEO"> |
| DeQS 2019 | CHE | CHE | <cases module="CHE" pseud_procedure="CHE"> |
| Regelbetrieb Testbetrieb | Nicht-PID | undefined | <cases module="PNEU" pseud_procedure="undefined"> |

Element body/data_container/cases/case

Das Element <case> entspricht einem Vorgang und enthält genau einen QS-Datensatz eines Moduls und abhängig vom Verfahren (direkt, indirekt, PID) die patientenidentifizierenden Daten "<patient>".

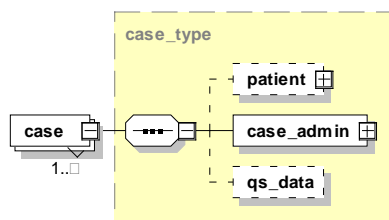


Abbildung 48: Aufbau des Elements case

Element body/data_container/case/case_admin

Das <case_admin>-Element enthält weitere Elemente, die einen Vorgang identifizieren. Zusätzlich legt das Element fest, was mit dem Vorgang geschehen soll. Auf Vorgangsebene (Datensatzebene) werden von jeder Prüfstelle der Status der Prüfung und ggf. die Fehler in das Element <protocol> eingetragen.

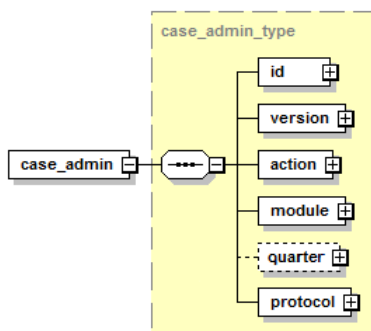


Abbildung 49: Aufbau des Elements case_admin

Im Folgenden werden die einzelnen Kind-Elemente beschrieben.

Tabelle 109: Kind-Elemente des Elements `case_admin`

| Kind-Elemente | Beschreibung |
|-------------------------------|--|
| <code><id></code> | <p>Vorgangsnummer oder Datensatznummer. Diese Nummer kennzeichnet jeden dokumentierten Datensatz eines Dokumentationssystems eindeutig und zwar unabhängig vom angewandten Modul.</p> <p>Im einfachsten Fall könnte also die Vorgangsnummer um 1 erhöht werden, wenn ein neuer Datensatz angelegt wird. Im stationären Bereich, z. B. wenn während eines Krankenhausaufenthalts zwei QS-Dokumentationen eines Falles angelegt werden, müssen auch unterschiedliche Vorgangsnummern vergeben werden.</p> <p>Insbesondere ist es falsch, einfach eine Patientenidentifikationsnummer oder die offizielle Fallnummer zu verwenden bzw. zu pseudonymisieren. Bei der Umsetzung hat der Softwareanbieter weitgehende Freiheit, vorausgesetzt die modulübergreifende Eindeutigkeit der Vorgangsnummer ist gewährleistet.</p> <p>Die Vorgangsnummer darf für die Datenannahmestelle nicht auf Personen zu beziehen sein.</p> <p>Da im ambulanten Bereich keine Registrierung der Softwareinstanzen vorgesehen ist, muss bezogen auf einen Leistungserbringer (BSNRAMBULANT) die Vorgangsnummer immer eindeutig sein. Hierfür ist die Vorgangsnummer als GUID von der QS-Software zu generieren.</p> |
| <code><version></code> | <p>Enthält eine Versionsnummer des Datensatzes. Sie gibt an, die wievielte Version des Datensatzes übertragen wird.</p> <p>In der Regel wird die Versionsnummer 1 lauten. D.h., dass der nach dem ersten Dokumentationsabschluss freigegebene Datensatz übertragen wird. Muss ein korrigierter Datensatz erneut eingesandt werden, so muss die Versionsnummer vom dokumentierenden System um 1 erhöht werden. Die neue Version des Datensatzes wird bei der Entgegennahme geprüft und überschreibt bei Korrektheit die alte Version des Datensatzes.</p> <p>Wenn die Datenannahmestelle einen Datensatz mit derselben Versionsnummer ein zweites Mal erhält, so wird dieser zurückgewiesen.</p> |
| <code><module></code> | <p>Verfahrensabkürzung. Hier ist zu beachten, dass dieser Wert identisch zu den Attributwerten im Element <code><cases></code> und <code><qs_data></code> sein muss.</p> <p>Wenn die Datenannahmestelle unterschiedliche Modulbezeichnungen innerhalb eines <code><cases></code>-Elements erhält, wird die ganze Datenlieferung zurückgewiesen.</p> |
| <code><quarter></code> | <p>Dieses Element ist optional und ermöglicht eine fallbezogene Quartalsangabe. Hiermit werden DAS (KV), welche keinen Zugriff auf die QS-Daten haben in die Lage versetzt quartalsbezogene Ist-Werte zu ermitteln.</p> |
| <code><protocol></code> | <p>Protokoll auf Vorgangsebene</p> |
| <code><action></code> | <p>Definiert die gewünschte Aktion, kann „create“, „update“ und „delete“ sein.</p> |

| Kind-Elemente | Beschreibung |
|---------------|---|
| | <p>„create“ ist beim ersten Export des Datensatzes zu verwenden, weitere Exporte des Datensatzes werden mit „update“ geliefert.</p> <p>Da nicht alle Datenexporte auch an die Datenannahmestelle verschickt werden (z. B. Testexporte usw.), muss die Datenannahmestelle „update“ und „create“ gleichbehandeln, wenn der erhaltene Datensatz nicht bereits im Datenpool vorhanden ist.</p> <p>Um den Datensatz zu stornieren, muss <action> auf „delete“ gesetzt werden.</p> <p>Die Datenannahmestelle wird dadurch veranlasst, den betreffenden Datensatz einschließlich aller Vorversionen und Teildatensätze als „storniert“ zu kennzeichnen. Der Stornovorgang wird in der Datenbestätigung protokolliert.</p> <p>Der zu stornierende Datensatz muss ebenfalls eine hochgezählte/fortgeschriebene Versionsnummer enthalten, um die Stornierung unabhängig von der Reihenfolge der Verarbeitung von Datensätzen sicherzustellen. Ein Storno mit einer bereits verwendeten Versionsnummer wird zurückgewiesen (Bestätigungsstatus ERROR, Fehlerart DOPPELT). Ein Stornoversuch eines noch nicht übermittelten Datensatzes wird ebenfalls zurückgewiesen.</p> <p>Zur Stornierung eines Datensatzes (Vorgang) genügt der Export der entsprechenden administrativen Daten <case>/<case_admin>. Sowohl die PID (<patient>) als auch die QS-Daten (<qs_data>) des zu stornierenden Datensatzes sind nicht erneut zu übermitteln.</p> |

Element patient (PID-Module)

Das Element enthält die patientenidentifizierenden Daten des übergeordneten Vorgangs. Das Kind-Element von <patient> ist entweder das Element <pid> oder das Element <perineo_pid>. Die Inhalte des <patient>-Elements ergeben sich aus Anwendung der Abfrage vExportZieleXml (Abschnitt B 2.8.1) in der QS-DOK-Datenbank. Das Attribut twodigitik ist verpflichtend und muss die ersten 2 Stellen des Institutionskennzeichens der Krankenkasse enthalten. Das Attribut ist nicht von der Verschlüsselung betroffen.

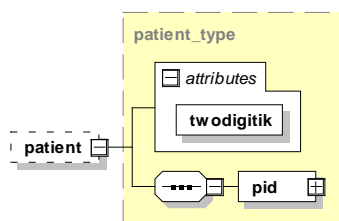


Abbildung 50: Aufbau des Elements patient



Achtung Umgang mit PID-Daten

QSKH-Verfahren: Es dürfen keine PID von nicht gesetzlich Versicherten exportiert werden, nur PID von GKV-Versicherten sind zulässig. QS-Daten (<qs_data>) sind davon nicht betroffen und müssen weiterhin unabhängig von dem Versicherungsstatus des Patienten exportiert werden.

Bei nicht gesetzlich Versicherten sind sämtliche Informationen im <patient>-Element leer. In diesem Fall ist das gesamte <patient>-Element wegzulassen.

Qesü- und DeQS-Verfahren: Für die derzeitigen Qesü- und DeQS-Verfahren werden sowohl die PID- als auch die QS-Daten von nicht GKV-Versicherten bereits bei der Auslösung ausgeschlossen. Grundsätzlich können gemäß Qesü- oder DeQS-RL auch Verfahren entwickelt werden, die auch nicht GKV-Versicherte einschließen.

richtlinienübergreifend gilt: Bei einer Stornierung ist das gesamte <patient>-Element wegzulassen.

Element patient_type/pid

Das Element <pid> dient dazu, die tatsächlichen PID aufzunehmen.

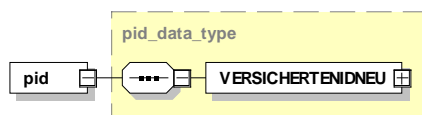


Abbildung 51: Aufbau des Elements pid

Diese PID bestehen aus dem Element <VERSICHERTENIDNEU>, welches die eGK-Versichertennummer enthält.

Element patient/perineo_pid

Das Element <perineo_pid> dient dazu, die für die Verknüpfung der Module Geburtshilfe (Modul 16/1) und Neonatologie (Modul NEO) des QS-Verfahrens Perinatalmedizin benötigten PID aufzunehmen.

Diese PID bestehen aus den Elementen <vorname_mutter> für den Vornamen der Mutter, <nachname_mutter> für den Nachnamen der Mutter, <GEBDATUMK> für das Geburtsdatum des Kindes sowie optional (Modul NEO) <VERSICHERTENIDNEUK> für die eGK-Versichertennummer des Kindes.

Element case_Datentyp/case_admin/protocol

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

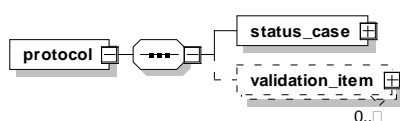


Abbildung 52: Aufbau des Elements case_admin/protocol

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 Vorgangsebene (Datensatzebene) erfolgen.
- Für erfolgreiche Prüfergebnisse (`status="OK"`) wird nicht explizit das Element `<validation_item>` erstellt. Für die Übermittlung des Status des Datensatzes dient weiterhin der implizite Wert des Elements `<status_case>` (`<status_case V="OK">`).
- `<status_case>` beinhaltet das schlechteste Ergebnis aller Prüfungen eines Datensatzes.
- Das Protokoll auf Vorgangsebene 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 `body/data_container/cases/statistic`

Das Element `<statistic>` dient dazu, Statistiken über die Datenlieferung des Absenders und über deren Verarbeitung durch die Datenannahmestelle aufzunehmen.

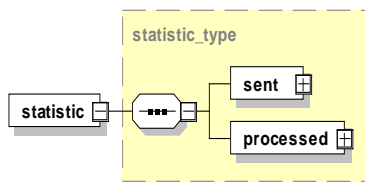


Abbildung 53: Aufbau des Elements `statistic`

Es ist nach der Prüfung bei der Datenannahmestelle ein Teil des Rückprotokolls und besteht aus ähnlichen Kind-Elementen.

Tabelle 110: Kind-Element des Elements `statistic`

| Kind-Element | Beschreibung |
|--------------------------------|---|
| <code><sent></code> | Statistik über die von dem Datenlieferanten exportierten Datensätze. Es muss daher vom Datenlieferanten selbst ausgefüllt werden. |
| <code><processed></code> | Hat dieselbe Struktur wie <code><sent></code> und enthält das Ergebnis der Verarbeitung durch die Datenannahmestelle. |

Element `statistic/sent`

Das Element nimmt Statistiken über die von dem Datenlieferanten exportierten Datensätze auf und muss vom Datenlieferanten selbst ausgefüllt werden.

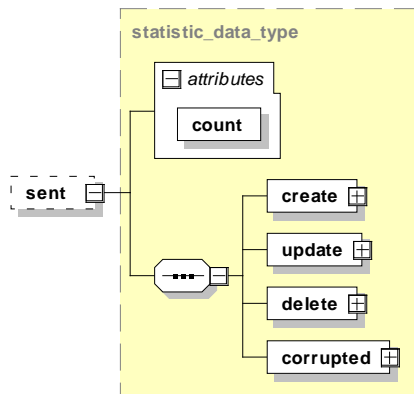


Abbildung 54: Aufbau des Elements `sent`

Es besteht aus vier Elementen und einem Attribut.

Tabelle 111: Attribut des Elements `sent`

| Name | Type | Use | Beschreibung |
|-------|---------------------------|----------|---|
| count | non_negative_integer_type | required | Gesamtzahl von Vorgängen (Ganze Zahl ≥ 0): Summe von (<code><create></code> + <code><update></code> + <code><delete></code> + <code><corrupted></code>) |

Tabelle 112: Kind-Elemente des Elements `statistic/sent`

| Kind-Element | Beschreibung |
|--------------------------------|--|
| <code><create></code> | Anzahl der Datensätze, die neu importiert werden sollen. |
| <code><update></code> | Anzahl der Datensätze, die aktualisiert werden sollen (z. B. nach einer Korrektur). |
| <code><delete></code> | Anzahl der Datensätze, die von der Datenannahmestelle/Bundesauswertungsstelle storniert werden müssen. |
| <code><corrupted></code> | Anzahl der Datensätze, die fehlerhaft sind. Der Datenabsender trägt hier „0“ ein. |

Element `statistic/processed`

Das Element `<processed>` hat dieselbe Struktur wie das Element `<sent>` mit dem Unterschied, dass der Datenempfänger nach der Prüfung der Exportdatei in das Element `<processed>` eintragen soll, wie viele Datensätze er tatsächlich neu importiert, überschrieben und storniert hat und ggf. wie viele Datensätze fehlerhaft sind. Außerdem soll er im Attribut `count` des Elements `<processed>` die Gesamtsumme eintragen.

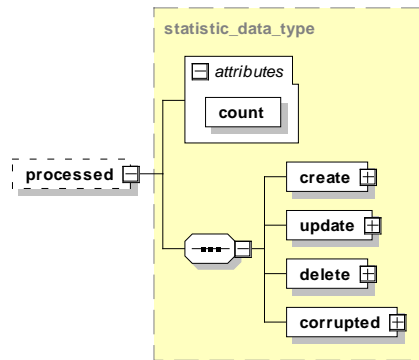


Abbildung 55: Aufbau des Elements processed

<processed> ist vom Datenlieferanten anzulegen und besteht aus vier Elementen und einem Attribut.

Tabelle 113: Attribute des Elements statistic

| Name | Type | Use | Beschreibung |
|-------|---------------------------|----------|---|
| count | non_negative_integer_type | required | Gesamtzahl von Vorgängen (Datensätzen): Summe von (<create>+<update>+<delete>+<corrupted>) |

Tabelle 114: Kind-Elemente des Elements statistic/processed

| Kind-Element | Beschreibung |
|--------------|--|
| <create> | Anzahl der Datensätze, die der Datenempfänger nach der Prüfung tatsächlich neu importieren konnte. |
| <update> | Anzahl der Datensätze, die der Datenempfänger nach der Prüfung tatsächlich aktualisieren konnte. |
| <delete> | Anzahl der Datensätze, die der Datenempfänger tatsächlich stornieren konnte. |
| <corrupted> | Anzahl der Datensätze, die wegen Mängeln nicht entsprechend der Absicht des Datenlieferanten in den Datenpool übernommen werden konnten. |

Alle Werte müssen vom Leistungserbringer mit der Zahl „0“ vorbelegt werden.

Zusammenfassend kann man die Angaben im Element <sent> als Absichtserklärung der Datenlieferung betrachten. Die Angaben im Element <processed> sind als Ergebnis der Verarbeitung in Bezug auf den Zieldatenpool zu verstehen und werden daher nur von der Datenannahmestelle gesetzt.

Element `qs_data`

Das `<qs_data>`-Element ist ein Container für die QS-Daten, die verfahrensspezifisch sind.

Der Datentyp der konkreten `<qs_data>`-Instanz ist jeweils als Attribut `"xsi:type"` bei der Dokumenterstellung zu definieren.

```
<qs_data xsi:type="qs_data_17n2_type" module="17/2">
```

Wie bereits oben erwähnt, würden ggf. vorhandene MDS (Minimaldatensätze) unter dem zugehörigen Modul definiert und eingehängt.

```
<qs_data xsi:type="qs_data_mds_type" module="17/2">
```

Die Abbildung zeigt als Beispiel, wie ein Minimaldatensatz zum Primärmodul NEO exportiert wird.

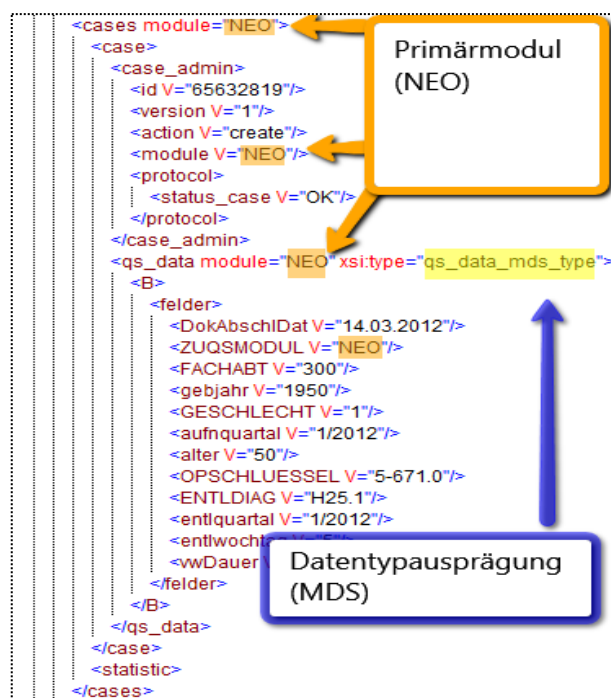


Abbildung 56: Integration des MDS in das Primärmodul

Minimaldatensatz des Hauptmoduls Neonatologie (NEO)

```
<qs_data xsi:type="qs_data_mds_type" module="NEO">
```

Die genaue Struktur eines Moduls ist der Access-Datenbank für die QS-Software und dem entsprechenden XML-Schema zu entnehmen.

Der Aufbau des XML-Elements `<qs_data>` ist variabel und abhängig von der Struktur des jeweiligen Erfassungsmoduls. Die genaue Struktur eines Moduls (nach dem Export) ist von der Spezifikationsdatenbank vorgegeben. Im Allgemeinen gilt Folgendes:

- Jedes Modul hat immer einen einzigen Basisdatensatz.
- Komplexe Module können zusätzlich mehrere Teildatensätze (Bögen) enthalten, die sich hierarchisch anordnen lassen (Basisbogen und Kinderbögen).

Abbildung 58: Ausprägungen des `qs_data`-Elements (Erfassungsmodule)

4.5 Aufbau XML-Dokument – Sollstatistik

Die XML-Struktur (`header/body`) der Sollstatistik hat denselben Aufbau wie die QS-Dokumentation. Die spezifischen Elemente der Sollstatistik sind wie ein QS-Modul (Ausprägung) festgelegt und müssen dementsprechend in das `<qs_data>`-Element integriert werden.

Folgende Tabelle zeigt, welche unterschiedlichen Ausprägungen definiert sind.

Tabelle 115: Schemata der Sollstatistik

| Schema | Beschreibung |
|-------------------------------|--|
| <code>soll_kv_type.xsd</code> | Inhalt des <code>qs_data</code> -Elements für die KV |
| <code>soll_sv_type.xsd</code> | Inhalt des <code>qs_data</code> -Elements für die DAS-SV |

Der Datentyp der konkreten `<qs_data>`-Instanz ist jeweils als Attribut `"xsi:type"` bei der Dokumenterstellung zu definieren.

Beispiel:

Datentypen für Sollstatistik

KV-Bereich: `<qs_data xsi:type="soll_kv_type" module="PCI_KV">`

SV-Bereich: `<qs_data xsi:type="soll_sv_type" module="PCI_SV">`

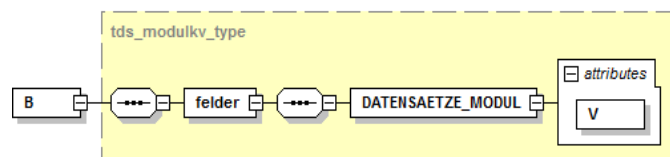


Abbildung 59: Sollstatistik-Modul

Das Element `DATENSAETZE_MODUL` nimmt die Anzahl der SOLL-Fälle für das entsprechende QS-Modul auf.



Hinweise

Die XML-Struktur ist ausschließlich für die KVen und DAS-SV für die Erstellung der Sollstatistik relevant. Für die LQSen/LKGen gilt für die Datenübermittlung weiterhin das CSV-Format.

Die KVen tragen in das Element `DATENSAETZE_MODUL` das Ergebnis der Vollständigkeitsprüfung ein.

5 Tools

Das vorliegende Kapitel beschreibt Spezifikationskomponenten, die als Hilfsprogramme Prozesse in der Qualitätssicherung unterstützen. Die Hilfsprogramme basieren auf der Programmiersprache Java. Dementsprechend ist ein Abschnitt enthalten, der die Installation einer Java-Laufzeit-Umgebung (JRE) beschreibt. Die Tools selbst umfassen derzeit ein Verschlüsselungsmodul und ein Datenprüfprogramm. Das LE-Pseudonymisierungsprogramm ist im in der Spezifikationskomponente enthaltenen Anwenderhandbuch und in der dazugehörigen Datenflussspezifikation beschrieben.

5.1 Java-Installation

Zur Ausführung von Java-Programmen wird das Java Runtime Environment (JRE) zur Verfügung gestellt. Das JRE kann „online“ und „offline“ installiert werden.

Offizielle Installationspakete können von der Website <http://www.java.com/de/download/manual.jsp> heruntergeladen werden.

Anleitungen zur Installation von Java unter verschiedenen Betriebssystemen sind auf der folgenden Website detailliert beschrieben:

https://www.java.com/de/download/help/download_options.xml

Aufruf

Der Aufruf des Java-Programms erfolgt über die Konsole des Betriebssystems. Die Rückmeldungen des Programms erfolgen ebenfalls über die Konsole und können dort abgefangen werden.

Die Ein- und Ausgabe der Konsole lässt sich in der Regel in jedes Programm integrieren. Ein Exitcode von 0 ohne Ausgabe entspricht einer fehlerfreien Verarbeitung des Programms. Bei Fehlern gibt es einen Exitcode von 1 und in der Regel eine Konsolen- bzw. eine Error-Ausgabe.

Aufruf: `java -jar Beispielprogramm.jar -g -o Dateiname 2> error.txt`

- `Java`: Aufruf der JVM mit dem Befehl `java`
- `-jar`: Parameter `-jar`, mit dem der JVM mitgeteilt wird, dass ein Java-Archiv aufgerufen wird
- `Beispielprogramm.jar`: Benennung des Archivs, mit vollem Pfad, wenn es nicht im aktuellen Verzeichnis liegt
- `-g -o`: Parameter, die an das Java Programm übergeben werden sollen
- `2>`: Mit `2>` Ziel der Error-Ausgabe spezifizieren
- `error.txt`: Dateiname der Error-Ausgabe – hier die Datei „error.txt“

Bei längeren Pfaden oder Dateinamen, die ggf. Leerzeichen oder andere Zeichen enthalten, sind diese in Anführungszeichen zu setzen. Dies gilt sowohl für Paketnamen als auch für Parameter-Dateien.

5.2 Datenprüfprogramm

Das Datenprüfprogramm wird vor dem Hintergrund der Anforderungen der Richtlinie zur einrichtungs- und sektorenübergreifenden Qualitätssicherung (Qesü-RL) und der Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) bereitgestellt, um nicht nur beim Leistungserbringer, sondern auch bei den Datenannahmestellen eine Datenprüfung durchzuführen. Das Datenprüfprogramm bezieht die Plausibilitätsregeln direkt aus der Spezifikation und testet Daten vor deren Verschlüsselung.

5.2.1 Umfang der Prüfungen

Es werden zwei wesentliche Bereiche mit diesem Programm geprüft:

- Schema-Konformität (Struktur)
- Die XML-Datei wird dabei auf Konformität mit dem zugrunde liegenden Schema (XSD) überprüft.
- Regel-Konformität (Inhalte)
- Die XML-Datei wird dabei auf Einhaltung der Regeln (XSLT) überprüft.

Die eigentliche Prüfung erfolgt in der XML-Export-Datei. Deren Struktur ist der entsprechenden Dokumentation bzw. dem gültigen XML-Schema zu entnehmen.

Die inhaltliche Prüfung selbst erfolgt über ein XSLT-Stylesheet und einen XSLT-Prozessor. Das Datenprüfprogramm verwendet die freie Version eines XSLT-Version 2.0-kompatiblen Programms (XSLT2). Die Einbindung von XSLT-Stylesheet und XSLT-Prozessor erfolgt über ein Java-Programm. Prinzipiell kann jeder XSLT2-fähige XSLT-Prozessor für die Prüfung auf dieser Grundlage verwendet werden. Das Datenprüfprogramm stellt eine Referenzimplementierung dar.

5.2.2 Ausgangskontrolle vor Versand

Eine Schemavalidierung der Ausgangsdateien vor dem Versand der XML-Daten wird aus folgenden Gründen als notwendig erachtet:

- Sicherstellung der Datenintegrität nach der Verarbeitung der Daten
- Frühe Feststellung von Fehlerquellen in der eigenen Datenverarbeitung
- Entlastung des nachfolgenden Datenservices von nicht validen Daten
- Vermeidung des Versands von Daten, die gegen den Datenschutz verstoßen

Aus diesen Gründen wurde das Datenprüfprogramm um die Möglichkeit erweitert die XML-Dateien auch nach der Verschlüsselung der XML-Elemente auf Schemavalidität zu prüfen.

Der Leistungserbringer verwendet abhängig vom Datenfluss die Schemata zur Übergabe an die Datenannahmestelle (LE-DAS) unter:

```
interface_LE_DAS\
```

Die Datenannahmestelle verwendet das folgende Schema zur Übergabe an die Vertrauensstelle (DAS-VST):

```
interface_DAS_VST\interface_DAS_VST.xsd
```

Diese Schemata können einfach in den Config-Dateien im Parameter `<xsd_path>` entsprechend angegeben werden. Dazu müssen nur mehrere Config-Dateien verwendet werden (je eine pro Schema). Des Weiteren besteht die Möglichkeit, beim Programmstart mit dem Parameter `-xsd-path` das Schema zu übergeben.

Diese Config-Dateien können entweder in verschiedenen Verzeichnissen abgelegt oder mit dem Parameter `-c` oder `--config` beim Start des Datenprüfprogramms über die Konsole angegeben werden. Da in diesem Fall keine inhaltliche Prüfung erfolgen soll (nur XSD, kein XSLT) muss zusätzlich der Parameter `--no-spez-val` angegeben werden:

```
java -jar datenpruefprogramm-4.2.0-jar-with-dependencies.jar -c
config_schema.xml --no-spez-val
```

Diese notwendige Ausgangskontrolle können die Softwareanbieter (bzw. die Leistungserbringer) und die Datenannahmestellen unabhängig von dem Datenprüfprogramm realisieren, indem sie eine Schemavalidierung gegen die o.g. Schemata durchführen.

Für eine Schemavalidierung gibt es zahlreiche Tools und Bibliotheken für alle bekannten Programmiersprachen (<http://www.w3.org/XML/Schema>).

5.2.3 Programmaufruf

Das Datenprüfprogramm erzeugt eine Ausgabe/Output-Datei, die der Eingabe/Input-Datei entspricht, die jedoch um die Ergebnisse der Prüfungen erweitert wird. Die durchgeführten Prüfungen entsprechen einer Prüfung auf Dokumenten- und Vorgangsebene (Datensatzebene).

Das Datenprüfprogramm kann mehrere Dateien in einem Aufruf prüfen. Daher gibt es entsprechende Ordner für die Ein- und Ausgabedateien. Sollten diese Ordner nach der letzten Prüfung nicht geleert worden sein, so werden die Dateien des Eingabeordners erneut geprüft und der Ausgabeordner wird parallel mit Datum und Uhrzeit gesichert und ein neues leeres Ausgabeverzeichnis angelegt.

Die Prüfungen umfassen die Schemaprüfung und die Überprüfung der Feldinhalte (auch feldübergreifend).

Parameter `-c` oder `-config`

Die Steuerung der Funktionen erfolgt über eine Konfigurationsdatei, deren Dateipfad dem Programm beim Programmstart mit dem Parameter `-c` oder `--config` beim Programmaufruf übergeben werden kann.

```
java -jar datenpruefprogramm-4.2.0-jar-with-dependencies.jar -c
C:/konfiguration/config.xml
```

Wenn keine Konfigurationsdatei übergeben wird, wird die Datei `./config.xml` gesucht und geladen. Wenn diese Datei nicht gefunden wird, wird eine Standard-`config.xml`-Datei im Start-Order angelegt.

Parameter --no-spez-val

Mit diesem Parameter wird das Prüfskript ausgeschaltet.

```
java -jar datenpruefprogramm-4.2.0-jar-with-dependencies.jar --no-spez-val
```

Parameter --no-schema-val

Mit diesem Parameter wird die Schemaprüfung ausgeschaltet.

```
java -jar datenpruefprogramm-4.2.0-jar-with-dependencies.jar --no-schema-val
```

Batch-Dateien

Beim Datenprüfprogramm werden beispielhafte Batchdateien mitgeliefert:

- datenpruefprogramm_schema.bat
- Hier wird eine reine Schemaprüfung (Nur XSD) anhand einer Konfigurationsdatei „config_schema.xml“ durchgeführt.
- datenpruefprogramm_<Datenfluss>_<Verfahrensart>.bat
- Hier wird sowohl die Schemaprüfung (XSD) als auch die Inhaltliche Prüfung (XSLT) anhand einer Konfigurationsdatei „config_<Datenfluss>_<Verfahrensart>.xml“ durchgeführt.

Beispiel einer Konfigurationsdatei:

```
<?xml version= "1.0 " encoding= "UTF-8 " standalone= "no ">
<config>
<provider>
<address>12345 Musterdorf</address>
<email>bernd.mustermann@musterfirma.de</email>
<fax>0123/456798</fax>
<function>Softwarehersteller</function>
<name>Mustermann</name>
<phone>0123/456789</phone>
<registration></registration>
</provider>
<gui>>false</gui>
<input_path recursive= "true">C:\pruefmodul\input</input_path>
<output_path>C:\pruefmodul\output</output_path>
<xsd_path>C:\pruefmodul\xsd\interface_LE_WEICH</xsd_path>
<xsl_path>C:\pruefmodul\xsl</xsl_path>
</config>
```

Die Konfigurationsdatei besteht aus den folgenden Bereichen:

Provider (Softwarehersteller)

Im Element `<provider>` werden Daten benötigt, aus denen hervorgeht, wer das Prüfmodul einbindet und ausführt. In der Regel ist dies der Softwarehersteller. Zu beachten ist, dass die Auswahlmöglichkeit im Element `<function>` auf Softwarehersteller eingeschränkt ist. Die Elemente `<fax>`, `<phone>`, und `<address>` sind optional, die anderen sind Pflichtelemente.



Achtung

In den Elementen `<fax>`, `<phone>`, und `<address>` dürfen auf keinen Fall die Angaben des Leistungserbringers eingetragen werden!

GUI (Konsole)

Für ein vereinfachtes Debugging gibt es die Möglichkeit, eine Konsole mit detaillierten Programmausgaben während der Verarbeitung über das Element `<gui>` und den Wert `true` zu öffnen. Der Standard-Wert ist `false`.

Input_Path (Eingabeverzeichnis) – überschreibbar mit Parameter `-input`

Im Element `<input_path>` kann der Eingabeordner für die zu überprüfenden Exportdateien festgelegt werden. Das Element ist optional. Ohne diesen Parameter ist der Ordner `<arbeitsverzeichnis>\input\` der Standard-Eingabe-Ordner. Es werden alle Dateien mit der Dateierweiterung `.xml` verarbeitet. Wenn das Attribut `recursive` auf `true` steht, werden auch alle entsprechenden Dateien in Unterordnern berücksichtigt. Die Standard-Einstellung von `recursive` ist `false`.

Output_Path (Ausgabeverzeichnis) – überschreibbar mit Parameter `-output`

Im Element `<output_path>` kann der Ausgabeordner festgelegt werden. Das Element ist optional. Ohne diesen Parameter ist der Standard-Ausgabeordner `<arbeitsverzeichnis>\output\`. Der Dateiname der Ausgabedatei ist dabei gleich dem der Eingabedatei. Ein ggf. nicht vorhandener Ordner wird angelegt.

XSD_Path (Schemaordner) – überschreibbar mit Parameter `--xsd-path`

Im Element `<xsd_path>` wird der Schemapfad gesetzt. Es wird dabei entweder auf ein Verzeichnis gezeigt, in dem genau eine Schemadatei erwartet wird, oder es wird direkt auf eine `xsd`-Datei gezeigt. Das Element ist optional. Ohne diesen Parameter wird das Schema im Verzeichnis `<arbeitsverzeichnis>\xsd\interface_LE_WEICH` gesucht. Eine Spezifikationskonforme Protokollierung kann vom Datenprüfprogramm sichergestellt werden, wenn die weiche Schemavariante verwendet wird.

Um nach einer Schemavalidierung der XML-Dateien, die Weiterverarbeitung und dementsprechend die spezifikationskonforme Protokollierung auf Datensatzebene weiterhin zu ermöglichen, wurden neben der harten Schemavariante ein weiches Schema für die Schnittstellen LE und DAS eingeführt. Diese weiche Variante wird ausschließlich mit dem Datenprüfprogramm verwendet (Abbildung 60). Eine detaillierte Übersicht über die Anwendung weicher Schemata mit dem Datenprüfprogramm finden sie im Abschnitt B 4.2.



Hinweis

Neben den Leistungserbringern sind nur die LQS/LKG und BAS (DAS) die Stellen, die QS-Daten gemäß den G-BA-RL entschlüsseln dürfen und die weiche Variante benötigen.

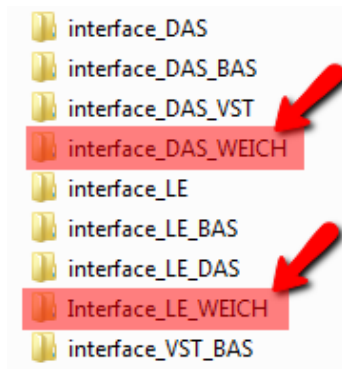


Abbildung 60: Weiche Schemavariante für das DPP

XSL_Path (XSLT-Stylesheet-Ordner) – überschreibbar mit Parameter --xsl-path

Im Element `<xsl_path>` kann der Quellordner für die XSLT-Stylesheets festgelegt werden. Das Element ist optional. Ohne diesen Parameter wird im Standard XSL-Ordner `<arbeitsverzeichnis>\xsl\` nach den XSLT-Stylesheets gesucht.

Mehrere Konfigurationsdateien können für dasselbe Datenprüfprogramm angelegt werden, um beispielsweise Dateien unterschiedlicher Spezifikationen zu validieren oder die Durchführung einer Eingangs- bzw. einer Ausgangskontrolle jeweils vor der Entschlüsselung und nach der Verschlüsselung zu ermöglichen.

5.2.4 Verzeichnisstruktur

Für das korrekte Funktionieren des Prüfprogramms ist neben den erforderlichen Dateien auch eine korrekte Verzeichnisstruktur notwendig.

In der config.xml wird der `<xsl_path>` definiert.

Wenn der Parameter `<xsl_path>` auf ein Verzeichnis zeigt, muss in diesem Verzeichnis eine Stylesheet-Datei der folgenden Art vorliegen:

```
xsl\<<Spezifikationsversion>.aqxsl
```

Kompiliertes Haupt-XSL-Stylesheet, das die Prüfung entsprechend der Spezifikation durchführt. Es enthält alle Tests auf Regeln und Wertebereichsverletzungen. Ansonsten kann der Parameter `<xsl_path>` auf eine beliebige Stylesheet-Datei verweisen.

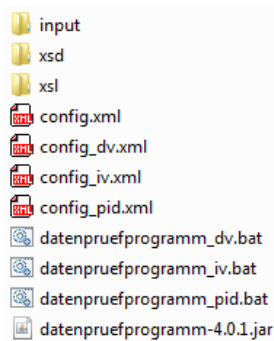


Abbildung 61: Beispiel einer typischen Verzeichnisstruktur

Die Verzeichnisstruktur kann über eine config.xml-Datei modifiziert werden. Ohne diese config.xml wird eine Standardkonfiguration angenommen, die das Prüfprogramm selbst in einer config.xml neu schreibt.

In Abbildung 61 ist eine typische Verzeichnisstruktur mit der config-Datei und dem Programm abgebildet. Unterhalb dieses Grundordners befinden sich die Verzeichnisse, die schon in der Beschreibung der Konfiguration angesprochen wurden.

5.2.5 Ausgabe

Nach dem Prüflauf wird ein Ordner <output> erzeugt, der die geprüften Dateien und deren Datenflussprotokolle beinhaltet.

Geprüfte Dateien

Im Order <output>/files liegen nun die geprüften Quell-Dateien, welche um das Ergebnis der Prüfung erweitert worden sind. Jeder Fall wird innerhalb der XML-Datei geprüft und in der XML-Struktur abgelegt. Zudem wird ein neuer Eintrag als validation_provider erzeugt.

Protokolle

Im Order <output>/protocol liegen die Datenflussprotokolle, die den entsprechenden Dateien im Ordner <output>/files entsprechen, in denen QS-Daten und Patientendaten entfernt wurden.

HTML-Protokolle

Im Order <output>/html liegt eine index.html, in der auf vereinfachte Sichten der im Ordner <output>/protocol erstellten Protokollen verwiesen wird.

| | | |
|--|--------------------------------------|---------|
| 5d29aac7-d480-4377-ad2b-070596b27dda.xml | 5d29aac7-d480-4377-ad2b-070596b27dda | OK |
| 65238192-1879-4fa2-84e6-a4323f52522c.xml | 65238192-1879-4fa2-84e6-a4323f52522c | OK |
| 8e5e8197-5478-46b5-8fd3-cb527d0141a2.xml | 8e5e8197-5478-46b5-8fd3-cb527d0141a2 | OK |
| 9afe7686-7f73-43f8-a5ce-27d5139064f4.xml | 9afe7686-7f73-43f8-a5ce-27d5139064f4 | OK |
| 9d6359cf-2b70-4706-8670-5dd51b67306f.xml | 9d6359cf-2b70-4706-8670-5dd51b67306f | WARNING |

Abbildung 62: Beispiel für eine index.html Datei im Ordner <output>/html

Da für die Rückprotokollierung nur die Übertragung der unter `<output>/files` abgelegten Datei spezifiziert ist, wird die Darstellung außerhalb von QS-Programmen beim Leistungserbringer durch ein eigenes Stylesheet ermöglicht, das sich an der Darstellung des Datenprüfprogramms orientiert. Die Dokumentation in Bezug auf dieses Stylesheet und dessen Einbindung ist im Abschnitt A „Lokale Transformation (Empfehlung)“ zu finden.

5.2.6 Grafische Oberfläche

Wird der Parameter GUI in der Konfigurationsdatei auf „true“ gesetzt, wird das Datenprüfprogramm mit einer einfachen grafischen Oberfläche gestartet.

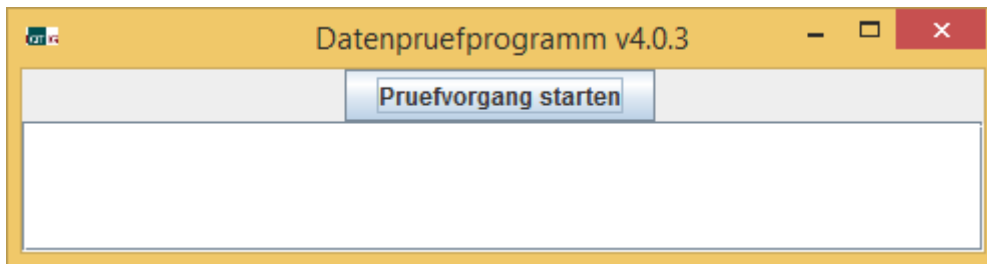


Abbildung 63: Grafische Oberfläche des Datenprüfprogramms

Bei der grafischen Oberfläche muss zum Starten die Schaltfläche „Pruefvorgang starten“ gedrückt werden.

Die grafische Oberfläche zeigt die Ausgabe im Fensterbereich direkt an. Am Inhalt des Ausgabe-Ordners ändert sich nichts; beide Laufvarianten (grafische Ausgabe oder Konsolenausgabe) erzeugen den gleichen Output.

5.2.7 Programmierschnittstelle – API

Sämtliche Funktionen des Datenprüfprogramms können über eine Programmierschnittstelle (API) aufgerufen und direkt in einem Java-Umfeld verwendet werden.

5.3 Verschlüsselungspaket

Das IQTIG stellt ein Ver- und Entschlüsselungspaket für die Anwendung in der externen Qualitätssicherung entsprechend QSKH-RL, Qesü-RL und DeQS-RL zur Verfügung. Die Ver- und Entschlüsselungsalgorithmen sind entsprechend der im Abschnitt „Gesicherte QS-Datenübertragung“ beschriebenen W3C-Standards¹³⁵ implementiert. Das bedeutet, dass an jeder beliebigen Stelle im Workflow auch jede andere Implementierung, die sich an diese Standards hält, verwendet werden kann.

Das Verschlüsselungspaket besteht aus dem TPACKER für die Transportverschlüsselung und dem XPACKER für die XML-Verschlüsselung. Für alle Anwender, die händisch die Verschlüsselung durchführen müssen, stellt der GPACKER mit seiner grafischen Oberfläche eine interaktive Alternative zur Verwendung der Programme XPACKER und TPACKER dar.

¹³⁵ RSA2048: <http://www.w3.org/TR/xmlenc-core/>

AES128: <http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf>

5.3.1 XPack – XML-Verschlüsselung

Das Programm XPack.jar dient zur Ver- und Entschlüsselung einzelner XML-Elemente innerhalb einer XML-Datei. Bevor diese verschlüsselt werden, wird jedes einzelne Element mit „base64“ komprimiert bzw. „gezippt“ und beim Entschlüsseln parallel wieder „entpackt“.

Eine weitere Funktion des Programms dient zur Generierung eines asymmetrischen Schlüssel-paares (privater und öffentlicher Schlüssel).

Die Verschlüsselung wird mit dem hybriden Verfahren (Abschnitt A „Die Verschlüsselung“ durchgeführt:

- Die Verschlüsselung der Daten erfolgt mit einem AES-128bit-Schlüssel im CBC-Mode (aes128-cbc).
- Die Verschlüsselung des symmetrischen Schlüssels erfolgt mit einem RSA-Schlüssel im OAEP-Mode (rsa-oaep-mgf1p).

Weitere technische Details beschreibt der W3C-Standard „XML Encryption Syntax and Processing“.¹³⁶

Die folgende Abbildung zeigt ein XML-Dokument nach der Verschlüsselung eines XML-Elements (QS-Daten):

```

<encryption>
  <ds:KeyInfo Id="Pub_key_Bundesauswertungsstelle.pub-qs_data-2.2.0">
    <xenc:EncryptedKey Id="qs_data" xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">
      <xenc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#rsa-oaep-mgf1p"/>
      <xenc:CipherData>
        <xenc:CipherValue>dybYkkepEipN48IE0mbj28aw83HDyMqJQxvT3nSof34SdLHLpZ+A==</xenc:CipherValue>
      </xenc:CipherData>
    </xenc:EncryptedKey>
  </ds:KeyInfo>
</encryption>
</header>
<body>
  <data_container>
    <care_provider>
    <cases module="KAT_FU_A" pseud_procedure="KAT">
      <case>
        <patient>
        <case_admin>
        <qs_data module="KAT_FU_A" xsi:type="qs_data_kat_fu_a_type">
          <xenc:EncryptedData Type="http://www.w3.org/2001/04/xmlenc#Content" xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">
            <xenc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#aes128-cbc"/>
            <xenc:CipherData>
              <xenc:CipherValue>xEO1RE59SeMKaRs1V4DIz2QJDDV</xenc:CipherValue>
            </xenc:CipherData>
          </xenc:EncryptedData>
        </qs_data>
      </case>
    </cases>
  </data_container>
</body>
  
```

Abbildung 64: Verschlüsselung eines XML-Elements (qs_data)

In diesem XML-Dokument existieren einige spezielle Elemente. Die wichtigsten sind:

- EncryptedData ist das einschließende Element für die XML-Verschlüsselung. Der gesamte Inhalt des übergeordneten Elements einschließlich der Attribute ist verschlüsselt.
- CipherData ist das verschlüsselte Element.
- CipherValue enthält die verschlüsselten Daten.
- KeyInfo enthält Informationen über den Schlüssel, mit dem die Daten verschlüsselt worden

¹³⁶ Ebd.

sind.

- Das Attribut `id` enthält den Namen des `PublicKeys` und den Namen des zu verschlüsselnden „Tags“ und die aktuelle Versionsnummer des jeweils aktuellen XPackers.

Syntax/Hilfe

Mit dem Befehl `-h` wird die Syntax des Programms und eine Parameterübersicht ausgegeben:

```
java -jar XPacker-4.1.2-jar-with-dependencies.jar -h
```

Verschlüsseln

Die Verschlüsselung wird durch den Parameter `-e` aktiviert.

Benötigt wird der Parameter `-k`, gefolgt von Dateinamen des öffentlichen Schlüssels, der Parameter `-t` mit durch Komma getrennten Element-Namen sowie der Parameter `-2` mit Tag-Name, in den der verschlüsselte Schlüssel aufgenommen wird. Optional sind die Parameter `-f` und `-o`.

`-f` gibt die einzulesende XML-Dateien an und `-o` die auszugebende Datei. Falls diese Parameter fehlen, wird die Standard Ein-/Ausgabe verwendet.

Beispiel:

```
java -jar XPacker-4.1.2-jar-with-dependencies.jar -e -f in-  
file.xml -o outfile.xml  
-k datenannahme.pub  
-t qs_data -2 encryption
```

Entschlüsseln

Die Entschlüsselung entspricht dem Verschlüsseln, nur dass der Parameter `-2` entfällt und bei dem Parameter `-k` der private Schlüssel angegeben werden muss und der Parameter `-e` durch `-d` ersetzt wird.

Beispiel:

```
java -jar XPacker-4.1.2-jar-with-dependencies.jar -d -f in-  
file.xml -o outfile.xml  
-k datenannahme.pri -t qs_data
```

Schlüsselpaare erzeugen

Für die Erzeugung eines Schlüsselpaares wird der Parameter `-g` verwendet. Optional kann mit `-o` ein Dateiname angegeben werden. An diesen Namen wird `.pub` für den öffentlichen Schlüssel und `.pri` für den privaten Schlüssel angehängt. Falls der Parameter `-o` fehlt, wird im aktuellen Verzeichnis ein `key.pub` für den öffentlichen Schlüssel und ein `key.pri` für den privaten Schlüssel erzeugt.

Beispiel:

```
java -jar XPacker-4.1.2-jar-with-dependencies.jar -g -k datenannahme
```

Erzeugt im aktuellen Verzeichnis die Dateien datenannahme.pub und datenannahme.pri.

5.3.2 TPacker – Transportverschlüsselung

Das Programm TPacker.jar dient zur Transportverschlüsselung. Neben dem Ver- und Entschlüsseln werden die Dateien auch ge- und entpackt (.zip).

Beim Ver- und Entschlüsseln wird aus dem symmetrischen Schlüssel (Passwort) über ein MD5-Hash ein AES-Schlüssel erzeugt. Die eigentliche Verschlüsselung erfolgt im ECB-Mode. Als Padding¹³⁷ verwendet das Programm den PKCS5-Standard.

Die Steuerung der Funktionen erfolgt über Parameter beim Programmaufruf.

Syntax/Hilfe

Mit dem Befehl `-h` werden die Syntax des Programms und eine Parameterübersicht ausgegeben

```
java -jar TPacker-4.1.2-jar-with-dependencies.jar -h
```

Isolierte Varianten

Das Packen und Verschlüsseln bzw. das Entschlüsseln und Entpacken können unabhängig voneinander durchgeführt werden.



Achtung **Anwender- und Übertragungsfehler**

Die im Folgenden beschriebenen Aufrufe von isoliertem Ver- und Entschlüsseln sind komplex und können schnell zu Anwender- und Übertragungsfehlern führen.

Es müssen die richtigen Schritte mit den richtigen Parametern exakt in der richtigen Reihenfolge durchgeführt werden. Dazu sind die im Folgenden beschriebenen Varianten zu nutzen.

Verschlüsseln

Die Verschlüsselung wird durch die Parameter `-e` aktiviert. Benötigt wird noch der Parameter `-f` mit dem Dateinamen der zu verschlüsselnden Datei und der Parameter `-p` mit dem Passwort.

Falls nichts anderes angegeben wird, entspricht der Name der Ausgabedatei dem mit dem Parameter `-f` angegebenen Eingabedateinamen.

Mit dem Parameter `-o` kann man den Namen der Ausgabedatei ändern. In beiden Fällen wird an den Dateinamen ein Zeitstempel und `.aes` hinzugefügt. Falls der Zeitstempel nicht gewünscht ist, wird dieser mit dem Parameter `-t` deaktiviert.

¹³⁷ Als „Padding“ wird hier das Auffüllen eines unvollständigen Blocks am Ende des Datenstroms bezeichnet.

Beispiel:

```
java -jar TPacker-4.1.2-jar-with-dependencies.jar -e -f Datei  
-o Dateiname -p geheim
```

würde die Datei Dateiname_2012_01_01_121212.aes erzeugen, wenn das Programm am 01.01.2012 um 12:12 Uhr und 12 Sekunden ausgeführt wird. Die Verschlüsselung würde mit dem Schlüssel „privat“ durchgeführt.

Entschlüsseln

Beim Entschlüsseln geht man genau wie beim Verschlüsseln vor, nur dass man den Parameter `-e` durch den Parameter `-d` ersetzt.

Der Dateiname der Ausgabedatei entspricht der von Parameter `-f`. Falls ein Suffix `.aes` enthalten ist, wird dieses entfernt.

Beispiel:

```
java -jar TPacker-4.1.2-jar-with-dependencies.jar -t -d -f Da-  
teii.aes -p geheim
```

würde die Datei „Datei“ erzeugen.

Packen

Beim Packen geht man genau wie beim Verschlüsseln vor, nur dass man die Parameter `-e` durch `-z` ersetzt und der Parameter `-p` überflüssig ist. Das Suffix der gebildeten Datei lautet `.zip`.

Der Parameter `-f` kann eine oder mehrere, durch Komma(,) getrennte Dateinamen (ohne Leerzeichen) enthalten. Bei Angabe eines Verzeichnisses gibt es einen Fehler.

Beispiel:

```
java -jar TPacker-4.1.2-jar-with-dependencies.jar -t -z -f  
Datei -o Dateiname
```

würde die Datei „Dateiname.zip“ erzeugen.

Entpacken

Beim Entpacken geht man genau wie beim Packen vor, nur dass man den Parameter `-z` durch den Parameter `-u` ersetzt.

Die Ausgabe über Parameter `-o` ist immer ein Verzeichnis. Falls kein Verzeichnis angegeben wird, wird das aktuelle Verzeichnis als Ausgabeverzeichnis verwendet.

Beispiel:

```
java -jar TPacker-4.1.2-jar-with-dependencies.jar -t -u -f Datei.zip -o Verzeichnis
```

würde die Datei „Datei(en)“ im Verzeichnis „Verzeichnis“ entpacken.

Integrierte Varianten

Das Packen und Verschlüsseln bzw. das Entschlüsseln und Entpacken kann in einem Schritt zusammengefasst werden.

Verschlüsseln mit integriertem Verpacken

Es ist auch möglich, in einem Schritt zu packen und zu verschlüsseln. Hier kombiniert man die Parameter zu `-ze`. Es wird immer zuerst gepackt und dann verschlüsselt. Als Suffix wird dann `.zip.aes` angehängt.

Beispiel:

```
java -jar TPacker-4.1.2-jar-with-dependencies.jar -t -ze -f Datei -o Dateiname -p geheim
```

würde die Datei „Dateiname.zip.aes“ erzeugen.

Entschlüsseln mit integriertem Entpacken

So wie das Packen und Verschlüsseln, kann auch das Entschlüsseln und Entpacken in einem Schritt zusammengefasst werden. Hier kombiniert man die Parameter zu `-ud`. Es wird immer zuerst entschlüsselt und dann entpackt. Die Ausgabe erfolgt auch hier immer in ein Verzeichnis.

Beispiel:

```
java -jar TPacker-4.1.2-jar-with-dependencies.jar -t -ud -f Datei.zip.aes -p geheim
```

würde die Datei „Datei.zip.aes“ im aktuellen Verzeichnis entschlüsseln und entpacken.

Der Modus „unsafe“

Der TPacker ist standardmäßig so eingestellt, dass vor der Verschlüsselung das zu verschlüsselnde Dokument darauf untersucht wird, ob ein Element `VERSICHERTENID` bzw. `VERSICHERTENIDNEU` vorhanden ist. Falls diese Prüfung positiv ausfällt oder kein wohlgeformtes XML gefunden wird, unterbricht eine entsprechende Fehlermeldung die Verschlüsselung. Durch diese Maßnahme soll verhindert werden, dass versehentlich eine Transportdatei VOR der XML-Verschlüsselung erzeugt werden kann.

Um diesen Mechanismus auszuschalten, kann der Parameter

```
--unsafe
```

gemeinsam mit dem Verschlüsselungsparameter `-e` verwendet werden.

Beispiel:

```
java -jar TPackter-4.1.2-jar-with-dependencies.jar --unsafe -e -f
Datei -o Dateiname -p geheim
```

würde beim Erzeugen der verschlüsselten Datei keine Prüfung auf den Inhalt vornehmen

5.3.3 TPackter und XPackter ohne externe Abhängigkeiten

XPackter und TPackter werden vom IQTIG bei Bedarf ohne externe Abhängigkeiten (ohne fremde Bibliotheken) zur Verfügung gestellt. Die aktuelle Version ist auf der Kommunikationsplattform des IQTIG zu finden.

**Hinweis**

Die Programmvarianten ohne externe Abhängigkeiten sind nur für die Softwareanbieter relevant, die selber die Programme in ein JAVA-Umfeld integrieren und dabei Konflikte mit bereits installierten Bibliotheken vermeiden wollen.

5.3.4 Programmierschnittstelle – API

Sämtliche Funktionen der Verschlüsselungsprogramme können über je eine Programmierschnittstelle (API) aufgerufen und direkt in einem Java-Umfeld verwendet werden.

5.3.5 GPackter

Für alle Anwender, die händisch die Verschlüsselung durchführen müssen, stellt der GPackter mit seiner grafischen Oberfläche eine interaktive Alternative zur Verwendung der Programme XPackter und TPackter dar. Der GPackter integriert dabei den XPackter und TPackter über die GUI, die als einzelne Programme damit nicht benötigt werden. Er ermöglicht die Verschlüsselung und Komprimierung von QS-Dateien im XML-Format, die im Rahmen der DeQS-RL, der Qesü-RL oder der QSKH-RL erstellt worden sind. Eine ausführliche Dokumentation liegt der Komponente bei.

5.4 LE-Pseudonymisierungsprogramm

Das LE-Pseudonymisierungsprogramm richtet sich ausschließlich an die Datenannahmestellen (DAS), um die leistungserbringeridentifizierenden Daten bei den Qesü-Modulen vor der Weiterleitung an das IQTIG nach einem einheitlichen Verfahren zu pseudonymisieren. Alle weiteren Informationen und die Beschreibung zur LE-Pseudonymisierung sind dem ZIP-Archiv und den darin enthaltenen Dokumentationen (Anwenderhandbuch und Datenflussspezifikation) auf der IQTIG-Website zu entnehmen.

Die Dokumentation besteht aus:

- Anwenderhandbuch mit der Beschreibung der einzelnen Programmkomponenten (grafische Benutzeroberfläche, Konsolenanwendung, Java-API) sowie die Funktionenbeschreibung (Konfiguration, Pseudonymisierung, Depseudonymisierung, Datenübertragung etc.).
- Datenflussspezifikation mit der Beschreibung der konkreten Datenflüsse bei der Pseudonymisierung und Datenübertragung der Zusatzinformationen.

C Anhang

Austausch von Pseudonymen im IQTIG

Das IQTIG betreibt für den G-BA eine Bundesauswertungsstelle (BAS). Diese BAS arbeitet mit sensiblen, personenbezogenen Daten von gesetzlich und privat Krankenversicherten Personen. Des Weiteren erhält die BAS hierzu Informationen über die entsprechenden Leistungserbringer. Zum Schutz der Daten arbeitet das IQTIG mit pseudonymisierten Datensätzen. Die patientenidentifizierenden Daten werden durch die unabhängige Vertrauensstelle des G-BA nach § 299 SGB V pseudonymisiert, die leistungserbringeridentifizierenden Daten werden gemäß den entsprechenden G-BA Richtlinien durch die jeweils zuständige Datenannahmestelle auf Landes- bzw. Bundesebene pseudonymisiert.

Die Pseudonymisierung erfolgt mit einer Kombination aus unterschiedlichen Verschlüsselungs- und Hashingalgorithmen, die nach aktuellem Kenntnisstand (Empfehlungen des Bundesamtes für Sicherheit in der Informationstechnik (BSI)) sicher sind. Mit steigender Rechenleistung und durch neue wissenschaftliche Erkenntnisse können diese Algorithmen jedoch kompromittierbar, also unsicher werden. Aus diesem Grund ist es notwendig, entsprechende Pseudonyme sowie die dahinterliegenden Algorithmen in regelmäßigen Abständen auszutauschen und den aktuellen Empfehlungen des BSI anzupassen. Ein weiterer Grund für einen Pseudonymwechsel, jedoch nur bezogen auf das Leistungserbringerpseudonym, ist der Wechsel einer Datenannahmestelle auf Landesebene zu einer anderen Institution/Organisation und der damit einhergehenden Nutzung eines anderen Schlüssels zur Pseudonymisierung der Daten.

Zeitfenster zum Austausch der Patientenseudonyme

Die Patientenseudonyme müssen aus Datenschutzgründen spätestens nach drei Jahren oder bei Bekanntwerden einer Kompromittierung ausgetauscht werden. Nach dem Wechsel des Geheimnisses bzw. des Pseudonymisierungsalgorithmus in der Vertrauensstelle erhält die BAS Daten zum gleichen Patienten/Fall unter einem anderen Pseudonym. Der Prozess, der zum Austausch der alten Pseudonyme in der BAS nach dem Geheimniswechsel führt, kann u. U. mehrere Wochen betragen. Um eine korrekte Zusammenführung von Patientenseudonymen für Auswertungen zu ermöglichen, sollte für die Zeiten einer Auswertungserstellung ein Pseudonymwechsel möglichst vermieden werden. Deshalb ist es notwendig, den Wechsel der Pseudonyme sorgfältig zu planen und eng zwischen der BAS und der Vertrauensstelle abzustimmen, um Verzögerungen im Verfahrensablauf zu vermeiden.

Es gibt keinen Stichtag bzw. Zeitraum, in dem der Pseudonymaustausch völlig reibungsfrei durchgeführt werden könnte. Ein günstiger Zeitraum besteht jedoch nach dem Datenannahmeschluss (23.03. für QS-Daten bzw. 30.04. für Sozialdaten (siehe Abbildung 65)) und der anschließenden Erstellung der Auswertungsdatenpools beim IQTIG. Zu diesen Zeitpunkten sind alle Daten für die Leistungserbringer-, Landes- und Bundesauswertungen im IQTIG eingetroffen und

konnten ohne Pseudonymmappings verarbeitet werden. Bei einer *Umstellung der Pseudonyme im Mai/Juni/Juli* besteht ausreichend zeitlicher Puffer, um den Pseudonymtausch im IQTIG durchzuführen, bevor die Daten für weitere Auswertungen benötigt werden.

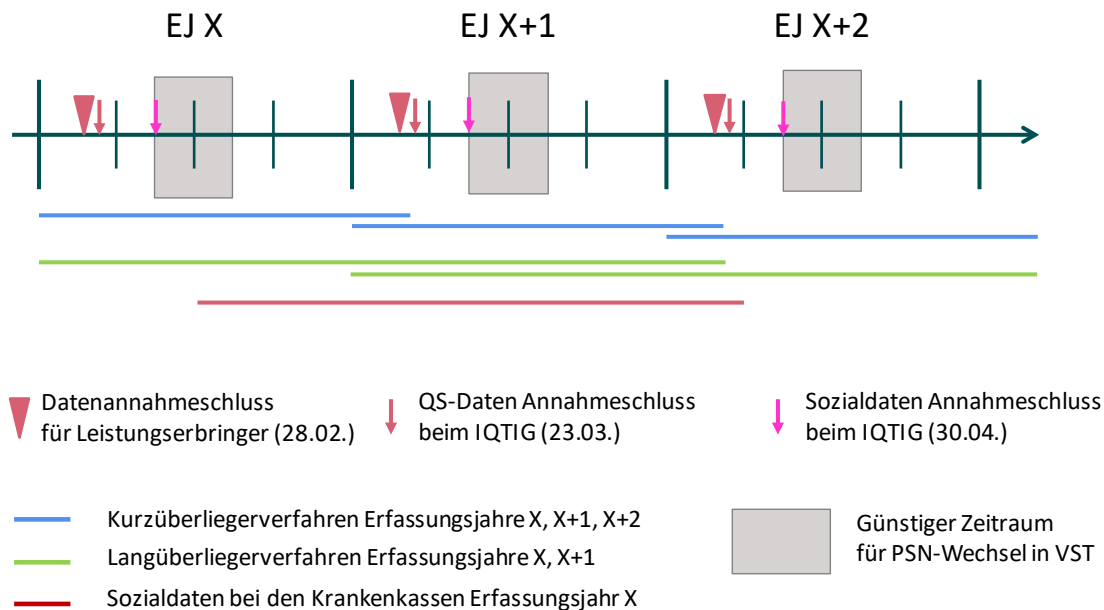


Abbildung 65: Darstellung der Überlappung von Datenlieferungen

Ein optimales Zeitfenster für den Austausch der Geheimnisse/Algorithmen in der Vertrauensstelle stellt somit der Juni (z. B. 01.06.) dar, so dass auch die Quartalslieferung für die QS-Daten des ersten Quartals abgeschlossen ist. Die Vertrauensstelle informiert den G-BA und das IQTIG im Januar über einen bevorstehenden Pseudonymwechsel, sofern

- es sich um einen regelhaften Wechsel handelt oder
- ein Wechsel der Vertrauensstelle bevorsteht

Im Falle einer Kompromittierung kann ein Wechsel jederzeit und umgehend von der Vertrauensstelle initiiert werden.

Spezifikation des Austauschverfahrens von Patientenpseudonymen

Der Austausch der Pseudonyme erfolgt nach folgendem Vorgehen:

- Die Vertrauensstelle wechselt das Geheimnis für die Pseudonymgenerierung
- Sie informiert die BAS, dass die Umstellung erfolgt ist, und übermittelt der BAS die Geheimnis-ID(s) des/der QS-Verfahren, bei denen ein Geheimniswechsel erfolgte.
- Das IQTIG erstellt anhand der Geheimnis-ID je QS-Verfahren eine Tabelle mit allen im IQTIG vorliegenden Patientenpseudonymen („alte“ Patientenpseudonyme).

Die Tabelle zur Übermittlung der alten Patientenpseudonyme vom IQTIG an die Vertrauensstelle entspricht folgendem Schema.

- Dateinamensschema:
 abgleich_<Verfahrenskennung>_<Erstellungsdatum: JJJJ-MM-TT-HH-

```
mm-ss>.csv
```

- Inhalt (Tabellenspalten)
 - <String: Verfahrenskennung>, <Boolean: Alte Versionennummer.>, <Long: Geheimnis-ID alt>, <String: Altes Pseudonym>
- Beispiel
 - PCI, FALSE, 151, 1af45178e01af45178e01af45178e01af45178

Die Tabelle(n) mit den alten Pseudonymen werden zwischen dem IQTIG und der Vertrauensstelle über einen SFTP-Server, der ausschließlich für den Austausch der Pseudonyme genutzt wird, übertragen.

Die Vertrauensstelle erzeugt die zu jedem alten Pseudonym gehörenden neuen Pseudonyme und speichert das Ergebnis wiederum in einer csv-Datei nach folgenden Vorgaben:

- Dateinamensschema:
 - abgleich_ergebnis_<Verfahrenskennung>_<Erstellungsdatum:JJJJ-MM-TT-HH-mm-ss>.csv
- Inhalt (Tabellenspalten)
 - <String: Verfahrenskennung>, <Long: Geheimnis-ID alt>, <String: Altes Pseudonym>, <Long: Geheimnis-ID neu>, <String: Neues PSN>
- Beispiel
 - PCI, FALSE, 151, 1af45178e01af45178e01af45178e01af45178, 317, ff76879678ff76879678ff76879678ff768796

Das IQTIG ersetzt nach Erhalt der Mappingtabelle umgehend alle alten Pseudonyme und Geheimnis-IDs durch die neuen Versionen und löscht anschließend die Mappingtabelle.

Austausch der Leistungserbringerpseudonyme beim Wechsel einer Datenannahmestelle

Beim Wechsel einer Datenannahmestelle (DAS) auf Landesebene kann es dazu kommen, dass die neue DAS aus Datenschutzgründen oder organisatorischen Gründen das Schlüsselpaar (öffentlicher und privater Schlüssel) zur Erzeugung der Leistungserbringerpseudonyme (LE-PSN) nicht übergeben bekommt. Die neue DAS ist somit gezwungen, sich ein neues Schlüsselpaar zu erstellen. Die damit erzeugten Leistungserbringerpseudonyme können im IQTIG nicht mit den alten Datensätzen verknüpft werden. Die Vertrauensstelle in der Funktion der Datenannahmestelle der Krankenkassen (DAS-KK) besitzt die öffentlichen Schlüssel aller DAS und kann somit die LE-PSN aller Bundesländer erstellen. Sollte ein Wechsel einer DAS erfolgen, übermittelt die neue DAS ihren öffentlichen Schlüssel an die DAS-KK. Die DAS-KK wäre somit in der Lage, das alte und neue LE-PSN zu erstellen und könnte somit beim Wechsel der Leistungserbringerpseudonyme unterstützen.

Erfolgt ein Wechsel der DAS, ist ein schnellstmöglicher Austausch der Leistungserbringerpseudonyme im IQTIG in enger Absprache mit der Vertrauensstelle und dem G-BA durchzuführen. Hierzu erstellt das IQTIG je QS-Verfahren eine Tabelle mit einer Liste aller LE-PSN des entsprechenden Bundeslandes/Sektors und übermittelt diese an die Vertrauensstelle. Hierfür wird der

SFTP-Server genutzt, der auch für den Austausch der Mapping-Tabellen zum Patientenpseudonymtausch genutzt wird. Die Tabellen haben folgende Vorgaben:

- Dateinamensschema:
abgleich_LE-PSN_<Verfahrenskennung>_<Landeskennung>_<DAS-Typ>_<Erstellungsdatum:JJJJ-MM-TT-HH-mm-ss>.csv
- Inhalt (Tabellenspalten)
<String: Verfahrenskennung>, <String: Landeskennung>, <String: DAS-Typ>, <String: Altes LE-Pseudonym>
- Beispiel
PCI, BE, KV, 1af45178e01af4517

Die Vertrauensstelle nutzt den alten sowie den neuen Schlüssel der entsprechenden DAS und erstellt das Mapping von altem zu neuem LE-PSN. Das Ergebnis wird mit Hilfe der folgenden csv-Datenstruktur über den gleichen SFTP-Server an das IQTIG übermittelt:

- Dateinamensschema:
abgleich_ergebnis_LE-PSN_<Verfahrenskennung>_<Landeskennung>_<DAS-Typ>_<Erstellungsdatum:JJJJ-MM-TT-HH-mm-ss>.csv
- Inhalt (Tabellenspalten)
<String: Verfahrenskennung>, <String: Landeskennung>, <String: DAS-Typ>, <String: Altes LE-Pseudonym>, <String: Neues LE-Pseudonym>
- Beispiel
PCI, BE, KV, 1af45178e01af4517, ff76879678ff768796

Nach Erhalt des Ergebnisses ersetzt das IQTIG die alten durch die neuen LE-PSN und löscht die Mappingtabelle.

Glossar

| Begriff | Beschreibung |
|--------------------------------|---|
| Abrechnungsdaten | Daten, die von Leistungserbringern über von ihnen erbrachte Leistungen zum Zweck der Abrechnung mit den Kostenträgern dokumentiert und übermittelt wurden. |
| Anonymisierung | Verändern personenbezogener Daten derart, dass die Einzelangaben über persönliche oder sachliche Verhältnisse nicht mehr oder nur mit einem unverhältnismäßig großen Aufwand an Zeit, Kosten und Arbeitskraft einer bestimmten oder bestimmaren natürlichen Person zugeordnet werden können. (§3 Abs. 6a BDSG) |
| Auslösekriterien | Algorithmus zur Auslösung der Dokumentationspflicht (QS-Filter). |
| Basisspezifikation | Gemeinsame Spezifikation für stationäre, einrichtungs- und sektorenübergreifende QS-Dokumentationen. |
| Bundesdatenpool | Zusammenführung aller bundesweit dokumentierten QS-Daten. |
| Datenannahmestellen | Stellen, an die die Leistungserbringer oder andere Daten liefernde Stellen (z. B. Krankenkassen) die erhobenen Daten übermitteln. Sie prüfen die übermittelten Daten auf Plausibilität, Vollständigkeit und Vollzähligkeit. |
| Datenbasis | Im Einzelfall festzulegende bzw. festgelegte Gesamtmenge von auszuwertenden bzw. ausgewerteten Daten. |
| Datenfeld | Kleinste Einheit eines Datensatzes (z. B. Angabe des Geschlechts im Dokumentationsbogen). |
| Datenfluss | Übermittlung der Daten der QS-Verfahren in einem festgelegten Format und Inhalt, die vom Leistungserbringer über eine Datenannahmestelle, Vertrauensstelle (nur einrichtungs- und sektorenübergreifende PID-Verfahren) bis zur Datenauswertungsstelle gelangen. Die Datenflüsse sind grundsätzlich in der QSKH-RL, der Qesü-RL und der DeQS-RL des G-BA festgelegt. |
| Datensatz | Eine zusammenhängende Menge von QS-Daten, die einem Fall (beispielsweise einem Patienten) zugeordnet wird. |
| Datenvalidierung | Verfahren zur Überprüfung von QS-Daten einerseits auf Vollzähligkeit, Vollständigkeit und Plausibilität (statistische Basisprüfung), andererseits ihre Übereinstimmung (Konkordanz) mit einer Referenzquelle wie bspw. der Krankenakte (Stichprobenverfahren mit Datenabgleich). |
| Diagnosis Related Groups (DRG) | Diagnosebezogene Fallgruppen zur pauschalierten Abrechnung stationärer Behandlungsfälle. Wesentliche Grundlage für die Ermittlung einer DRG, das sogenannte Grouping, sind Diagnosen- (ICD) und Prozeduren-Kodes (OPS). |

| Begriff | Beschreibung |
|--|---|
| Dokumentation | siehe: QS-Dokumentation |
| Einheitlicher Bewertungsmaßstab (EBM) | Vergütungssystem für die Abrechnung von Leistungen im vertragsärztlichen Bereich. |
| Einrichtung | siehe: Leistungserbringer |
| Erfassungsjahr | Das Jahr, in dem die Daten erhoben werden. Hierauf beruhen die Ergebnisse der Indikatoren. Die Kriterien für die Abgrenzung des Erfassungsjahres sind in der Spezifikation zur Sollstatistik definiert. |
| Follow-up | auch: Mehrpunktmessung, Mehrzeitpunktmessung. Versorgungsleistungen, deren Qualität (z. B. ein Behandlungserfolg) nicht einmalig, sondern zu unterschiedlichen weiteren Zeitpunkten gemessen wird. Die Messungen können in einem bestimmten oder auch in unterschiedlichen Sektoren (vgl. QSKH-RL) stattfinden. |
| 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 etwa 70 Millionen Versicherte und legt damit fest, welche Leistungen der medizinischen Versorgung von der GKV erstattet werden. |
| Indikator | „Anzeiger“. Auch: Qualitätsindikator. Quantitatives Maß zur Bewertung der Erreichung eines Qualitätsziels einer bestimmten gesundheitlichen Versorgungsleistung. |
| Instrument | siehe: Erfassungsinstrument. In einem weitergehenden Verständnis beschreibt ein Instrument im Bereich der Qualitätsförderung eine Methode bzw. ein Werkzeug, das mit dem Ziel der Verbesserung der Versorgungsqualität eingesetzt wird. |
| Leistungsbereich | hier: Leistungsbereich der gesetzlichen Qualitätssicherung (auch: QS-Verfahren). Medizinischer Themen- oder Behandlungsbereich, der im Rahmen der bundesweiten Qualitätssicherung dokumentationspflichtig ist. |
| Leistungserbringer | Personen und Einrichtungen, die medizinische Versorgungsleistungen erbringen bzw. bereitstellen. Der Begriff wird im SGB V auch für Ärzte und ärztliche Einrichtungen sowie für zugelassene Krankenhäuser gem. §108 SGB V genutzt. |
| Leistungserbringeridentifizierende Daten (LID) | Daten, die eindeutig einen bestimmten Leistungserbringer identifizieren (z. B. Institutionskennzeichen oder Betriebsstättennummer). |
| Minimaldatensatz | Wenn in der klinischen Praxis Konstellationen auftreten, in denen ein Fall durch den QS-Filter ausgelöst wird, obwohl der Datensatz |

| Begriff | Beschreibung |
|--|---|
| | des entsprechenden Leistungsbereichs nicht für die Dokumentation geeignet ist, kann ein Ersatzdatensatz mit minimalen Angaben ausgefüllt werden. |
| Missing Values | „Fehlende Werte“, z. B. fehlende Antworten und nicht auswertbare Antworten bei der Auswertung eines Fragebogens. |
| Operationen- und Prozedurenschlüssel (OPS) | Kodierungssystem für medizinische Leistungen bzw. verbindlicher Abrechnungsschlüssel zwischen medizinischen Leistungserbringern und Kostenträgern (Gesetzliche Krankenkassen). |
| 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. |
| Probetrieb | Erprobung eines QS-Verfahrens in einer begrenzten Anzahl von Einrichtungen. Ziel ist die Prüfung, ob die für das QS-Verfahren benötigten Daten entsprechend der vom Auftragnehmer vorgesehenen Planungen für die vorgelegten Indikatoren und Instrumente erhebbar und die Ergebnisse umsetzbar sowie für die Durchführung der Qualitätssicherung verwertbar sind. |
| Pseudonymisierung | Ersetzen des Namens und anderer Identifikationsmerkmale durch ein Kennzeichen zu dem Zweck, die Bestimmung des Betroffenen auszuschließen oder wesentlich zu erschweren (§3 Abs. 6a BDSG). |
| Pseudocode | Programmcode, der das zugrunde liegende Prinzip eines Algorithmus beschreibt, selbst aber nicht lauffähig ist. Er dient zur Veranschaulichung, unabhängig von der konkret zu verwendenden Programmiersprache. |
| Qualität | Bezogen auf die Gesundheitsversorgung: Grad, in dem versorgungsrelevante Ergebnisse, Prozesse und Strukturen bestimmte, definierte Anforderungen erfüllen. |
| Qualitätsindikator | siehe: Indikator |
| Qualitätssicherung | Sammelbegriff für unterschiedliche Ansätze und Maßnahmen zur Sicherstellung festgelegter Qualitätsanforderungen bzw. zur Erreichung bestimmter Qualitätsziele. Hier: Gesetzliche Qualitätssicherung im Gesundheitswesen nach §§135-139 SGB V. |
| Qualitätssicherung, externe stationäre | Einrichtungsübergreifende Qualitätssicherung für medizinisch-pflegerische Leistungen, die ausschließlich im stationären Sektor erbracht werden. |
| QS-Auslösung | Initiierung einer Dokumentationspflicht zu Zwecken der Qualitätssicherung (QS-Dokumentation). Bei einer Erhebung vorhandener Daten (z. B. Sozialdaten bei den Krankenkassen) analog das |

| Begriff | Beschreibung |
|--------------------|--|
| | Kriterium, das die Lieferung eines bestimmten Datensatzes auslöst. |
| QS-Daten | Sammelbegriff für alle Daten, die im Zuge eines QS-Verfahrens erhoben und ausgewertet werden. |
| QS-Dokumentation | Gesonderte Erhebungen der Leistungserbringer zu Diagnose- und Behandlungsdaten der Patienten durch die Leistungserbringer für die Qualitätssicherung. |
| QS-Filter | Algorithmus, der auf Grundlage festgelegter Kriterien die für die Qualitätssicherung durch die Leistungserbringer zu dokumentierenden Patienten und deren Daten „filtert“. Die Kriterien hierzu werden in einer Spezifikation definiert. |
| QS-Filter-Software | Implementierung der Spezifikation für den QS-Filter. |
| QS-Verfahren | siehe: Leistungsbereich |
| Regelbetrieb | auch: Routinebetrieb oder Echtbetrieb. Verpflichtende und flächendeckende Umsetzung eines QS-Verfahrens. |
| Reliabilität | Maß für die Zuverlässigkeit einer Messmethode bzw. für die mit dieser Methode gewonnenen Daten, wobei die Messung (Datenerhebung und Auswertung) reproduzierbar sein muss. |
| Risikoadjustierung | Methode zur Ermittlung und Gewichtung wesentlicher Einflussfaktoren (individueller Risiken) auf die Ausprägung einer Messgröße. Eine Risikoadjustierung ist vor allem bei Qualitätsindikatoren erforderlich, die sich auf Behandlungsergebnisse und einen Einrichtungsvergleich beziehen. Um hier einen fairen Vergleich zu erhalten, dürfen nur in gleichem Umfang erkrankte Patienten in Einrichtungen mit einer vergleichbaren Klientel miteinander verglichen werden. |
| Routinedaten | hier: Daten, die wesentlich zur Abwicklung von Geschäfts- und Verwaltungsabläufen erhoben werden (z. B. Abrechnungsdaten, personenbezogene administrative Daten). Abseits des uneinheitlichen Sprachgebrauchs stehen die Sozialdaten bei den Gesetzlichen Krankenkassen (auch: GKV-Routinedaten) im Vordergrund des Interesses, da sie gem. §299 Abs. 1a SGB V zu Zwecken der Qualitätssicherung verwendet werden dürfen. Diese beinhalten insbesondere die abrechnungsrelevanten Daten für ambulante und stationäre Versorgungsleistungen (§§295 und 301 SGB V), für Arznei-, Heil- und Hilfsmittel (§§300 und 302 SGB V) sowie die Versichertenstammdaten (§284 SGB V). |
| Sektor | Institutionell, d.h. durch unterschiedliche Abrechnungsmodalitäten getrennte Bereiche der medizinisch-therapeutischen Versorgung im deutschen Gesundheitswesen (z. B. ambulante/stationär). |

| Begriff | Beschreibung |
|-----------------------|--|
| Sollstatistik | Aufstellung der im Erfassungsjahr zu dokumentierenden Leistungen, die vom Krankenhaus durch Konformitätserklärung schriftlich bestätigt wird. Sie gibt die zu erwartende Anzahl von Fällen in den einzelnen Leistungsbereichen (Soll) an und bildet zusammen mit der Anzahl der tatsächlich durch die Leistungserbringer gelieferten Daten (Ist) die Grundlage der Vollzähligkeitsprüfung. |
| Sozialdaten | Einzelangaben über die persönlichen und sachlichen Verhältnisse (personenbezogene Daten), die von den sozialrechtlichen Leistungsträgern zur Erfüllung ihrer gesetzlichen Aufgaben gesammelt und gespeichert werden. |
| Spezifikation | Datensatzbeschreibung. Festlegung, welche Daten für die Qualitätssicherung erhoben bzw. übermittelt werden müssen, welche Prüfalgorithmen zur Anwendung kommen (z. B. für Plausibilitätsprüfungen) und wie die QS-Auslösung operationalisiert ist. Im Rahmen der Neuentwicklung von QS-Verfahren ist die Spezifikation als das Ergebnis der informationstechnischen Aufbereitung zu betrachten. |
| Strukturierter Dialog | Instrument der Qualitätsförderung. Strukturiertes Verfahren, das Einrichtungen bei der kontinuierlichen Verbesserung von Prozessen und Qualität unterstützt. Auslöser des Strukturierten Dialogs sind rechnerische Auffälligkeiten im Ergebnis eines Qualitätsindikators im Rahmen der statistischen Basisprüfung der übermittelten QS-Daten (QSKH-RL). Abweichend von der Definition in der QSKH-RL bezeichnet der Begriff „Strukturierter Dialog“ in der Qesü-RL nur den Prozessschritt, der dem schriftlichen Teil der Durchführung von QS-Maßnahmen folgt. |
| Systempflege | Routinemäßige und kontinuierliche Evaluation und Anpassung der Qualitätsindikatoren, der Softwarespezifikation usw. |
| Verfahren | siehe: Leistungsbereich |
| Vertrauensstelle | Institution, die im Rahmen der einrichtungs- und sektorenübergreifenden Qualitätssicherung erhobene patientenidentifizierende Daten pseudonymisiert. Näheres regeln die die entsprechenden Richtlinien (QSKH-RL, Qesü-RL oder DeQS-RL) sowie themenspezifische Bestimmungen. |
| Verweildauer | Dauer des stationären Aufenthalts eines Patienten, Abstand zwischen Aufnahme- und Entlassungsdatum. |
| Vollständigkeit | Erfassung aller zu einem einzelnen Behandlungsfall erforderlichen Angaben (Daten). |
| Vollzähligkeit | Erfassung aller dokumentationspflichtigen Behandlungsfälle zu einer bestimmten Versorgungsleistung. |

| Begriff | Beschreibung |
|------------------------|---|
| Vollzähligkeitsprüfung | Abgleich der laut QS-Filter-Software zu erwartenden Anzahl von Fällen in einem Leistungsbereich (Soll) mit der Anzahl der tatsächlich durch die Leistungserbringer gelieferten Daten (Ist). |
| 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. |
| Zählleistungsbereich | Bei Leistungsbereichen, die über einen gemeinsamen Dokumentationsbogen erfasst werden (beispielsweise Herzchirurgie), dient der Zählleistungsbereich der Zuordnung zu einer definierten Teilmenge eines Leistungsbereichs (beispielsweise kathetergestützte vs. offen-chirurgische vs. kombinierte Eingriffe im Leistungsbereich <i>Aortenklappenchirurgie</i>). |